



16th Brazilian Congress of Mechanical Engineering

Engineering for the New Millennium

ROBOTICS AND CONTROL

ISBN 85-85769-06-6



Robotics and Control

TRB0010	9
METHOD OF CAMERA'S CALIBRATION TO BUILT A VIRTUAL ENVIRONMENT	
Silva, Silvana Aparecida Graminho da Okamoto-Jr., Jun	
TRB0021	19
THREE-AXIS FORCE SENSOR WITH STRAIN GAGES FOR ROBOTICS APPLICATIONS	
Villaça, Rodolfo da Silva Amaral, Paulo Faria Santos Pinto, Benedito Geraldo Miglio Bento Filho, Antônio	
TRB0073	28
DEVELOPMENT OF AN OMNIDIRECTIONAL VISION SYSTEM	
Grassi Junior, Valdir Deccó, Cláudia C. G. Okamoto Junior, Jun Porto, Arthur José Vieira	
TRB0103	38
BASIC PRINCIPLES AND CONCEPTS OF BALLISTIC LOCOMOTION	
Dutra, Max Suell Pina Filho, Armando Carlos de	
TRB0121	45
GENERATION OF WALKING PATTERNS OF AN HEXAPOD ROBOT THROUGH SCHEDULING ALGORITHMS	
Dutra, Max Suell Mello, Vinícius da Costa	
TRB0140	55
MONOCULAR VISION SYSTEM FOR AUTOMATIC GUIDANCE OF A WELDING ROBOT	
Alfaro, Sadek Absi Siqueira, Milton L. Motta, José M. S. T.	
TRB0150	61
INSTRUMENTED VEHICLE FOR INSPECTION AND MAINTENANCE OF PIPES	
Dutra, Max Suell Catunda, Carlos Eduardo Guedes Ferreira, Cesar Gomes	
TRB0158	71
PETRI NETS AND OBJECT-ORIENTATION FOR HYBRID CONTROL SYSTEM DEVELOPMENT	
Villani, Emilia Miyagi, Paulo Eigi Valette, Robert	
TRB0219	81
ULTIMATE BOUNDEDNESS POLYHEDRA FOR DISCRETE-TIME LINEAR SYSTEMS WITH DEADZONES	
Milani, Basilio E. A. Coelho, Alessandra D.	

TRB0268	DYNAMIC MODELING AND SIMULATION OF A BIPED ROBOT	91
Campos, Daniel Carmona de Bezerra, Carlos Andre Dias Zampieri, Douglas Eduardo Mendeleck, Andre		
TRB0284	POSITION CONTROL OF A FLEXIBLE STRUCTURE WITH PIEZOELECTRIC ACTUATORS	100
Tamai, Edilson Hiroshi		
TRB0289	OPTIMUM DESIGN OF GENERAL 3R MANIPULATORS BY USING TRADITIONAL AND RANDOM SEARCH OPTIMIZATION TECHNIQUES	107
Lanni, Chiara Saramago, Sezimaria F. P. Ceccarelli, Marco		
TRB0363	NUMERICAL SIMULATION AND EXPERIMENTAL VALIDATION OF DYNAMIC CHARACTERISTICS OF CAPAMAN (CASSINO PARALLEL MANIPULATOR)	117
Carvalho, João Carlos Mendes Ceccarelli, Marco		
TRB0366	OBSTACLE AVOIDANCE PROCEDURES FOR MOBILE ROBOTS	126
Becker, Marcelo		
TRB0404	DESIGN AND IMPLEMENTATION OF A HAPTIC INTERFACE FOR TELEPRESENCE	136
Rodrigues, Denilson Laudares Campos, Mario Fernando Montenegro Kumar, Vijay		
TRB0479	PIEZOELECTRIC ACTIVE SYSTEMS FOR MONITORING SURFACE BREAKING DEFECTS	146
Gama, Antonio Lopes Morikawa, Sérgio Ricardo Kokai Braga, Arthur Martins Barbosa Correia, Rafael Ribeiro		
TRB0537	THERMAL ALARM USING A SHAPE MEMORY ALLOY HELICAL SPRING	157
de Araujo, Carlos Jose Rocha, Edgard Augusto Silva Souto, Cícero da Rocha Gonzalez, Cezar Henrique		
TRB0551	THE CONTROL DESIGN OF A ROTARY PNEUMATIC POSITIONING SYSTEM	164
Hunold, Marcos Costa Cabral, Eduardo Lobo Lustosa		
TRB0624	A DISTRIBUTED CONTROL ARCHITECTURE FOR COOPERATIVE ROBOTIC SYSTEMS	174
Bergerman, Marcel Manrich, Christian		

TRB0782	184
APPLICATION OF FUZZY CONTROLLERS IN MICRO-CONTROLLERS TO THE CONTROL OF VIBRATIONS IN FLEXIBLE BEAM TYPE STRUCTURE	
Abreu, Gustavo Luiz Chagas Manhães Cardoso, Patrick Magalhães	
TRB0799	194
STRUCTURAL VIBRATION CONTROL USING THICKNESS-SHEAR MODE OF PIEZOELECTRIC ACTUATORS	
Trindade, Marcelo A.	
TRB0829	204
DIRECT ADAPTIVE NEURAL CONTROL APPLIED TO A HYDRAULIC ACTUATOR	
Riul, José Antônio Silva, José Felício Cavalcanti, José Homero Feitosa	
TRB0853	210
IMPROVING THE POSITIONING ACCURACY OF POWERFUL MANIPULATORS WITH APPLICATION IN NUCLEAR MAINTENANCE	
Meggiolaro, Marco Antonio Dubowsky, Steven	
TRB0854	220
MODELING, SIMULATION AND FORCE CONTROL OF A FLEXIBLE ROBOTIC ARM IMPACTING ITS ENVIROMENT	
Valer, Carlos Ingar Sampaio, Rubens	
TRB0860	230
POSITION CONTROL OF A MILLING MODULE USING PROPORTIONAL HYDRAULIC	
Schneider, Fernando Buendgens Cunha, Mauro André Barbosa De Negri, Victor Juliano	
TRB0906	240
A GLOBAL VISION SYSTEM FOR MOBILE MINI-ROBOTS	
Aires, Kelson Rômulo Teixeira Alsina, Pablo Javier Medeiros, Adelardo Adelino Dantas De	
TRB0918	248
WORKSPACE FOR A QUADRUPED ROBOT CONSIDERING YOUR LEG MECHANISM	
Pizzuolo, Tarcísio de Assunção Resende, Peterson de Silvino, José Luis Martins-Filho, Luiz De Siqueira	
TRB0920	256
A CONTROLLER AREA NETWORK FIELD BUS FOR AN UNMANNED ROBOTIC AIRSHIP	
Faria, Bruno Guedes Maeta, Silvio Mano Ramos, Josué Jr. Guimarães Bergerman, Marcel Bueno, Samuel Siqueira	

TRB0959	CONTROL OF A FLEXIBLE MANIPULATOR USING A NEURAL NETWORK FOR NON LINEAR FRICTION IDENTIFICATION AND COMPENSATION	266
Gomes, Sebastião Cícero Pinheiro Gervini, Vitor Irigon Machado, Celiane Costa		
TRB0976	HIERARCHICAL SINGULARITY ANALYSIS OF AN ARTICULATED ROBOT	274
Martins, Daniel Guenther, Raul Simas, Henrique		
TRB1010	FRICTION COMPENSATION OF ROBOTIC ACTUATORS THROUGH NEURAL NETWORK	284
Gervini, Vitor Irigon Gomes, Sebastião Cícero Pinheiro		
TRB1012	IMPLEMENTATION OF BOOLEAN OPERATION USING INTERVAL ARITHMETIC IN B-REP SOLID MODELLING	291
Shimada, Marcelo Tsuzuki, Marcos de Sales Guerra		
TRB1060	DETERMINATION OF OPERATION PERFORMANCE OF AN ULTRASONIC MOTOR FOR LINEAR DISPLACEMENT	301
Ibrahim, Ricardo Cury Rodrigues, Hugo Leonardo Paiva		
TRB1067	MULTIPLE NEURAL NETWORKS IN FLEXIBLE LINK CONTROL USING FEEDBACK-ERROR-LEARNING	307
De Almeida Neto, Areolino Góes, Luiz C. S. Nascimento Jr., Cairo L.		
TRB1144	VIBRATION CONTROL USING FUZZY LOGIC AND IMPLEMENTED IN MICROCONTROLLERS	316
Nuñez, Israel Jorge Cárdenas Ribeiro, José Francisco		
TRB1315	DESIGN METHODOLOGY OF A ROBOT FOR SURGICAL APPLICATIONS	323
Vidal, Walter de Britto Lima, Raul Gonzales Moscato, Lucas A. T		
RB1326	VISUAL SERVO CONTROL OF NONHOLONOMIC MOBILE ROBOTS	333
Carvalho, José Reginaldo H. Silveira, Geraldo F. Rives, Patrick Shiroma, Pedro Bueno, Samuel Siqueira		
TRB1329	DESIGN AND IMPLEMENTATION OF A TACTILE SENSOR FOR A ROBOTIC GRIPPER	342
Madani, Fernando Silveira Adade Filho, Alberto		

TRB1332	352
PROPOSAL OF UPPER LIMB PROSTHESIS MOVEMENT FROM THE MODELING FORWARD AND INVERSE KINEMATICS	
Casemiro, Edna Rodrigues Hermini, Helder Anibal Rosário, Joao Mauricio	
TRB1627	360
LASER INTERFEROMETRY TECHNIQUE APPLIED TO THE MEASUREMENT OF FLEXTENSIONAL TRANSDUCERS DISPLACEMENTS	
Nader, Gilder Shirahige, Alessandro Adamowski, Julio Cezar Silva, Emilio Carlos Nelli	
TRB1688	369
A DESIGN METHODOLOGY FOR THE ESTIMATION OF POSITIONING DEVIATION FOR GANTRY MANIPULATORS	
Ferreira, Claudio Violante Romano, Vitor Ferreira	
TRB1893	379
KINEMATICS MODELLING OF AN HEXAPOD ROBOT	
Dutra, Max Suell Netto, Salvador M. C.	
TRB1930	388
PROJECT OF A CONTROLLER LQG FOR A PNEUMATIC SYSTEM	
Oliveira, Benedito Santana Araújo, Clivaldo/Silva Silva, Simplicio Arnaud Riul, José Antônio	
TRB2050	397
CASCADE CONTROL OF A PNEUMATIC SERVODRIVE	
Guenther, Raul Perondi, Eduardo André	
TRB2121	407
ANALYSIS AND EXPERIMENTAL DEVELOPMENT OF A PLOTTER FOR DRAWING ON VERTICAL SURFACES	
Segenreich, Solly A. Fontanella, Jorge Luiz	
TRB2164	415
AN APPROACH TO STATE SPACE COMPUTATIONAL MODELING AND PREDICTION OF TIME SERIES IN PARALLEL AND DISTRIBUTED COMPUTERS	
Barreto, Gilmar Bottura, Celso Pascoli Bordon, Mauricio Jose Tamariz, Annabell Del Real	
TRB2216	426
PETRI NET TOOLS: A GRAPHICAL OPEN SYSTEM EDITOR AND SIMULATOR	
Soares, João Borsoi Inamasu, Ricardo Yassushi Silva, Andrea Ribari Yoshizawa Ferraz, Patricia Porto, Arthur Jose Vieira	

TRB2226	435
END MILL TOOL WEAR MONITORING THROUGH TORQUE MEASUREMENT	
Moscato, Lucas A. Cunha, Vinicius L. C.	
TRB2278	445
SUBSPACE IDENTIFICATION METHOD FOR MULTIVARIABLE SYSTEMS WITH INTERCONNECTED STRUCTURE	
Torricono, Angel Fernando Cáceres Bottura, Celso Pascoli	
TRB2315	454
PARALLEL AND DISTRIBUTED MULTIVARIABLE IDENTIFICATION VIA THE MOESP APPROACH.	
Barreto, Gilmar Bottura, Celso Pascoli Tamariz, Annabell Del Real Fonseca Neto, Joao Viana Da	
TRB2331	464
SEMI-ACTIVE SUSPENSION FOR PASSENGER VEHICLES USING FUZZY CONTROLLERS	
Teixeira, Rafael Luis Ribeiro, José Francisco	
TRB2376	474
SCREW-BASED JACOBIAN ANALYSIS OF A 3-DOF PARALLEL MANIPULATOR	
Valdiero, Antonio Carlos Campos, Alexandre Guenther, Raul Martins, Daniel	
TRB2378	484
METHODOLOGY FOR MODELING FLEXIBLE SYSTEMS OF MATERIALS MOVEMENT	
Junqueira, Fabrício Miyagi, Paulo Eigi	
TRB2396	494
DYNAMICS AND CONTROL OF A FLEXIBLE ROTATING ARM THROUGH THE MOVEMENT OF A SLIDING MASS	
Oliveira, Frederico Ricardo Ferreira de Fleury, Agenor de Toledo	
TRB2401	504
AUTOMATIC GENERATION OF CONTROL RULES FOR RESOURCE ALLOCATION IN PRODUCTION SYSTEMS WITH CONCURRENT PROCESSES	
Nakamoto, Francisco Yastami Maruyama, Newton Miyagi, Paulo Eigi Santos Filho, Diolino José dos	
TRB2410	514
GENERALIZED PREDICTIVE CONTROL FOR COOLING SYSTEMS BASED IN VAPOR COMPRESSION	
Galvez, Jose Maria Silva, Agostinho Gomes	



TRB2426	524
STRUCTURED MODEL BASED CONTROL ARCHITECTURE FOR PRODUCTION SYSTEMS	
SantosFilho, Diolino José dos	
Matsusaki, Cristina Toshie Motohashi	
Maruyama, Newton	
Miyagi, Paulo Eigi	
TRB2428	534
MICRO-ROBOT SOCCER TEAM - MECHANICAL AND HARD-WARE IMPLEMENTATION	
Alsina, Pablo Javier	
Vieira, Frederico Carvalho	
Medeiros, Adelardo Adelino Dantas de	
TRB2653	541
SPECIFIC PROBLEMS OF THE CAM MECHANISMS KINETO-ELASTODYNAMICS	
Oprisan, Cezar	
Leohchi, Dumitru	
TRB2668	551
NEURAL NETWORKS APPLIED TO THE PATTERN RECOGNITION	
Nunes, Luiz Eduardo Nicolini do Patrocinio	
Prado, Pedro Paulo Leite do	
TRB2681	558
KINEMATIC MODEL FOR A QUADRUPED ROBOT	
Bento Filho, Antônio	
Amaral, Paulo Faria Santos	
Pinto, Benedito Geraldo Miglio	

MÉTODO DE CALIBRAÇÃO DE CÂMERAS PARA CRIAÇÃO DE AMBIENTE VIRTUAL**Silvana A. G. da Silva**

Escola Politécnica da Universidade de São Paulo
Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Av. Prof. Mello Moraes, 2231, 05508-900, São Paulo, SP, Brasil
e-mail: silvanaa@usp.br

Jun Okamoto Jr.

Escola Politécnica da Universidade de São Paulo
Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Av. Prof. Mello Moraes, 2231, 05508-900, São Paulo, SP, Brasil
e-mail: jokamoto@usp.br

Resumo. Para controlar o movimento dos robôs a distância utiliza-se a teleoperação e para se ter acesso ao local de trabalho destes são utilizadas câmeras de vídeo e outros sensores. Para qualquer aplicação onde exista uma relação entre a imagem 2D e a realidade 3D deve ser feita a calibração da câmera, que corresponde a obtenção dos seus parâmetros intrínsecos e extrínsecos. O trabalho apresenta um novo método de calibração explícita de câmeras de vídeo fixas para aplicações em teleoperação de robôs móveis. Através do processamento de quatro imagens do robô, capturadas pela câmera, obtém-se os dados necessários para determinar a posição e a orientação da câmera em relação a um sistema de coordenadas globais e, também, construir o mapa virtual do ambiente. O método pode ser adaptado para qualquer formato de robô, não utiliza nenhum padrão de referência ou marcas no robô e baseia-se no movimento do robô, sendo dados o ângulo e a distância a ser percorrida. Com o mapa virtual e conhecendo-se a altura e os ângulos de inclinação da câmera é possível determinar as coordenadas de qualquer posição do robô no ambiente de trabalho. Os resultados obtidos comprovam a eficiência das equações desenvolvidas para calibração da câmera.

Palavras-chave: Calibração de câmeras, ambiente virtual, mapa virtual

1. Introdução

Com o aumento do número de robôs industriais, a robótica está se tornando uma área da engenharia com identidade própria. O avanço e a redução de custos da tecnologia de sensores viabiliza, cada vez mais, a construção de robôs com maior capacidade de realimentação de dados sobre o ambiente de trabalho, aumentando, desta maneira, a acuidade de posicionamento, a velocidade de operação e a flexibilidade dos mesmos. É consensualmente aceito que a visão computacional é a mais poderosa e flexível forma de realimentação de informações disponíveis sobre o ambiente e que tem despendido um considerável esforço em pesquisa e desenvolvimento para a área (Kabayama; Trabasso (1999)).

1.1. Calibração de Câmeras: Conceito

Calibração de câmera, segundo Lang (1971) é um processo preliminar para calcular os parâmetros físicos da câmera (parâmetros extrínsecos), como centro da imagem, posição e orientação, etc., (Calibração Explícita). Alguns parâmetros ópticos e geométricos internos da câmera (parâmetros intrínsecos) podem ser calibrados para uso em medições 3D, esta é a Calibração Implícita. Aplicações: Medidas 3D, monitoramento de segurança e movimentação de objetos em trajetórias. As pesquisas atuais sobre calibração de câmeras consistem basicamente de um objeto de referência com forma geométrica simples ou padrão colocado em uma certa posição e, então, se estabelece uma relação entre os parâmetros extrínsecos e as coordenadas dos objetos de calibração de acordo com os princípios de projeção perspectiva.

Num sistema de teleoperação de robôs móveis, o robô deve reconhecer e identificar o seu ambiente de trabalho. Uma das formas pela qual ele pode obter isso é através de uma câmera pelo processamento das imagens obtidas de seu ambiente. Em situações onde envolve a relação entre a imagem 2D (advinda da câmera) e a realidade 3D (ambiente de trabalho do robô) é necessário que a câmera seja calibrada. A câmera deve estar calibrada para que o operador tenha um referencial da posição e da orientação do robô em relação ao sistema de coordenadas considerado. Com isto é possível, através das imagens 2D coletadas pela câmera calibrada, saber com certa precisão a posição em que se encontra o robô possibilitando ao operador determinar as novas posições nas quais o robô deve se deslocar.

O processo de calibração da câmara, segundo Grosky; Tamburino (1990) analisa as medidas do sistema da câmara (pixels) para se obter os pontos de referência no sistema de coordenadas 3D. Nesta correspondência, o processo de calibração consiste de duas fases independentes:

1ª remover as distorções geométricas da câmara;

2ª usar uma transformação linear entre as coordenadas da câmara e as coordenadas dos pontos de referência conhecidos projetados sob uma imagem plana.

Na 2ª fase é concebido um modelo genérico que inclui 12 variáveis extrínsecas e 5 parâmetros intrínsecos. As variáveis extrínsecas incluem uma matriz de rotação que descreve a orientação do eixo óptico e os deslocamentos do ponto focal da câmara num sistema de coordenadas global. As variáveis intrínsecas corrigem os desvios do eixo óptico e dos eixos das coordenadas da câmera. Existem 3 importantes ângulos de rotação que são: *swing*, *pan* e *tilt*. Muitos

trabalhos tem sido desenvolvidos na área de calibração de câmeras abrangendo os mais diversos métodos. Alguns métodos usados para obter as equações de estimativa dos parâmetros da câmara utilizam objetos de calibração de formas variadas como círculos, elipses, linhas, pontos, pontos de referência 3D, objetos de calibração planar arbitrário, papel quadriculado 2D, cartas padrões, entre outros. A seguir são apresentados alguns estudos desenvolvidos nesta área que ilustram os métodos de calibração e utilizam os objetos acima citados.

Han; Rhee (1992) descrevem um método para calcular os parâmetros da câmara a partir da imagem de um círculo usando um padrão circular. Usam equações simples para determinar os ângulos *pan*, *tilt* e *swing* e a distância da câmara ao ponto de referência no sistema de coordenadas globais. Comparado com métodos atuais, este é um dos mais simples e precisos, sendo até trivial. O método consiste de um círculo de raio arbitrário com 2 furos que é usado como padrão para a calibração. O furo central indica a posição central do círculo e o outro, o furo direcional, indica informações direcionais. O círculo central é posicionado na origem do sistema de coordenadas global, e o furo direcional sobre o eixo X_w (eixo X do sistema de coordenadas da base). Então, a direção da câmara (equivalente ao eixo óptico da câmara) é ajustada manualmente até que o furo central coincida com o centro da imagem na tela. O objetivo é calcular os parâmetros a partir da imagem do círculo. Outro exemplo do uso do padrão circular pode ser visto em Ma (1993).

Métodos que utilizam linhas podem ser vistos em Vaillant (1990). Com os pontos, segundo a modelagem e o critério de adequação, o problema pode ser formulado linearmente ou não, com ou sem restrições. Isto conduz aos métodos analíticos TSAI (1986), ou aos algoritmos iterativos Chaumette; Rives (1990).

Hong; Yang (1993) desenvolveram um algoritmo para calibração da câmara usando um ponto de referência 3D, onde a idéia principal do método é que a relação entre os parâmetros de duas câmeras, (o ângulo *pan*, o *tilt* e as coordenadas globais de um dado ponto de calibração 3D), possa ser estabelecida pelos princípios da projeção perspectiva e ópticos geométricos e a solução das equações transcendentais possa ser obtida por algoritmos iterativos para calibrar os parâmetros da câmara.

Utilizando um objeto de calibração planar arbitrário, Chen et all (1991) apresentaram uma metodologia para estimar os parâmetros da câmara. As coordenadas do objeto de calibração em relação a câmara sofrem uma combinação arbitrária de transformações em termos de 3 rotações e 3 translações e, então, o objeto é gravado numa imagem. Todas as possíveis vistas do objeto de calibração são geradas para construir uma biblioteca. Ao obter-se uma imagem do objeto de calibração, esta corresponderá a uma das vistas armazenadas. Quando uma imagem é apresentada, são feitas comparações com as da biblioteca, procurando outra idêntica, ou aproximadamente idêntica a esta imagem.

Uma carta padrão simples de calibração composta de linhas retas paralelas é usada como referência para calibração em Nomura et all (1992). Seu artigo apresenta um método simples e útil de calibração para câmara de TV com lentes de alta distorção. Os parâmetros a serem calibrados são o comprimento focal efetivo, a largura de um pixel em uma imagem plana, o centro de distorção da imagem e o coeficiente de distorção. Alguns parâmetros são eliminados com o ajuste preciso da carta de calibração e com a utilização da baixa distorção próximo ao centro da imagem. Com isso, o número de parâmetros desconhecidos a serem calibrados são reduzidos drasticamente, possibilitando uma calibração simples e útil.

O item 2 faz uma breve descrição do método proposto e o item 3 descreve um processo utilizado para aprimorar este método. Seguem a análise dos resultados e as conclusões deste trabalho.

2. Descrição do Método Proposto

Os objetivos deste trabalho são: apresentar um novo método para calibrar os parâmetros extrínsecos da câmara, ou seja, calcular os ângulos *pan* e *tilt* e a altura h da câmara; a construção de um mapa virtual que represente o ambiente de trabalho do robô e a obtenção da localização do robô através das coordenadas 2D da imagem após calibrada a câmara. Não se considera a distorção das lentes, por se tratar de calibração extrínseca. O método proposto não utiliza nenhum padrão de calibração e nem marcas de referência no robô. A câmara, a princípio, abrange a área de atuação do robô e está direcionada a ele. Não se tem informações a priori a respeito do eixo óptico da câmara e nem da posição do robô. Na Fig. (1) pode-se visualizar os ângulos *pan* e *tilt*, as suas áreas de abrangência e a altura h da câmara.

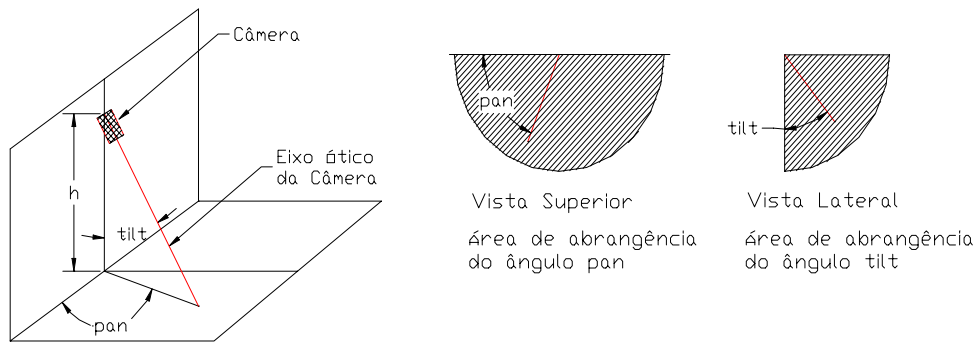


Figura 1. Representação dos ângulos *tilt* e *pan* e a altura *h* da câmera.

Para elaboração de um mapa virtual que represente o ambiente de trabalho são necessárias 4 imagens provenientes dos vértices de um quadrado formado pelo movimento do robô, cujas distâncias são conhecidas. Estas imagens são processadas e, então, são calculadas as posições centrais do robô que servirão de base para a elaboração do mapa virtual.

Tendo-se os parâmetros da câmera calibrados, através do mapa virtual pode-se obter a localização do robô através das coordenadas 2D da imagem. Estas informações sobre a localização do robô são imprescindíveis para a teleoperação do robô pois, o operador do robô utiliza-se delas para decidir a melhor trajetória a ser seguida. As imagens obtidas do robô apresentam um ambiente em 3D, de onde não se pode extrair diretamente, em milímetros, a posição do centro do robô. Em uma imagem 3D a relação entre os pixels e os milímetros não é única e varia ao longo da imagem. A solução apresentada considera estas variações e a correlação entre os pixels e milímetros é determinada através de equações proporcionais.

Para realizar a conversão de pixels para milímetros, construir o mapa virtual do laboratório e encontrar as coordenadas X e Y para qualquer nova posição do robô no ambiente de trabalho, foi necessário capturar quatro imagens provenientes dos vértices de um quadrado formado pelo deslocamento do robô, sendo dados a distância entre estas posições e o ângulo de 90° . A única condição é que o robô se desloque formando um quadrado perfeito (dentro das limitações do robô). São capturadas as posições dos cantos deste quadrado e utilizadas para a calibração da câmera.

3. Fundamentos do Método Proposto

O método de calibração proposto está fundamentado em estudos de perspectiva, apresentados em Doria (1958): Construção de Perspectivas Lineares Exatas de Elementos Geométricos situados sobre o Plano Geometral: Processo dos Pontos de Distância.

Considera-se (Fig. 2) A' a épura do ponto A , onde P é o ponto principal, LT é a linha de terra e LH é a linha do horizonte. A execução da perspectiva linear exata faz-se a partir desses dados e, para encontrar a perspectiva do ponto A , nota-se que: Os pontos de distância D e D' estão afastados de P de uma distância igual ao afastamento do observador ao quadro; As inclinações de 45° em relação ao quadro (ou a LT) tem seu ponto de fuga nos pontos de distância; As retas horizontais perpendiculares ao quadro concorrem para o ponto principal.

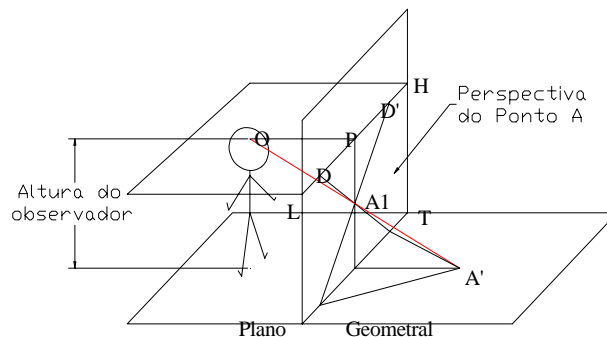


Figura2. Representação da perspectiva do ponto A

A Fig. (3) representa o rebatimento do plano geometral, baixa-se de A' uma perpendicular $A'R$ à LT ; seu ponto de fuga será P e tem-se a perspectiva da reta $A'R$. Com o centro do compasso em R e raio RA' , traça-se o arco $A'B$ que

corresponde a uma reta inclinada de 45° ; a perspectiva da reta $A'B$ será a reta BD . A interseção das perspectivas das retas RP e BD nos dá o ponto $A1$, que é a perspectiva procurada. Variando-se a cota e o afastamento do observador, variará a posição da perspectiva $A1$.

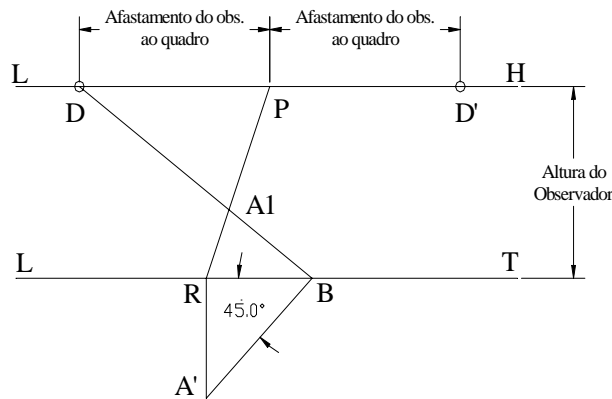


Figura 3. Rebatimento do plano geometral.

A Fig. (4) mostra as 4 posições do robô no ambiente real e como elas aparecem na imagem.

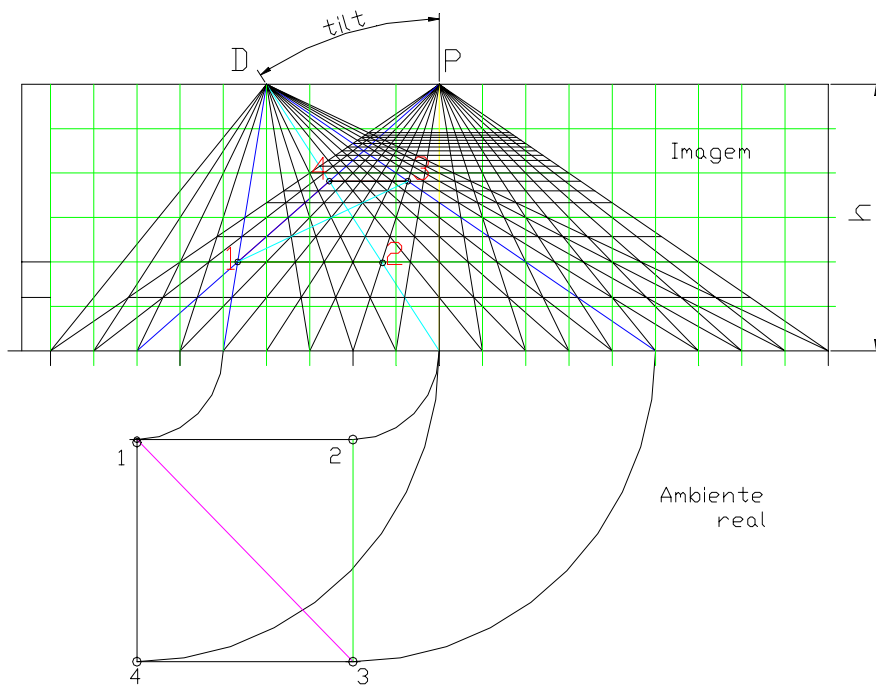


Figura 4. Representação perspectiva linear.

O desenho acima foi elaborado com base nos conceitos de perspectivas lineares, onde um ponto no ambiente real é representado em perspectiva. Neste caso, para a sua construção, é necessário conhecimento dos valores de h e da distância entre P e D . Calculada a posição central do robô nas 4 imagens, tem-se as coordenadas x e y das 4 posições no mapa virtual. Baseado na Fig. (4) foram feitas algumas considerações que estão apresentadas a seguir:

P é o ponto de interseção entre o prolongamento das retas 1-4 e 2-3. Para construção do mapa virtual divide-se a reta 2-1 em n partes. A partir de P traça-se $n-1$ retas passando pelas n divisões (retas verticais);

D é o ponto de interseção da reta 2-4 com a reta que passa por P e é paralela a reta 2-1. Para construção do mapa deve-se traçar as demais retas do ponto D até os pontos das n divisões na reta 2-1 e marcar os pontos de contato (k, b, z) entre as retas (Fig. 5);

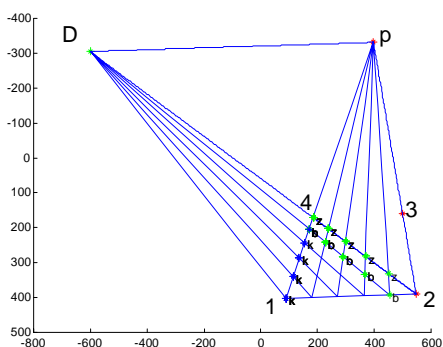


Figura 5. Representação do ponto P e do ponto D.

A distância y do ponto 1 à linha inferior na imagem, representa a distância do ponto 1 à linha inferior no mapa virtual, e esta posição em y está representada abaixo pelo ponto s . A reta AB é paralela a reta $2-1$ e está distante desta $vproj$. $vproj$ é a posição da próxima linha horizontal a partir de s , e seu valor é encontrado extrapolando-se a equação vertical (explicada no próximo item), ver Fig. (6);

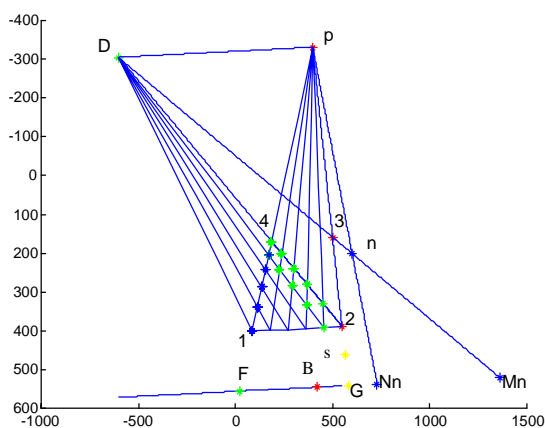


Figura 6. Pontos de contato entre as retas verticais e representação dos pontos Nn e Mn no mapa virtual.

Cálculo das coordenadas verticais (dv , coordenadas y nos pontos z) e das coordenadas horizontais (dx , coordenadas x nos pontos z). Nota-se que estes valores diminuem ao longo da reta 1-4. Com os valores de dv encontra-se a equação vertical e com os valores de dx encontra-se a equação horizontal, estas duas equações regem o comportamento das distâncias entre as linhas horizontais no mapa virtual. A distância da linha inferior à linha que passa por P e D no mapa virtual representa a altura h da câmera; O ângulo formado pela linha 2-4 e a linha P -perpendicular à linha inferior representa o ângulo $tilt$;

A seguir são apresentados os passos para calcular os valores de $Xreal$ e $Yreal$ de uma posição qualquer do robô no ambiente de trabalho em relação a posição 1 do robô:

Calcula-se a posição x e y na imagem da nova posição do robô (representada na Fig. (6) pelo ponto n); F é o ponto de interseção da reta 1-4 com a reta AB . Projetando-se uma reta do ponto P passando pelo ponto n até a linha AB tem-se o ponto Nn , a distância de F a Nn é $dFNn$. G é o ponto de interseção da reta 2-3 com a reta AB . A distância de 1 a 2 é o valor deslocado pelo robô da posição 1 até a posição 2 ($desloc$). A distância de F a G é um valor proporcional a distancia 1-2 (dFG). Se n estiver a direita de F , $Xreal$ será positivo, se estiver a esquerda, $Xreal$ será negativo. Vide Eq. (1).

$$Xreal = \frac{dFN_n \cdot desloc}{dFG} \tag{1}$$

Projetando-se uma reta do ponto D passando pelo ponto n até a linha AB tem-se o ponto Mn . $Base$ é o valor de dFG/n . i é o número de linhas horizontais da reta 2-1 à reta AB . Se n estiver acima da posição 1, então $Yreal$ será positivo, se estiver abaixo, $Yreal$ será negativo. Vide Eq. (2).

$$Y_{real} = \frac{(dMN_n - i.base).desloc}{dFG} \tag{2}$$

Construção do mapa virtual:

- dividir dFG em n partes e extrapolar valores anteriores e posteriores através da equação vertical e da horizontal. Traçar a linha superior do mapa e, também, extrapolar os valores para determinar as posições das linhas verticais, (Fig. 7);

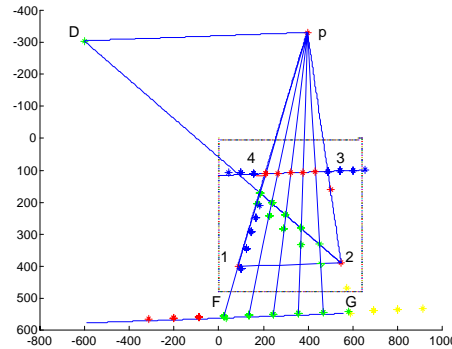


Figura 7. Extrapolação das equações verticais e horizontais

- Traçar as linhas verticais dos pontos superiores aos pontos inferiores e marcar os pontos laterais esquerdos e direitos, (Fig. 8);

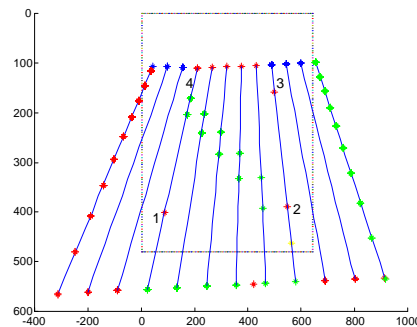


Figura 8. Traçado das linhas verticais

- Finalmente traçar as linhas horizontais e as linhas que representam as paredes. Um exemplo do mapa virtual pode ser visto nos resultados experimentais.
Se a câmera estiver posicionada exatamente no meio do mapa virtual e o seu eixo óptico estiver na mesma direção, então o ângulo pan é zero. O ângulo formado pela lateral esquerda chão-parede e a linha vertical subtraído de 45° é o ângulo pan (Fig. 9).

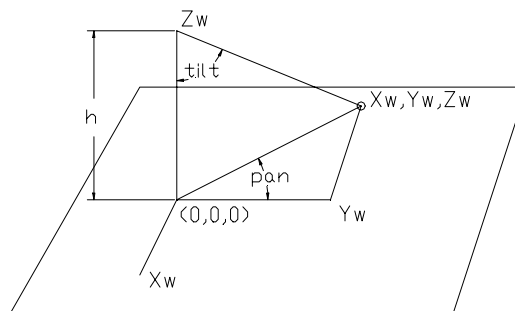


Figura 9. Representação da altura e dos ângulos da câmera.

Com os valores de h , $tilt$ e pan , pode-se calcular as coordenadas da câmera XW (Eq. (3)), YW (Eq. (4)) e ZW (Eq. (5)).

$$Xw = \sin(pan).h.\tan(tilt) \tag{3}$$

$$Yw = \cos(pan).h.\tan(tilt) \tag{4}$$

$$Zw = h \tag{5}$$

4. Resultados Experimentais

Para os testes finais foram capturadas várias imagens do robô no ambiente de trabalho para testar a veracidade do programa desenvolvido e apresentado. Será exposto um caso onde o robô se desloca para 4 posições com os seus respectivos resultados.

A Fig. (10) mostra as imagens processadas, primeiramente foi determinado um limiar para binarização e após filtrado pelo método de dilatação e erosão. Em seguida, foram calculadas as posições centrais e traçadas as retas dos eixos maior e menor.

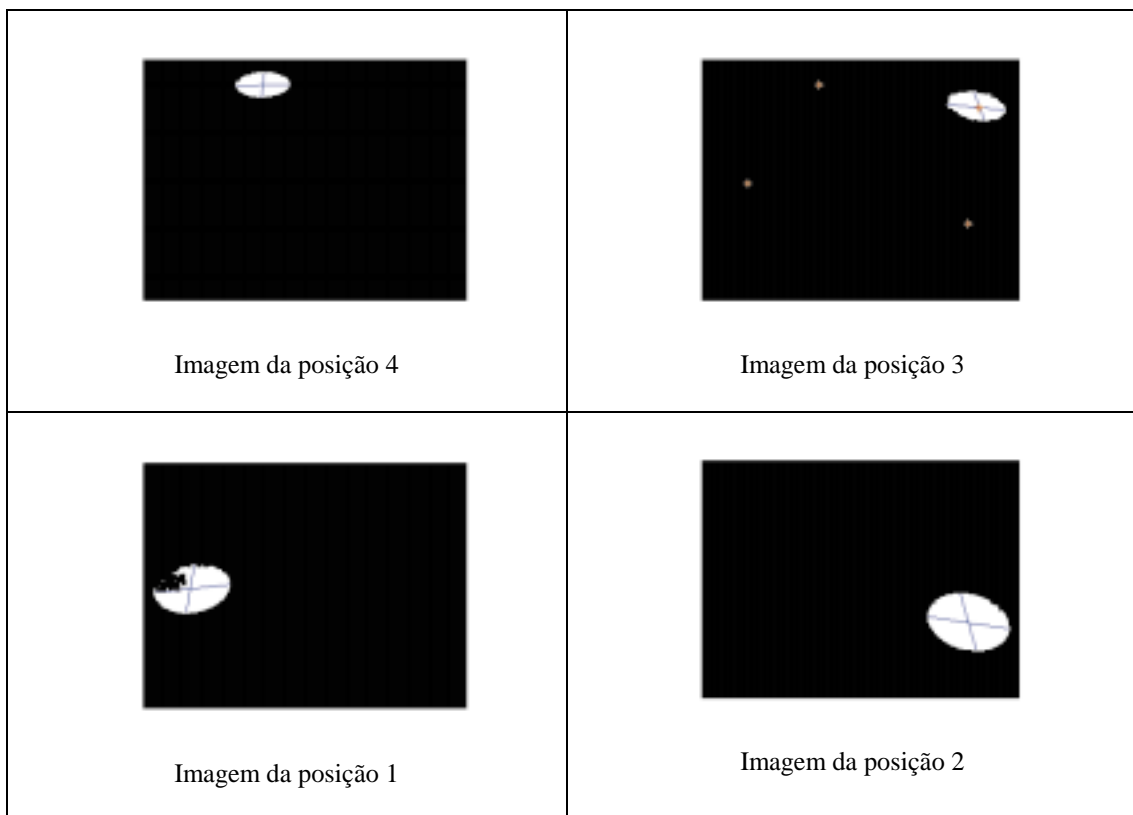


Figura 10. Imagens processadas.

A Fig. (11) representa a fase inicial da construção do mapa virtual, onde pode-se visualizar as quatro posições do robô, os pontos P e D , os pontos de interseção entre as retas verticais e, neste caso, a nova posição do robô após calibrada a câmera coincide com a posição 2 do robô, para as quais estão representados os pontos N e M , que servirão de base para os cálculos de $Xreal$ e $Yreal$.

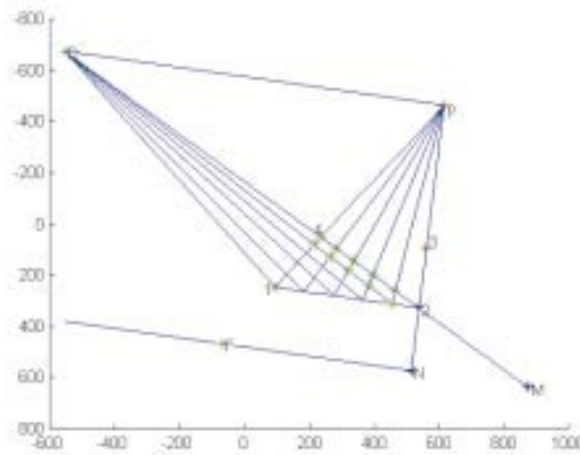


Figura 11. Mapa virtual inicial.

Finalmente, a Fig. (12) mostra o mapa virtual na forma como ele foi calculado, isto é, considerando a inclinação entre as posições 1 e 2 do robô, e do qual se utiliza para realizar todos os cálculos devidos. Na Fig. (12a) não foi considerada a inclinação mencionada, para que se tenha uma melhor visualização do mapa, o objetivo deste é apenas visual. São mostrados nos mapas as 4 posições do robô que serviram de base para a sua elaboração. A partir deste mapa, que representa a calibração da câmera, pode-se calcular a posição X_{real} e Y_{real} para qualquer nova posição do robô no ambiente de trabalho.

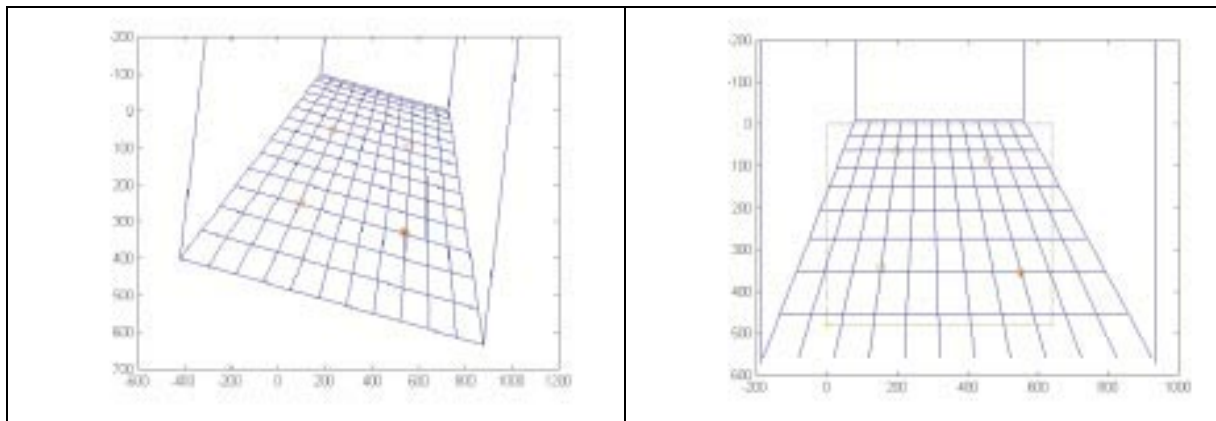


Figura 12. Mapa virtual: a) com inclinação, b) sem inclinação entre as posições 1 e 2.

A Tab. (1) mostra os resultados obtidos em dois processos de calibração da câmera. Após calculados os ângulos de inclinação pan e $tilt$ e a altura h da câmera, foram capturadas 6 novas imagens do robô em diversas posições (para cada processo de calibração) e calculadas as coordenadas do centro do robô nestas imagens (x_n, y_n). Com estas coordenadas foram calculadas as respectivas posições do robô no ambiente de trabalho. Os instrumentos de medição (trena e transferidor) utilizados não apresentavam grande precisão, o que acarreta um certo erro nas medidas reais apresentadas.

Os valores X_{real} e Y_{real} são calculados em relação à posição 1 do robô.

x_n, y_n são as coordenadas da nova posição do robô;

X_{calc}, Y_{calc} são os valores calculados para a nova posição;

X_{med}, Y_{med} são os valores reais medidos pelos instrumentos de medição;

$erroX$ e $erroY$ são os erros percentuais entre os valores calculados e os medidos.

Tabela 1. Resultados obtidos no processo de calibração da câmera

		x_n (mm)	y_n (mm)	X_{calc} (mm)	Y_{calc} (mm)	X_{med} (mm)	Y_{med} (mm)	$ErroX$ (%)	$ErroY$ (%)
1	1	92.8468	247.8999	1.29e-013	16.79	0	0	0	-1.86
	2	538.7178	328.2827	900.00	16.79	900	0	2.5e-014	-1.86
	3	237.3170	51.5051	3.97e-013	916.79	0	900	0	1.86
	4	311.2321	116.0344	274.65	653.77	280	630	-1.90	3.77
	5	406.6699	207.9741	575.73	338.77	580	330	-0.73	2.65
	6	141.8194	37.8671	-258.15	900	-270	900	4.38	-0.34
2	1	123	276	9.607e-015	148.766	0	155	0	4.15
	2	538	272.5	900	148.766	900	160	0	4.15
	3	448	158	900	1022.1	900	980	0	4.29
	4	168	158	9.607e-015	1048.8	0	1000	0	4.88
	5	323.67	211.63	464.86	538.73	450	520	3.3	3.6
	6	84.023	210.28	-170.99	564.48	-170	550	0.58	2.63

A Tab. (2) mostra os parâmetros de calibração obtidos para os dois processos de calibração. Estes valores são comuns a todas as novas posições do robô e só são alterados se a câmera mudar de posição, nesta situação, eles precisam ser recalculados. h é a altura da câmera, pan e $tilt$ são os ângulos da câmera, ccx e ccy são as coordenadas em x e em y da posição da câmera em relação a posição 1 do robô.

Tabela 2. Parâmetros de calibração obtidos

calibração		Valor calculado (mm)	Valor medido (mm)	Erro (%)
1	h (calc) + h(robô)	1068.4+240=1308.4	1315	0.5
	tilt	49.25	47	4.78
	pan	5.1973	5	3.94
	ccx	112.3221	120	6.39
	ccy	1234.9	1190	3.77
2	h (calc) + h(robô)	1475	1490	1.00
	tilt	33.5630	32.47	3.36
	pan	1.9612	1	3.88
	ccx	758.4907	750	1.13
	ccy	2215	2300	3.69

A técnica de calibração proposta apresentou ótimos resultados, com erros na ordem de 3%, como pode ser visto na Tab. (1), e erros médios de 5% na Tab. (2). Abaixo encontram-se as equações Verticais e Horizontais que regem o comportamento das distâncias verticais no mapa virtual para esta situação apresentada.

$$f(X) := 0.013X^4 + 0.051X^3 + 2.087X^2 + 36.331X + 13.023 \quad \text{Equação Vertical}$$

$$f(X) := 0.014X^4 + 0.055X^3 + 2.273X^2 + 39.562X + 195.412 \quad \text{Equação Horizontal}$$

5. Conclusão

Foi proposto um novo método para calibração de câmeras fixas no ambiente de trabalho tendo como objetivos finais: determinar os ângulos de inclinação e a altura da câmera; construir o mapa virtual do robô que forneça uma relação métrica entre os pixels da imagem; determinar as coordenadas de qualquer posição do robô no seu ambiente de trabalho após calibrada a câmera e a orientação e posição da câmera em relação a um sistema de coordenadas. Para realizar a conversão de pixels para milímetros, construir o mapa virtual do laboratório e encontrar as coordenadas X e Y para qualquer nova posição do robô no ambiente de trabalho foi necessário capturar quatro imagens provenientes dos vértices de um quadrado formado pelo deslocamento do robô, sendo dados a distância entre estas posições e o ângulo de 90° . A única condição é que o robô se desloque formando um quadrado perfeito (dentro das limitações do robô). São capturadas as posições dos cantos deste quadrado e utilizadas para a calibração da câmera.

O método proposto está baseado no processo dos pontos de distância das perspectivas lineares exatas de elementos geométricos situados sobre o plano geométrico. Utilizando-se este processo verificou-se que é possível não somente determinar uma relação métrica entre os pixels e os milímetros através da construção do mapa virtual mas, também,

encontrar todos os parâmetros necessários para calibrar a câmera e, após calibrada, descobrir as coordenadas de qualquer posição do robô no ambiente de trabalho. O método depende apenas da posição central do robô nas imagens, por isso pode ser facilmente adaptável a qualquer formato de robô. Algumas das causas de erro nos resultados finais obtidos são: pequena variação nos cálculos da posição central da imagem; o não paralelismo entre os lados do quadrado formado pelos deslocamentos do robô, isto depende das limitações e da precisão do robô em manter esta condição. Os valores experimentais obtidos apresentaram resultados satisfatórios, com erros na ordem de 3%. Como trabalho futuro visa-se a teleoperação do robô móvel com enfoque em realidade virtual.

6. Agradecimentos

Às instituições de fomento CAPES e FAPESP por propiciarem condições financeiras aos pesquisadores para desenvolverem este trabalho e a FINEP, pelo projeto RECOPE por equipamentos utilizados no desenvolvimento deste trabalho. Também, aos professores pelo incentivo e acompanhamento despendidos.

7. Referências

- Chen, Z., Wang, C-M. and Ho, S-Y., 1993, "An Effective Search Approach to Camera Parameter Estimation Using an Arbitrary Planar Calibration Object", *Pattern Recognition*, Vol. 26, No. 5, China, pp. 655-666.
- Doria, C., 1958, "Curso de Perspectivas e Sombras", Curitiba, Brasil.
- Grosky, W. I. and Tamburino, L. A., 1990, "A Unified Approach to the Linear Camera Calibration Problem", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 12, No. 7, pp. 663-671.
- Han, M. and Rhee, S., 1992, "Camera Calibration for Tree-Dimensional Measurement", *Pattern Recognition*, Vol. 25, No. 2, Korea, pp. 155-164.
- Haumette, F. and Rives, P., 1990, "Modélisation et Calibration D'une Caméra", AFCET RFIA.
- Hong, Z. and Yang, J., 1993, "An Algorithm for Camera Calibration using a Three-Dimensional Reference Point", *Pattern Recognition*, Vol. 26, No. 11, China, pp. 1655-1660.
- Kabayama, A. M. and Trabasso, L. G., 1999, "Análise de Desempenho de Técnicas de Visão Computacional", COBEM Congresso de Engenharia Mecânica.
- Lang, S., 1971, "Álgebra Linear", Ed. Edgard Blücher Ltda.
- Ma, S. D., 1993, "Conics-Based Stereo, Motion Estimation and Pose Determination", *IJCV*, 10(1).
- Nomura, Y., Sagara, M., Naruse, H. and Ide, A., 1992, "Simple Calibration Algorithm for High-Distortion-Lens Camera", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 14, No. 11, Japan, pp. 1095-1099.
- Tsai, R. Y., 1986, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision", In *CVPR, USA*, pp. 364-374.
- Vaillant, R., 1990, "Géométrie Différentielle et Vision par Ordinateur: Détection et Reconstruction des Contours d'ocultation de la Surface d'un Objet Non-Polyédrique", PhD thesis, Université Paris-Sud.

METHOD OF CAMERA'S CALIBRATION TO BUILT A VIRTUAL ENVIRONMENT

Silvana A. G. da Silva

Escola Politécnica da Universidade de São Paulo
Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Av. Prof. Mello Moraes, 2231, 05508-900, São Paulo, SP, Brasil
e-mail: silvanaa@usp.br

Jun Okamoto Jr.

Escola Politécnica da Universidade de São Paulo
Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Av. Prof. Mello Moraes, 2231, 05508-900, São Paulo, SP, Brasil
e-mail: jokamoto@usp.br

Abstract. *To control the movement of teleoperated robots, the robot is monitored by computer to accomplish the prior-defined movements and, to have access to the place of the robot's work, video cameras and other sensors are used. This work presents a new method of explicit calibration of fixed video cameras for applications in mobile robots' teleoperation. The method does not use any reference pattern or marks in the robot. To camera's calibration are necessary four images from vertexes of the square formed by the robot's displacement, in which the central coordinates of robot's four positions are calculated. Starting from these coordinates, the virtual map of the work environment is built, which can be obtained a metric relationship among the pixels of the images, the camera's height and angles of inclination. Being known these values, it is possible to determine the coordinates of any robot's position in its work environment. Being used four images, the method is easily adapted for any robot format. The obtained results verify the efficiency of the equations developed for calibration of the camera*

Keywords: Camera calibration, virtual environment, virtual map

SENSOR DE FORÇA TRIDIMENSIONAL COM STRAIN GAGES PARA APLICAÇÕES EM ROBÓTICA

Villaça, Rodolfo da Silva

Universidade Federal do Espírito Santo
Programa de Pós Graduação em Engenharia Elétrica
Av. Fernando Ferrari S/N -
Campus de Goiabeiras - Vitória - ES
rodolfo@tropical.com.br

Amaral, Paulo Faria Santos

Universidade Federal do Espírito Santo
Departamento de Engenharia Elétrica
Av. Fernando Ferrari S/N -
Campus de Goiabeiras - Vitória - ES
automatica@automatica.com.br

Pinto, Benedito Geraldo Miglio

Universidade Federal do Espírito Santo
Departamento de Engenharia Mecânica
Av. Fernando Ferrari S/N -
Campus de Goiabeiras - Vitória - ES
automatica@automatica.com.br

Bento Filho, Antônio

Universidade Federal do Espírito Santo
Departamento de Engenharia Mecânica
Av. Fernando Ferrari S/N -
Campus de Goiabeiras - Vitória - ES
automatica@automatica.com.br

Resumo. Este artigo descreve a implementação de um sensor de força tridimensional utilizando strain gages. O sensor será utilizado para medição das forças de reação nas pernas do robô quadrúpede ROQUE da UFES. O sensor é formado por uma cruzeta plana simétrica, feita de aço com pequena espessura onde são afixados os strain gages e, no centro da cruzeta, é fixada uma haste com a ponta de prova do sensor. A cruzeta permite elevado grau de desacoplamento entre as componentes da força aplicada na ponta de prova, de modo que a obtenção do modelo teórico das relações força deformação e os procedimentos de calibração são simplificados. O sinal de força é obtido através de pontes de Weathstone, sendo que a ligação dos strain gages é feita de modo a maximizar o sinal e anular as deformações axiais. Um circuito de precisão alimenta a ponte e permite obter o sinal de saída amplificado para entrada em um conversor A/D. O sinal obtido é processado em uma placa com um microcontrolador INTEL 80C196KB, conectada serialmente a um PC. Finalmente, são apresentados os resultados da aplicação de forças em condições variadas, permitindo concluir que o sensor é adequado para a medição de forças tridimensionais.

Palavras chave: força, tensionamento, medições

1. Introdução

Um requisito básico na indústria de robótica atual é a capacidade de trabalhar sob quaisquer condições, sejam estas conhecidas ou não. Assim, o sucesso de um projeto de robótica depende, primordialmente, de um bom nível de adaptabilidade do sistema. O robô deve estar equipado por sensores para avaliação de sua posição de trabalho em relação ao ambiente, assim o mesmo poderá tomar decisões acertadas a fim de corrigir as suas ações e torná-las mais precisas. Uma das aplicações típicas para esse sensor é a detecção de forças de reação em patas de robôs quadrúpedes, como o ROQUE, da UFES, a fim de tornar o seu movimento mais preciso e menos dependente das condições do seu ambiente de trabalho.

A utilização de sensores do tipo *strain gages* como transdutores em medições de força tridimensionais foi proposta inicialmente pelo artigo '*A three axis tactile sensor probe for robotic use*' de Havlík (1985), quando o autor propôs a metodologia e aplicou a mesma em soldadores e traçadores de contorno. O trabalho realizado na UFES foi a adaptação dessa metodologia à medição de forças de reação de robôs. As medidas obtidas sob a forma de diferenças de potencial elétrico e então encaminhadas a uma placa de aquisição de dados que os digitalizaria e transmitiria de forma serial a um microcomputador PC. O microcomputador receberia então esses valores e os converteria para unidades de força usando a equação de conversão proposta em Havlík (1985).

O projeto, especificação, testes e conclusões do trabalho realizado serão descritos nas próximas seções deste artigo. A Seção 2 aborda uma rápida introdução aos *strain gages*. Na Seção 3 trata da metodologia de medição de forças usando *strain gages*. A Seção 4 descreve o circuito eletrônico para obtenção de dados e o tratamento dos dados e na Seção 5 são apresentados alguns resultados obtidos em experimentos. Finalmente segue-se na Seção 6 as conclusões finais.

2. Strain Gages

O meio usado na vasta maioria das medições de força atualmente são os *strain gages*. Estes consistem essencialmente em um filme resistivo, de pequena área de seção transversal, montados sob a superfície de um sólido cuja deformação em suas dimensões quando submetido a forças de tensão (tração ou compressão) quer ser medida. Por estar montado junto a superfície do sólido, as deformações sofridas por ele serão as mesmas sofridas pelos *strain gages*. Essa deformação sentida pelo transdutor causa variação em sua resistência elétrica.

As mudanças na resistência de um condutor quando submetido a esforços mecânicos foram inicialmente observadas por Lord Kelvin, em 1856, entretanto a primeira aplicação baseada nesse princípio veio ocorrer somente em 1939 quando houve a construção dos primeiros *strain gages* Hermann (1967). Sua evolução deu-se rapidamente e vários tipos de *strain gages* surgiram a partir dessa data, como por exemplo os *strain gages* de fio metálico, os *strain gages* de folha metálica, os *strain gages* de metal depositado e os *strain gages* baseados em semicondutores.

As variações de resistência dos transdutores *strain gages* geralmente são medidas sob a forma de variação de tensão elétrica, conectando-os em pontes, comumente conhecidas como *Pontes de Weathstone* e aplicando-se uma excitação (tensão elétrica) em uma das extremidades das pontes e monitorando a variação sentida na outra extremidade. Maiores detalhes sobre a construção dessas pontes, formas de medição e esquematização serão dadas nas seções posteriores.

2.1. Relações importantes e unidades de medida

gauge factor (1):

$$k = GF = \frac{\Delta R / R}{\varepsilon} = \frac{\Delta R / R}{\Delta L / L} \quad (1)$$

$k, GF = \text{gauge factor}$

$\Delta R / R = \text{variação na resistência do transdutor}$

$\varepsilon = \text{deformação (strain)}$

$\Delta L / L = \text{variação na dimensão do sólido}$

módulo de elasticidade (2):

$$E = \frac{s}{\varepsilon} = \frac{F / A}{\Delta L / L} \quad (2)$$

$E = \text{módulo de elasticidade [N/m}^2\text{]}$

$s = \text{tensão, pressão [N/m}^2\text{]}$

$\varepsilon = \text{deformação (strain)}$

$F = \text{força [N]}$

$A = \text{área [m}^2\text{]}$

$\Delta L = \text{alongamento na contração [m]}$

$L = \text{comprimento original [m]}$

2.2. Strain gages de superfície metálica

Dentre as diversas opções e tipos de *strain gages* disponíveis, escolheu-se esta pois era a que apresentava mais vantagens para a realização do projeto, sendo sua tecnologia já dominada pelos pesquisadores envolvidos e pela Automática Tecnologia que patrocinou o projeto, e também por ser a que mais se encaixava nas características do que pretendemos medir que é a deformação na superfície dos braços da cruzeta metálica. Uma outra característica dos *strain gages* de superfície metálica é o seu baixo custo e tamanho reduzido. Sua resistência nominal e *gage factor* a 24°C é de $350\Omega \pm 0,15\%$ e 2,1nom respectivamente [2].

3. Usando strain gages na medição de forças tridimensionais

A construção e o método de medição e conversão de um sensor capaz de medir de forma ortogonal as forças sob os braços de uma cruzeta metálica tridimensional, com o mínimo de componentes indesejadas em cada eixo baseia-se no artigo referido em Havlík (1985). Na Figura 1 está um esboço da cruzeta, mostrando também o eixo de referência e uma posição aproximada dos *strain gages* em sua superfície.

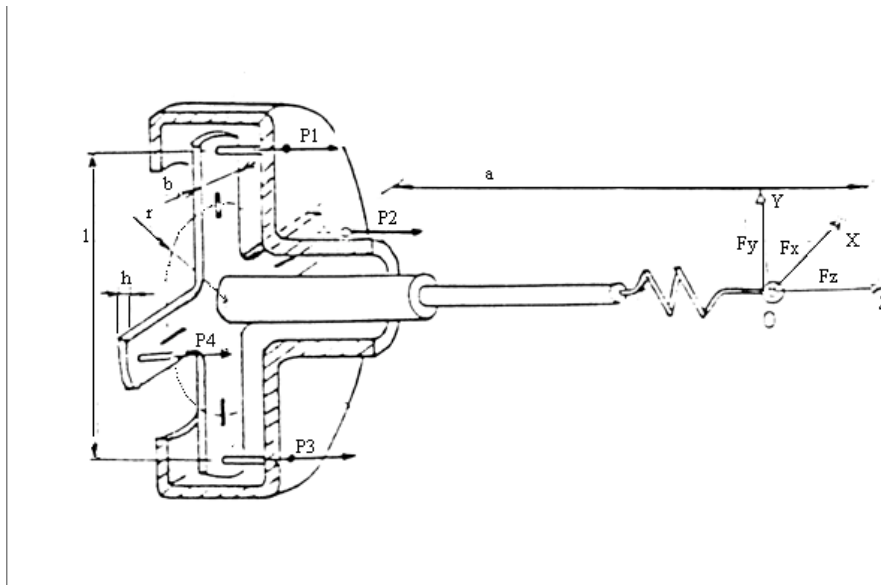


Figura 1. Esboço da cruzeta a ser montada, destacando suas dimensões e referenciais.

Assumindo que sejam lineares as características dos transdutores *strain gages* na faixa de operação desejada pelo sistema, têm-se a equação mostrada a seguir (3). Observe também que se a matriz de transformação M for diagonal, não haverá cruzamento de componentes entre os eixos e assim está obtida a ortogonalidade.

$$\begin{bmatrix} U_X \\ U_Y \\ U_Z \end{bmatrix} = M \cdot d \tag{3}$$

Na equação (3), U_x , U_y e U_z representam o vetor com as tensões de saída obtidas pelos sensores dos três eixos, e d representa o vetor de deslocamento da ponta da haste da cruzeta, partindo de sua posição inicial, também medido nos três eixos, ou seja, pode-se dizer que há uma proporcionalidade entre a variação de tensão medida na saída das pontes de *strain gages* e o deslocamento sofrido pela haste da cruzeta quando submetida a uma força em sua ponta.

Baseado no cálculo dos momentos das forças aplicadas nos quatro pontos de fixação da cruzeta, $P1$, $P2$, $P3$, $P4$ obtemos a seguinte relação descrita em (4):

$$\begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \end{bmatrix} = \begin{bmatrix} 0 & \frac{a}{2l} & \frac{-1}{4} \\ \frac{a}{2l} & 0 & \frac{-1}{4} \\ 0 & \frac{-a}{2l} & \frac{-1}{4} \\ \frac{-a}{2l} & 0 & \frac{-1}{4} \end{bmatrix} * \begin{bmatrix} Fx \\ Fy \\ Fz \end{bmatrix} \tag{4}$$

De acordo o esquema proposto, os *strain gages* são colados na superfície da cruzeta obedecendo um círculo de diâmetro $2r$, seguindo orientação radial. Agora deduz-se uma relação para as tensões radiais sofridas pelos *strain gages* nos quatro braços da cruzeta. A equação (5) ficaria assim:

$$\begin{bmatrix} \epsilon_{11} \\ \epsilon_{12} \\ \epsilon_{23} \\ \epsilon_{24} \\ \epsilon_{35} \\ \epsilon_{36} \\ \epsilon_{47} \\ \epsilon_{48} \end{bmatrix} = \begin{bmatrix} \cdot c & 0 & 0 & 0 \\ c & 0 & 0 & 0 \\ 0 & c & 0 & 0 \\ 0 & \cdot c & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & \cdot c & 0 \\ 0 & 0 & 0 & c \\ 0 & 0 & 0 & \cdot c \end{bmatrix} * \begin{bmatrix} P1 \\ P2 \\ P3 \\ P4 \end{bmatrix} \tag{5}$$

Nesta equação, c é o coeficiente de força, tração e/ou compressão, para uma superfície que apresenta um lado livre de carga e o outro fixo, e ϵ_{ij} representa a tração e/ou compressão em uma fibra radial da superfície. O índice i representa o número da força P_i e j representa o número de seqüência no vetor ϵ , sendo ímpar para o lado esquerdo e par para o lado direito da cruz. No trabalho proposto o esquema, onde a seção da cruzeta é retangular, o valor de c pode ser obtido através da seguinte relação (6):

$$c = \frac{6 \cdot (l - r)}{Ebh^2} \tag{6}$$

Na equação anterior, b , l , r e h são parâmetros geométricos da Fig. 2, E é o *Módulo de Elasticidade de Young* para o material utilizado (l é o comprimento do braço da lâmina, e r a distância do centro do *strain gage* à base da lâmina (superfície)).

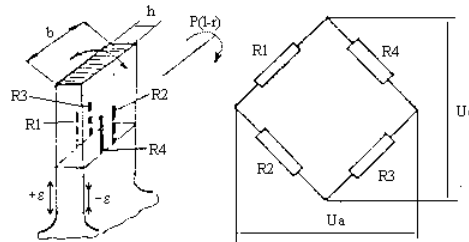


Figura 2. Flexão aplicada a uma viga engastada somente em um de seus lados e arranjo dos *strain gages* na viga e na Ponte de Weathstone.

No esquema acima, apresentado na Fig. 2, que representa uma forma simplificada de aplicação de uma única carga de flexão em uma viga de seção retangular apresentando uma extremidade livre e uma *Ponte de Weathstone* completa formada por *strain gages*, pode-se deduzir também a seguinte equação (7):

$$U_A = a_0 + \frac{1}{4} \left[\frac{\Delta R_1}{R_1} - \frac{\Delta R_2}{R_2} + \frac{\Delta R_3}{R_3} - \frac{\Delta R_4}{R_4} \right] * U_E \tag{7}$$

fazendo $U_E = 1$, tem-se a equação (8), simplificação de (7).

$$U_A = a_0 + \frac{k}{4} \cdot \epsilon \cdot E \tag{8}$$

a_0 representa o valor de tensão medido nos terminais da ponte quando esta estiver sem carga, idealmente zero.

Reescrevendo (8):

$$U_A = a_0 + \frac{k}{4} E \cdot C \cdot H \cdot F \tag{9}$$

Nesta equação, E é um vetor de controle, escolhido a fim de tornar a matriz M_F descrita em (10) ortogonal.

$$M_F = \frac{k}{4} E \cdot C \cdot H \tag{10}$$

Através de uma análise matemática do vetor E a fim de satisfazer as necessidades de projeto, podemos deduzir E a seguir:

$$E = \begin{bmatrix} 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 \\ 1 & -1 & 0 & 0 & -1 & 1 & 0 & 0 \\ -1 & 1 & 0 & 0 & -1 & 1 & 0 & 0 \end{bmatrix}$$

Analisando o vetor acima e posicionando-se os elementos dos seus termos não nulos em *Pontes de Weathstone* obtêm-se o seguinte esquema para obtenção de medidas (Fig. 3), nos três eixos, obedecendo a ortogonalidade de seus resultados:

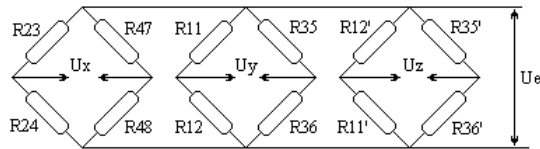


Figura 3. Pontes de Weathstone para medição ortogonal de esforços na cruzeta.

Da Figura 3, de acordo com a equação (7) pode-se escrever então (12):

$$\begin{bmatrix} U_X \\ U_Y \\ U_Z \end{bmatrix} = \begin{bmatrix} a_x \\ a_Y \\ a_Z \end{bmatrix} + k \cdot \begin{bmatrix} \frac{a.c}{2l} & 0 & 0 \\ 0 & \frac{a.c}{2l} & 0 \\ 0 & 0 & \frac{c}{4} \end{bmatrix} \cdot \begin{bmatrix} F_X \\ F_Y \\ F_Z \end{bmatrix} \tag{12}$$

onde l é o comprimento da cruzeta, conforme a Figura 1.

Esta então é a relação utilizada nesse trabalho para fazer a conversão do valor obtido nos sensores em forças nas três direções ortogonais. O arranjo final dos *strain gages* na superfície da cruzeta ficaria então assim como mostrado na Fig. 4.

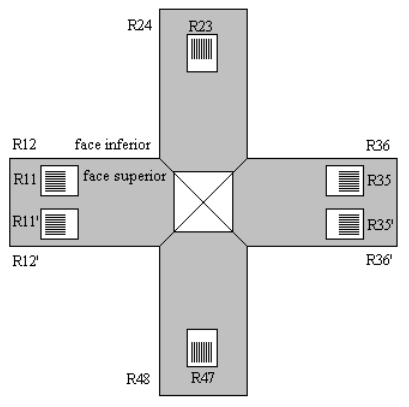


Figura 4. Posicionamento dos *strain gages* na cruzeta metálica.

4. O circuito de aquisição de dados.

A família de microcontroladores 80C196KB pertencem à uma parte CHMOS da família MCS®96 da Intel. Os CHMOS compartilham da mesma arquitetura interna e conjunto de instruções com os outros membros da família MCS®96, entretanto incluem alguns avanços a fim de aumentar sua performance e consumir menos energia.

Os membros da família de microcontroladores Intel MCS®96 possuem CPU de 16 bits e no mínimo 230 bytes de memória RAM incorporada Intel (1990). São máquinas que operam registrador a registrador, sem necessidade de acumuladores, de forma que a maioria das operações podem ser rapidamente efetuadas utilizando-se quaisquer dos registradores, podendo inclusive obter acesso aos dispositivos periféricos presentes no chip. Dentre esses dispositivos têm-se portas seriais, conversores A/D, saídas PWM, 48 linhas de entrada/saída e um sub-sistema de entrada de saída de alta velocidade.

Aplicações típicas para os produtos da família MCS-96 seriam controladores em malha fechada, processamento digital de sinais. São utilizados em produtos tais como modem, controladores de motores, impressoras, fotocopiadoras, disk drivers, instrumentação médica em geral, etc.

A placa de aquisição de dados foi construída tomando-se como base a placa originalmente desenvolvida para o sensor TOKTELA® produto da Automática Tecnologia. Devido às semelhanças entre os dois circuitos a adaptação da placa seria mais prático, fácil e barato do que o desenvolvimento e confecção de uma nova placa voltada especialmente para o uso no circuito.

4.1. Diagrama de blocos.

A placa de aquisição de dados é composta das seguinte etapas: aquisição dos dados, amplificação, amostragem, conversão A/D, serialização, conversão RS-232 e comunicação com o PC. A seguir, na Fig. 5 tem-se o diagrama de blocos simplificado.

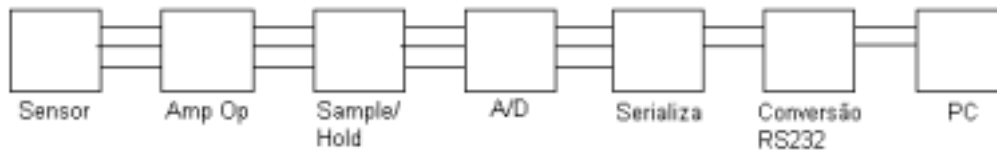


Figura 5. Diagrama de blocos simplificado.

- Amplificador de sinais de entrada: A entrada dos dados é feita através de três pares de linhas, provenientes das três pontes e *strain gages* montadas na cruzeta Para amplificar os sinais provenientes das pontes de *strain gages* utiliza-se amplificadores operacionais de alto ganho, que proporcionam ganhos absolutos em torno de 1000. Esse ganho pode ser precisamente ajustado com auxílio de resistores variáveis.
- Amostragem e retenção (*sample and hold*): A próxima etapa do circuito é onde o sinal lido dos sensores e amplificado é amostrado e retido (*sample and hold*) em intervalos regulares, para então ser encaminhado à conversão A/D. Cabe aqui ressaltar a importância de se ter uma etapa de amostragem externa, que permite a leitura simultânea dos três sinais, ao passo que se utilizássemos o amostrador interno ao 80C196KB teríamos que efetuar as leituras seqüencialmente, resultando em possíveis erros caso os valores das componentes de força variassem com frequência alta.
- A etapa de conversão analógico-digital: O circuito responsável por essa etapa é o microcontrolador 80C196KB, que possui internamente um módulo conversor A/D. A conversão é controlada via software interno programado no microcontrolador.
- A etapa de serialização. Assim como na conversão analógico-digital, a etapa de serialização é realizada por um módulo interno ao microcontrolador utilizado no circuito, o 80C196KB preparando a leitura para ser enviada via transmissão serial para as próximas etapas do circuito.
- A conversão RS-232: A fim de compatibilizar os níveis de tensão obtidos na saída do serializador interno ao 80C196KB com o padrão RS-232, utilizado para a comunicação de dados entre PC's e dispositivos periféricos (o circuito para medição de forças 3D), a próxima etapa do circuito é a conversão RS-232 No PC utiliza-se uma porta serial COM1 ou COM2, à escolha do usuário.
- PC: Responsável pela exibição e manipulação matemática das medidas.

4.2. Programação

No projeto a programação do microcontrolador será feita usando-se a linguagem ANSI C, a qual possui um compilador capaz de gerar código compatível com o 80C196KB.

Em linhas gerais, o microprocessador deverá estar preparado para ler os dados recolhidos pelo sensor, realizar a conversão analógico-digital, serializar os dados obtidos e transmiti-los. O programa então deverá ser responsável por orientar o funcionamento do microprocessador em cada uma dessas etapas. A seguir, na Fig. 6 tem-se um fluxograma de funcionamento do mesmo.

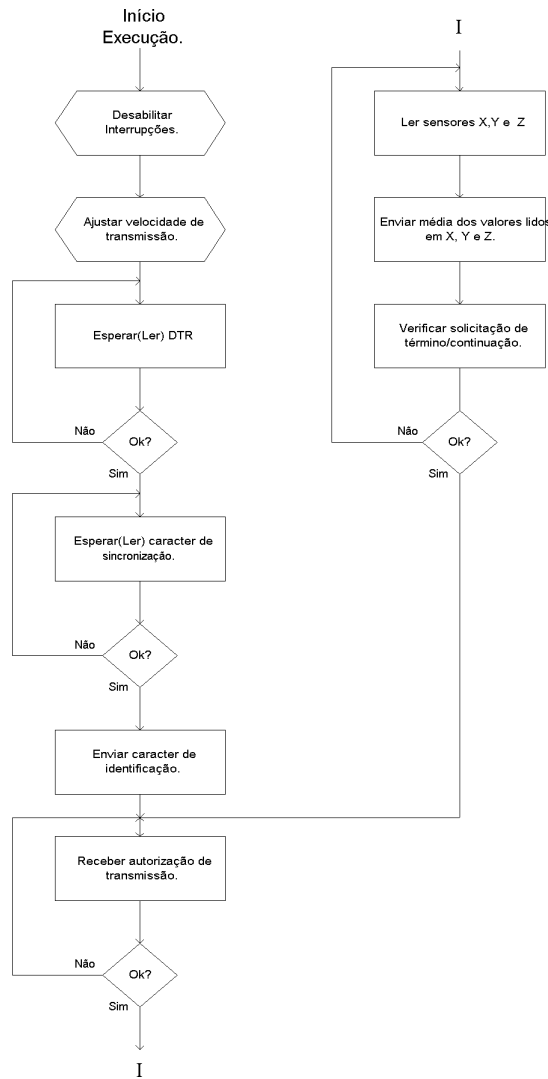


Figura 6. Fluxograma do programa residente no 80C196KB.

5. Testes e Calibração

A calibração seguiu a seguinte seqüência de passos:

- Ajuste dos potenciômetros de offset;
- Ajuste dos potenciômetros de equilíbrio das pontes. Um desbalanceamento ocorre em consequência do não equilíbrio perfeito das pontes montadas. Esse desbalanceamento ocorre por diversos fatores, como por exemplo a colagem não perfeita, imperfeições na superfície da lâmina, margem de erro na resistência dos *strain gages*, etc. Os potenciômetros são ajustados de forma a obter equilíbrio total da ponte quando sem carga em sua extremidade.
- Ajuste de escala.

Essa etapa foi, sem dúvida, uma das mais importantes de todo projeto, onde seria decidido o sucesso ou fracasso do mesmo.

Passadas as etapas de calibração, partiu-se para ensaios, com levantamento de curvas Força x Saída do conversor A/D, a fim de verificar a linearidade do mesmo e o fator real de conversão entre tensão e força, comparando-o com o obtido matematicamente.

Nos ensaios, pesos foram adicionados ao sistema gradativamente nas 3 direções e observando-se a resposta do sistema, em cada eixo, para cada conjunto de peso adicionado produziu-se uma tabela. De posse dessas tabelas foram levantadas as curvas de resposta do sistema comparados com o valor real dos pesos medidos, obtendo-se aí o valor dos erros de medição e o que chamamos de matriz real do conversão do sistema.

Na figura a seguir (Fig. 7) é mostrado, como exemplo, o resultado da medição de pesos no eixo X e as variações presentes nos demais eixos devido a não ortogonalidade ideal do sistema.

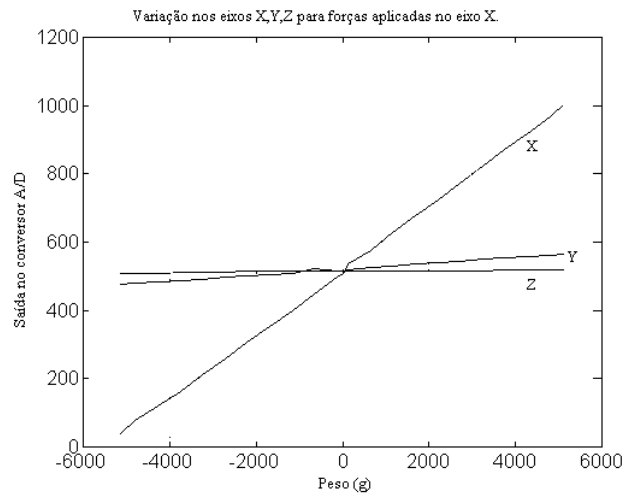


Figura 7. Gráfico de resposta (saída digital do conversor A/D) para os eixos com força aplicada em X.

Curvas semelhantes à da Fig. 7 foram levantadas para os eixos Y e Z, variando-se o ponto de aplicação das forças entre extremidades e meio da haste. Na Fig., a seguir, há uma foto dos ensaios e da cruzeta construída.

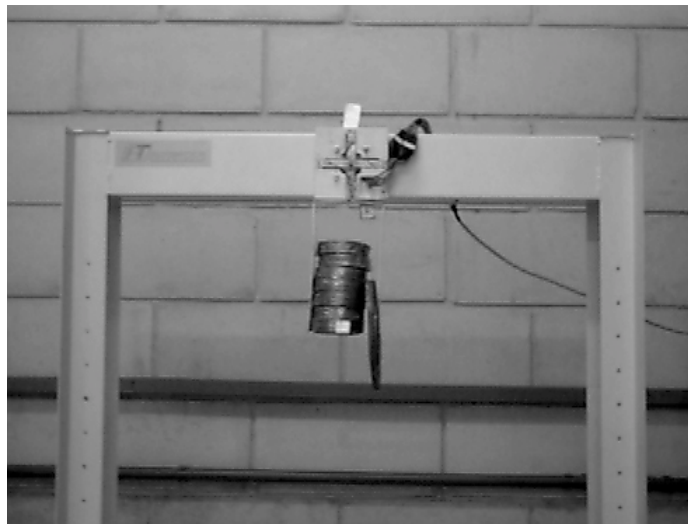


Figura 8. Medição de força nas direções X e Y e a cruzeta.

A observação do gráfico e os resultados obtidos permite a conclusão de que há uma relação linear entre força e tensão enquanto se está respeitando os limites plásticos e de saturação do material. Os gráficos também permitem a conclusão de que não foi obtida a ortogonalidade total entre os eixos. Estimou-se o erro máximo através de manipulação matemática das curvas como sendo inferior a 10% do valor real, obedecendo os limites de saturação do material.

6. Conclusão.

Em vista do que foi apresentado, pode-se concluir que o trabalho obteve o sucesso esperado, confirmando a teoria de que é possível montar-se um sensor de força tridimensional usando-se *strain gages* como transdutor. Tal sensor estaria apto a medidas qualitativas e quantitativas com boa precisão.. Dessa forma as aplicações em que o trabalho estaria afim seriam as aplicações para determinar a direção e sentido, com margem de erro pequena, de uma força aplicada ao sensor. A robótica permanece entre uma das aplicações mais afins deste projeto.

Pode-se afirmar, com segurança muito grande, que uma montagem mais cuidadosa do sensor resultará em resultados ainda mais satisfatórios. Uma montagem mais cuidadosa, cabe aqui explicar, seria aquela em que o processo de colagem dos *strain gages* fosse feito de tal forma evitar completamente a formação de pequenas bolhas de ar entre o seu filme resistivo e a superfície da cruzeta e cuidados com as dimensões e o material da cruzeta de forma que ela seja perfeitamente simétrica e sem nenhuma superfície curvada, ainda que minimamente. Tal montagem levaria a erros menores, menos dificuldade na calibração e leituras mais precisas.

7. Referências.

- Havlík, Stefan. A three axis tactile sensor probe for robotic use. Tchecoslováquia, 1985. [1]
Philips. Guide to strain gages. Holanda. [2]
Hermann K., P. Neubert. Strain Gages: Kinds and Uses. London: Macmillan,1967. Estados Unidos. [3]
Intel Corporation. 80C196KB User's Guide. Estados Unidos, 1990. [4]

THREE-AXIS FORCE SENSOR WITH STRAIN GAGES FOR ROBOTICS APPLICATIONS

Villaça, Rodolfo da Silva

Federal University of Espírito Santo
The Department of Electrical Engineering Postgraduate Program
Fernando Ferrari Av.
Campus de Goiabeiras - Vitória - ES
Brazil
rodolfo@tropical.com.br

Amaral, Paulo Faria Santos

Federal University of Espírito Santo
The Department of Electrical Engineering
Fernando Ferrari Av.
Campus de Goiabeiras - Vitória - ES
Brazil
automatica@automatica.com.br

Pinto, Bendito Geraldo Miglio

The Department of Mechanical Engineering
Fernando Ferrari Av.
Campus de Goiabeiras - Vitória - ES
Brazil
automatica@automatica.com.br

Bento Filho, Antônio

The Department of Electrical Engineering
Fernando Ferrari Av.
Campus de Goiabeiras - Vitória - ES
Brazil
automatica@automatica.com.br

Abstract. *This paper describes the implementation of a three-axis force sensor using strain gages. The sensor will be used for the acquisition of the reaction forces in the legs of the quadruped robot ROQUE of UFES. The sensor is a symmetrical and flat crosspiece, made of steel with small thickness with the ends fixed. The strain gages are bonded at the surface of the crosspiece in points of maximum strain. The sensor probe is a rod connected perpendicularly to the center of the crosspiece. The crosspiece allows high degree of uncoupling between the applied force components, simplifying the attainment of the theoretical force strain relations model and the calibration procedures. The force signal is gotten through Weathstone bridges. The connection of strain gages to the bridge was made to maximize the signal and to extinguish the axial strain from the measurements. An accuracy circuit feeds the bridge and allows to get the output signal amplified for input in an A/D converter. The signal is then processed in a INTEL 80C196KB micro controller board connected serially to a PC. The results of the application of forces in various conditions are presented, showing satisfactory accuracy, repeatability and precision for the measurement of three-dimensional forces.*

Keywords. *force, strain, measurement*

DESENVOLVIMENTO DE UM SISTEMA DE VISÃO OMNIDIRECIONAL

Valdir Grassi Junior

Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231
05508-900 São Paulo, SP
vgrassi@usp.br

Cláudia C. G. Decco

Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231
05508-900 São Paulo, SP
ccgdecco@usp.br

Jun Okamoto Junior

Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231
05508-900 São Paulo, SP
jokamoto@usp.br

Arthur José Vieira Porto

Núcleo de Manufatura Avançada
Escola de Engenharia de São Carlos da Universidade de São Paulo
Av. Trabalhador São-carlense, 400 CP. 359
13560-970 São Carlos, SP
ajvporto@sc.usp.br

Resumo. *Sistemas de visão omnidirecional produzem imagens de 360° do ambiente podendo ser utilizados em navegação, tele-operação e controle servo visual de robôs. Este tipo de sistema dispensa o movimento da câmera para determinada direção de atenção mas requer processamento não convencional da imagem, uma vez que a imagem adquirida se encontra mapeada em coordenadas polares não lineares. Uma maneira efetiva de se obter uma imagem em um sistema omnidirecional é com o uso combinado de lentes e espelhos. Várias formas de espelhos convexos podem ser utilizadas montando-se uma câmera com o seu eixo óptico alinhado com o centro do espelho. Dentre as formas usadas, tem-se os cônicos, parabólicos, hiperbólicos e esféricos. Foram implementados dois tipos de espelhos para o sistema: um hiperbólico e um esférico, sendo que o espelho hiperbólico foi feito usando-se usinagem de ultra-precisão. Também foi desenvolvido um programa para transformar a imagem omnidirecional adquirida pelo sistema em uma imagem plana, obtendo-se imagens panorâmicas do ambiente. Este trabalho mostra o sistema de visão omnidirecional desenvolvido, apresentando os cálculos para determinação de uma forma adequada de espelho, as soluções mecânicas adotadas para a construção do sistema, e os resultados do programa desenvolvido.*

Palavras chave: *visão omnidirecional, processamento digital de imagens, usinagem de ultra-precisão, sistemas catadióptricos*

1. Introdução

Na área de robótica móvel, a utilização de sistemas de visão pode permitir a realização de certos tipos de aplicações com alto grau de autonomia, tais como, navegação, controle servo visual e tele-operação. Para estas aplicações, normalmente deseja-se obter, através do sistema de visão, imagens com a maior quantidade de informação possível do ambiente no qual o robô está inserido para melhorar sua base de dados para tomada de decisões. No intuito de desenvolver e aplicar sistemas de visão omnidirecional, isto é, que possuem campo de visão de 360°, vários trabalhos foram realizados por pesquisadores nas áreas de visão computacional, óptica aplicada e robótica. Yagi (1999) apresenta uma revisão bibliográfica voltada para aplicações de diversos tipos de sistemas de visão omnidirecional.

Uma maneira de se obter uma imagem com campo de visão de 360° é através do uso de múltiplas câmeras apontadas cada uma para direções diferentes, ou até mesmo através da rotação de uma câmera em torno de um eixo vertical e posterior composição das imagens obtidas. Ambas as soluções são maneiras de se obter uma imagem com alta resolução, mas requerem um processamento para compor as imagens obtidas em uma imagem panorâmica final. Além disso, o uso do primeiro método pode resultar em um sistema de custo elevado, devido ao uso de múltiplas câmeras, com montagem de difícil calibração, e um tamanho não muito compacto. Já com o segundo método, a imagem final seria obtida somente após a captura de vários quadros até que a câmera complete uma volta, o que tornaria inviável o seu uso em aplicações dinâmicas que requerem uma rápida taxa de atualização da imagem omnidirecional obtida.

Outra maneira de se construir um sistema omnidirecional é através do uso de uma lente do tipo olho de peixe em uma única câmera fixa. Dessa forma é possível adquirir em tempo real uma imagem com um campo de visão de 360°

abrangendo um hemisfério completo. Cao et al (1985) usa este tipo de sistema omnidirecional para controlar um robô móvel. Entretanto, estes tipos de lente costumam ser caras enquanto que apresentam boa resolução somente no centro da imagem. Na região periférica da imagem, onde na maioria das vezes se encontram os objetos de interesse da cena, a resolução é baixa.

Uma maneira efetiva de se obter uma imagem omnidirecional é através do uso combinado de lentes e espelhos convexos. Este tipo de sistema omnidirecional recebe o nome de sistema catadióptrico, e assim como o sistema omnidirecional que utiliza lente olho de peixe, com este tipo de sistema é possível adquirir imagens omnidirecionais em tempo real sem ter a necessidade de se movimentar a câmera ou o espelho. Através deste tipo de sistema pode-se obter, dependendo do tipo de espelho utilizado, um campo de visão até mesmo mais abrangente que o obtido pelos outros sistemas citados até aqui. Através deste tipo de solução pode-se obter imagens cujas propriedades dependem da forma da superfície do espelho, e também pode-se obter sistemas de visão de tamanho adequado para aplicações em robótica.

Este artigo descreve o sistema de visão omnidirecional catadióptrico que foi desenvolvido pelo Grupo de Percepção Avançada (GPA) do Laboratório de Automação e Sistemas (LAS) do Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos da EPUSP. Este sistema foi desenvolvido para ser usado em um robô móvel para aplicações de tele-operação e controle servo visual. Na seção 2 descreve-se algumas das possíveis formas de espelho que podem ser utilizadas em sistemas de visão omnidirecional catadióptrico, e algumas propriedades conseguidas com estes espelhos. Na seção 3 apresenta-se as formas de espelho escolhidas, na seção 4 o projeto e a fabricação dos espelhos e das peças que compõe o sistema omnidirecional. A seção 5 apresenta o programa desenvolvido para tratamento da imagem omnidirecional adquirida de forma a facilitar a sua visualização pelo usuário humano. Finalmente, na seção 6 é apresentada uma conclusão deste artigo.

2. Sistemas de visão omnidirecional catadióptricos

Tipicamente, um sistema de visão omnidirecional catadióptrico consiste em se utilizar um espelho convexo fixo em frente à uma câmera que também permanece fixa com sua lente voltada para cima. Várias formas diferentes de espelho podem ser usadas, como também dois tipos diferentes de modelos de câmera podem ser adotados: câmeras de projeção perspectiva ou de projeção ortográfica. As propriedades das imagens obtidas pelo sistema de visão dependem do espelho e do modelo de câmera utilizado.

Baker e Nayar (1998) descrevem um conjunto de sistemas catadióptricos que possuem a propriedade de ter um centro único de projeção. Este centro único de projeção torna capaz a geração, a partir da imagem capturada pelo sistema de visão, de qualquer imagem projetada em qualquer plano definido pelo usuário. Assim é possível a geração de imagens perspectivas ou panorâmicas livres de distorções. Em particular Nayar (1997) e Peri (1997), desenvolvem um sistema de visão que usa uma câmera de projeção ortográfica e um espelho parabólico, e desenvolvem também um programa capaz de simular uma câmera de projeção perspectiva virtual onde a direção de visão, o campo de visão e a magnificação da câmera podem ser ajustadas pelo usuário. Neste programa, a imagem perspectiva mostrada ao usuário como sendo proveniente da câmera virtual é calculada em tempo real a partir da imagem adquirida pelo sistema de visão omnidirecional com centro único de projeção. Chang e Hebert (1998) utilizam o sistema omnidirecional desenvolvido por Nayar em uma aplicação de controle servo visual de um robô móvel, evitando assim o uso de um sistema de visão composto de uma câmera que se movimenta através de um mecanismo de *pan/tilt* para apontar sempre na direção do alvo de interesse.

Svododa (1997) também desenvolve um sistema catadióptrico com centro único de projeção. Este sistema é composto por uma câmera de projeção perspectiva e um espelho hiperbólico e também pode ser usado para obtenção de projeções perspectivas e panorâmicas livres de distorção. Svododa também mostra as curvas epipolares para este tipo de sistema, possibilitando o uso das imagens adquiridas por esse sistema para cálculo de distâncias baseado em visão stereo.

Além dos sistemas catadióptricos com centro único de projeção, formados basicamente por espelhos hiperbólicos e parabólicos, outros sistemas podem ser montados utilizando-se espelhos cônicos, esféricos, e outros espelhos projetados para fornecerem imagens com propriedades específicas.

Yagi et al (1994) utiliza um espelho cônico no sistema de visão omnidirecional de um robô móvel usado para navegação e construção de mapas do ambiente. Neste sistema de visão, as linhas verticais de objetos do ambiente, tais como portas, mesas e armários, são mapeadas como linhas radiais na imagem capturada. Através da detecção destas linhas radiais, é possível o mapeamento do ambiente e até mesmo o desvio de obstáculos por parte do robô.

Em um trabalho de Chahl e Srinivasan (1997), eles descrevem uma família de superfícies que possuem a propriedade de preservar a relação linear entre o ângulo de incidência da luz na superfície do espelho e o ângulo de reflexão no dispositivo de aquisição da imagem, como acontece em um espelho plano. Dessa forma, a elevação e o ângulo em coordenadas do mundo são mapeadas linearmente na imagem polar omnidirecional como sendo raio e ângulo. Um trabalho semelhante foi feito por Hicks e Bajcsy (1999), onde eles desenvolvem uma equação que descreve uma família de superfícies reflexivas que possuem um campo de visão abrangente e que possui a propriedade de preservar a geometria de um plano perpendicular ao eixo de simetria da superfície do espelho.

A maioria dos sistemas de visão omnidirecional catadióptricos apresentados aqui requerem um processamento muitas vezes não convencional da imagem capturada, por esta estar mapeada em um sistema de coordenadas polares não linear. No entanto, ao se usar um sistema de visão omnidirecional catadióptrico em um robô móvel, o ambiente inteiro pode ser visualizado em um único instante de tempo em uma única imagem, o que dispensa o movimento de

uma câmera em direção ao objeto de interesse. Em geral, é melhor evitar o controle de movimento de câmera pois este controle muitas vezes possui limitações de velocidade e de acuracidade devido a problemas de calibração. Por alguns destes motivos, escolheu-se implementar um sistema de visão omnidirecional para o robô móvel que será utilizado para aplicações de navegação, controle servo visual e tele-operação no GPA do Departamento de Engenharia Mecatrônica da EPUSP.

3. Solução escolhida

Dentre as formas de espelho possíveis de se utilizar em um sistema de visão omnidirecional catadióptrico, decidiu-se por construir um espelho esférico, e um espelho hiperbólico semelhante ao utilizado por Svoboda (1997).

O espelho esférico foi escolhido por sua facilidade de fabricação. Embora não apresente nenhuma propriedade especial, um sistema omnidirecional que utiliza este tipo de espelho é relativamente fácil de ser construído, e também pode ser usado para navegação robótica e tele-operação.

O espelho hiperbólico foi escolhido pois utilizando-se este tipo de espelho com uma câmera que possui projeção perspectiva, pode-se obter um sistema omnidirecional com centro único de projeção, permitindo a criação de imagens panorâmicas ou perspectivas a partir da imagem capturada pelo sistema. Essa característica é importante pois, além das imagens panorâmicas e perspectivas serem mais facilmente compreendidas por uma pessoa, pode-se fazer uso dos muitos algoritmos bem conhecidos de processamento de imagem que foram desenvolvidos para trabalharem com imagens com projeção perspectiva.

O espelho hiperbólico foi projetado de forma a se obter um sistema de visão omnidirecional compacto o suficiente para ser montado sobre o robô móvel. Para projeto deste espelho, seguiu-se a metodologia apresentada por Svoboda (1997).

3.1 Formulação para especificação do espelho hiperbólico

Um espelho hiperbólico pode ser definido no sistema de coordenadas centrado no ponto focal F' , como mostrado na Fig. (1), pela Eq. (1) em função dos parâmetros do espelho a e b . O valor máximo do ângulo de visão α do espelho pode ser dado pela Eq. (2).

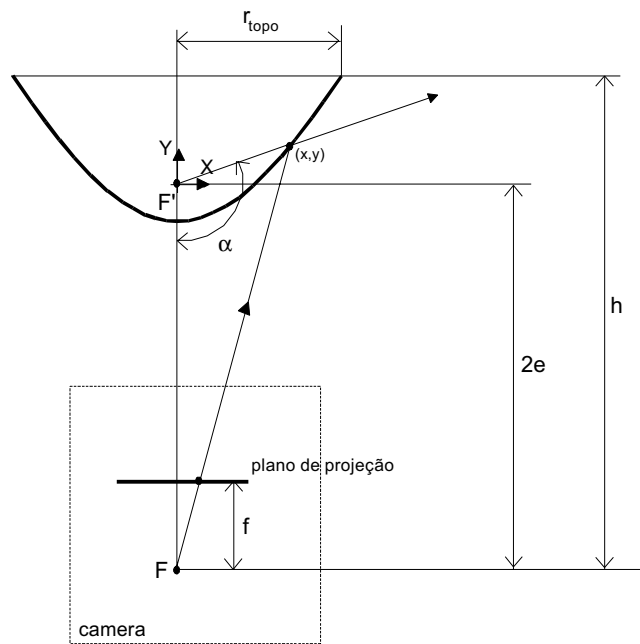


Figura 1. Geometria do espelho hiperbólico e da câmera

$$y = \sqrt{a^2 \cdot \left(1 + \frac{x^2}{b^2}\right)} - \sqrt{a^2 + b^2} \tag{1}$$

$$\alpha = \frac{\pi}{2} + \operatorname{atan}\left(\frac{h - 2\sqrt{a^2 + b^2}}{r_{\text{topo}}}\right) \tag{2}$$

Na Figura (1), h é a distância entre a borda do espelho e o centro da câmera, $e = \sqrt{a^2 + b^2}$ representa a excentricidade do espelho, r_{topo} é a coordenada x do topo do espelho, e $y_{\text{topo}} = h - 2e$ é a coordenada y do topo do espelho.

Para se chegar à equação final do espelho é necessário estabelecer uma relação entre os parâmetros a e b . Da Equação (1), obtém-se a Eq. (3) que relaciona b em função de a/b .

$$b = h \cdot \sqrt{\left(\frac{a^2}{b^2} + 1\right)} - \frac{a}{b} \cdot \sqrt{h^2 + r_{\text{topo}}^2} \quad (3)$$

Para que o sistema de visão omnidirecional seja compacto, deve-se fixar um valor adequado para r_{topo} e h . Considerando-se que a projeção do espelho na imagem é um círculo e não uma elipse, ou seja, considerando-se que os fatores de escala vertical e horizontal da projeção da imagem na câmera são iguais, a altura h pode ser calculada em função da distância focal da câmera (f), da medida em pixel do raio da borda do espelho (r_{pixel}) e da medida em milímetros do tamanho de cada pixel do CCD da câmera (t_{pixel}). A Equação (4) expressa h em função destes parâmetros.

$$h = \frac{f \cdot r_{\text{topo}}}{t_{\text{pixel}} \cdot r_{\text{pixel}}} \quad (4)$$

A câmera e lente que foram utilizadas no sistema de visão possuem os seguintes parâmetros aproximados: $f=12\text{mm}$, e $t_{\text{pixel}}=0,01\text{mm}$. A imagem adquirida pela câmera possui tamanho de 640×480 pixels. Assim, deseja-se que $r_{\text{pixel}}=240$. Por fim, para tornar o sistema compacto, escolheu-se $r_{\text{topo}}=20\text{mm}$. Substituindo estes valores na Eq. (4) obtém-se $h=100\text{mm}$.

Analisando a forma do espelho e o ângulo de visão para algumas possibilidades de a/b , optou-se por construir um espelho com $a = 2b$. Portanto, os valores calculados através das Eq. (3) e (2) foram: $b=19,646$, $a=39,292$, e $\alpha=121,3^\circ$. Substituindo estes valores de a e b na Eq. (1), obtém-se a expressão final do espelho no sistema de coordenadas centrado em F' , dada pela Eq. (5).

$$y = 39,3 \sqrt{\left(1 + \frac{x^2}{186}\right)} - 43,9 \quad (5)$$

Determinada a equação do espelho, é possível sua fabricação através do torneamento de um tarugo de alumínio, e posterior polimento da superfície da peça até que ela se torne reflexiva como um espelho. O Departamento de Engenharia Mecânica da USP de São Carlos possui um torno CNC de ultra precisão que consegue dar às peças usinadas o mesmo acabamento de uma peça polida, conseguindo chegar a uma superfície reflexiva como um espelho. Optou-se pela fabricação do espelho nesta máquina devido às pequenas dimensões da peça, evitando o polimento manual que poderia imprimir na peça erros de forma que prejudicariam a imagem adquirida pelo sistema de visão.

4. Implementação do sistema

O sistema de visão omnidirecional implementado é composto de uma câmera, um espelho e peças para fixação deste espelho e câmera na posição correta.

Para que as peças de fixação causassem mínima obstrução na imagem adquirida, ou seja, o reflexo destas peças no espelho não diminuíssem a região de reflexo do ambiente exterior, que contém informações úteis para a aplicação na qual o sistema de visão será utilizado, utilizou-se um tubo de acrílico para sustentar o espelho na frente da câmera na distância desejada. Um dos inconvenientes desta solução é que dependendo da luminosidade do ambiente, alguns reflexos de luz no acrílico podem aparecer na imagem capturada. Entretanto, testes feitos no laboratório apresentaram bons resultados, mostrando que, para os propósitos das aplicações do sistema de visão, o acrílico não interfere significativamente na imagem adquirida, seja através de reflexos ou de distorções devido a possível irregularidade no acrílico.

Para obtenção do espelho esférico, depositou-se uma camada de metal na parede interior (parte côncava) de uma cúpula semi esférica de vidro. O vidro apresenta uma irregularidade no ápice da semi esfera. Entretanto, esta região geralmente reflete a própria câmera apontada para o espelho, não sendo utilizada para obter informação útil do ambiente. Também não se pode garantir, devido ao processo de fabricação da cúpula, uma boa precisão de forma, mas apesar disso, as imagens obtidas com este espelho apresentaram um resultado adequado.

A peça que contém a superfície hiperbólica foi pré-usinada em torno CNC convencional, e o acabamento da superfície hiperbólica foi feito em um torno de ultra-precisão no Departamento de Engenharia Mecânica da USP de São Carlos para deixar a superfície espelhada.

Para que a superfície do espelho pudesse ser usinada no torno de ultra-precisão, foi necessário calcular adequadamente os pontos programados no torno. A superfície usinada é formada pela interpolação linear dos pontos

programados e dependendo da quantidade e posição dos pontos, a superfície pode apresentar um erro de forma que prejudique a formação de uma imagem contínua. O cálculo adequado da posição destes pontos garante um bom resultado final da superfície.

Utilizou-se o método proposto por Palma e Porto (1995) para determinar os pontos de maneira correta. Este método calcula os pontos que devem ser programados garantindo o erro de forma desejado e fazendo a compensação do raio de ferramenta. Esta compensação é necessária pois ao longo da usinagem, o contato da ferramenta com a peça não ocorre na ponta da ferramenta, que é a referência usada para zerar o sistema de coordenadas do torno.

A peça foi usinada garantindo um erro de forma de 0.0001mm, que é adequado para produzir uma superfície que reflita a luz visível.

O protótipo do sistema de visão foi montado no robô móvel, e algumas imagens foram adquiridas utilizando os espelhos esférico e hiperbólico. O protótipo construído do sistema de visão omnidirecional que utiliza o espelho hiperbólico possui 285 mm de comprimento, e o protótipo do sistema de visão que utiliza o espelho esférico possui 410 mm de comprimento. No entanto, o tamanho final de ambos pode ser diminuído se for utilizado uma câmera de dimensões menores.

A Figura (2) mostra o protótipo do sistema de visão omnidirecional construído, os espelhos esférico e hiperbólico, e as peças de fixação construídas.

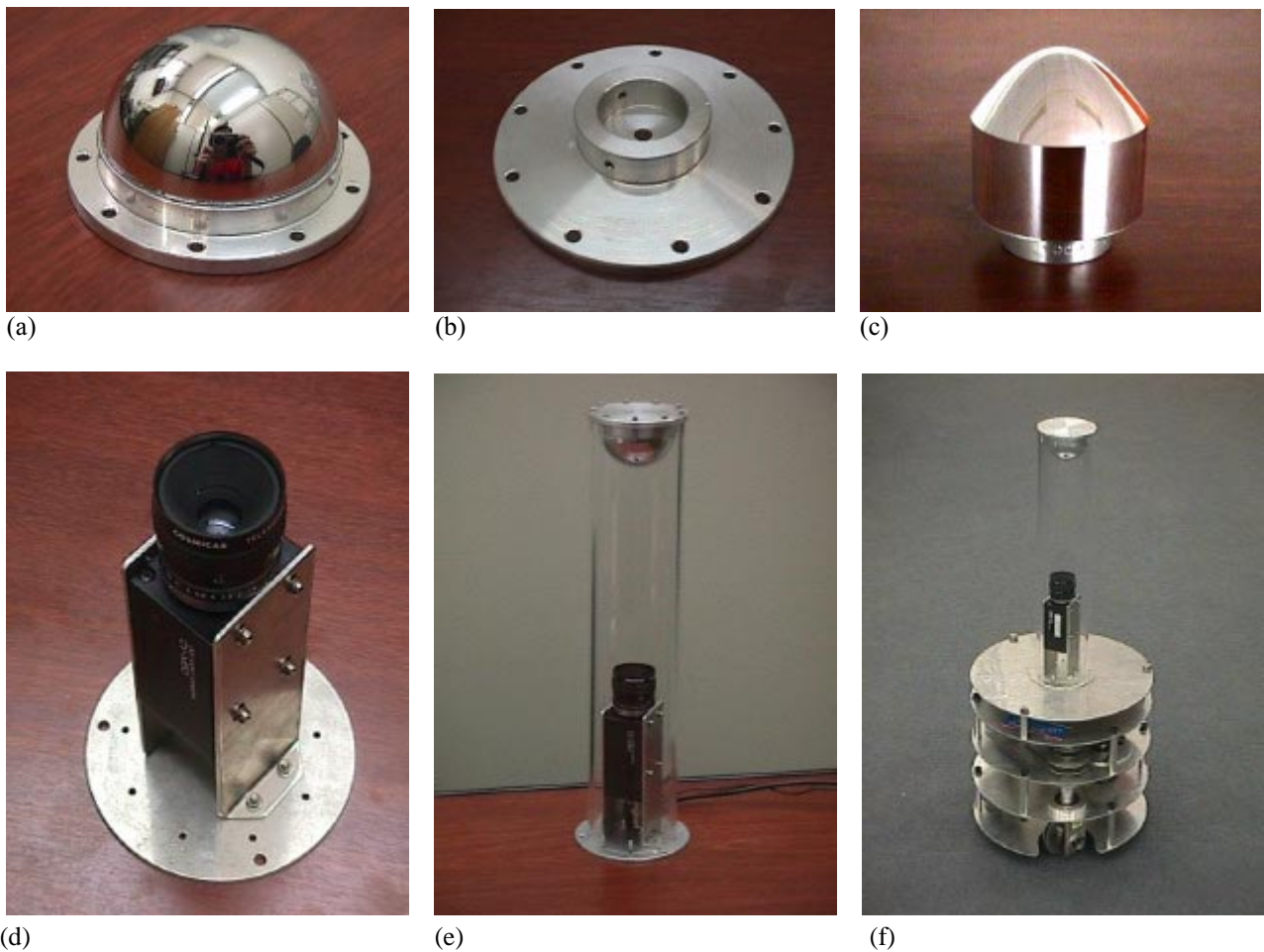


Figura 2. Imagens do sistema de visão omnidirecional catadióptrico construído. (a) Espelho esférico de 70mm de diâmetro construído em vidro. (b) Base para fixação do espelho hiperbólico. (c) Espelho hiperbólico de alumínio usinado em torno de ultra-precisão. (d) Peças de fixação da câmera. (e) Protótipo do sistema de visão omnidirecional com espelho esférico. (f) Sistema de visão omnidirecional montado em robô móvel.

5. Retificação da imagem

O ambiente ao redor do sensor de visão omnidirecional é mapeado na imagem adquirida em coordenadas polares não lineares, dependendo da forma do espelho convexo utilizado. Muitas vezes, conforme a aplicação, se faz necessário transformar esta imagem obtida através do sistema de visão omnidirecional em uma imagem panorâmica, ou se possível, em imagens de projeção perspectiva. Esse processo de transformação é denominado retificação da imagem e através dele obtemos imagens que são de fácil observação pelo homem, ou imagens sobre as quais podem ser utilizados algoritmos de processamento de imagem desenvolvidos para serem usados com imagens de projeção perspectiva.

Foram implementados dois métodos de retificação da imagem: um método baseado em transformação direta de coordenadas polares para retangulares, e outro método que considera a equação do espelho hiperbólico no processo de retificação.

5.1 Mapeamento baseado em transformação direta de coordenadas

O primeiro método de retificação implementado se baseia em uma transformação de coordenadas polares para coordenadas retangulares. Nesta transformação, o raio na imagem omnidirecional adquirida pelo sistema é mapeamento linearmente como sendo a coordenada y da imagem retificada, e o ângulo na imagem omnidirecional é mapeado na coordenada x da imagem retificada como representado pela Fig. (3). Para este método de retificação, se o centro da imagem for considerado como sendo a origem (0,0) e todas as variáveis forem escaladas para assumirem valores entre 0 e 1, as Eq. (6) podem ser usadas para fazer a retificação.

$$x_{\text{entrada}} = y_{\text{saída}} \cdot \cos(2\pi \cdot x_{\text{saída}}) \quad ; \quad y_{\text{entrada}} = y_{\text{saída}} \cdot \sin(2\pi \cdot x_{\text{saída}}) \quad (6)$$

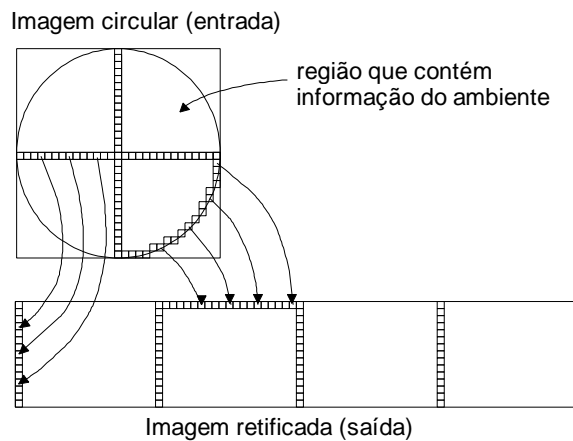


Figura 3. Esquema representando o processo de retificação de uma imagem

Nas equações (6), x_{entrada} e y_{entrada} são as coordenadas do pixel na imagem capturada pelo sistema de visão omnidirecional e correspondem às coordenadas $x_{\text{saída}}$ e $y_{\text{saída}}$ da imagem retificada.

É interessante notar que é feito um mapeamento inverso. Para cada pixel na imagem retificada, busca-se um correspondente na imagem circular capturada. Isso garante que todos os pixels na imagem retificada sejam preenchidos, e também permite que vários pixels na imagem retificada correspondam a um mesmo pixel na imagem capturada.

5.2 Mapeamento dado pela equação da superfície do espelho hiperbólico

No segundo método de retificação implementado, são consideradas a equação da superfície do espelho hiperbólico utilizado e a propriedade de centro único de projeção para que se possa obter uma imagem panorâmica livre de distorções.

No espelho hiperbólico, os raios ópticos refletidos se interceptam em um único ponto, o foco da hipérbole, que é também chamado de centro de projeção. Este centro de projeção único é interessante pois funciona como se estivéssemos vendo o ambiente em todas as direções através deste ponto. Dessa forma, a partir da imagem obtida por um sistema omnidirecional com esta propriedade, é possível construir imagens perspectivas projetando a imagem capturada para um plano qualquer no espaço, ou construir imagens panorâmicas através da projeção em um cilindro.

A Figura (4) mostra como um raio de pixels da imagem capturada com o sistema de visão omnidirecional pode ser transformada em uma coluna na imagem panorâmica obtida através da projeção em um cilindro considerando-se a geometria do espelho hiperbólico.

Através de algumas manipulações trigonométricas feitas a partir do esquema representado na Fig. (4), obtém-se as relações expressas nas Eq. (7) e (8) que são usadas no processo de retificação.

$$p_{\text{entrada}} = \frac{x \cdot D \cdot r_{\text{pixel}} \cdot (2e + y_{\text{topo}})}{p_{\text{saída}} \cdot x \cdot y_{\text{topo}} + D \cdot 2e \cdot r_{\text{topo}}} \quad (7)$$

$$y = \frac{x \cdot p_{\text{saída}} \cdot y_{\text{topo}}}{D \cdot r_{\text{topo}}} \quad (8)$$

Nas Equações (7) e (8), $p_{saída}$ representa um pixel em uma coluna na imagem retificada, $p_{entrada}$ representa um pixel de um raio na imagem capturada pelo sistema de visão omnidirecional, x e y representam um par de coordenadas da superfície do espelho, D representa a altura em pixel, definida pelo usuário, da parte na imagem retificada, que se estende da linha formada pelo mapeamento dos pixels correspondentes a borda do espelho, $x = r_{topo}$, até a linha formada pelo mapeamento dos pixels correspondentes a $y = 0$. Os demais parâmetros das Eq. (7) e (8), a excentricidade do espelho $2e$, r_{pixel} , y_{topo} e r_{topo} , são conhecidos.

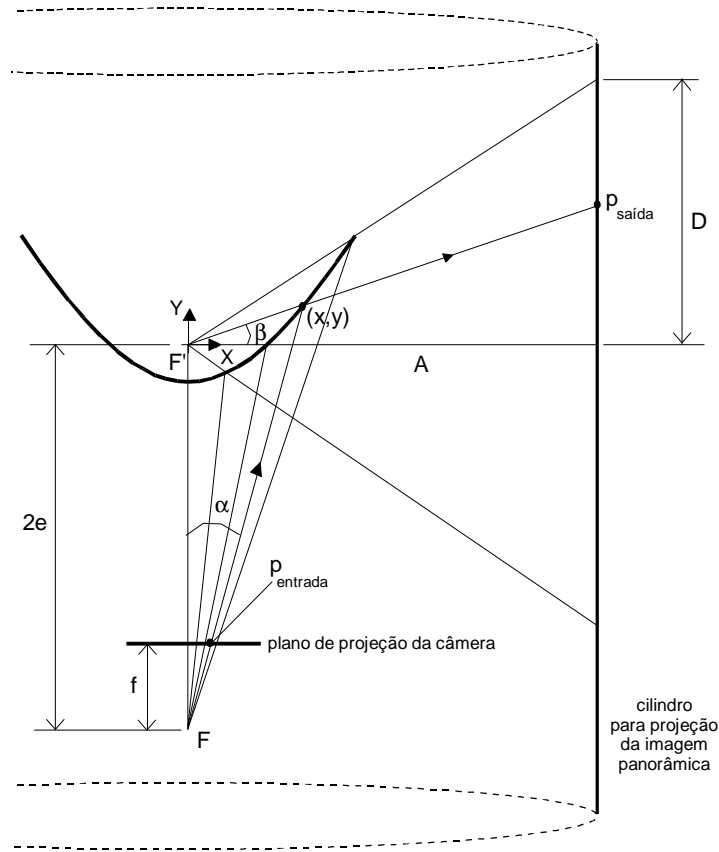


Figura 4. Geometria do espelho hiperbólico e da câmera usada para retificação da imagem

A Equação (8) representa uma reta que passa pelo foco da hipérbole e também passa por $p_{saída}$. Para um dado $p_{saída}$, iguala-se a Eq. (8) com a Eq. (5) do espelho e determina-se a raiz do polinômio através do método numérico de Newton, obtendo-se o valor de x onde a reta e o espelho se cruzam. Substituindo este valor de x na Eq. (7), obtém-se o pixel $p_{entrada}$ que equivale a um dado $p_{saída}$.

Observe que, assim como no outro método de retificação, aqui é feito um mapeamento inverso, buscando-se para cada pixel da imagem final retificada, um correspondente na imagem capturada pelo sistema de visão.

A formulação apresentada até aqui faz o mapeamento inverso apenas de uma coluna da imagem retificada para um raio na imagem capturada pelo sistema de visão, Fig. (3). As Eq. (9) estendem o mapeamento para toda a imagem. Nestas equações, $x_{saída}$ é um valor de 0 a 1 que representa qual a coluna na imagem retificada que está sendo mapeada; $p_{entrada}$ é o valor obtido através da Eq. (8) em função de $p_{saída}$; $x_{entrada}$ e $y_{entrada}$ representa a coordenada do pixel na imagem capturada pelo sistema de visão que corresponde a um dado pixel na imagem retificada. Na Equação (9) assume-se que a origem (0,0) da imagem capturada pelo sistema de visão se encontra no centro da imagem.

$$x_{entrada} = p_{entrada} \cdot \cos(2\pi \cdot x_{saída}) \quad ; \quad y_{entrada} = p_{entrada} \cdot \sen(2\pi \cdot x_{saída}) \quad (9)$$

5.3 Resultados dos Métodos de Retificação

Os métodos de retificação foram implementados para utilizarem imagens salvas em arquivo.

A Figura (5.a) mostra uma imagem capturada pelo sistema de visão omnidirecional com o espelho esférico, e a Fig. (6) mostra o resultado obtido através do processo de retificação direta de coordenadas descrito na seção 5.1 aplicado na imagem mostrada na Fig. (5.a).

A Figura (5.b) mostra uma imagem capturada pelo sistema de visão omnidirecional com o espelho hiperbólico, sendo que as Fig. (7) e (8) mostram, respectivamente, o resultados obtidos através da retificação pelo método que usa a equação do espelho descrito na seção 5.2, e pelo método de retificação direta de coordenadas descrito na seção 5.1.

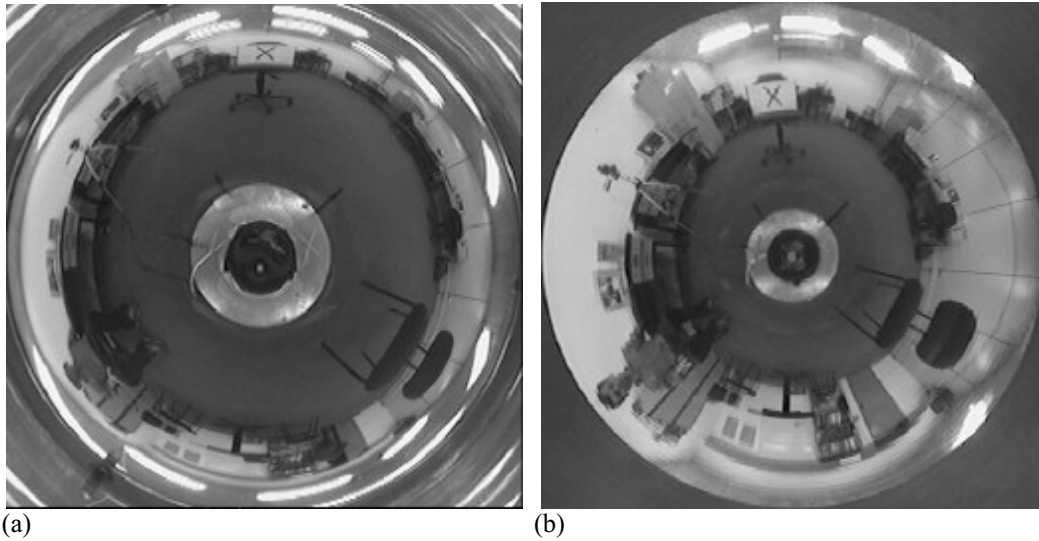


Figura 5. Imagens adquiridas pelo sistema de visão omnidirecional. (a) Imagem adquirida com o espelho esférico. (b) Imagem adquirida com o espelho hiperbólico



Figura 6. Imagem adquirida com o espelho esférico retificada pelo método de transformação direta de coordenadas



Figura 7. Imagem adquirida com o espelho hiperbólico retificada pelo método desenvolvido que leva em consideração a equação do espelho hiperbólico e a propriedade de centro único de projeção



Figura 8. Imagem adquirida com o espelho hiperbólico retificada pelo método de transformação direta de coordenadas

Verifica-se que na imagem capturada pelo sistema de visão omnidirecional com o espelho esférico, Fig. (5.a), os objetos que aparecem na periferia da imagem são representados por uma quantidade de pixels menor que a usada para representar os objetos que aparecem na periferia da imagem capturada com o espelho hiperbólico, Fig. (5.b). Isso acontece devido a forma do espelho esférico, e a característica mencionada pode ser observada também ao comparar-se a Fig. (6) e a Fig. (8) que correspondem as imagens capturadas com os dois espelhos retificadas usando o mesmo método. As duas imagens da Fig. (5) podem ser comparadas pois os objetos que são mostrados em ambas as imagens capturadas se encontram na mesma posição no ambiente.

Comparando-se entre si os resultados apresentados nas Fig. (7) e (8) referentes a aplicação dos métodos de retificação na imagem capturada pelo sistema omnidirecional com o espelho hiperbólico, comprova-se que ao se usar, no processo de retificação, a equação da superfície do espelho juntamente com a propriedade de centro único de projeção do espelho hiperbólico, é possível obter-se uma imagem livre de certas distorções. Dessa forma, pode-se perceber que na Fig. (7) houve uma maior preservação da proporção na imagem da cadeira à esquerda, e da pessoa no centro da imagem.

Além da implementação do método de retificação usando-se imagens salvas em arquivo, Chahl (1997) implementou com sucesso o método de retificação direta de coordenadas da imagem em tempo real, ou seja, cada quadro de imagem obtido pela câmera é retificado. Para isso, Chahl calculou previamente como os pixels são mapeados de uma imagem para outra e guardou este resultado em tabelas (“*lookup tables*”) que são acessadas de forma rápida, tornando possível a visualização de imagens panorâmicas em tempo real. Contudo, para as aplicações em robótica móvel atualmente em desenvolvimento, pretende-se trabalhar com as imagens capturadas pelo sistema omnidirecional sem que seja necessário retificá-las, os algoritmos de retificação desenvolvidos operam em imagens salvas em arquivo visando a avaliação da qualidade dos espelhos, a validade da metodologia utilizada para sua construção e verificar a propriedade de centro único de projeção do espelho hiperbólico.

6. Conclusão

Foi desenvolvido um sistema de visão omnidirecional composto por uma câmera de projeção perspectiva montada em frente a um espelho convexo. Duas implementações foram feitas, uma usando um espelho hiperbólico de 40mm de diâmetro e outra usando um espelho esférico de 70mm de diâmetro. O sistema de visão que se utiliza do espelho hiperbólico possui 285 mm de altura, sendo possível tornar o sistema ainda mais compacto se uma câmera de menores dimensões for usada.

O espelho esférico foi construído a partir da deposição de metal na superfície interna de uma cúpula semi esférica de vidro. O espelho hiperbólico foi construído através do torneamento de um tarugo de alumínio em uma máquina CNC convencional, e posterior usinagem da superfície hiperbólica por um torno de ultra-precisão a fim de conferir a peça o acabamento espelhado. A peça do espelho hiperbólico foi usinada no Departamento de Engenharia Mecânica da USP de São Carlos.

Foram implementados dois métodos de retificação da imagem adquirida pelo sistema de visão omnidirecional. Estes métodos transformam a imagem adquirida pelo sistema em uma imagem panorâmica, facilitando a interpretação da imagem por seres humanos. Para a imagem adquirida com o uso do espelho hiperbólico, é possível gerar uma imagem panorâmica ou até mesmo uma imagem perspectiva livre de certas distorções já que este tipo de espelho possui centro único de projeção.

Algumas imagens obtidas pelo sistema de visão omnidirecional desenvolvido foram apresentadas juntamente com os resultados obtidos pelos métodos de retificação.

O sistema omnidirecional desenvolvido será usado em um robô móvel para aplicações de tele-operação, navegação e controle servo visual.

7. Agradecimentos

Os autores agradecem à FAPESP por propiciar condições financeiras aos pesquisadores para que este trabalho pudesse ser desenvolvido. Agradecimentos também à RECOPE pela aquisição de alguns dos equipamentos que foram utilizados para desenvolver este trabalho, a Roberto Verzini da Engfilme Ind. e Com. Ltda. por ter custeado o espelhamento da cúpula de vidro, ao técnico Adilson Fernandes Felix Filho por ter usinado as peças de fixação do sistema de visão omnidirecional, ao Núcleo de Manufatura Avançada do Departamento de Engenharia Mecânica da USP de São Carlos por terem cedido recursos humanos e equipamentos essenciais para a usinagem do espelho hiperbólico, e aos professores pelo incentivo e acompanhamento despendidos.

8. Referências

- Baker, S., Nayar, S.K., 1998, “A Theory of Catadioptric Image Formation”, Proceedings of the 6th International Conference on Computer Vision, pp. 35-42, Bombay, India.
- Cao, Z.L., Oh, S.J., Hall, E.L., 1985, “Dynamic Omnidirectional Vision for Mobile Robots”, Proc. SPIE Intelligent Robots and Computer Vision, Vol. 579, pp. 405-414.
- Chahl, J.S., Srinivasan, M.V., 1997, “Reflective Surfaces for Panoramic Imaging”, Applied Optics, Vol. 36, No. 31, pp. 8275-8285.

- Chang, P., Hebert, M., 1998, "Omni-directional Visual Servoing for Human-Robot interaction", Proc. of IEEE/RSJ International Conference on Intelligent Robotic Systems (IROS), Victoria, B.C., Canada.
- Hicks, A., Bajcsy, R., 1999, "Reflective Surfaces as Computational Sensors", Workshop on Perception for Mobile Agents at CVPR99, Fort Collins, Colorado, USA.
- Nayar, S.K., 1997, "Catadioptric Omnidirectional Camera", Proc. of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 482-488.
- Palma, J.G., Porto, A.J.V., 1995, "Sistema de Apoio a Programação Comando Numérico para Usinagem de Ultraprecisão", Dissertação de mestrado, Escola de Engenharia de São Carlos da USP, São Carlos.
- Peri, V.N., Nayar, S.K., 1997, "Generation of Perspective and Panoramic Video from Omnidirectional Video", Proc. of DARPA Image Understanding Workshop, pp. 243-245.
- Svoboda, T., Pajdla, T., Hlavac, V., 1997, "Central Panoramic Cameras: Geometry and Design", Research Report K335/97/147, Czech Technical University, Faculty of Electrical Engineering, Center for Machine Perception, available at <ftp://cmp.felk.cvut.cz/pub/cmp/articles/svoboda/TR-K335-97-147.ps.gz>
- Yagi, Y., Kawato, S., Tsuji, S., 1994, "Real-Time Omnidirectional Image Sensor (COPIS) for Vision-Guided Navigation", IEEE Transactions on Robotics and Automation, Vol. 10, No. 1, pp. 11-22.
- Yagi, Y., 1999, "Omnidirectional Sensing and Its Applications", IEICE Transactions on Information and Systems, Vol. E82-D, No. 3, pp. 568-579.

DEVELOPMENT OF AN OMNIDIRECTIONAL VISION SYSTEM

Valdir Grassi Junior

Departamento de Engenharia Mecânica e de Sistemas Mecânicos
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231
05508-900 São Paulo, SP
vgrassi@usp.br

Cláudia C. G. Decco

Departamento de Engenharia Mecânica e de Sistemas Mecânicos
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231
05508-900 São Paulo, SP
cegdecco@usp.br

Jun Okamoto Junior

Departamento de Engenharia Mecânica e de Sistemas Mecânicos
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231
05508-900 São Paulo, SP
jokamoto@usp.br

Arthur José Vieira Porto

Núcleo de Manufatura Avançada
Escola de Engenharia de São Carlos da Universidade de São Paulo
Av. Trabalhador São-carlense, 400 CP. 359
13560-970 São Carlos, SP
ajvporto@sc.usp.br

Abstract. *Omnidirectional vision systems can get images with a 360-degree of field of view. For this reason, they can be useful to robotic navigation, tele-operation and visual servoing. This type of system does not require the movement of the camera to the direction of attention of the robot. On the other hand, it is required a non-conventional image processing once the image captured by this vision system is mapped on a non-linear polar coordinate system. An effective way to obtain an image in an omnidirectional system is with the combined use of lenses and mirrors. Several shapes of convex mirrors can be used mounting the center of the mirror aligned with the camera optics axis. The most common used mirror shapes are conic, parabolic, hyperbolic and spherical. Two types of mirror were constructed for the initial tests: a hyperbolic and a spherical mirror. The hyperbolic mirror was made using ultra-precision machining. A program was also developed to transform the acquired omnidirectional image into a plane image, resulting on a panoramic image of the environment. This work shows the development of an omnidirectional vision system, presenting the calculations used to determine a suitable mirror shape, the mechanical solutions used to build a fully operational system, and the results of the developed algorithm.*

Keywords: omnidirectional vision, digital image processing, ultraprecision machining, catadioptric systems

BASIC PRINCIPLES AND CONCEPTS OF BALLISTIC LOCOMOTION

Max Suell Dutra

Universidade Federal do Rio de Janeiro, COPPE/PEM, C.P. 68503 – CEP. 21945-970, Rio de Janeiro, RJ, Brazil
E-mail: max@serv.com.ufrj.br

Armando Carlos de Pina Filho

Universidade Federal do Rio de Janeiro, COPPE/PEM, C.P. 68503 – CEP. 21945-970, Rio de Janeiro, RJ, Brazil
E-mail: pina-filho@mailbr.com.br

***Abstract:** Human and animal locomotion is performed by means of rhythmic and synchronized movements. Regarding the motion of the legs, a large number of degrees of freedom is involved, as well as a series of particularities related to the study of kinematics and dynamics, to the influence of friction, to the possibility of impacts and instability, to the essential coordination between the segments, etc. The study of the ballistic locomotion is a particular case of the locomotion study, presenting models with characteristics similar to those of the “natural” locomotion and having, for this reason, large application in designing autonomous robots and developing prostheses and orthoses. In this work are presented and discussed many topics relevant to the project of ballistic walkers. This research, containing several concepts and principles, represents a starting-point to the project of bipedal ballistic walkers.*

***Keywords:** Ballistic Walker, Biped, Locomotion.*

1. Introduction

Studies related to the functionality of the muscles of the leg during locomotion in normal speeds demonstrate that it requires a very little activity from them during the phase of swinging the leg, that is, the phase in which one of the legs is on the floor and the other one effectuates the rhythmic movement to perform the desired step (phase of single support). The muscles are required only in the beginning and in the end of the phase of swinging motion. When both feet are on the floor it is the so-called phase of double support.

A cyclic movement like the locomotion can be performed by a mechanic system oscillating in its natural scope, applying the non-forced oscillation, which manages the locomotion not to be defined only by the operators, but largely by the passive properties inherent to the system (Van der Linde, 1998). A model of ballistic locomotion (or ballistic walker) follows such principle, moving itself entirely under the action of gravity, since it sets in its rhythmic movement by means of a certain propulsion of energy. The appropriate analysis for the project of a ballistic walker must take into account certain aspects related to the gait of a biped, as well as factors concerning the study of friction, damping and control.

2. Objective

The study of the ballistic locomotion is of large application for the project of autonomous robots and in the technology of rehabilitation (production of prostheses and orthoses). In both areas it exists an inevitable restriction of energy for autonomous functionality.

Presently, in the project of prostheses and orthoses the results in terms of restitution of the motor function are still very limited. The study of the dynamic interaction between parts of the body is very complex, but it is clear that the mechanic properties of many parts are finely related, leading to a very efficient way of locomotion. A more coherent understanding on the dynamic of human gait can help considerably in the project of prosthetic and orthotic apparatus.

3. Advantages and Disadvantages

Ballistic walkers have big advantages, such as: very little energy expended, mechanic simplicity, low weight, and behavior of “natural” locomotion.

Notwithstanding it also must be taken into account severe restrictions. One disadvantage of the bipedal gait is the instability. Walking on two feet means that part of the time the body is equilibrated on one leg. This hints the necessity of a detailed analysis of mass distribution and “displacement” of the center of the bodily mass.

Since the pattern of ballistic locomotion is determined by the natural frequency of the mechanic system, one has another restriction concerning the changes in the limit cycle, which can only be achieved through the application of an additional force or torque to the system, or actively modifying the intrinsic parameters of the system. In this last case, the most reasonable intrinsic parameter to be varied is the stiffness of the articulation (Van der Linde, 1998).

4. Determinants of Gait

There is not only one way to describe the movements of the limbs during locomotion, but a description, given by Saunders *et al* (1953), is applied thanks to its simplicity and totality of description. In this are distinguished six determinants of gait: 1 - The compass gait that is performed with stiff legs like an inverted pendulum. The pathway of the center of gravity is a series of arcs, 2 - pelvic rotation about a vertical axis. The influence of this determinant flattens the arcs of the pathway of the center of gravity, 3 - pelvic tilt, the effects on the nonweight-bearing side further flatten the arc of translation of the center of gravity, 4 - knee flexion of the stance leg. The effects of this determinant combined with pelvic rotation and pelvic tilt achieve minimal vertical displacement of the center of gravity, 5 - plantar flexion of the stance ankle. The effects of the arcs of foot and knee rotation smooth out the abrupt inflexions at the intersection of the arcs of translation of the center of gravity, and 6 - lateral displacement of the pelvis. These determinants smooth out the abrupt changing in the support leg.

Figure (1) presents a bipedal toy that goes down inclined surfaces by the performance of lateral swinging movements synchronized with the swing of its pendular legs. The energy necessary to wear down the friction is provided by the fact that the toy goes down a little bit more in each step ahead. Such toy represents a classic model of a ballistic walker.

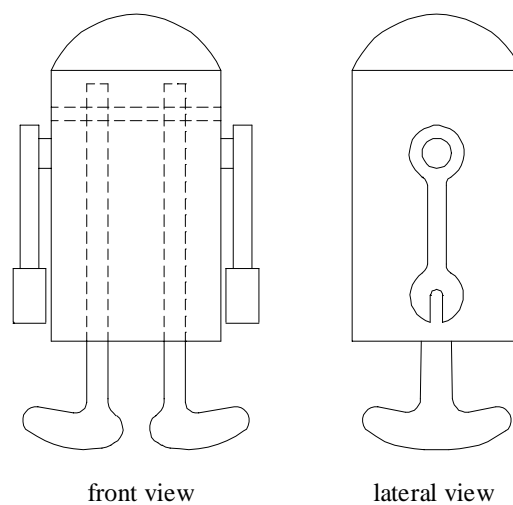


Figure 1. Classic model of a ballistic walker.

5. Simple Model of Ballistic Walker

The model that represents the least complicated mechanic configuration when one thinks about the dynamic of locomotion consists of three articulations (one in the hip and one for each knee), being that the foot is connected to the leg rigidly. The mass of the superior part of the body is gathered in the hip articulation, while the masses of the inferior limbs are really distributed. The motion performed in locomotion is similar to an inverted pendulum, where the hip articulation fulfils a series of arcs in the course of the movement.

Such model provides a good approximation of the horizontal efforts acting upon the mechanism, nonetheless the foreknowledge of the vertical efforts is very poor. With the purpose of solving this problem it must be added to the model new determinants of gait, able to define more precisely the vertical efforts, which are: the pelvic tilt, providing an increase of the vertical force in the beginning of the phase of swinging the leg, the plantar flexion of the supporting leg, increasing the force in the end of the movement, and the knee flexion of the supporting leg.

6. "Real" Model of Ballistic Walker

The ballistic walker that has the configuration of the simple model plus the three determinants of gait above-mentioned is very alike to the configuration of human locomotion, and that is why it was entitled "real" (placed between inverted commas, since the analysis presented here is two dimensional, moreover it has a series of restrictions that will be commented at the end of this work).

According to McMahon (1984), it can be imaged mechanisms that provide real functional relations between the angle of the leg and the length of the vertical component of the pelvic articulation. This way, all the articulations of the leg work by means of a characteristic movement that depends on the knee angle only. Again, the hip and the knee keep moving freely under the action of gravity without any muscular torque.

7. Cinematic Analysis

The cinematic analysis is the starting point for the project of any mechanism. There are different procedures to specify the expressions of displacements of the system, such as the geometric analysis and the analysis by means of the so-called matrices of translation and rotation (ideal for more complex mechanisms). In accordance with the configuration adopted to the model, through the specification of lengths and/or angles, one achieves the desired results.

For a better perception of these factors, a three dimensional model with 15 degrees of freedom is necessary; nonetheless, in a way of simplifying the technical study, it is admitted to adopt a two-dimensional model (Dutra, 1995), which develops its movement on a plane, with three principal determinants of gait (Figure 2).

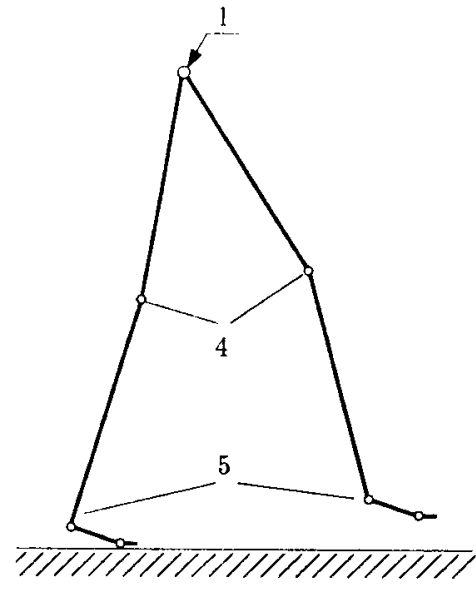


Figure 2. Two-dimensional model of a biped robot. Principal determinants of gait: compass (1), knee flexion (4) and plantar flexion of the stance ankle (5).

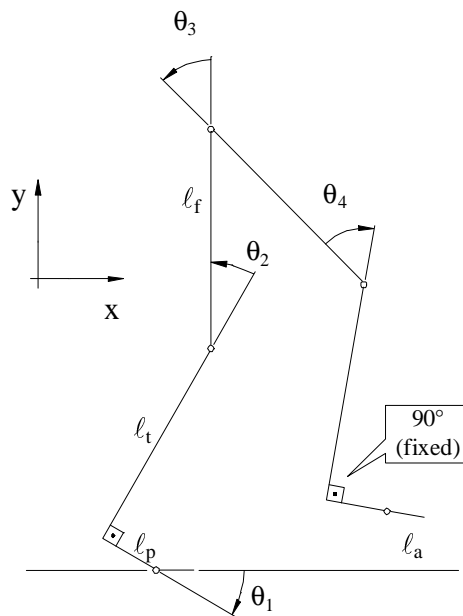


Figure 3. Model of the ballistic walker intended to be analyzed.

Thus, adopting the model presented in Fig. (3), one can determine the coordinates of the tiptoe of the swinging leg in relation to the initial point of movement, in this case the tiptoe of the supporting leg. Since the analysis is two dimensional the coordinate z will be zero, and the coordinates x and y will be given by the following equations:

$$x = \ell_a \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) + \ell_p \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) + \ell_t \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4) - \ell_f \sin(\theta_1 + \theta_2 + \theta_3) - \ell_f \sin(\theta_1 + \theta_2) - \ell_t \sin(\theta_1) - \ell_p \cos(\theta_1) - \ell_a \quad (1)$$

$$y = \ell_a \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4) + \ell_p \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4) - \ell_t \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) - \ell_f \cos(\theta_1 + \theta_2 + \theta_3) + \ell_f \cos(\theta_1 + \theta_2) + \ell_t \cos(\theta_1) - \ell_p \sin(\theta_1) \quad (2)$$

where: ℓ_a is the length of the part of the foot responsible for the support; ℓ_p is the length of the part of the foot that rises from the floor; ℓ_t is the length of the tibia; and ℓ_f is the length of the femur. Starting from the equations of x and y one can determine the speeds and accelerations, as well as specify the other coordinates of each one of the articulations:

Coordinates of the heel of the supporting leg:

$$x_{ca} = -\ell_p \cos(\theta_1) - \ell_a \quad (3)$$

$$y_{ca} = -\ell_p \sin(\theta_1) \quad (4)$$

Coordinates of the knee of the supporting leg:

$$x_{ja} = -\ell_t \sin(\theta_1) - \ell_p \cos(\theta_1) - \ell_a \quad (5)$$

$$y_{ja} = \ell_t \cos(\theta_1) - \ell_p \sin(\theta_1) \quad (6)$$

Coordinates of the hip:

$$x_q = -\ell_f \sin(\theta_1 + \theta_2) - \ell_t \sin(\theta_1) - \ell_p \cos(\theta_1) - \ell_a \quad (7)$$

$$y_q = \ell_f \cos(\theta_1 + \theta_2) + \ell_t \cos(\theta_1) - \ell_p \sin(\theta_1) \quad (8)$$

Coordinates of the knee of the swinging leg:

$$x_{jb} = -\ell_f \sin(\theta_1 + \theta_2 + \theta_3) - \ell_f \sin(\theta_1 + \theta_2) - \ell_t \sin(\theta_1) - \ell_p \cos(\theta_1) - \ell_a \quad (9)$$

$$y_{jb} = -\ell_f \cos(\theta_1 + \theta_2 + \theta_3) + \ell_f \cos(\theta_1 + \theta_2) + \ell_t \cos(\theta_1) - \ell_p \sin(\theta_1) \quad (10)$$

Coordinates of the heel of the swinging leg:

$$x_{cb} = \ell_t \sin(\theta_1 + \theta_2 + \theta_3 + \theta_4) - \ell_f \sin(\theta_1 + \theta_2 + \theta_3) - \ell_f \sin(\theta_1 + \theta_2) - \ell_t \sin(\theta_1) - \ell_p \cos(\theta_1) - \ell_a \quad (11)$$

$$y_{cb} = -\ell_t \cos(\theta_1 + \theta_2 + \theta_3 + \theta_4) - \ell_f \cos(\theta_1 + \theta_2 + \theta_3) + \ell_f \cos(\theta_1 + \theta_2) + \ell_t \cos(\theta_1) - \ell_p \sin(\theta_1) \quad (12)$$

This is very important for the determination of the trajectories that allow the biped robot to walk more skillfully (Shih, Gruver, and Lee, 1993). In addition, note that, in dynamic walking, each time one foot touches the floor, the system may suffer impacts and incurs on additional accelerations that influence the forward velocity (Zheng & Hemami, 1984). For this reason, one must impose some continuity conditions on the leg velocities to avoid impacts when the foot of the swinging leg touches the floor.

For example, adopting the model dimensions presented in Tab. (1), and the angles used by Dutra (1995), namely: $\theta_1 = -15^\circ$, $\theta_2 = 3^\circ$, $\theta_3 = 30^\circ$, and $\theta_4 = -3^\circ$, will have the coordinates showing in Tab. (2) and step $x = 0,363$ m.

Table 1. Model dimensions.

part of model	length [m]	mass [kg]
Thumb [ℓ_a]	0,03	0,2
Foot [ℓ_p]	0,11	0,6
Leg (below the knee) [ℓ_t]	0,37	4
Thigh [ℓ_f]	0,37	6
Body (above the hip)	0,8	20

Table 2. Coordinates of each one of the points of articulation.

supporting leg				hip		swinging leg			
x_{ca}	y_{ca}	x_{ja}	y_{ja}	x_q	y_q	x_{jb}	y_{jb}	x_{cb}	y_{cb}
-0,136	0,028	-0,040	0,386	0,036	0,748	0,132	0,390	0,228	0,033

8. Dynamic Structural Analysis

One of the main points in the project of a ballistic walker is the dynamic analysis of the structure. Such analysis can be accomplished by applying many different methods, but with the purpose of getting a mathematical model with the following configuration:

$$M\ddot{U} + KU = F \tag{13}$$

where: M represents the mass matrix; K is the stiffness matrix; F is the vector of forces; U is the vector of displacements. As examples, Hurmuzlu (1993) and Katoh & Mori (1984) present two models of biped with their respective dynamic analyses.

In this work, the selected method intended to be applied to analyse the structure was the finite element method, since it represents a quite efficient modeling technique, being largely applied to the analysis of structures containing complex and irregular geometric forms. Some main aspects of this method are presented by Inman (1994) and Meirovitch (1986).

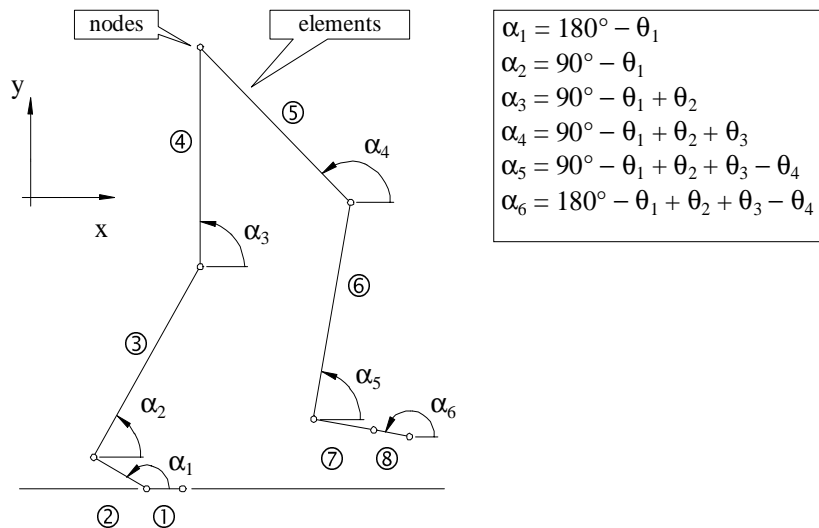


Figure 4. Angles (in absolute values), nodes and elements adopted for analysis.

Therefore, considering Fig. (4) (phase of single support) to be the model which will be studied, we wish to come to the dynamic equation of the walker. The type of element adopted is the so-called “bar element”, the same one applied in the analysis of trellis. By means of the finite element methods one can determine the mass and stiffness matrices for each element:

$$M_{(e)} = \frac{ml}{6} \begin{bmatrix} 2\cos^2 \phi & 2\sin \phi \cos \phi & \cos^2 \phi & \sin \phi \cos \phi \\ 2\sin \phi \cos \phi & 2\sin^2 \phi & \sin \phi \cos \phi & \sin^2 \phi \\ \cos^2 \phi & \sin \phi \cos \phi & 2\cos^2 \phi & 2\sin \phi \cos \phi \\ \sin \phi \cos \phi & \sin^2 \phi & 2\sin \phi \cos \phi & 2\sin^2 \phi \end{bmatrix} \tag{14}$$

$$K_{(e)} = \frac{EA}{l} \begin{bmatrix} \cos^2 \phi & \sin \phi \cos \phi & -\cos^2 \phi & -\sin \phi \cos \phi \\ \sin \phi \cos \phi & \sin^2 \phi & -\sin \phi \cos \phi & -\sin^2 \phi \\ -\cos^2 \phi & -\sin \phi \cos \phi & \cos^2 \phi & \sin \phi \cos \phi \\ -\sin \phi \cos \phi & -\sin^2 \phi & \sin \phi \cos \phi & \sin^2 \phi \end{bmatrix} \tag{15}$$

where: m is the mass of the element; l is its length; E is the elasticity modulus; A is the cross section area; and ϕ represents the angle formed by each element with the horizontal.

And then determine the matrices for the structure:

$$M = \sum_{e=1}^8 M_{(e)} \tag{16}$$

$$K = \sum_{e=1}^8 K_{(e)} \tag{17}$$

The vector of forces is formed by reactions in each point. Like in the phase of single support, where it occurs the rhythmic movement of the leg under the action of gravity, the forces are, in hypothesis, “null” (no muscular effort), then the dynamic equation of the walker in this phase will be:

$$\frac{1}{6} \begin{bmatrix} m_{1,1} & m_{1,2} & \dots & m_{1,14} \\ m_{2,1} & m_{2,2} & & \\ \dots & & \dots & \\ m_{14,1} & & & m_{14,14} \end{bmatrix} \mathbf{U}'_{14 \times 1} + \begin{bmatrix} k_{1,1} & k_{1,2} & \dots & k_{1,14} \\ k_{2,1} & k_{2,2} & & \\ \dots & & \dots & \\ k_{14,1} & & & k_{14,14} \end{bmatrix} \begin{Bmatrix} U_5 \\ U_6 \\ U_7 \\ \dots \\ U_{18} \end{Bmatrix}_{14 \times 1} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ \dots \\ 0 \end{Bmatrix}_{14 \times 1} \tag{18}$$

where: \mathbf{U}' is the vector of accelerations and the displacements (U_5, U_6, \dots, U_{18}) can be observed in Fig. (5).

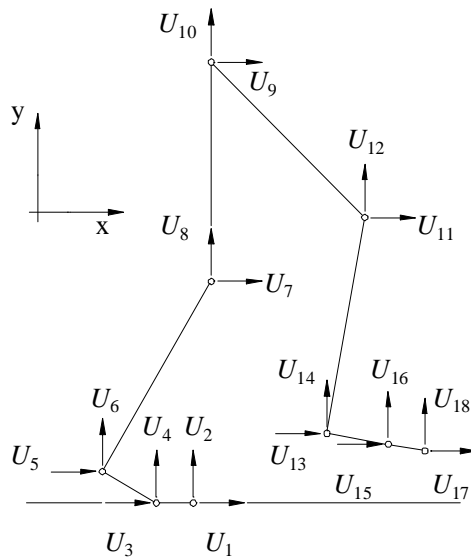


Figure 5. Displacements adopted for analysis.

To specify the dynamic equation for the phase of double support we just have to consider the new boundary conditions. In addition to $U_1 = U_2 = U_3 = U_4 = 0$, we also will have $U_{13} = U_{14} = U_{15} = U_{16} = U_{17} = U_{18} = 0$. After the calculation of the mass and stiffness matrices, we can determine the proper modes and frequencies of the mechanism solving the problem of eigenvalues and eigenvectors given by the equation:

$$[K]\{U\} = \Lambda[M]\{U\} \tag{19}$$

where: $\Lambda = \omega^2$ represents the eigenvalues (ω is the frequency), and $\{U\}$ represents the eigenvectors.

Such problem can be solved by means of the application of a computational program of calculation. This way, the frequency values should be found starting from the eigenvalues, and the modes starting from the eigenvectors, which used in conjunction with the interpolation functions given by Meirovitch (1986) could provide the approximate modes of the mechanism.

More details about the application of the finite element methods in the dynamics study of a ballistic walker are presented by Pina Filho (2001) and Dutra *et al* (2001).

9. Conclusion

As for the “real” model of ballistic walker, the addition of the knee flexion, plantar flexion of the supporting leg, and pelvic tilt does not cause drastic changes in the swinging period, but the last two ones are important for the estimation of the vertical reactions.

The action of gravity is so important in the dynamics of locomotion that a model that does not include any muscular torque during the whole phase of swing can fairly represent human locomotion in normal speeds. In spite of this, the ballistic walker presented here has its limitations. It does not assume a significant role for the arms and the trunk in the dynamics of locomotion. It is limited to the two dimensional plane and thus it does not take into account the relations between the movements of lateral swing and of front swing that the toy of Fig. (1) permits, and that is a very important aspect to be considered in human locomotion, specially in low speeds. Besides one must put a stable walker model in presence of problems, what was not evaluated.

The accurate study of the ballistic locomotion will be able to provide the opportunity to experiment functional concepts for new prostheses and orthoses leading to improved projects of orthopedic apparatus. Indeed, this is one of the major motivations for the study presented here and for other future works.

The topics presented in this work provide the acknowledge necessary to start a series of more complex and specific works aiming at the project of ballistic walkers, which embodies studies on topics not presented here, such as: the influence of the friction for the motion, a study on the damping, and possible forms for controlling the motion.

10. Acknowledgments

The authors would like to express their gratitude to CNPq, CAPES and FAPERJ for the financial support provided during the course of this present research.

11. References

- Dutra, M. S., 1995, “Bewegungskoordination und Steuerung einer zweibeinigen Gehmaschine”, *Shaker Verlag, Aachen, Germany*.
- Dutra, M. S., Pina Filho, A. C. de, Romano, V. F., 2001, “Dynamics Study of a Ballistic Walker Using the Finite Element Methods”, *IX International Symposium on Dynamic Problems of Mechanics - DINAME 2001*, UFSC, Florianópolis - SC, Brazil.
- Hurmuzlu, Y., 1993, “Dynamics of Bipedal Gait: Part I – Objective Functions and the Contact Event of a Planar Five-Link Biped”, *ASME Journal of Applied Mechanics*, Vol. 60, pp. 331-336.
- Inman, D. J., 1994, “Engineering Vibration”, *Prentice-Hall, USA*.
- Katoh, R., Mori, M., 1984, “Control Method of Biped Locomotion Giving Asymptotic Stability of Trajectory”, *Automatica*, Vol. 20, Pergamon Press, Great Britain, pp. 405-414.
- McMahon, T. A., 1984, “Muscles, Reflexes and Locomotion”, *Princeton University Press, USA*.
- Meirovitch, L., 1986, “Elements of Vibration Analysis”, *McGraw-Hill, Singapore*.
- Pina Filho, A. C. de, 2001, “Study of the Ballistic Locomotion in Bipedal Robots” (in portuguese), *M. Sc. dissertation*, COPPE/UFRJ, Rio de Janeiro - RJ, Brazil.
- Saunders, J. B., Inman, V., Eberhart, H., 1953, “The Major Determinants in Normal and Pathological Gait”, *J. Bone Jt. Surgery* 35 A.
- Shih, C. L., Gruver, W. A., Lee, T. T., 1993, “Inverse Kinematics and Inverse Dynamics for Control of a Biped Walking Machine”, *Journal of Robotic Systems*, Vol. 10, pp. 531-555.
- Van Der Linde, R. Q., 1998, “Active leg compliance for passive walking”, *Proceedings IEEE International Conference on Robotics and Automation*, Leuven, Belgium, pp. 2339-2344.
- Zheng, Y. F., Hemami, H., 1984, “Impact Effects of Biped Contact with the Environment”, *IEEE Transactions on Systems, Man and Cybernetics*, Vol. 14, pp. 437-443.

GENERATION OF WALKING PATTERNS OF AN HEXAPOD ROBOT THROUGH SCHEDULING ALGORITHMS

Max Suell Dutra

Universidade Federal do Rio de Janeiro - C.P. 68503 - CEP. 21945-970 Rio de Janeiro, RJ, Brasil
COPPE – Programa de Engenharia Mecânica
maxdutra@ufrj.br

Vinicius da Costa Mello

Universidade Federal do Rio de Janeiro - C.P. 68503 - CEP. 21945-970 Rio de Janeiro, RJ, Brasil
EE – Departamento de Engenharia Mecânica
vcmello@ufrj.br

Abstract: *This work presents a leg synchronization method of a six-legged (hexapod) robot based on strategies of sharing resources, named SMER (Scheduling by Multiple Edge Reversal). The synchronization of all legs is obtained through this method according to the desired locomotion pattern of such robots. This locomotion is very complex because there are different kinds of stability and the number of degrees of freedom is changed while the robot moves. There must be a method to generate the correct cyclic pattern and maintain it throughout motion. This generation can be implemented in a simple concentrated algorithm or, through SMER, for example, in a distributed one. The advantage of distributing the system is that each leg will be responsible for its own movement and the locomotion continues even with a dysfunctional leg. To make possible the characterization of hexapod gait patterns, being possible the application in the control of such robots, the ways of walking of six-legged insects were studied by many researchers, who defined some important characteristic patterns of locomotion of these insects. For this coordination, it is not necessary the characterization, because of their quantity and complexity, of all gait patterns of hexapod insects. Through novel strategy of sharing resources, the SMER strategy, there is the possibility of reaching any gait pattern determined in advance. The main goal is to mimic the most important biological characteristic of insect's locomotion system, named CPG (Central Pattern Generator) by many researchers. This CPG is responsible for oscillatory coordination in living beings, e.g.: walking, breathing, swimming, etc. The method SMER can be used to reproduce the characteristics mentioned before, through a simple oscillatory structure, composed of interconnected neurons which the importance lies on the mutual inhibition between them. As it is in Yang (1999), this method is interesting because it can be built as an asymmetric Hopfield network and its hardware implementation is facilitated because of such possibility. First, in this work, some studies in animal locomotion area are pointed, and then, some hexapod gaits are presented. After that, the SMER ideas are introduced with some formal description and finally it is showed the SMER description of one gait as an interesting example of its application in locomotion.*

Keywords: *Hexapod, Locomotion, Oscillators, Gaits, SMER.*

1. Introduction

1.1. Central pattern generators

Rhythmic behavior of biological structures is present everywhere. Some examples of this behavior are digestion, breathing and locomotion in living beings. Many people are interested in investigations of the neurophysiological mechanisms underlying the generation and control of walking; a list of references can be found in Collins et al. (1993).

Central pattern generators (CPGs), which are described in Collins et al. (1994) as groups of neurons located in the central nervous system, are believed to be involved in the production and regulation of various cyclic motor patterns in animals, like the locomotion itself.

Walking mechanisms, which have legs instead of wheels or tracks, present some advantages over other kinds of propulsion systems when autonomous intelligent robots have to move in a non flat or irregular terrain like the bottom of the sea, for example. Together with path planning, the correct coordination of the legs of such robots is a crucial point in its task of achieving a predefined point. Therefore, the understanding of CPGs is very important for the purpose of controlling legged robots, which can exchange between its gaits according to its necessities.

1.2. Robots biologically inspired

Robots inspired in biological models are an interesting branch of mobile machines research. This researching strategy lies on the reproduction of the main characteristics of insects and mammals. Some studies and mechanical models were done in order to mimic the interesting biological characteristics of these animals. Amendt (1995) examined the locomotion of an insect, *Carausius Morosus*, Golubitsky et al. (1998) presented studies of quadruped gaits of horses. Some works (Dutra, 1997; Raptopoulos, 1998; Oliveira, 1999) studied the kinematics and dynamics of hexapods robots inspired on, already mentioned, *Carausius Morosus* model. Costa Netto (Costa Netto et al., 2000; Costa Netto, 2001) proposed a mechanical model for the robot, and also, the generation of a step through neural networks and fuzzy logic.

One of the most important, and complex, rhythmic behavior is locomotion, which is essential for walking robots with legs. Then, the reproduction of biological characteristics of CPGs present in stick insects and some animals like horses, is a common approach. Collins et al. (1993) studied walking patterns of cockroaches and proposed some mathematical description. In various works (Collins, 1994; Buono et al., 2000) rings of coupled nonlinear oscillators are shown to be useful for reproducing some qualities of locomotor central pattern generations.

Yang (Yang et al., 1998; Yang, 1999) introduced in his work a novel algorithm, the SMER, which can be applied to predict or reproduce the interesting behavior of many biological oscillatory neural networks of arbitrary topology, specially CPGs, just assuming some form of *postinhibitory rebound (PIR)*.

Hence, this work is a continuation of CPGs study. The target is to show the reproduction of hexapodal gaits using SMER strategy. Yang showed that SMER can be constructed as an asymmetric Hopfield neural network, which is interesting because it can simplify the circuitry implementation. This neural network is robust and small deviations can be corrected.

A distributed algorithm is used in this case (leg coordination) to generate the activation sequence of the legs and, together with sensory feedback, maintain the motion being the "neural" structure of each leg responsible by its motion. This is more powerful than a table or a concentrated algorithm that only generates the sequence and some kind of control has to be implemented to sustain the motion.

In the next section, three of the most common and characteristic gaits of hexapods (*tripod*, *tetrapod* and *methacronal*) are seen. These gaits were studied based on those insects mentioned before: *Carausius Morosus* and cockroaches (Collins et al., 1993; Amendt, 1995; Costa Netto, 2001).

2. Animal gaits: primary gaits of a hexapod

One of the advantages of hexapod robots (over robots like bipeds and quadrupeds) is that they can move around maintaining its static stability during all motion. Hexapods guarantee its static stability during locomotion maintaining at least three legs on the ground supporting its weight; more detailed description can be viewed in (Oliveira, 1999; Costa Netto, 2001). Therefore, with this statement in mind, it can be imagined three ways of walking, which were described by Golubitsky et al. (1998) as primary gaits.

Golubitsky studied the reproduction of these gaits by pattern formation in a structure composed of coupled nonlinear oscillators and has described this pattern formation by the symmetries in the network of such oscillators.

These primary gaits were studied in many other works (Collins et al., 1993; Raptopoulos, 1998; Oliveira, 1999; Costa Netto, 2001). Here, is followed the gait description presented in Costa Netto's work, which describe three gaits: *tripod*, *tetrapod* and *wave gait* (or *methacronal*).

In locomotion, the animals have two distinct phases: the support phase and the transfer phase. The support phase is the period of the movement in which the leg is on the ground and is supporting robot's body mass and the transfer phase is the period which the leg is been moving into another position, or in other words, a leg is lifted off the ground, translated and finally positioned on the ground again. A descriptive figure, Fig.(1), of the three gaits studied here is adapted from Costa Netto (2001) and is showed in the sequel:

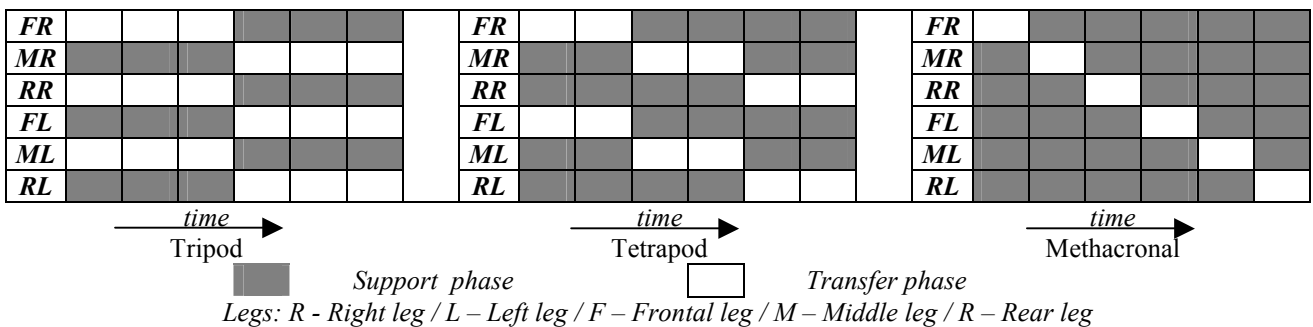


Figure 1. Descriptive phase relation between legs in different gaits (for one cycle).

All three are for only one cycle. The first one, is the *tripod* gait and it has three legs in contact with the ground during all movement. As showed by Collins et al. (1993) the *tripod gait* has three legs moving in phase and the other three moving with half phase (180°) difference. The middle graphic is one described as the *tetrapod gait*. In this gait two legs (each one in the opposite sides of the body, e.g., frontal left and frontal right) move in phase while all the others are in contact with the ground; the next pair of legs will move with one third (120°) phase lag. The last one is called *wave gait* (or *methacronal*), this gait is characterized by one six (60°) of a period difference between two adjacent legs (of the same side, e.g., left side or right side of animal's body). Legs on opposite sides, but in the same segment of the body, are half of period out of phase (e.g., left frontal and right frontal).

Insects choose between these gaits in order to change the parameters of locomotion, i.e., velocity, stability, etc. So, these gaits can be named as: fast speed (or running), medium speed and slow speed (or walking); these gaits are *tripod*, *tetrapod* and *metacronal*, respectively. In order to have an efficient hexapod robot biologically inspired these insect's gait patterns, and their exchanging, must be reproduced.

Our strategy for producing and maintaining each of these gaits in a robot is to apply SMER distributed algorithm to reproduce the phase relations of the locomotor CPG for each gait.

3. Scheduling by Multiple Edge Reversal: fundamentals

This section presents the basics of an algorithm named scheduling by multiple edge reversal (SMER). It is a simple but powerful distributed algorithm. First, the description of the basics of scheduling by edge reversal (SER), proposed by Barbosa, whose his work, in part, can be viewed in Barbosa et al. (1996), its generalization, which is the SMER proposed by França (1994), and then, Yang's contribution which is the Oscillatory Building Blocks (OBBs) network (Yang, 1999).

3.1. Scheduling by Edge Reversal (SER)

The elemental components in this kind of distributed algorithms are: *processes* and *atomic resources*. Here, the processes are the steps of each leg of the hexapod and atomic resources are the signals sent to the legs, atomic in the sense that they are processed by a single process.

To explain the mechanism of SER, consider a system by a connected graph $G=(N,E)$, where N is the finite set of nodes, which represent the processes, and E is the finite set of edges representing shared resources. An edge is present between any two nodes if and only if the two corresponding processes share at least one atomic resource. A *sink* is any node which all edges connecting itself with its neighbors are directed to itself and these neighbors are all idle, similarly a *source* is any node which all the edges connecting itself with its neighbors are directed to them, so in any acyclic orientation ω the set of nodes are marked as *sinks*(ω) or *sources*(ω) depending if they are with all their connected edges directed to itself or not.

SER works in the following way: starting from any acyclic orientation ω on G there is at least one sink node, i.e., a node that has all its edges directed to itself. All sink nodes are allowed to operate while other nodes remain idle. This obviously ensures mutual exclusion at any access made to shared resources by sink nodes. After operation a sink node will reverse the orientation of its edges, becoming a source and thus releasing the access to resources to its neighbors. A new acyclic orientation is defined and the whole process is then repeated for the new set of sinks. Let $\omega'=g(\omega)$ represent this greedy operation, SER can be regarded as the endless repetition of the application of $g(\omega)$ upon G . It's easily seen that this operation for a finite set of nodes and edges will be repeated defining a period of length p and this simple dynamics ensures that no deadlock, all nodes will operate in a cycle, or starvation, every node under any schedule eventually becomes a sink, will occur since in every acyclic orientation there is at least one sink and it will operate exactly m times. Figure (2) illustrates SER dynamics under an acyclic orientation, reproduced from Yang (1999).

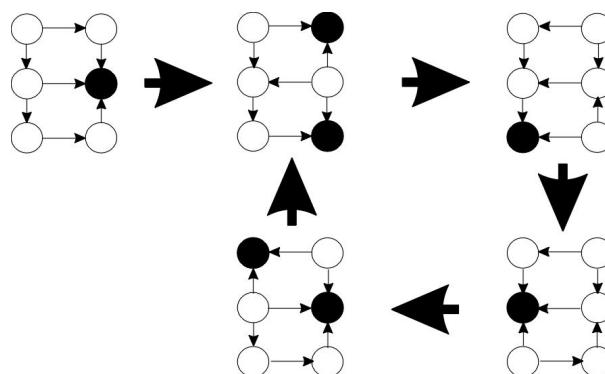


Figure 2. A graph G under SER, with $m=1$ and an operating cycle $p=4$.

3.2. Scheduling by Multiple Edge Reversal (SMER)

SMER is a generalization of SER, as stated above, where is allowed nodes to have different access rates to atomic resources. Let a multigraph $M(N,E)$ represent the system under SMER dynamics. Differently from SER, which all nodes have the same access rate and a node only can operate once a cycle, SMER allows the nodes operate more than once consecutively, yielding different access rates for each node.

Between any two nodes i and j where $i, j \in N$, there can exist $e_{ij} \geq 0$ (zero or more edges). The reversibility of node i is r_i , i.e., the number of edges that will be reversed by i towards each of its neighboring nodes, indiscriminately, at the end of operation. Node i is a sink if it has, at least, r_i edges directed to itself from each of its neighbors. Each sink i operate and reverse its r_i edges towards its neighbors at a definite cycle, and so, a new definite set of sinks will operate, and after that, reverse its edges forming a cycle of operation. With different number of edges, i.e., different number of r_i 's, each node can operate more than once consecutively.

Lemma (1), shown in França (1994), states a basic topologic constraint towards the definition of M , where gcd is defined as the greatest common divisor and f_{ij} is the sum of the greatest multiple of $\text{gcd}(r_i, r_j)$ that does not exceed the number of shared resources oriented from n_i to n_j , and from n_j to n_i , respectively in the initial orientation of this graph (Yang, 1999).

Lemma 1. (França, 1994) Let nodes i and j be two neighbors in M . If no deadlock arises for any initial orientation of the shared resources between i and j , then $\max\{r_i, r_j\} \leq e_{ij} \leq r_i + r_j - 1$ and $f_{ij} = r_i + r_j - \text{gcd}(r_i, r_j)$.

Finally, it is important to know that there is always at least one SMER solution for any target system's topology, which has arbitrary pre-specified reversibilities at any of its nodes (Barbosa et al., 1996), and specification of these parameters is crucial for the solution.

4. Oscillatory neural network implementation of SMER

It is a common technique the use of nonlinear oscillators to approximate biological characteristics such as digestion, breathing and the most important for robotic purposes locomotion. Three coupled Van der Pol oscillators were used by Dutra (2001) to mimic the human biped way of walking, Collins et al. (1993) studied pattern formation in six coupled nonlinear oscillators, each one as a leg of an hexapod, and a general model of coupled nonlinear oscillators was proposed by Golubitsky et al. (1998) to reproduce the gaits of $2n$ legged animals, such as: quadrupeds, hexapods, centipedes, etc.

Yang (1999) proposed a novel oscillatory mechanism named OBB, oscillatory building block, which is connected in a network for an implementation of SMER as an asymmetric Hopfield network, and this network is capable to reproduce the phase relations of animals gaits.

4.1. OBB characteristics

Being M^{ij} a sub-multigraph of $M(N, E)$ representing a single pair of neighboring nodes n_i and n_j in $M(N, E)$, with r_i and r_j as their respective reversibilities. Let e_{ij} be the number of shared resources between the two neighboring nodes, the arrangement of shared resources must obey Lemma 1, i.e., $e_{ij} = r_i + r_j - \text{gcd}(r_i, r_j)$. In this simple, nonlinearly coupled system, it is also assumed that the postsynaptic potential (PSP) of each cell depends on three factors: (i) its former PSP state; (ii) the impact of coupled neighboring cells and; (iii) the negative feedback of this cell towards itself.

Choosing correct values for the system parameters is the main point to achieve correct cyclic behavior. The synapse weights, cell thresholds and scheduling criteria are crucial for a desired pattern formation in a two node model SMER-based OBB network as coupled neurons i and j .

Let $r = \max(r_i, r_j)$ and $r' = f(r)$, where $f(r) = 10$, if $0 < r < 10$ and $f(r) = 10^{n+1}$, if $10^n < r < 10^{n+1}$, n is a natural number. Therefore, as in Yang et al. (1998), cells i and j 's thresholds θ_i , θ_j and their synapse weights w_{ij} , w_{ji} are defined in Eqs. (1)-(4):

$$\theta_i = \frac{\max(r_i, r_j)}{r_i + r_j - \text{gcd}(r_i, r_j)} \quad (1)$$

$$W_{ij} = \frac{\max(r_i, r_j)}{r'} \quad (2)$$

$$\theta_j = \frac{\min(r_i, r_j) - 1}{r_i + r_j - \text{gcd}(r_i, r_j)} \quad (3)$$

$$W_{ji} = \frac{\min(r_i, r_j)}{r'} \quad (4)$$

The SMER-based Hopfield network is, thus, defined by these system parameters and it guarantees the SMER-like behavior of this novel network developed by Yang and França (Yang et al., 1998); details and proofs in Yang (1999).

4.2. OBBS network

A network composed of a set of OBBS can exhibit more complicated rhythmic patterns than simple ones. This characteristic is essential for generating different gait patterns. Therefore, this section presents some fundamentals of this SMER-like neural network, which facilitates the SMER implementation.

Let, once more, $M(N,E)$ be a multigraph with $\|N\| > 2$ (node set N has more than 2 nodes). Thus each node in M that has more than one coupled neighbor is called *macroneuron*, which is composed of the same *clones*; a *clone* is a subset that has the same reversibility and threshold as the *macroneuron* and is the one that effectively couples with another neuron. Each *clone* is a unique neuron in a simple OBB. In Fig. (3), reproduced from Yang (1999), a diagram of *macroneurons* coupled through its clones and the original SMER description.

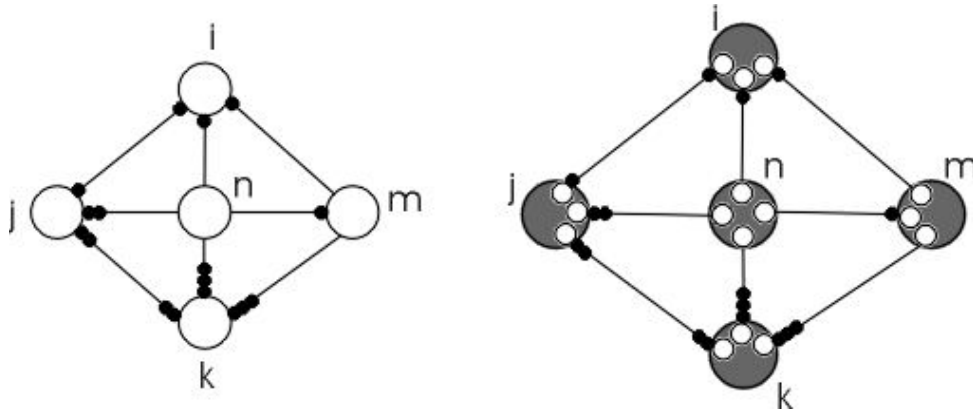


Figure 3. Two equivalent illustrations: Left - the original SMER graph. Right - the coupled network with its clones. Network parameters: $r_i=r_m=r_n=1$, $r_j=2$, $r_k=3$.

The specificity of this network lies on that a *macroneuron* has n clones if it has n coupling *macroneurons*. Then each *clone* of a *macroneuron* is directly in charge of the coupling with the clone of another *macroneuron* to form a simple OBB. Each *macroneuron* will fire only if all its clones fire.

Hence, this novel neural network, the SMER-like Hopfield network, is composed of a membrane potential matrix, which is a collection of states of clones determining the output of each *macroneuron*, a biases matrix, which is the threshold matrix, and a weight matrix responsible for the strengths of couplings between clone neurons of the entire net. The dynamics of the OBB is shown in the next two equations, Eq.(5) and Eq.(6).

$$\begin{cases} M_i(k+1) = M_i(k) + W_{ji}U_j(k) + W_{ii}U_i(k) \\ M_j(k+1) = M_j(k) + W_{ij}U_i(k) + W_{jj}U_j(k) \end{cases} \quad (5)$$

Where,

$$\begin{cases} U_i(k) = \max(0, \text{sgn}(M_i(k) - \theta_i)) \\ U_j(k) = \max(0, \text{sgn}(M_j(k) - \theta_j)) \end{cases} \quad (6)$$

In these equations the terms are: $M_i(k)$ - membrane potential, W_{ij} - coupling strength, θ_j - bias or threshold, $U_i(k)$ - output of neuron i , which the firing rate function is defined as the heavyside type (in Eq.(6)).

The main interesting characteristic of this network is that it works under a local clock pulse, not being necessary a global one. A firing *macroneuron* will release its resources, i.e., revert its edges, only at the end of its operational cycle. This is important in gait research if a step is considered as a process and the driving signals as the shared resources.

5. Hexapodal gaits implemented with OBB nets

The general locomotion model, developed by Golubitsky et al. (1998) for a $2n$ -legged animal, has the topology of a complete graph $M(N,E)$ with $\|N\|=4n$ and $\|E\|=C_{4n}^2$. For an Hexapod, with $2n=6$, there are 12 neurons divided in two layers, which the upper layer is composed by *macroneurons* called flexors and lower one is composed by extensors. The extensors are in charge with the body mass re-scheduling (or the support phase of a step) and flexors are charged with the translation of a leg (or the transfer phase). Fig. (4) shows the topology referred before. For different gaits

there can exist different couplings between the neurons within this unified architecture in the sense that some couplings are kept while the other couplings are blocked. It seems reasonable that the gait transition is controlled by the biological signals from animal's *central nervous system* (CNS) for avoiding risk or obeying animal's willingness. Due to coordination phase relations and duty factors of the locomotion itself the SMER strategy, and so, SMER-like neural network, is presented as a good approach; the phase relations of locomotion of some $2n$ -legged animals can be viewed with more details in some other studies (Collins et al., 1993; Golubitsky et al., 1998; Buono et al., 2000).

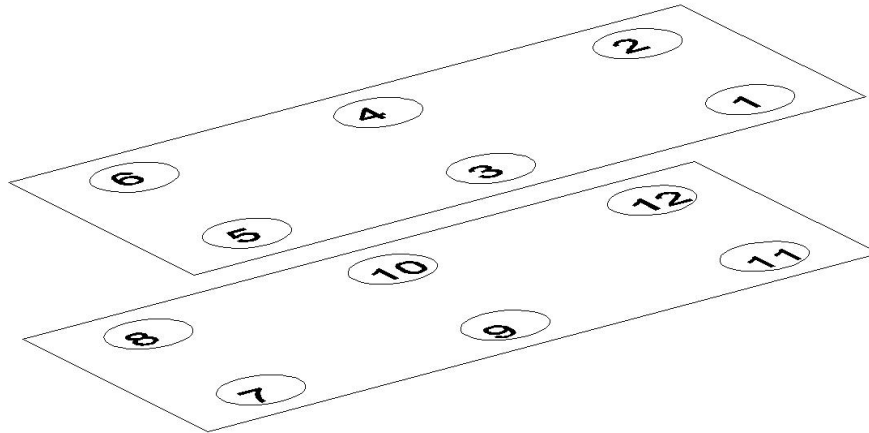


Figure 4. Flexors and extensors of a six-legged robot arranged in two layers.

5.1 Hexapod *Tripod* gait reproduction using SMER strategy

The cyclic behavior of locomotion has been presented, and also, that the SMER strategy is perfect for reproducing these cyclic behaviors. So, in this section, it is introduced some description of this strategy applied to gait reproduction.

As our objective is not to give mathematical descriptions of all three gaits presented here, only the *tripod* gait will be analyzed. Another gait can have its cyclic characteristics described, by its symmetries for example, and can be easily implemented through SMER.

In *tripod* gait the coupled *macroneurons* in the opposite sides and in the opposite surfaces are out of phase, as shown in Fig.(1) and Fig.(6), or in other words: while one is operating all its neighbors are idle. Thus, in order to get the correct phase relation between the coupled neurons for a *tripod* gait, any other out of phase *macroneuron* should be directly coupled. Figure (6) presents the two layers of Fig.(4) arranged in a flat way, which the phase relations between the neurons are easily shown.

Figure 5 presents a scheme of the insect studied here. There are shadows in the picture in order to make easy to see which legs are in contact with the ground. On the model the yellow spheres are the joints, the blue rectangle is the body and the segments linked by the joints are the legs (all the three segments together, as the sketch suggests). The legs are colored in different ways if they are in different states. Red ones are in the same state in transfer phase, they are off the ground, and are not support any weight. Their task is only to place the tip of the leg in a forward position. In this phase, transfer phase, the flexor *macroneuron* of this leg is in its operational cycle and the extensor is idle. Like the legs before the other three legs behave equally, or are in the same state (all three in phase), and their task is to support the robot's weight and translate the entire body (support phase). This cycle of operation is performed by extensors, so, for these legs, the extensor *macroneuron* of each leg is in its operational cycle and its counterpart (the flexor) is idle.

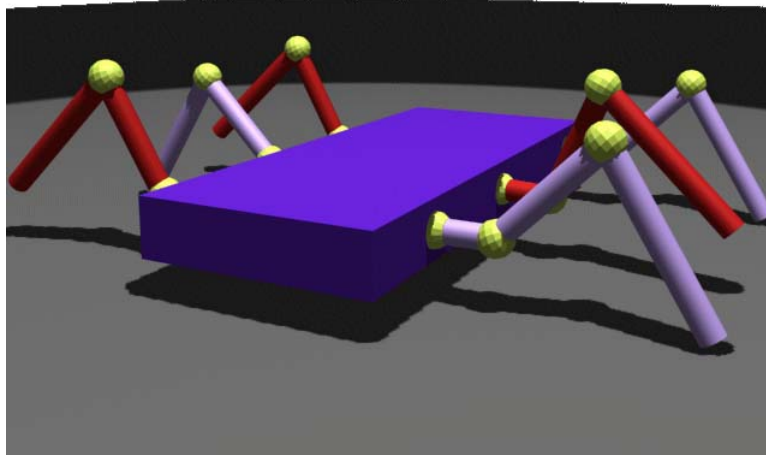


Figure 5. The scheme of an hexapod.

Each pair of flexor and extensor forms a stable oscillatory system and each neuron of this pair is modeled as the biological group of neurons responsible for the driven signals that are sent to muscle legs and are responsible for the leg’s movements. So, flexor is responsible for the muscles that are in charge with the movement of lifting the leg and extensor is the driver of the muscles that put the leg in contact with the ground, support weight and moves the leg backwards (and so the body forwards). Each pair of upper flexor and lower extensor of Fig.(4), e.g., 5 and 7, is responsible for the leg locomotion and this pair is in the internal leg structure.

5.2 Mathematical description of tripod gait

Here, the mathematical description and computation of this gait in the SMER sense are shown. The sketch shown in Fig.(6) is the first step. From the figure, it can be seen the phase relation between the neurons, this is crucial for deciding which ones must be coupled.

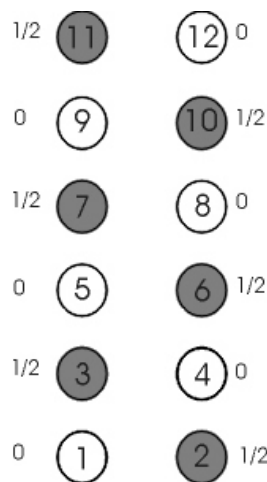


Figure 6. Phase relation between each neuron. Neurons numbered from 1 to 6 are flexors and from 7 to 12 are extensors.

The neurons that must be coupled (i.e., all out of phase ones) are seen in Fig.(6). The ones which are exactly in phase have the same color and are: gray - {2, 3, 6, 7, 10, 11} and white - {1, 4, 5, 8, 9 12}. Therefore, the coupled network can be achieved just connecting the two groups, all gray neurons have to be coupled to white ones. Once, the connections are determined, the values of coupling strengths, thresholds and membrane potentials are the next step. The membrane potentials (or PSP) is the most important because it has to be consistent with the SMER dynamics. Two coupled neurons cannot be active at the beginning of the cycle. Concerning these statements, there may be a convergence delay at the beginning of operation until a desired periodic state is achieved, since the PSP values are randomly chosen and carried out through SMER scope.

The matrixes of membrane potentials, biases and coupling strengths are presented in the next equations (Eq.(7) up to Eq(9)).

$$M_{ij}(0) = \begin{pmatrix} 0 & 0.04 & 0.04 & 0 & 0 & 0.04 & 0.04 & 0 & 0 & 0.04 & 0.04 & 0 \\ 0.96 & 0 & 0 & -0.04 & -0.04 & 0 & 0 & -0.04 & -0.04 & 0 & 0 & -0.04 \\ 0.96 & 0 & 0 & -0.04 & -0.04 & 0 & 0 & -0.04 & -0.04 & 0 & 0 & -0.04 \\ 0 & 1.04 & 1.04 & 0 & 0 & 0.04 & 0.04 & 0 & 0 & 0.04 & 0.04 & 0 \\ 0 & 1.04 & 1.04 & 0 & 0 & 0.04 & 0.04 & 0 & 0 & 0.04 & 0.04 & 0 \\ 0.96 & 0 & 0 & 0.96 & 0.96 & 0 & 0 & -0.04 & -0.04 & 0 & 0 & -0.04 \\ 0.96 & 0 & 0 & 0.96 & 0.96 & 0 & 0 & -0.04 & -0.04 & 0 & 0 & -0.04 \\ 0 & 1.04 & 1.04 & 0 & 0 & 1.04 & 1.04 & 0 & 0 & 0.04 & 0.04 & 0 \\ 0 & 1.04 & 1.04 & 0 & 0 & 1.04 & 1.04 & 0 & 0 & 0.04 & 0.04 & 0 \\ 0.96 & 0 & 0 & 0.96 & 0.96 & 0 & 0 & 0.96 & 0.96 & 0 & 0 & -0.04 \\ 0.96 & 0 & 0 & 0.96 & 0.96 & 0 & 0 & 0.96 & 0.96 & 0 & 0 & -0.04 \\ 0 & 1.04 & 1.04 & 0 & 0 & 1.04 & 1.04 & 0 & 0 & 1.04 & 1.04 & 0 \end{pmatrix} \quad (7)$$

$$\theta_{ij} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix} \quad (8)$$

$$W_{ij} = \begin{pmatrix} -0.1 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 \\ 0.1 & -0.1 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 \\ 0.1 & 0 & -0.1 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 \\ 0 & 0.1 & 0.1 & -0.1 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 \\ 0 & 0.1 & 0.1 & 0 & -0.1 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 \\ 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 \\ 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & -0.1 & 0.1 & 0.1 & 0 & 0 & 0.1 \\ 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & -0.1 & 0 & 0.1 & 0.1 & 0 \\ 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & -0.1 & 0.1 & 0.1 & 0 \\ 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0 & -0.1 & 0 & 0.1 \\ 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & -0.1 & 0.1 \\ 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & 0 & 0 & 0.1 & 0.1 & -0.1 \end{pmatrix} \quad (9)$$

The membrane potentials in Eq.(7) are not randomly chosen, they were chosen, and used by, in a C++ program developed for simulate this gait. It allows the neurons fire in this cycle: (2-3-6-7-10-11) (1-4-5-8-9-12).

Each row and column of the matrixes can be seen as elements of the multigraph. Each element in the weight matrix (the coupling strengths matrix) is about the i^{th} neuron connected to the j^{th} neuron and the elements of the main diagonal are interpreted as the self-feedback connection, which is related as the amount of reverted resources after its operational cycle. The elements in the biases matrix can be seen as the activation threshold of each clone that is contained in the macroneuron. In the PSP matrix (membrane potential matrix) each element of a i^{th} row is a shared resource that is owned by the i^{th} neuron and is connected with j^{th} neuron.

Therefore, the output of each *marconeuron* is characterized by the activation of each of its clones. If the j^{th} neuron is active then all its *clones* are active. A *clone* will only be active only if the i^{th} element of the PSP matrix, that represents the ownership of the resources shared with the i^{th} neuron of the j^{th} column is greater than the correspondent θ_{ij} . So, for a *macroneuron* be active all the elements in its representative column of PSP matrix have to be greater than the elements of biases matrix.

The main advantage of SMER is that an edge will only be reversed, and so, a leg will change from support to transfer phase (or the opposite – from transfer to support), if the leg is placed in a desired position. Sensors of position and force are needed. The signals only are sent if the leg is moving correctly. The SMER interacts with the sensory feedback to achieve the correct synchronization.

In the matrixes used for simulation of the gait, which are presented in Eq.(7) up to Eq.(9), the *macroneurons* that are active at the beginning of the cycle are: 2-3-6-7-10-11.

Figure 7 presents the simulation results obtained utilizing the matrixes above. Only the activation of the upper layer neurons, flexors, are present. This simulation was done in a digital computer and the output of each neuron is the binary value: 1 or 0. This digital implementation was used for computer simulation of walking and the analogic neural network implementation of SMER, as demonstrated in Yang (1999), can present a more complex behavior and preserve the movement even though small perturbations are present.

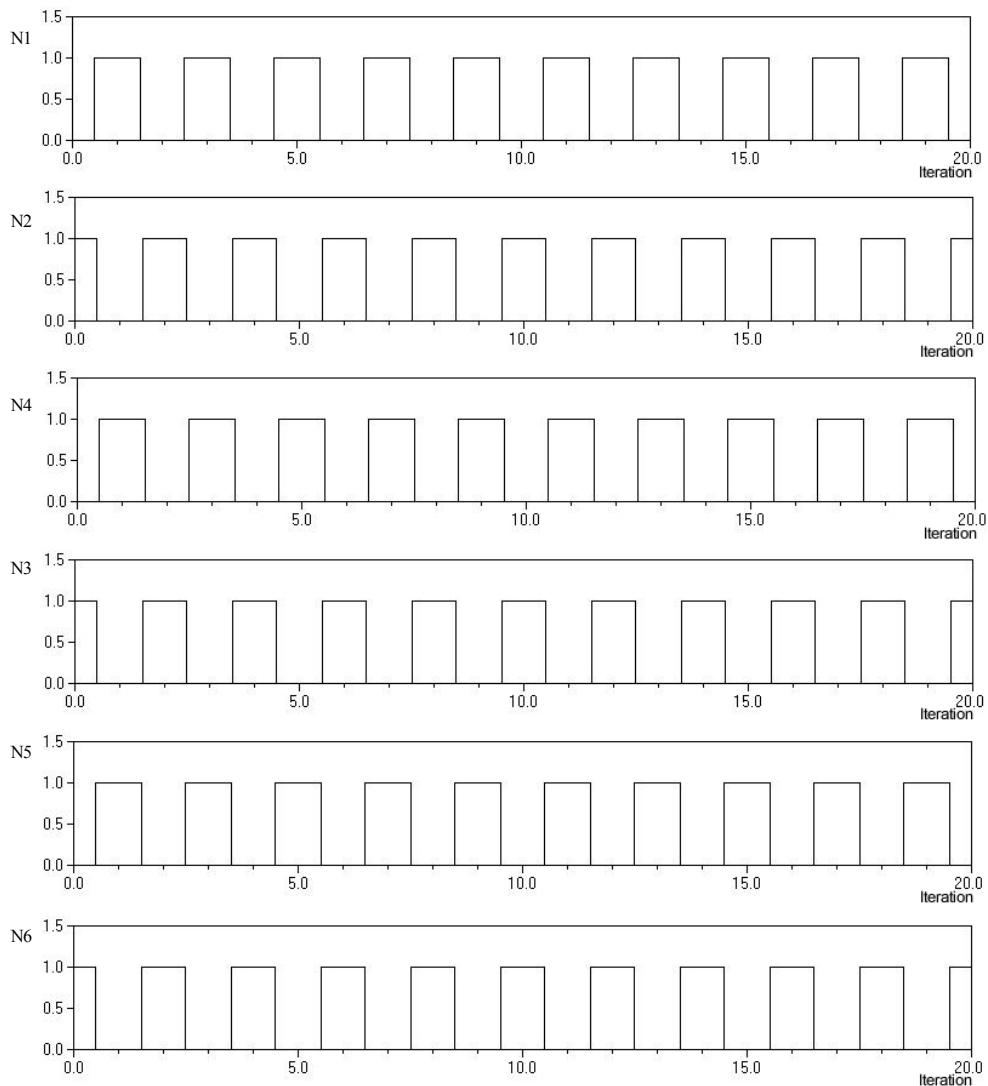


Figure 7: Digital simulation results.

The results shown in fig.(7) are coherent with the sketch of fig.(6). The obtained synchronization, allows the legs in support phase and in transfer phase to be correctly out of phase.

6. Conclusion

A novel approach for the synchronization of the legs of an hexapod robot has been presented together with mathematical description of the method. An hexapod's gait pattern has been easily reproduced through the SMER showing some of its capabilities. The methodology for creation of this gait can be applied for reproducing any other gait in the robotics scope.

The SMER algorithm is shown as a simple but powerful strategy for reproducing any cyclic behavior determined in advance. Such ability is good for reproducing the hexapod's, and any other legged mammal or insect, rhythmic behavior for robotic implementation. Even the locomotion of snakes could be reproduced if the supposition of waves traveling along its body is made, just assuming that each segment plays the role of the *macroneurons*.

The asymmetric Hopfield-like neural network based on SMER can be configured for most complex rhythmic patterns independently from their biological neuronal models, just being concerned with the phase relations of the locomotion itself.

Therefore, the control of the hexapod's legs can be done utilizing sensors for feedback information about the leg's state, and SMER as the synchronization method. In this control, each leg is responsible for its own movement, and also, for permitting other legs to move, reaching the desired gait.

7. Acknowledgements

We thank the LabRob group and Felipe M. G. França for helpful discussions. This research was supported by FAPERJ, grant: E26/150.971/2000.

8. References

- Amendt, O., 1995, "Neuronale Steuerung Eines Insektenartigen Schreitroboters". Dr-Ing. Dissertation, VDI Fortschrittberichte Reihe 8. Gerhard-Mercator-Universität Duisburg, VDI Verlag GmbH, Düsseldorf, Germany.
- Barbosa, V.C., Benevides, M.R.F. and França, F.M.G., 1996, "Sharing Resources at Nonuniform Access Rates", Technical Report COPPE/UFRJ, ES-412/96, Rio de Janeiro, Brazil, November.
- Buono, P.L., Golubitsky, M., 2000, "Models of Central Pattern Generators for Quadrupedal Locomotion: I. Primary Gaits", J. Math. Biol., Preprint.
- Collins, J.J. and Stewart, I., 1993, "Hexapodal Gaits and Coupled Nonlinear Oscillator Models", Biol. Cybern., Vol. 68, pp. 287-298.
- Collins, J.J. and Stewart, I., 1994, "A Group-theoretic Approach to Rings of Coupled Biological Oscillators", Biol. Cybern., Vol. 71, pp. 95-103.
- Costa Netto, S. M. da, Evsukoff, A., Dutra, M. S., 2000, "Fuzzy Systems to Solve Inverse Kinematics Problem in Robots Control: Application to an Hexapod Robot's Leg", Proceedings of the 8^o SBRN, v.1, pp. 150-155, Rio de Janeiro, Nov.
- Costa Netto, S. M. da, 2001, "Estudo e Projeto de um Robô Hexápode com Atuadores Utilizando Materiais com Memória de Forma", M.Sc. Thesis, COPPE/UFRJ, Rio de Janeiro, RJ, Brazil.
- Dutra, M. S., Pina Filho, 2001, A. C., "Modeling a Bipedal Locomotor Using Coupled Nonlinear Oscillators of Van der Pol", Biological Cybernetics, to appear.
- Dutra, M. S., 1997, "Kinematical Analysis of a Six-Legged Walking Machine", Anais do Diname 97, Angra dos Reis, Rio de Janeiro, Brazil, March.
- França, F.M.G., 1994, "Neural Networks as Neighborhood-Constrained Systems", PhD. Thesis, Imperial College, London, England.
- Golubitsky M., Stewart I., Buono P.L. and Collins J.J., 1998, "A Modular Network for Legged Locomotion", Physica D., Vol. 115, pp. 56-72.
- Haykin, S., 1994, "Neural Networks: A Comprehensive Foundation", 1 Ed., Ontario, IEEE Press.
- Oliveira, A. M., 1999, "Cinematica de um Robô Hexápode", M.Sc. Thesis, COPPE/UFRJ, Rio de Janeiro, RJ, Brazil.
- Raptopoulos, L. S. C., 1998, "Estudo de um Robô Hexápode", Mechanical Engineering Bach. Thesis, DEM/UFRJ, Rio de Janeiro, RJ, Brazil.
- Yang, Z.J. and França, F.M.G., 2000, "Building Artificial CPGs With Asymmetric Hopfield Networks", IEEE/INNS/ENNS - Proc. of Intl. Joint Conf. on Neural Networks 2000, Vol. IV, pp. 290-295, Italy, Jul.
- Yang, Z.J., 1999, "Produzindo Padrões Arbitrários: Características Biológicas em Sistemas Neurolocomotores Artificiais", D.Sc. Thesis, COPPE/UFRJ, Rio de Janeiro, RJ, Brazil.

MONOCULAR VISION SYSTEM FOR AUTOMATIC GUIDANCE OF A WELDING ROBOT

Sadek C. Absi Alfaro

Automation and Control Group (GRACO), Mechanical Engineering Department, FT, University of Brasilia, DF
sadek@unb.br

Milton L. Siqueira

Automation and Control Group (GRACO), Mechanical Engineering Department, FT, University of Brasilia, DF

José M. S. T. da Motta

Automation and Control Group (GRACO), Mechanical Engineering Department, FT, University of Brasilia, DF

Abstract. *In automated production lines using welding processes, robots are used to move the welding torch. But for the robot to weld it is normally necessary that it has been previously programmed to follow a predetermined trajectory. If the location of the pieces to be welded changes for any reason, all the programming will have to be redone. One way to eliminate the robot programming is to use a vision system attached to the robot, which captures images from the area to be welded, interprets them in a computer, detecting the joint line position, and then positions automatically the robot to follow the trajectory to be welded. The system developed here consists basically of a CCD video camera and a computer. Its algorithm analyses images in the simplest way possible, in other words, exclusively through a camera attached to the robot arm, without using any additional devices to "enhance" them, say, with the help of laser or filters. The joint line position is detected using a mix of comb scanning and circular searching. Thus we can establish the welding trajectory to be followed by the robot. The system was implemented in an industrial robot ABB-IRB 2000. The results show that it is feasible to automate welding cells, with no robot programming.*

Keywords. *Vision System, Welding Robot.*

1. Introduction

In industrial automation, the use of sensors is a key for the acquisition of parameters and information, which can be utilized in the control of the manufacturing process. In the case of automation with robots, sensors can measure internal parameters, like joint angles, but also external parameters like torch position. The attachment of sensors to a robot expands its capability, reducing its dependence on other machines or people (Groover, 1986).

In automated production lines by welding, robots are used to move the welding torch. However, for the robot to weld, it is normally necessary that it has been previously programmed to follow a predetermined trajectory. If the location of the pieces to be welded changes for any reason, all the programming will have to be redone. This happens because robots are usually unable to see where the pieces to be welded are, and are unable to detect the shifting of joint line trajectory (Araya, 1991).

By the installation of a vision system, the robot can handle this problem, and so becomes more versatile. By doing so, the robot programming is not necessary. This article describes a monocular vision system for automatic off-line guidance of welding robots. The system is able to acquire and process images in the simplest way possible, just using one monochromatic CCD video camera, without use of any extra devices, such as lasers, special illumination and/or filters. It detects the absolute position of the joint line, identifying the trajectory to be welded. With the trajectory identified, the system acts in the robot's controller, moving it to the start point, initiating the welding arc, and controlling its displacement along the full length of the trajectory, depositing the weld bead at the same time.

By using laser, it is also needed to use filters. The filters depend on welding parameters and chosen filters become hard for easy industrial application. The technique present in this paper, remove the laser and filter and once the robot "learn" the welding path (off-line operation), all the vision system is removed in the actual welding, making ease the process. One drawback of this technique is that special attention is needed in the welding joint setup and clamping.

2. The System

The vision system can be described from the illustration shown in Fig. 1. Successive images of the work area are acquired by a CCD monochromatic video camera and sent to the frame grabber (in the computer) as a compact video signal. In the PC, with the help of the developed search algorithm, it is possible to detect the joint path configuration, thus identifying the trajectory to be welded (Alfaro, 1997). The coordinating data of the trajectory are sent to the robot's controller, using serial communication between PC and controller, which controls the displacement of the robot arm, moving it (and the welding torch attached to it) along the joint trajectory. The system presented here was implemented in ABB IRB-2000-S3 industrial robot.

2.1. Data Acquisition Step

The acquisition is the step where data necessary for the robot guidance are collected and stored in frame-grabber's memory. These data are images of the work area (plates to be welded). After the capture of the images, they are sent to the PC. The method utilized for data acquisition involves a monochromatic CCD camera, which captures images, and sent to the frame grabber (in the computer) as a compact video signal. In the PC, with the help of the developed search algorithm, it is possible to detect the joint path configuration, thus identifying the trajectory to be welded. The coordinating data of the trajectory are sent to the robot's controller, using serial communication between PC and controller, which controls the displacement of the robot arm, moving it (and the welding torch attached to it) along the joint trajectory. The system presented here was implemented in ABB IRB-2000-S3 industrial robot.

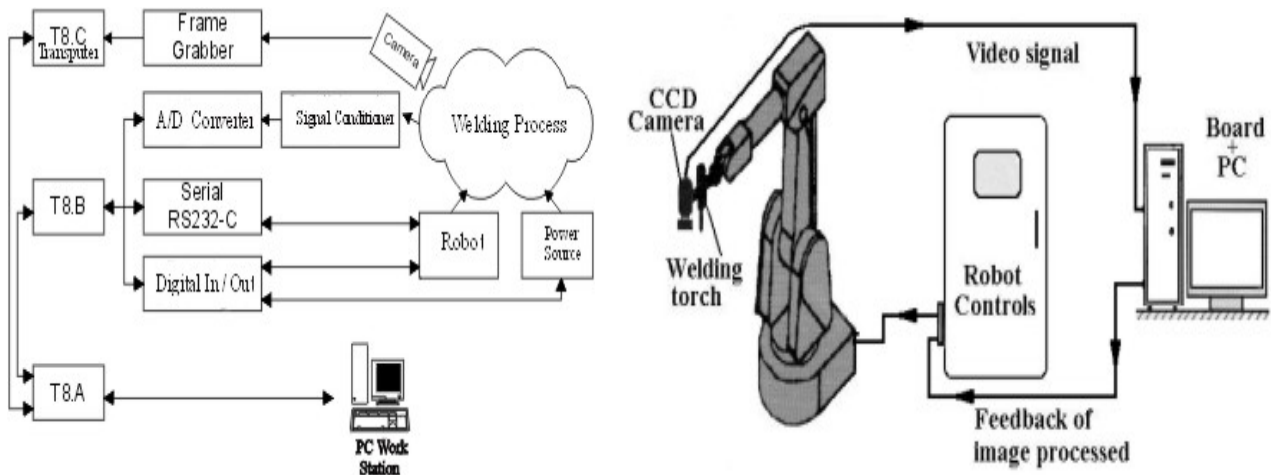


Figure 1. Schematic illustration of the vision system.

The frame-grabber, 16MB memory, 16 bit resolution, digitised the captured images. The camera is attached to the robot's arm, in an "eye-in-hand" configuration (Fig. 2). The images are digitized in grey level and stored in the frame-grabber's memory. Fig. 2 shows the CCD camera, high resolution 725 (H) x 582 (V), 625 lines (1), the weld torch (2) and support (3) used to attach the camera and torch to the robot arm (4). This figure also shows the two pieces (plates) and the trajectory to be welded (5) (Gomide, 1997).



Figure 2. Set up for data acquisition.

In Fig. 3 is shown a captured image, wherein can be seen the plates and the joint configuration. The images were acquired frame by frame by activating the "search algorithm" as described in the following Analysis step.

2.2. Analysis Step

After one image has been captured, it is necessary to analyse it to obtain the data to be used in the subsequent guidance. This step takes place in the PC, where an algorithm for the joint path detection in one image was developed. The algorithm reads image by image from the frame-grabber memory. The output data are several absolute points (x, y) that describe the trajectory of the joint path to be welded in work plane of each image.

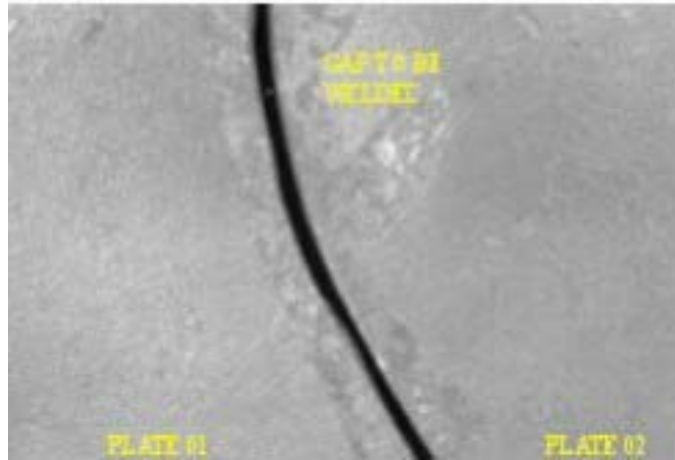


Figure 3. Captured image showing the work area.

The images are analysed in grey level, where each pixel has a value between 0 (black) and 255 (white). The experience shows that the pixels in the joint path area have a less intense level than the other regions, i.e., their colour is closer to the black. By the establishment of grey level threshold, the joint path configuration can be detected. Another important characteristic of the joint configuration is the common length, L , of the gap between two plates. Thus, in the image, one joint path can be considered to have " L " pixels (or more) in sequence, if they have a grey level lower than the threshold (Bauchspiess, 1997).

The algorithm works in the following way. It initiates the run from the bottom of the picture (one end of the joint line, from left to right, analysing pixels one by one, until it finds a point on the joint path (small area with a grey level lower than the threshold). The joint path has a characteristic number of pixels in sequence (length of the gap). The medium point, which corresponds to the centre of the gap, is stored. As the joint path is continuous, it's not necessary to search it from the beginning of the picture, like it happened in the first run. In the next run, the starting point jumps from the centre point to a certain distance Δx and forward in y direction (Fig. 4), where it is still closer to the joint path to reduce the processing time. Another search begins from this point until it finds the joint path again. Successive runs are achieved in this way and the same is shown in Fig. 4. It can be observed that the successive runs left traces in the acquired image, making "triangles" along the joint path. This method is a mix of comb scanning and circular searching, and it is called "triangular scan" (Soares, 1997).



Figure 4. Algorithm of "triangular scan"

For each analysed image, a certain number of centre points (of the path) were obtained. This number can be changed, depending of the accuracy required. To map the full length of the joint path, successive images must be ac-

quired. After one image has been processed, another one is acquired, so that the beginning of the newer one is the last centre point obtained in the previous picture. Consequently, data from one analysis has to be sent to the robot, controlling and moving it to right position for the acquisition of the next image. After the acquisition and analysis of each image, one loop is concluded. With consecutive loops (Fig. 5), full length of the joint path trajectory will be mapped and identified (Schilling, 1990).

2.3. Welding Step

In this step, the data about the mapped trajectory is sent to the robot’s controller. It can be handled by serial communication between PC and robot. The robot is moved to the beginning of the trajectory previously identified, followed by the initiation of the welding arc, and so controlling its displacement along this trajectory, depositing the weld bead. This process is called an off-line method, i.e., the welding is done only after the trajectory has been identified. It’s important to note that the welding parameters, like current and feed rate, for example, can also be set by serial communication (Norrish, 1996 and Zhang et al., 1993).

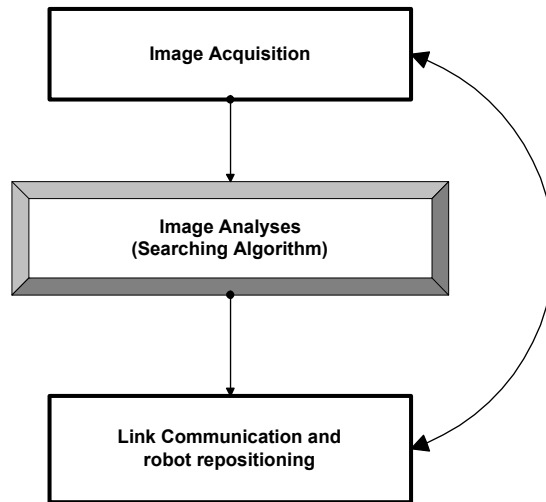


Figure 5. Loops for detection of the entire trajectory

3. Results

Figure 6 shows an image acquired and processed by the algorithm of search of the vision system. After its analysis, the system plots, on the screen, a curve that represents the trajectory previously identified in that image. The curve for the scanned path can be seen in Fig. 6, with the centre of the gap of the joint to be welded clearly identified. This is for only one image, i.e., just a portion of the joint length.

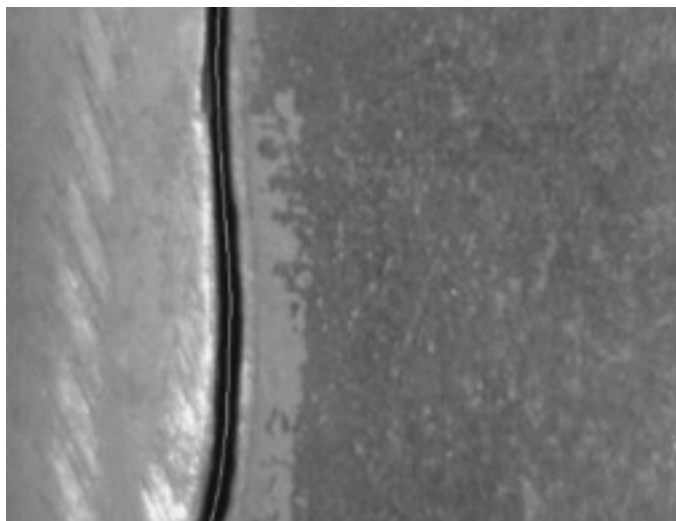


Figure 6. Trajectory identified for one image

In sequence, new images are captured and analysed, generating the data that depicts the complete trajectory to be welded. Figure 7 presents the real trajectory (photograph of the two plates and the gap between them) and the trajectory mapped by the vision system. The modelled trajectory describes correctly the real one. This was further confirmed by several other tests carried out on other trajectories of different configurations. In all these tests the welding arc followed, without any deviations, the route set by initial image processing of the joint path.

Using “triangular scan” method on a common PC, the time required to conclude the acquisition and analysis of one image, extracted for a trajectory, is about 0.50 s. Considering that the image has approximately 30 mm of length, the robot’s rate displacement is about 60 mm/s when detecting the trajectory in off-line way. Just after that the actual welding follows.

4. Conclusion

The vision system described in this article can automatically identify joint configuration to be welded in images acquired only by the use of one CCD camera, with no extra devices. The system is also able to communicate with the robot’s controls, moving it successfully along the trajectory previously scanned.

This method utilized to search for the joint path configuration, called “triangular scan”, was efficient and robust enough to detect the joint gap between two plates, even if in bad situations, such as poor local illumination and oxidized (or too dirty) plates.

The use of the vision system brings a reduction of work and time for the welding process. It is not necessarily a “hard” procedure to place the plates, to be welded, in the right position, and no programming of the robot is needed. The system detects automatically the trajectory and updates the robot’s position in order to follow it.

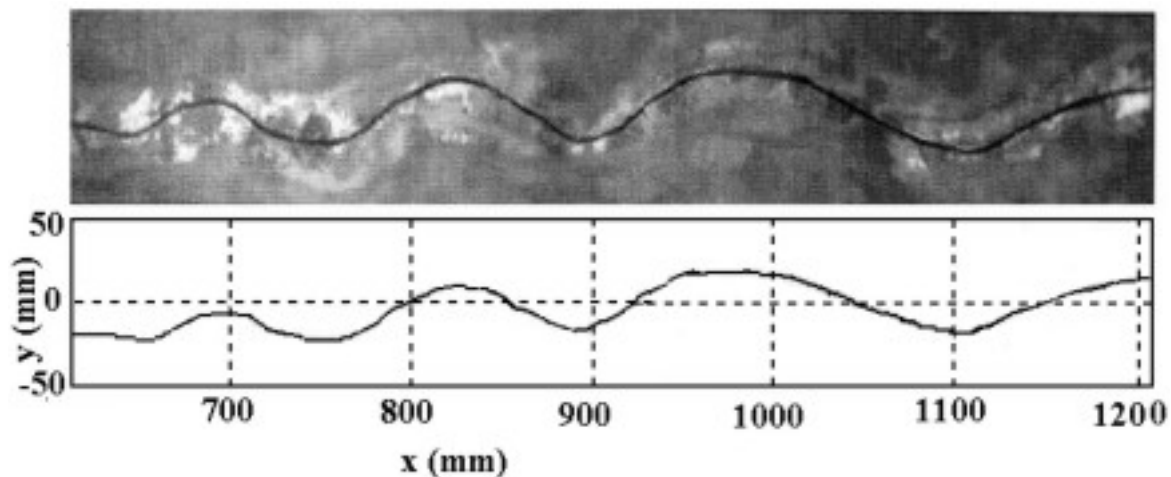


Figure 7. (a) Real trajectory and (b) the identified and mapped trajectory.

5. Acknowledgement

The authors would like to thank the following researching bureaus: FINEP, RHAЕ, CNPq and FAP-DF for the support to the GRACO in this and others activities.

6. References

- Alfaro, S. A., Soares Jr., L.R., et al., 1997, “Conception and development of an integrated welding cell”, *Eighth International Conference on the Joining of Materials*, Helsingor. pp. 14-17. Denmark, May.
- Araya, T., 1991, “Sensing and Adaptive Control of Arc Welding”, *Proceedings of Inst. Mechatronics & Robotics*, pp. 36-41. Amsterdam.
- Bauchspiess, A., Alfaro, S.A., Trier, W. et al., 1997, “Sistema de visão computacional inteligente para robôs de soldagem”, *Anais do III Simpósio Brasileiro de Automação Inteligente*, Vitória-ES.
- Gomide, R.B., Guillén, L.L., 1997, “Sistema de visão mono-ocular para guiagem automática de robôs de soldagem”, *Projeto Final*, Depto. de Engenharia Mecânica, Universidade de Brasília.
- Groover, M. P.; Weiss, M.; Nagel, R. N. and Odrey, N. G. *Industrial Robotics: Technology, Programming and Applications*. McGraw-Hill Book Co., NY, 1986.
- Norrish, J. *Advanced Welding Process*. Institute of Physics Publishing Ltd, UK, 1992.
- Schilling, R. *Fundamentals of Robotics: Analysis and Control*. Prentice-Hall Inc., N.J., 1990.
- Soares Jr., L.R., 1997, “Implementação de um sistema de visão por computador para guiagem de um manipulador robótico”, *Relatório de Atividades/CNPq, Grupo de Automação e Controle*, Universidade de Brasília.

Zhang, Y.M; R. Kovacevic, R. & Wu, L Closed-loop control of weld penetration using front-face vision sensing. Proceedings of the Institute of Mechanical Engineers, Part B, Vol.207 (B3), 1993, pp. 27-34.

MODELAGEM MECÂNICA DO MÓDULO DE LOCOMOÇÃO DE UM VEÍCULO INSTRUMENTADO PARA INSPEÇÃO EM DUTOS

Carlos Eduardo Guedes Catunda

Max Suell Dutra

Cesar Gomes Ferreira

Universidade Federal do Rio de Janeiro, COPPE – Programa de Engenharia Mecânica.

C.P. 68503 - CEP. 21945-970, Rio de Janeiro, RJ, Brasil.

Resumo. Um dos meios utilizados para a manutenção e inspeção de dutos, atualmente, é a utilização de PIG. Existem casos em que este sistema não pode ser utilizado, como é o do chamado duto “não pigável”. O duto “não pigável” é aquele que oferece riscos de aprisionamento de um PIG. Frente a esta situação, é proposto o robô VERME (Veículo de Exploração Robotizada com Movimentação Eletromecânica). Este veículo será utilizado para a fiscalização de uma malha de dutos, propiciando uma alternativa viável para a redução dos altíssimos custos de inspeção e manutenção de dutos. Inspirado no estudo da Zoologia dos Invertebrados, mais especificamente a dos vermes anelídeos, surgiu a proposta do veículo em questão. Os anelídeos são morfológicamente caracterizados pelo corpo segmentado em anéis (metamerismo). O veículo proposto reproduz um sistema semelhante ao utilizado na locomoção destes seres. Este veículo possui a sua estrutura física dividida em módulos independentes (anéis) interligados por atuadores mecânicos. Seu sistema de locomoção consiste em um mecanismo eletromecânico que, ao entrar em contato com as paredes do duto de aço, fixam o módulo correspondente e possibilita a movimentação do módulo posterior para a orientação desejada.

Palavras-chave: PIG Instrumentado, Sistema Multi-corpos, Mecatrônica, Dutos.

1. Introdução

Nos últimos anos, nossas cidades têm sofrido com um intenso derramamento de resíduos tóxicos nos rios, lagos e praias; desta forma, envenenando o meio ambiente que nos cerca. Já é mais do que notória a gravidade de tal conjuntura. Diante de tal circunstância, torna-se necessária a vistoria e a identificação de todos os pontos fonte de derramamento de dejetos poluidores nas vias hídricas.

O objetivo deste projeto é, verificar a viabilidade do emprego deste veículo na identificação de inúmeros problemas internos nos dutos que possam vir a causar falhas em sua operação ou riscos no que diz respeito à poluição ambiental em comparação com a atual utilização de PIGs Instrumentados. A origem do termo PIG possui algumas versões, sendo as mais aceitas as que relacionam o dispositivo ao animal de mesmo nome, tanto pela semelhança entre o grunhido dos porcos e o som emitido pelo dispositivo ao se deslocar na linha, como pela estreita relação entre PIG e detritos, tal qual os suínos [1]. O sistema de PIG é uma técnica que consiste no lançamento de qualquer corpo monitorado no interior de tubulações. A natureza destes corpos é de grande variedade e dependendo da técnica utilizada existem alguns exemplos: PIGs líquidos, gel, espuma, macro-esferas, infláveis, instrumentados, etc. Cada tipo é responsável pela realização de uma tarefa específica no interior de uma tubulação assim, o trabalho requerido determina o tipo de PIG a ser utilizado [2].

O sistema de PIG é usado em um grande número de aplicações industriais, sob inúmeras condições. Algumas aplicações envolvem condições severas de operação, exigindo das tubulações uma alta resistência a temperaturas e pressões elevadas. Os sistemas de PIGs podem ser utilizados em tubulações subterrâneas, subaquáticas e elevadas. Nestas aplicações, este sempre constitui um componente de suma importância para assegurar o pleno funcionamento e a integridade do sistema.

A inspeção visual por um sistema de PIG Instrumentado pode ter grandes vantagens, tais como: aumento da segurança, redução de custos e precisão da inspeção. Atualmente, sistemas de inspeção guiados manualmente são usados com mais frequência para realização da inspeção visual. A desvantagem é a de que, por depender de um operador, está sujeito a erros no seu posicionamento.

O veículo proposto é um tipo de PIG Instrumentado que pode ser capaz de realizar os mais variados ensaios não destrutivos no interior das tubulações. São três os ensaios mais comuns. Primeiramente, a leitura da variação do fluxo magnético para verificação da espessura de parede da tubulação. Em segundo lugar, a aplicação das diversas técnicas ultra-sônicas tais como a *Tandem*, *Delta* e *TOFD*, para detecção e dimensionamento de trincas internas, micro-inclusões, falhas em soldas radiais e longitudinais, e corrosão. E por último, a utilização do método de correntes parasitas para detecção de qualquer defeito sub-superficial em tubulações.

Este artigo descreve os aspectos de projeto que estão atualmente sob testes e avaliações. A maior ênfase está sendo dada na configuração global do sistema, e na abordagem que concerne a transposição de obstáculos internos aos dutos.

2. Áreas de Utilização do Veículo Proposto

Os tópicos que se seguem relatam algumas possíveis áreas de utilização do veículo proposto; tanto na fase de pré-operação dos dutos (montagem), quanto na fase operacional (inspeção e manutenção).

- Remoção de resíduos durante a montagem e a operação dos dutos;
- Levantamento Geométrico (vista do perfil interno dos dutos);
- Levantamento Topográfico;
- Verificação da espessura das paredes da tubulação;
- Averiguar a existência de corrosão; e
- Inspeção visual interna dos dutos.

As figuras a seguir apresentam os prováveis ambientes operacionais do veículo proposto.



Fig. 1.: Aqueduto em bom estado

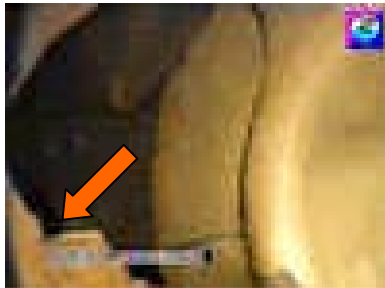


Fig. 2.: Detecção de corrosão

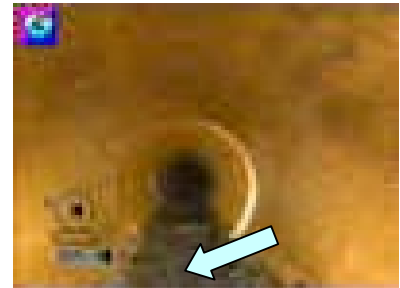


Fig. 3.: Assoreamento

2.1. Fase de Pré-Operação

Durante a montagem dos dutos este tipo de veículo torna-se necessário para detecção de possíveis danos ocorridos durante os vários estágios de montagem. Após o estágio de construção, é comum a existência de corpos estranhos no interior dos dutos (ferramentas, pedras, pontas de eletrodos e resíduos em geral). Podemos citar como danos mais comuns a obstrução de filtros, quebra de impelidores de bombas e travamento de válvulas [3]. A remoção destes materiais poderia ser realizada pela passagem deste veículo após a conclusão dos diversos trechos de dutos.

2.2. Fase de Operação

2.2.1. Remoção de Resíduos

Na operação de dutos, de maneira geral, ocorre incrustação de resíduos nas paredes internas dos mesmos, que varia de acordo com o produto deslocado e as condições operacionais. A remoção destes resíduos tem como objetivo conservar a eficiência do escoamento e amenizar transientes de pressão. Cada tipo de incrustação exige uma ferramenta específica, como acessório, acoplada ao veículo para sua remoção. As escovas de aço são acessórios que, por sua rigidez, removem incrustações mais duras tais como carbonatos e sulfatos. As facas são acessórios para remoção de depósitos macios tais como a parafina e lama. Os discos são usados para remoção de resíduos líquidos.

2.2.2. Levantamento Geométrico

Quando se torna necessário a verificação da existência de defeitos em um duto e determinar a sua localização exata, podemos utilizar o veículo acoplado com um gabarito que registra a variação do fluxo magnético de acordo com a proximidade da parede da tubulação. Este sistema deve ser utilizado em conjunto com um sistema de localização. A qualidade deste sistema de localização pode ser Magnética, Eletromagnética, Sonora ou Radioativa. O estudo destes diferentes sistemas possibilitará a seleção do mais adequado.

2.2.3. Levantamento Topográfico

Consiste em uma ferramenta para o levantamento da configuração espacial de um duto. Este tipo de ferramenta consiste em um sistema de localização fina. Como citado anteriormente, o estudo dos diferentes sistemas mencionados possibilitará a seleção do mais adequado.

2.2.4. Verificação da Espessura da Parede

Para determinação da perda de espessura da parede de dutos, podemos utilizar uma técnica chamada de fuga de fluxo magnético, capaz de detectar defeitos como trincas e inclusões. O princípio de funcionamento deste método é baseado na indução de um fluxo magnético na parede do duto e, caso haja alguma falha, ocorre uma perturbação de fluxo magnético que é percebida pelo sistema. Existe um outro método possível de ser utilizado; é o chamado método ultra-sônico que realiza a leitura das ondas reverberantes emitidas por um transdutor acoplado ao veículo.

2.2.5. Inspeção Visual Interna dos Dutos

Este sistema permite a confirmação dos dados obtidos nos outros processos de inspeção e manutenção. Pode ser efetuado através da fixação de câmaras de alta resolução no primeiro módulo do veículo.

3. Capacidades em Diferentes Ambientes

Tendo em vista que o veículo trabalhará em um ambiente inóspito, inacessível aos seres humanos, a plataforma operacional do veículo deve ser capaz de transpor obstáculos que possam vir a ser encontrados. Estes obstáculos podem ser desde o desnível entre as conexões dos dutos até o assoreamento causado por detritos acumulados no interior dos mesmos (lama, objetos sólidos, matéria orgânica, superfícies alagadas/escorregadias etc.) [4] [5]. O veículo deve ser capaz de vencer estes obstáculos para efetuar plena operação dentro da malha a ser analisada.

O perfil de uma malha de dutos pode ser tão variável que é impossível assegurar que o veículo em questão será capaz de percorrer todas as configurações possíveis de tubulações. As aplicações do robô VERME estão em diferentes configurações de tubulações, sobre uma grande variedade de condições.

Uma vez que o veículo deriva sua força de locomoção da pressão exercida pelo seu corpo contra as paredes da tubulação associada à aderência eletromagnética, então, não há nenhuma restrição com relação a inclinação do sistema. Uma outra vantagem é o baixo peso do veículo, característica de sua estrutura. Com uma grande capacidade de compressão da tubulação e seu baixo peso, o veículo pode executar suas tarefas de uma forma muito estável mesmo quando operando em instalações com dutos verticais.

3.1. Descrição do Sistema

O ambiente de trabalho do veículo proposto consiste quase que na totalidade por dutos de Aço Carbono, Aço Ligado ou Aço Inoxidável; portanto, nossas suposições e simplificações serão realizadas de acordo com estes modelos de tubulações.

Este modelo de sistema de locomoção é muito semelhante ao dos seres que possuem sua estrutura corporal subdividida em anéis; os chamados anelídeos. Os anelídeos, Fig. (4), são morfologicamente caracterizados pelo corpo segmentado (metamerismo) equivalente aos anéis de compressão da tubulação e o sistema nervoso com um filamento principal ventral correspondente ao atuador de ligação [6].

A modelagem simplificada consiste na análise do sistema de locomoção do veículo. O veículo é composto por módulos interligados por atuadores hidráulicos, Fig. (5). Cada módulo possui um sistema de locomoção. O sistema de locomoção consiste em anéis bipartidos acionados por pistões hidráulicos de encontro à parede da tubulação. Estes anéis possuem pastilhas com um alto coeficiente de atrito em sua área de contato com a tubulação; são semelhantes a um sistema de frenagem, Fig. (6).

O surgimento de uma tensão na superfície interna da tubulação gera a força de atrito necessária para fixar um (ou mais) módulo(s) do veículo na parede do duto e por intermédio do atuador de ligação, impulsionar o(s) módulo(s) seguinte(s) para o sentido desejado.

Certo de poder contribuir para o aumento da tensão superficial, o veículo possui eletroímãs associados aos anéis de compressão da tubulação. Os eletroímãs contribuem para o aumento da aderência às paredes da tubulação, facilitando a movimentação do veículo.

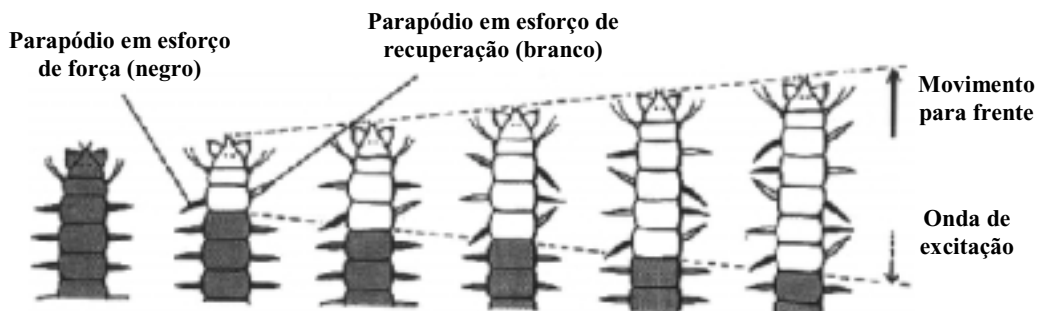


Fig 4.: Esquema de movimentação dos anelídeos

Nota-se que a tensão na parede interna da tubulação formada pelo atuador do interior do anel de compressão juntamente com a força produzida pelos eletroímãs, gera uma força normal F_N na direção radial à tubulação, produzindo assim, a força de atrito F_{at} necessária para fixar o módulo em operação e permitir a movimentação do módulo seguinte.

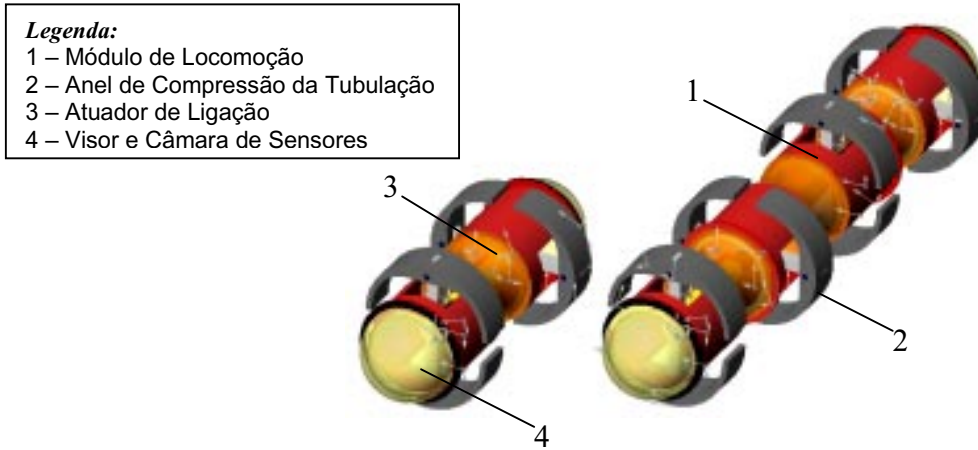


Fig. 5.: Esquema simplificado do veículo com dois e quatro módulos

4. Modelagem dos Módulos de Locomoção

4.1. Conceito de Aderência por Compressão da Tubulação

Tomando-se como base os atuais sistemas de frenagem de eixos, os anéis de compressão da tubulação podem ser modelados. O sistema de freio apresentado na Fig. (6), consiste basicamente de dois elementos: uma superfície de alto atrito (sapata de freio) e um atuador. A superfície de atrito se encarrega da fixação do módulo em operação fazendo uso do atuador para geração de uma força normal que se transformará em força de atrito quando o módulo seguinte for deslocado.

Algumas hipóteses tornam-se necessárias para o procedimento de análise deste sistema:

- [1] A pressão da superfície de contato do sistema (sapata de freio) com a tubulação é uniforme em qualquer ponto de contato.
- [2] A sapata de freio é adotada como sendo rígida. Uma vez que isto não é verdadeiro, alguma deflexão ocorrerá, dependendo da carga, pressão e cisalhamento aplicados à sapata. A distribuição de pressão resultante será diferente da adotada para a modelagem.
- [3] Toda análise será baseada no coeficiente de atrito, Tab.(1), o qual não variará com as diferenças de pressão, temperatura e desgaste das pastilhas. Adotaremos as variações destes parâmetros como sendo desprezíveis.

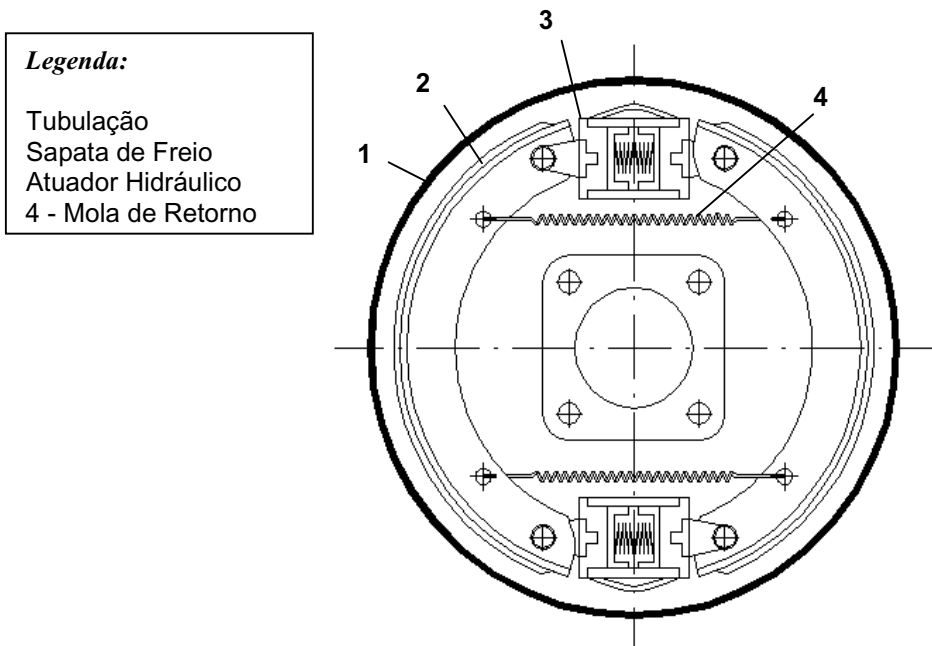


Fig 6.: Módulo de locomoção ou anel de compressão da tubulação

Tabela 1.: Materiais para as pastilhas de freio [7].

Material	Coeficiente de atrito		Máx. Temperatura (°C)	Máx. Pressão (Kpa)
	Molhado	Seco		
Ferro fundido em ferro fundido	0,05	0,15-0,20	320	1000 – 1750
Metal sinterizado em ferro fundido	0,05 – 0,01	0,1 – 0,4	540	1000
Metal sinterizado em metal duro	0,05 – 0,01	0,1 – 0,3	540	2100
Madeira em aço ou ferro fundido	0,16	0,2– 0,35	150	400 – 620
Couro em aço ou ferro fundido	0,12	0,3 – 0,5	100	70 – 280
Contiça em aço ou ferro fundido	0,15 – 0,25	0,3 – 0,5	100	50 – 100
Feltro em aço ou ferro fundido	0,18	0,22	140	35 – 70
Amianto trançado em aço ou ferro fundido	0,1 – 0,2	0,3 – 0,6	175 – 260	350 - 700
Amianto moldado em aço ou ferro fundido	0,08 – 0,12	0,2 – 0,5	260	350 – 1000
Amianto impregnado em aço/ f. fundido	0,12	0,32	260 – 400	1000
Grafite Carbono em aço	0,05 – 0,1	0,25	370 - 540	2100

Contudo, as tubulações são feitas de materiais que possuem determinadas limitações. Não podemos, portanto, aplicar uma força de compressão muito grande contra a parede interna do duto, pois, temos que atuar abaixo do limite de resistência do mesmo para que não haja risco de deformação plástica ou rompimento da tubulação inspecionada.

4.1.1. Modelagem Mecânica da Tubulação

A solução completa deste problema consiste em encontrar a distribuição das tensões e deformações em função do carregamento. Esta solução tem de satisfazer as equações de equilíbrio em todos os pontos do contorno, assim como as relações tensão-deformação-temperatura, e as condições geométricas associadas com a definição de tensão e o conceito da elongação contínua. Essas relações, Eq.(1-a) tida como equação de equilíbrio, Eq.(1-b) que representa a equação de compatibilidade geométrica e a Eq.(1-c) que são as relações de tensão-deformação-temperatura, fornecem 15 equações que são os fundamentos do que é geralmente chamado de *teoria elástica-linear* [8].

$$\begin{aligned}
 \frac{\partial \sigma_x}{\partial x} + \frac{\partial \tau_{xy}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + X &= 0 \\
 \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \sigma_y}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + Y &= 0 \\
 \frac{\partial \tau_{zx}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \sigma_z}{\partial z} + Z &= 0
 \end{aligned}$$

Onde X, Y, Z são forças de corpo distribuídas pelo volume de controle com intensidade X, Y, Z por unidade de volume.

(1-a)

$$\begin{aligned}
 \epsilon_x &= \frac{\partial u}{\partial x} & \gamma_{xy} &= \frac{\partial v}{\partial x} + \frac{\partial u}{\partial y} \\
 \epsilon_y &= \frac{\partial v}{\partial y} & \gamma_{yz} &= \frac{\partial w}{\partial y} + \frac{\partial v}{\partial z} \\
 \epsilon_z &= \frac{\partial w}{\partial z} & \gamma_{zx} &= \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}
 \end{aligned}$$

(1-b)

$$\begin{aligned}
 \epsilon_x &= \frac{1}{E} [\sigma_x - \nu (\sigma_y + \sigma_z)] + \alpha (T - T_o) \\
 \epsilon_y &= \frac{1}{E} [\sigma_y - \nu (\sigma_z + \sigma_x)] + \alpha (T - T_o) \\
 \epsilon_z &= \frac{1}{E} [\sigma_z - \nu (\sigma_x + \sigma_y)] + \alpha (T - T_o) \\
 \gamma_{xy} &= \frac{\tau_{xy}}{G} \\
 \gamma_{yz} &= \frac{\tau_{yz}}{G} \\
 \gamma_{zx} &= \frac{\tau_{zx}}{G}
 \end{aligned}$$

(1-c)

Para a modelagem da tubulação, a solução adotada será a linear-elástica para um cilindro sujeito a pequenas deformações com a teoria da elasticidade para sólidos homogêneos isotrópicos (direções principais de tensão são as mesmas de deformação). Não é necessário o envolvimento de outras teorias que envolvam materiais anisotrópicos e/ou ortotrópicos, pois, como mencionado anteriormente, o ambiente de trabalho consiste quase que na totalidade por dutos de Aço Carbono, Aço Ligado ou Aço Inoxidável (materiais isotrópicos).

Consideremos um cilindro longo de raio interno r_i e raio externo r_o . Determinaremos a distribuição de tensão do cilindro. Existe uma pressão interna uniforme p_i dado pelo acionamento do anel de compressão da tubulação, uma pressão externa p_o dado pelo ambiente que cerca a tubulação (ex.: tubulações submersas), e uma tensão longitudinal uniforme σ_z .

Adotaremos a simetria cilíndrica para o uso de coordenadas apropriadas r , θ e z (coordenadas cilíndricas) e a tensão longitudinal como sendo uniforme. São 15 as equações de elasticidade em coordenadas cilíndricas que regem esta modelagem juntamente com as seguintes condições de contorno:

$$\sigma_r = -p_i \quad \tau_{rz} = 0 \quad \tau_{r\theta} = 0 \quad (2)$$

na superfície interna $r = r_i$,

$$\sigma_r = -p_o \quad \tau_{rz} = 0 \quad \tau_{r\theta} = 0 \quad (3)$$

na superfície externa $r = r_o$, e

$$\sigma_z = \sigma_o \quad \tau_{rz} = 0 \quad \tau_{rz} = 0 \quad (4)$$

O problema agora pode ser simplificado, neste caso, em decorrência da simetria radial de carregamento. Encontraremos, então uma solução independente de θ . Também adotamos a hipótese de que todas as tensões e deformações são independentes da coordenada z .

Resolvendo a equação de equilíbrio com as devidas condições de contorno e colocando-se σ_r e σ_θ em função de u , teremos uma equação diferencial para $u(r)$ com solução para as tensões. Nas equações a seguir os esforços positivos indicam tração e os negativos indicam compressão.

$$\sigma_r = -\frac{p_i \left[\left(\frac{r_o}{r} \right)^2 - 1 \right] + p_o \left[\left(\frac{r_o}{r_i} \right)^2 - \left(\frac{r_o}{r} \right)^2 \right]}{\left(\frac{r_o}{r} \right)^2 - 1} = \frac{p_i r_i^2 - p_o r_o^2 + r_i^2 r_o^2 \left(\frac{p_o - p_i}{r^2} \right)}{r_o^2 - r_i^2} \quad (5-a)$$

$$\sigma_\theta = \frac{p_i \left[\left(\frac{r_o}{r} \right)^2 + 1 \right] - p_o \left[\left(\frac{r_o}{r_i} \right)^2 + \left(\frac{r_o}{r} \right)^2 \right]}{\left(\frac{r_o}{r} \right)^2 - 1} = \frac{p_i r_i^2 - p_o r_o^2 - r_i^2 r_o^2 \left(\frac{p_o - p_i}{r^2} \right)}{r_o^2 - r_i^2} \quad (5-b)$$

4.1.2. Aproximação para Tubulação de Paredes Finas

Quando a espessura da parede do cilindro for em torno de 5% de seu raio, a tensão radial resultado da pressão interna é muito menor comparada com a tensão tangencial. Dentro destas condições a tensão tangencial pode ser adotada como sendo uniformemente distribuída ao longo da espessura do cilindro. Sendo $r_o \approx r_i$, então adota-se $(r_o + r_i)$ como sendo $2r$ e $(r_o - r_i)$ como sendo t . Substituindo estas suposições nas equações de tensão para este caso, temos:

$$\begin{aligned} \sigma_r &\approx 0 \\ \sigma_\theta &= \frac{(p_i - p_o)d}{2t} \\ \sigma_z &= \frac{(p_i - p_o)d}{4t} \end{aligned} \quad (6)$$

Onde:

d é o diâmetro médio (2r) da tubulação;

t é a espessura da mesma;

p_o é a pressão atuante na parte externa da parede do duto; decorrente do ambiente que cerca a tubulação;

p_i é a pressão atuante na parte interna do tubo; decorrente da atuação do módulo de locomoção.

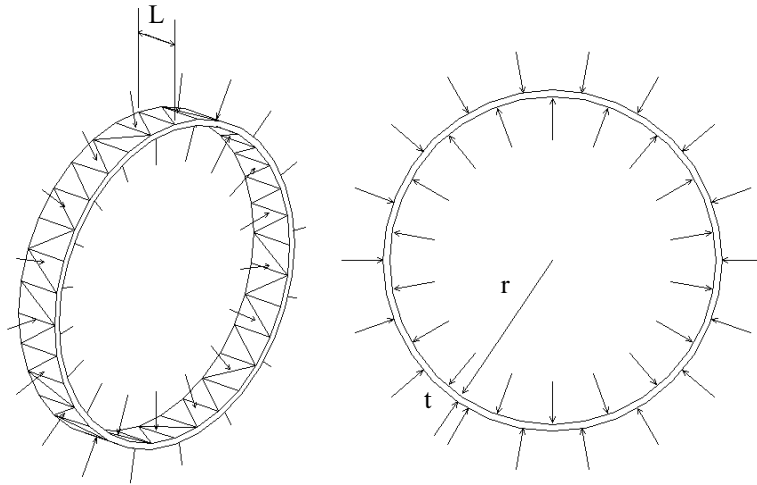


Fig 7.: Seção da tubulação em contato com o módulo de locomoção

4.1.3. Critérios de Resistência

Logicamente, existe um limite para a tensão que irá ser aplicada às paredes da tubulação para que não haja a deformação plástica da mesma. Para manter as tubulações estritamente no regime elástico faremos uso das teorias de resistência dos materiais mais significativas para este caso; a *teoria da máxima tensão de cisalhamento* que deu origem ao critério de *Tresca*, Eq.(7-a) e a *teoria da máxima energia de distorção* (também designada teoria do cisalhamento octaédrico) dando origem ao critério de *Mises-Hencky*, Eq.(7-b). Estes critérios são os mais indicados para materiais dúcteis [8].

$$\frac{1}{2}(\tau_{\max} - \tau_{\min}) = \frac{Y}{2} \tag{7-a}$$

$$\sqrt{\frac{1}{2}[(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2]} = Y \tag{7-b}$$

Podemos concluir que a pressão no início do escoamento, de acordo com o critério de *Mises-Hencky*, é:

$$p_{M-H} = \frac{4 Y t}{\sqrt{3} d} \tag{8}$$

Quanto à condição de *Tresca*, no início do escoamento, temos:

$$p_{Tr} = \frac{2 Y t}{d} \tag{9}$$

Onde Y é o limite de escoamento tabelado para diferentes tipos de aço utilizados nas tubulações.

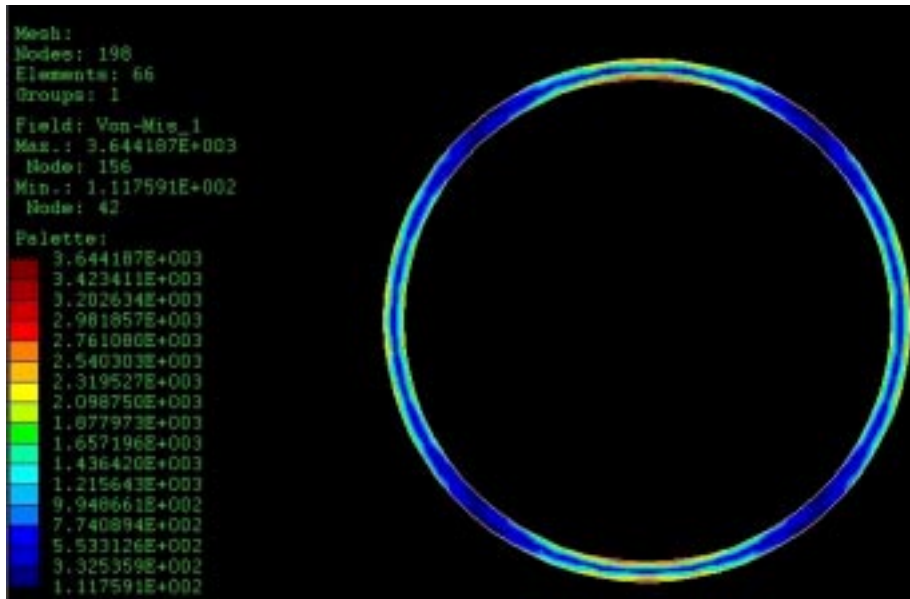


Fig 8.: Aplicação do critério da máxima energia de distorção para o caso em estudo

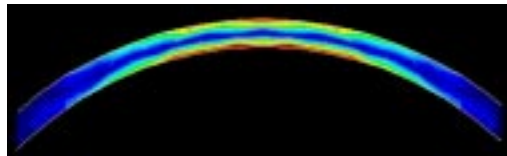


Fig 9.: Detalhe de iminência de falha do material

Com a distribuição de tensão máxima que a parede de uma determinada tubulação é capaz de suportar (com os devidos critérios de resistência e fatores de segurança), estima-se qual deve ser a força máxima para os atuadores. Sabe-se que apenas a força de atrito impulsiona o veículo para o sentido desejado, recebendo como empecilho à movimentação apenas a força resistiva dos cabos de alimentação de energia. A Figura (8) e (9) retratam a análise computacional das tensões [9] na parede da tubulação. Para o cálculo da tensão máxima admissível foi utilizado o programa de elementos finitos *Fiesta* com limitação para 200 nós. Com isso, podemos estimar o comprimento de cada módulo de locomoção para atingir a força de atrito desejada.

4.2. Conceito de Aderência Eletromagnética

O conceito da aderência eletromagnética contribui para o aumento da tensão superficial no interior da parede da tubulação, ou seja, contribui para o aumento da força normal F_N na direção radial à tubulação. Quanto maior for a força normal à tubulação, maior será a força de atrito que contribui para a movimentação do veículo em questão. Os eletroímãs deverão ser colocados no interior das sapatas de freio [10], junto às pastilhas com alto coeficiente de atrito.

Como o sistema de locomoção do VERME está em fase de testes, o projeto completo do formato otimizado dos eletroímãs ainda está sendo concluído; portanto, a equipe está trabalhando num mecanismo que melhor se adapte às condições inóspitas do ambiente.

5. Plataforma Operacional

O sistema completo está subdividido em seis partes principais. São elas: (i) sistema de locomoção, (ii) sensores de bordo de visão e ultra-som, (iii) controle de bordo e sistema de telemetria, (iv) sistema de navegação, (v) console de operação. Cada um destes sub-itens está descrito em detalhes nas seções seguintes.

5.1. Estrutura do veículo

A carcaça do veículo mostrada na Fig. (5) é confeccionada de alumínio anodizado com um sistema a prova d' água que contém o controle, e os equipamentos eletrônicos de força e telemetria, os sensores ultra-sons, a câmera CCD (*Charged Coupled Device*) e o sistema de iluminação, assim como o sistema de navegação e seus respectivos equipamentos.

O sistema de locomoção está situado em um compartimento separado contendo os motores e as transmissões os quais são conectados ao restante do sistema por cabos de aço e conectores. O sistema de cabos é totalmente formado por cabos de fibra óptica, acoplado ao veículo por uma conexão flexível para permitir a realização de curvas sem desgastar ou dobrar o cabo. O umbilical também é constituído de fibra óptica inserida em uma matriz impermeável e envolvido por fibras trançadas de Kevlar e uma capa polimérica antiabrasiva. Os conectores e as ligações dos cabos asseguram fácil manutenção e montagem em campo.

5.2. Equipamentos eletrônicos e de Telemetria

A comunicação com o veículo é feita por um sistema umbilical de fibra óptica especialmente desenvolvida para este fim. Um sistema de leitura de ondas, miniaturizado desenvolvido a partir de componentes comerciais disponíveis no mercado, foi integrado com um computador de bordo disposto no interior do compartimento da carcaça do veículo. O sistema de telemetria é integrado com um microprocessador de bordo para um melhor monitoramento.

O microprocessador de bordo monitora o sistema de telemetria, os sensores de bordo e também os chips de controle dos motores do sistema de locomoção. O microprocessador comunica-se com o *host* via placa de rede. O *host* é um computador pessoal e possui uma tela para que o operador planeje o controle das funções.

O veículo inteiro opera com energia gerada através dos cabos.

5.3. Sensores

Os sensores usados no veículo consistem em parte por uma câmera CCD (Sony) iluminada por LEDs montados em uma placa, para iluminar a trajetória frontal do veículo e permitir a tele-operação.

O sistema de medição ultra-sônico trabalha com cerâmicas piezelétricas dentro do princípio do tempo de percurso da onda difratada (*TOFD*), acoplado da parte posterior do veículo e consiste num jogo de transdutores em linha (*Multiplexer*) montados em um elevador. Os dados dos sensores de ultra-som e as respostas do sistema visual, são continuamente armazenadas e processadas para análise posterior.

5.4. Sistema de Navegação

O sistema de navegação está baseado em sistema comercial existente, porém modificado, de posicionamento acústico, o qual usa o tempo de percurso do som nos líquidos a fim de triangular a posição dos transceptores montados na carcaça do veículo. A rede de navegação consiste em (pelo menos) três transdutores.

Os impulsos sonoros são recebidos e retornados por dois transceptores na parte superior do veículo, onde é feita a leitura pelo computador de bordo o qual faz a triangulação para localizar cada transceptor, dando ao operador a posição exata do veículo dentro da malha de dutos. Este posicionamento é transmitido ao *host* para permitir a visualização, o planejamento e o controle das funções do veículo.

5.5. Conectores e Cabos

Os cabos foram projetados para minimizar o atrito com as paredes dos dutos. Desta forma, os cabos representam um componente chave para o sistema, uma vez que estes fornecem a energia para o sistema. Os cabos são forrados por uma capa polimérica resistente, que contém pares de cabos menores. Os cabos menores são envolvidos por uma resina polimérica flexível encapada por Kevlar.

Os conectores são projetados baseados nos conceitos de conexão subaquática. O projeto destas conexões pode ser encontrado comercialmente.

5.6. Painel de Controle

O painel de controle consiste em um sistema *host* de computação, um sistema de condicionamento de força, monitores de TV e uma interface de telemetria montados em um compartimento, com um controle remoto portátil contendo um *joystick*, uma chave geral, e um monitor de vídeo para o controle das funções.

5.7. Resultados Experimentais

O veículo de inspeção VERME está atualmente em fase de testes no Laboratório de Projeto de Máquinas/Robótica da Universidade Federal do Rio de Janeiro. Todos os sistemas de locomoção, navegação, orientação e sensoriamento ainda estão sendo testados. Os resultados adquiridos até esta etapa do projeto, mostram a viabilidade deste tipo de veículo na operação e inspeção de tubulações.

6. Conclusões

Devido às condições inóspitas do ambiente em que o sistema de PIG é utilizado, este tipo de inspeção ainda não se tornou um padrão de manutenção em tubulações, com exceção de algumas áreas específicas como a inspeção de tubulações de esgoto em alguns países. Entretanto, quanto maior for a dificuldade de locomoção de um PIG no interior de uma tubulação, maior será a necessidade de se garantir a segurança durante o serviço. A conclusão final deste artigo mostra que a premissa básica deste projeto, até o presente momento, está sendo cumprida de maneira satisfatória: propiciar uma alternativa viável para redução dos altíssimos custos de inspeção e manutenção de dutos.

7. Referências

- [1] MELLO, L. A. M., “Manual de Utilização de PIG” . CENPES/DIPREX/SESUB, Junho –1995.
- [2] KIRCHNER, F. ; HERTZBERG, J., “A Prototype Study of an Autonomous Robot Platform for Sewerage System Maintenance” . GMD – German National Research Center for Information Technology, n°01IW501A.
- [3] TIRATSOO, J.N.H., “Pipeline Pigging Technology” , Gulf Publishing Company, 2^d edition, 1992.
- [4] CATUNDA, C.E.G. ; M.S. DUTRA; R.F. de MORAES; W. WILLS; “Estudo de um veículo para Inspeção e Coleta Amostral de Resíduos em Dutos em Operação” , anais do 7 ° Congresso de Estudantes de Engenharia Mecânica – CREEM’2000; agosto/00, pp. 185 – 186.
- [5] FUJIWARA, S.; “Development of an In-Pipe Inspection Vehicle” , Materials Evaluation, May/95, pp. 566-569.
- [6] RUPPERT, E.E.; BARNER, R.D., “Invertebrate Zoology” , Saunders College Publishing, 6th edition, 1994.
- [7] SHIGLEY, J.E.; “Mechanical Engineering Design” ; 1st edition, 1986, pp. 610.
- [8] CRANDALL, S.H.; DAHL, N.C.; LARDNER, T.J.; “An Introduction to the Mechanics of Solids” , 2^d edition, 1978, pp.287-299.
- [9] GOMES, M.G.F.M.; “Finite Element Analysis of Pig Cups for Pipeline Cleaning” , M.Sc. Thesis, Universidade Federal do Rio de Janeiro – UFRJ – COPPE, 1994.
- [10] VACODYM – VACOMAX, Catálogo – *Rare-Earth Permanent Magnets*” , Edition 2000.

Instrumented Vehicle for Inspection and Maintenance of Pipes

Abstract.. Nowadays, one of the procedures applied for pipe inspection and maintenance is the PIG technique. In some cases, this technique cannot be used, such as in a “non pigable” pipe. A “non pigable” pipe offers risks of retaining a PIG. According to the situation, the robot known as VERME [port.] (Robotic Exploring Vehicle with Electromechanical Motion) was proposed. This vehicle will be used for inspecting a pipe mesh, which will provide a viable alternative to reduce the high cost of pipe inspection and maintenance. Based on the Invertebrate Zoology Study, specially annelids (worms), VERME was developed. The annelids are morphologically characterized by their ring-segmented body. The proposed vehicle reproduces a locomotion system similar to these beings. The physical structure of the robot is subdivided into many independent modules (rings) which are connected to mechanical actuators. Its locomotion system consists of an electromechanical device that, when in contact with the steel pipe’s wall, attaches the correspondent module and permits the movement of the subsequent module to the desired orientation.

Keywords: Instrumented PIG, Multi-body System, Mechatronic, Pipeline.

REDES DE PETRI E ORIENTAÇÃO A OBJETOS PARA O DESENVOLVIMENTO DE SISTEMAS HÍBRIDOS

Villani, Emilia

Dept. Eng. Mecatrônica e de Sistemas Mecânicos, Escola Politécnica, USP
Av. Prof. Mello Moraes, 2201 São Paulo Brasil
emiliav@usp.br

Miyagi, Paulo Eigi

Dept. Eng. Mecatrônica e de Sistemas Mecânicos, Escola Politécnica, USP
Av. Prof. Mello Moraes, 2201 São Paulo Brasil
pemiyagi@usp.br

Valette, Robert

LAAS – CNRS (Laboratoire d'Analyse et d'Architecture des Systèmes)
7, Avenue du Colonel Roche, 31077 Toulouse Cedex 4 FRANCE
robert@laas.fr

Resumo. Este artigo propõe uma nova abordagem para projeto de sistemas de controle híbrido, onde “híbrido” indica a presença simultânea de características de sistemas a eventos discretos (sistemas dinâmicos com estados discretos e transições instantâneas) e de sistemas contínuos (sistemas dinâmicos com variáveis de estado e eventos contínuos). Partindo de uma proposta já existente de Rede de Petri associada a equações diferenciais, considera-se uma solução para modelagem de sistemas industriais complexos através de uma visão estruturada. Assim, esta nova abordagem introduz o conceito de orientação a objetos à utilização de Rede de Petri associada a equações diferenciais. A notação UML é utilizada para auxiliar a representação de diferentes aspectos do sistema. Através de um exemplo, verifica-se que a utilização em conjunto da UML e de redes Predicado Transição Diferenciais se complementam, possibilitando o enfoque das características híbridas do sistema.

Palavras chave: sistemas híbridos, orientação a objetos, redes de Petri, UML.

1. Introdução

Dentro do conceito de sistemas mecatrônicos, um aspecto importante é a integração de sistemas de diferentes tecnologias (Fraser et al., 1993). Esta integração tem sido objeto de diversos estudos e, em muitos casos, é um fator limitante para o aperfeiçoamento de sistemas de controle. Neste contexto, um problema frequentemente abordado, que é um resultado direto desta necessidade de integração, é o desenvolvimento de sistemas híbridos.

O conceito de “sistemas híbridos” se insere dentro do contexto de modelagem de sistemas, onde uma possível classificação refere-se aos tipos de variáveis encontradas no modelo. Neste sentido, são definidos os sistemas a eventos discretos (tempo e variáveis de estado são representados por números inteiros ou variáveis lógicas), os sistemas de variáveis contínuas (tempo e variáveis de estado são representados por números reais) e os sistemas híbridos, onde encontra-se a presença simultânea de características de sistemas a eventos discretos e de sistemas contínuos.

Em particular, no que se refere ao projeto de sistemas supervisórios híbridos, o aumento da demanda por produtividade e eficiência tem resultado no aumento da sofisticação e complexidade dos processos supervisionados. Associada à evolução da tecnologia de automação disponível, os profissionais devem assim lidar com tomadas de decisões cada vez mais complexas. A ausência de uma abordagem efetiva para o tratamento da complexidade conceitual relacionada ao projeto de sistemas supervisórios híbridos é apontada em [Nimmo, 1999] [Garcia, 1997].

Neste sentido, partindo de uma proposta já existente de rede de Petri associada a equações diferenciais para modelagem de sistemas híbridos, este trabalho busca uma solução para sistemas de elevada complexidade através de uma visão mais estruturada. Assim, apresenta-se uma nova abordagem onde introduz-se o conceito de orientação a objetos à utilização de rede de Petri associada a equações diferenciais.

Este trabalho está organizado da seguinte forma. No item 2 apresenta-se uma introdução sobre sistemas híbridos. O item 3 introduz a orientação a objetos e como os conceitos relacionados à orientação a objetos podem ser incorporados à teoria de redes de Petri. O item 4 apresenta os principais aspectos da abordagem proposta para sistemas híbridos. O item 5 apresenta algumas considerações finais e indica as principais direções nas quais este trabalho deve evoluir.

2. Sistemas Híbridos

Enquanto a teoria de controle clássico dispõe de técnicas e ferramentas para análise e síntese de sistemas contínuos, e abordagens equivalentes apresentam soluções similares para sistemas a eventos discretos, não existe uma abordagem formalizada e consolidada para a análise de sistemas com comportamento dinâmico discreto e contínuo simultaneamente, isto é, as soluções são abordadas de modo *ad hoc*, evidenciando a necessidade de desenvolvimento e consolidação de uma abordagem sistemática para o tratamento de sistemas híbridos [Lemmon, He & Markovsky, 1999].

Um ponto de importância crucial no desenvolvimento de sistemas de controle é a decisão de qual modelo adotar. A escolha da linguagem de modelagem influencia diretamente as possibilidades de análise, verificação e validação e tem uma profunda influência em como um problema é abordado e como sua solução é emoldurada.

No que se refere a modelagem de sistemas híbridos, muitos formalismos têm sido propostos. Para uma revisão bibliográfica detalhada ver [Guéguen & Lefebvre, 2000], [Champagnat, 1998]. De uma forma geral, algumas abordagens consistem em extensões de modelos contínuos, como equações diferenciais ordinárias nas quais são incluídas variáveis cujo valor pode ser modificado de forma descontínua no tempo [Antsaklis & Nerode, 1998]. Outras abordagens consistem na modificação de técnicas de modelagem utilizadas em sistemas a eventos discretos, onde são introduzidos novos elementos para a representação da dinâmica contínua do sistema, como nas Redes de Petri Híbridas [Alla & David, 1998]. Existem também abordagens intermediárias que combinam modelos de sistemas contínuos, descritos por equações diferenciais, e de sistemas discretos, descritos por autômatos finitos ou Redes de Petri, onde é introduzida uma interface para a comunicação entre os dois tipos de modelos, como em [Valentin-Roubinet, 1998].

Entre estas abordagens, este trabalho considera particularmente as abordagens do último grupo, pois a extensão de formalismos discretos leva, em geral, a uma ferramenta com capacidade de modelagem da parte contínua relativamente restrita e específica. O equivalente pode ser afirmado para as extensões de formalismos contínuos. Por outro lado, as abordagens que especificam uma solução a partir da mistura de um formalismo discreto com um formalismo contínuo apresentam maior flexibilidade e poder de modelagem. Entre as abordagens deste grupo, considerou-se em especial aquelas onde o formalismo discreto é baseado em redes de Petri. Esta escolha foi realizada considerando-se as já bem conhecidas propriedades deste formalismo, como habilidade para representar a sincronização de processos, eventos concorrentes, causalidade, compartilhamento de recursos, presença de conflitos, entre outros.

Entre as principais abordagens para sistemas híbridos derivadas de redes de Petri que definem uma interface entre a parte discreta e a parte contínua, tem-se as redes de Petri Mistas [Valentin-Roubinet, 2000] e as redes Predicado Transição Diferencial [Champagnat, 1998]. Estes formalismos, no entanto, não apresentam recursos para a decomposição do sistema e para uma modelagem progressiva, por exemplo, através de uma abordagem hierárquica, onde os modelos podem ser expressos com diferentes níveis de detalhamento. A modelagem é realizada de uma forma plana, o que dificulta o estudo de sistemas complexos, onde não é possível compreender o sistema como um todo na profundidade necessária.

Buscando uma solução para estes problemas, este trabalho introduz os conceitos de orientação a objetos para as redes de Petri associadas a equações diferenciais. Entre as vantagens da orientação a objetos ([Booch, Rumbaugh & Jacobson, 1998], [Douglass, 1998], [Paludetto, 1991]) encontram-se a reutilização, a facilidade de modificação, revisão e manutenção dos modelos, e a maior correspondência entre os objetos do sistema de controle e dispositivos reais relacionados ao problema, pois a estrutura e o comportamento dos dados estão diretamente vinculados.

Entre as duas propostas de redes de Petri associadas a equações diferenciais citadas anteriormente, selecionou-se para os propósitos deste trabalho a rede de Petri Predicado Transição Diferencial pois considera-se que esta rede apresenta características que facilitam a sua aplicação para uma abordagem orientada a objetos, como a não utilização de variáveis globais e a não imposição de capacidade unitária para os *lugares*.

De uma forma geral, nas redes Predicado Transição Diferenciais tem-se que:

- à cada *marca* é associado um vetor de variáveis contínuas;
- à cada *lugar* é associado um sistema de equações diferenciais algébricas que definem a evolução do vetor de variáveis contínuas quando uma *marca* está naquele *lugar*;
- à cada *transição* é associada uma *função de junção*, que define o valor do vetor de variáveis contínuas após o disparo, e uma *função de habilitação*, que habilita ou inibe o *disparo* de *transições* de acordo com os valores do vetor de variáveis contínuas.

3. Orientação a Objetos

De uma forma genérica um objeto pode ser definido como uma entidade que possui atributos e comportamento [Douglass, 1998]. Os atributos representam os dados relativos ao objeto. O comportamento define como o estado do objeto evolui de acordo com a interação do objeto com outros objetos do sistema ou com a evolução do tempo.

Um conceito importante relacionado ao comportamento é o de encapsulamento. Através do encapsulamento define-se um objeto como sendo formado por uma estrutura interna e por uma interface oferecida pelo objeto ao ambiente externo, representada por métodos, através dos quais pode-se ter acesso aos atributos do objeto. A estrutura interna é escondida a fim de garantir a independência da visão externa que se tem do objeto em relação à sua estrutura interna.

[Paludetto, 1991] adiciona ainda a esta definição o conceito de coesão interna. O objeto deve apresentar uma forte coesão interna, no sentido que os dados e as operações que ele contém devem ser fortemente ligados e representar um subconjunto lógico significativo do sistema. Da mesma forma os diferentes objetos devem ser fracamente ligados entre si, no sentido que a interface entre os objetos deve ser a mais simples possível. Esta característica é essencial para que o ganho em simplicidade no modelo do sistema devido a decomposição do sistema em objetos seja superior a complexidade introduzida pela comunicação entre os objetos.

3.1. UML – Unified Modeling Language

Dentro do contexto de modelagem orientada a objetos a UML (Unified Modeling Language) está se tornando, de

fato, um padrão para representação deste tipo de sistema. Entre suas vantagens estão a capacidade de expressar modelos em diferentes níveis de precisão e a capacidade de representar visões diferentes de um mesmo sistema.

UML é uma linguagem de modelagem que visa definir a semântica do modelo objeto e fornecer uma notação para capturar e comunicar a estrutura e o comportamento do objeto. É o resultado da união dos esforços de muitos especialistas em orientação a objeto, entre eles se destacam Grady Booch (Método Booch), Jim Rumbaugh (Object Modeling Technique (OMT)), Ivar Jacobson (Object-Oriented Software Engineering (OOSE)) e David Harel (Statecharts). Observa-se que UML não é uma linguagem visual de programação, no entanto os seus modelos podem ser diretamente conectados a uma variedade de linguagens de programação.

De uma forma geral, pode-se descrever a UML como uma linguagem que, a partir de alguns elementos básicos e dos tipos de relação entre estes elementos, define uma série de diagramas que têm como finalidade a representação de diferentes aspectos de um mesmo sistema. Este texto não pretende apresentar uma revisão bibliográfica de todos os aspectos de todos os diagramas, para tanto cita-se como referência [Booch, Rumbaugh & Jacobson, 1998]. Serão apresentadas apenas algumas noções e conceitos da notação UML.

Na UML, os principais tipos de elementos básicos são elementos estruturais (classes, interfaces, componentes, casos de utilização, entre outros) e elementos relacionados ao comportamento do sistema (interações, estados, etc.). As principais relações definidas são de *associação* e *generalização*. A *associação* descreve uma relação estrutural entre dois elementos. Um tipo de *associação* é a *agregação*, onde um objeto é considerado como “uma parte” de outro objeto. Outro tipo é a *composição*, considerada uma relação de agregação mais forte, onde a “vida” do objeto agregado deve estar contida na “vida” do objeto que o agrega. A *generalização*, também chamada de *especialização* ou *herança*, é uma relação onde objetos especializados (filhos) compartilham a mesma estrutura de um objeto generalizado (mãe).

A linguagem UML define, ao todo, nove diagramas, entre eles, destacam-se aqueles utilizados neste trabalho:

- *Diagrama de classes (Class diagram)*: mostra um conjunto de classes que fazem parte do sistema e as relações entre estas classes. Apresenta uma visão estática do projeto de um sistema.
- *Diagrama de interação (Interaction diagram)*: mostra as interações entre objetos, é constituído de conjunto de objetos e suas relações e inclui as mensagens que podem ser trocadas entre eles. Apresenta uma visão dinâmica de um sistema. Pode ser dividido em *diagrama de seqüência (Sequence diagram)* e *diagrama de colaboração (Collaboration diagram)*. O primeiro enfatiza a ordenação no tempo das mensagens e o segundo a organização estrutural dos objetos que enviam e recebem mensagens.

3.2 UML e Redes de Petri

De acordo com diversos autores ([Douglass, 1998], [Baresi & Pezzè, 1998], [Giese, Graf & Wirtz, 1999]), a UML apresenta ainda muitos pontos a serem evoluídos. Entre os problemas apontados tem-se a não representação de forma satisfatória do comportamento do sistema (visão dinâmica com o tempo). Os recursos da UML para descrever aspectos de comportamento são fracos quando comparados aos recursos para descrever estruturas, em particular no que se refere a descrição de paralelismos, sincronismos, concorrência, etc. entre processos.

Visando solucionar, entre outras coisas, este problema foram desenvolvidos alguns trabalhos onde a UML é utilizada em conjunto com redes de Petri. Em [Giese, Graf & Wirtz, 1999] propõe-se a utilização em conjunto da UML com redes OCN (Object Coordination Net). Em [Baresi & Pezzè, 1998] é proposto o mapeamento automático dos modelos em UML em redes de Petri de Alto Nível, usando a abordagem CR (Customization Rules Approach), que permite aos seus usuários ajustar suas interpretações de uma notação informal a uma notação formal através da definição de um conjunto particular de regras. Na mesma linha que o trabalho de [Baresi & Pezzè, 1998], o trabalho de [Gehrke, Goltz & Wehrheim, 1998] propõe um método para traduzir os *diagramas de atividade* e de *colaboração* em redes de Petri (Lugar/Transição), dando uma interpretação precisa da sua semântica. O trabalho discute como usar os dois tipos de diagramas juntos num processo de desenvolvimento de software. Outra contribuição nesta área se refere ao trabalho de [Paludetto & Delatour, 1999], onde os autores propõem uma metodologia UML/OPN (Object Petri Net) com regras para tradução dos *diagramas de casos de utilização* em redes de Petri. Estes diagramas são um meio relativamente simples de representar informalmente os requisitos de comportamento relacionados a um determinado caso de utilização. Eles constituem o ponto de partida para a geração de redes de Petri, levando a uma modelagem formal onde a validação e verificação podem ser realizadas.

4. Metodologia de Projeto de Sistemas Híbridos

Baseando-se nos trabalhos citados no item 3 foi considerada a incorporação dos conceitos de orientação a objeto às redes Predicado Transição Diferencial para projeto de sistemas supervisórios híbridos.

Para complementar o poder de representação gráfica das redes de Petri no que se refere à representação dos aspectos relacionados à orientação a objetos, e considerando a futura implementação prática do sistema, utilizam-se recursos da UML, introduzindo modificações para a melhor representação dos aspectos híbridos do sistema.

Utilizando como exemplo o sistema de controle de um ar condicionado, apresentam-se as principais características da abordagem proposta.

4.1 O Sistema de Ar Condicionado

O exemplo considerado refere-se ao sistema supervisor do sistema de ar condicionado de um edifício de 3 andares, onde cada andar é dividido em 2 zonas (totalizando 6 zonas no edifício). O sistema de ar condicionado é do tipo VAC (Volume de Ar Constante) e permite apenas resfriamento da zona. Um esquema é apresentado na Figura 1.

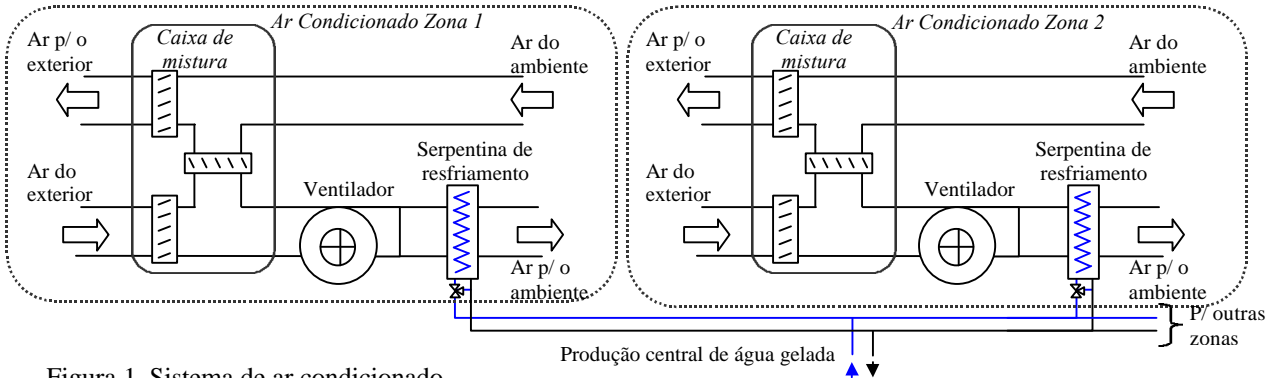


Figura 1. Sistema de ar condicionado.

Cada zona possui:

- Um ventilador, de velocidade constante.
- Uma serpentina de resfriamento cujo fluxo de água é variado de acordo com a posição de uma válvula de 3 vias controlada por um regulador PI em função da temperatura na zona.
- Uma caixa de mistura de 3 posições: sem renovação (0%), renovação parcial (P%) e renovação total (100%).
- Três sensores de temperatura: um para temperatura exterior, um para temperatura na zona e um para temperatura do ar na saída da serpentina, e um sensor de CO₂.

A interface com o usuário é realizada através de um painel de controle presente em cada zona (Figura 2):

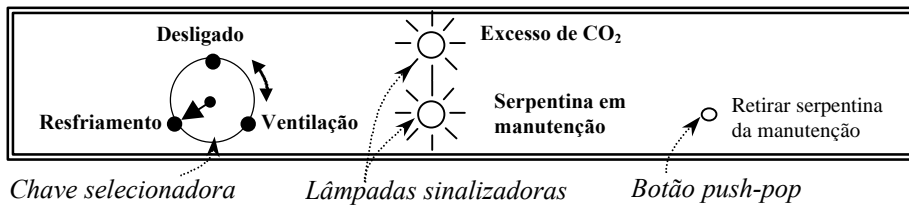


Figura 2. Painel de controle de uma zona.

Este painel possui os seguintes dispositivos:

- Uma chave seccionadora que permite a seleção do modo de operação do ar condicionado entre desligado, apenas ventilação e resfriamento.
- Duas lâmpadas sinalizadoras que indicam excesso de CO₂ na zona e necessidade de manutenção na serpentina.
- Um botão do tipo *push-pop* que recoloca a serpentina em operação após a realização de manutenção.

O sistema de controle a ser projetado deve ter as seguintes características:

- Em cada zona, quando o ar condicionado não estiver desligado e quando a porcentagem de CO₂ no ar ultrapassar um determinado limite, deve-se modificar a posição da caixa de mistura para renovar 100% do ar.
- Quando o condicionamento de uma zona é iniciado, a serpentina deve ser acionada após o ventilador.
- Quando o condicionamento de uma zona é desligado, a serpentina deve ser desligada antes do ventilador.
- A serpentina tem um sistema de detecção de falhas que consiste na simulação de um modelo contínuo onde se avalia a temperatura de saída do ar. Esta temperatura também é monitorada por um sensor. Quando a diferença entre a temperatura prevista e a temperatura real ultrapassa um determinado limite considera-se que ocorreu uma falha, assim o sistema de condicionamento da zona é desligado e a lâmpada sinalizadora correspondente é acesa.

4.2. Definição do Objeto Híbrido

Um primeiro passo antes de abordar o problema de decomposição do sistema é a definição de o que é um objeto do ponto de vista de sistemas híbridos. Para a decomposição de sistemas híbridos, um cuidado particular deve ser tomado no que se refere a interação contínua entre dois objetos. Tem-se então dois tipos de interação:

- **Interações discretas:** a interação pode ser considerada e modelada como eventos discretos (envio de mensagem, recebimento da resposta, etc.). Uma interação discreta corresponde a solicitação realizada por um objeto para execução de um método oferecido pela interface de outro objeto.
- **Interações contínuas:** Uma interação contínua corresponde ao caso em que o estado de um objeto evolui de forma contínua ao longo do tempo e esta evolução contínua depende da evolução ao longo do tempo do estado de um outro

objeto. Um exemplo é apresentado na Figura 3a, onde no modelo de um objeto, as variáveis contínuas que determinam seu estado são calculadas através de equações diferenciais que têm como parâmetros de entrada o valor de variáveis contínuas de outro objeto (x_1 é variável de saída do Objeto 1 e variável de entrada do Objeto 2).

No que se refere a interações contínuas uma atenção especial deve ser dada para o compartilhamento de variáveis contínuas e para os ciclos fechados. Quando a evolução do estado de um objeto depende da evolução do estado de um segundo objeto e a evolução deste depende da evolução do primeiro tem-se um ciclo fechado. Na verdade o que tem-se é que as equações dos dois objetos fazem parte de um único sistema de equações e não podem ser resolvidas de forma independente. Isto interfere diretamente no conceito de independência do objeto. Nestes casos, a evolução do estado dos dois objetos está de tal forma interligada que não existe sentido na definição de dois objetos distintos, sendo mais adequado considerar-se os dois objetos iniciais como um único objeto. Um exemplo é apresentado na Figura 3b. Considera-se, portanto, que a definição dos objetos deve ser realizada de tal forma a evitar as interações contínuas do tipo não permitido.

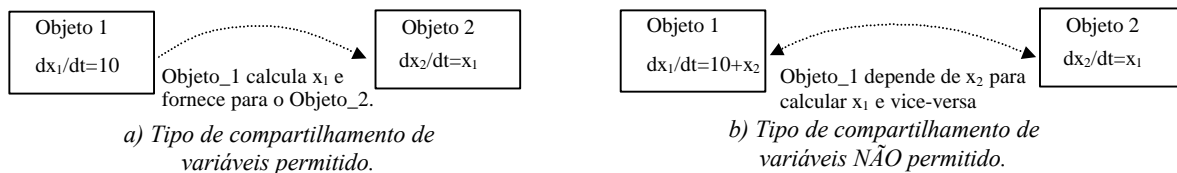


Figura 3. Compartilhamento de variáveis contínuas.

4.3. Decomposição do Sistema

Considerando as diversas etapas do desenvolvimento de um software de controle, a modelagem é uma atividade que pertence principalmente às fases de análise dos requisitos (onde identifica-se objetos e classes do modelo) e de projeto (onde o modelo é construído e, eventualmente, simulado, analisado, validado, etc.). A fase de programação (onde implementam-se os modelos construídos em uma linguagem orientada a objetos) não é considerada neste trabalho. No que se refere a decomposição do sistema, para a modelagem do sistema durante o projeto não aborda-se inicialmente o conceito de herança, que corresponde a inserção do objeto dentro de uma hierarquia de classe, uma vez que esta pode ser considerada uma atividade predominantemente relacionada à fase de programação [Booch, 1994].

Em contrapartida, para a definição de objetos híbridos, prioriza-se a decomposição do sistema segundo relações de agregação e composição entre os objetos. Neste sentido deve-se identificar quais são os candidatos a objetos do sistema e como estes candidatos podem ser decompostos em novos objetos. Desta forma busca-se, através de um refinamento sucessivo, trabalhar a complexidade do sistema.

Tomando como exemplo o sistema de ar condicionado de uma zona, este pode ser inicialmente dividido em três objetos, cuja descrição simplificada é apresentada abaixo:

Objeto 1: Ventilação:

Responsabilidade: circulação e renovação do ar.

Métodos: circular ar com renovação parcial / circular ar com renovação total / não circular o ar.

Objeto 2: Resfriamento

Responsabilidade: controlar a temperatura do ambiente através do resfriamento do ar

Métodos: resfriar o ambiente / suspender o resfriamento.

Objeto 3: Interface

Responsabilidade: indicar a ocorrência de problemas e possibilitar ao usuário modificar a configuração do sistema.

Métodos: acionar resfriamento e ventilação / acionar só resfriamento / desligar o sistema.

A interação entre estes objetos é do tipo discreto, a única exceção se refere a quando o sistema estiver no modo resfriamento, neste caso o Objeto 1 deve fornecer ao Objeto 2 a temperatura do ar na saída da caixa de mistura, para que o Objeto 2 possa estimar a temperatura na saída da serpentina e assim detectar falhas através da comparação desta temperatura estimada com a temperatura real do ar.

Cada um destes sub-sistemas pode ainda ser decomposto em objetos que tenham uma maior relação com os dispositivos físicos do sistema (dispositivos de atuação e de monitoração). Tem-se então uma nova decomposição do sistema onde cada objeto é responsável por fazer a comunicação do sistema de controle com os dispositivos da planta. Ele deve também fornecer para o restante do sistema os métodos correspondentes as possíveis modificações que o sistema de controle possa fazer no modo de operação de cada dispositivo. Além disso, ele deve armazenar as informações relativas ao dispositivo que são de interesse para o sistema de controle. Uma proposta de decomposição é apresentada no diagrama da Figura 4.

Para cada objeto do Nível 2 da Figura 4 apresenta-se a descrição de sua interface, usando a notação UML. Esta descrição é composta por atributos, que podem ser variáveis discretas, variáveis contínuas ou constantes e pelos métodos oferecidos.

Para a classe “Caixa de Mistura” (Figura 5) os atributos disponíveis são a porcentagem de renovação do ar quando a caixa está na posição de renovação parcial (P , constante), a temperatura do ar na saída da caixa de mistura calculada pelo objeto ($T_{\text{caixa_calc}}$, variável contínua), e a quantidade máxima de CO_2 permitida no ambiente ($Q_{\text{CO}_2\text{max}}$, constante). Os métodos disponíveis se referem a modificação da posição da caixa de mistura e a leitura da variável $T_{\text{caixa_calc}}$.

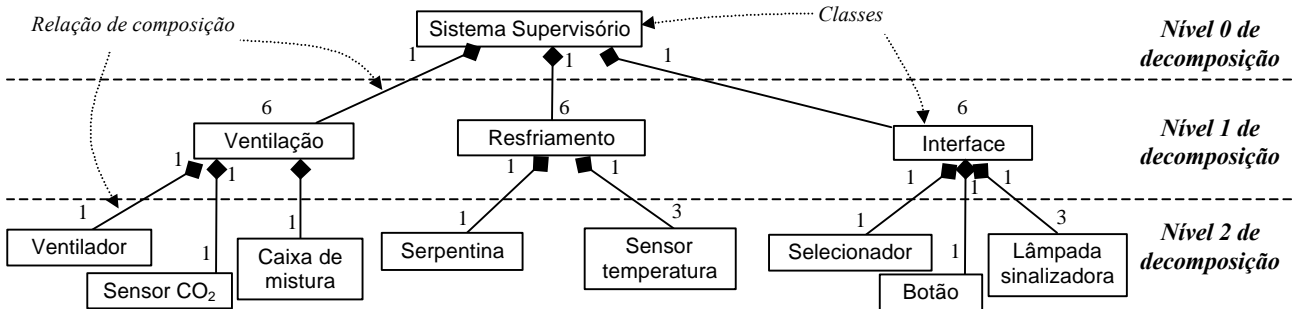


Figura 4. Decomposição estrutural do sistema supervisório representada em um diagrama de classes.

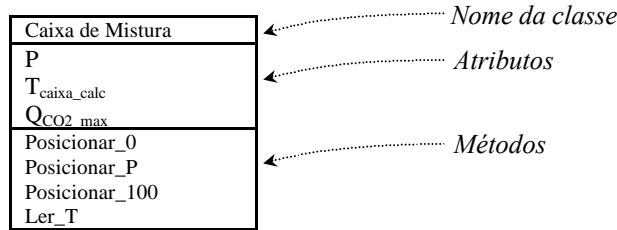


Figura 5. Definição da classe “Caixa de Mistura”.

De modo semelhante define-se os demais objetos (Figura 6):

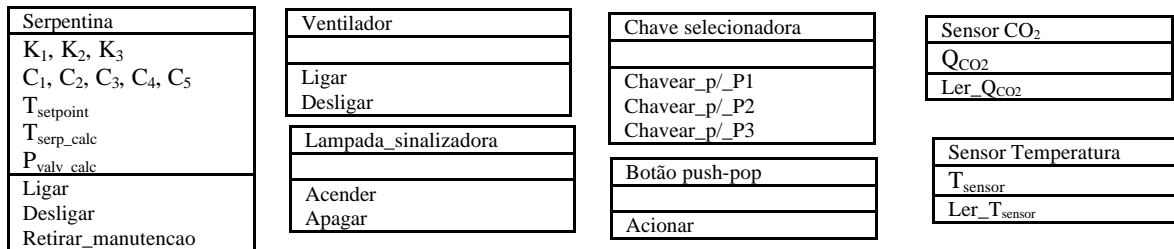


Figura 6. Definição das demais classes.

Nas descrições acima tem-se que

- K_1, K_2, K_3 representam as constantes do controlador da serpentina;
- C_1, C_2, C_3, C_4 e C_5 são constantes utilizadas no cálculo da temperatura do ar na saída da serpentina;
- $T_{setpoint}$ (constante) é o *set-point* da temperatura do ambiente utilizado pelo regulador PI;
- T_{serp_calc} (variável contínua) representa a temperatura calculada do ar na saída da serpentina;
- P_{valv_calc} (variável contínua) é a posição da válvula calculada pelo objeto;
- T_{sensor} (variável contínua) representa a temperatura lida pelo sensor;
- Q_{CO_2} (variável contínua) representa a quantidade de CO_2 detectada pelo sensor;
- Posições da chave selecionadora: P1 – desligado, P2 – resfriamento, P3 – ventilação.

Uma vez definidos os candidatos iniciais a objetos, passa-se então a modelagem dos seus comportamentos em Redes de Petri. Observa-se que, neste exemplo, a decomposição do sistema foi realizada de forma intuitiva. Neste caso através da fusão dos modelos dos objetos de um nível inferior, é possível verificar-se a coerência do nível superior. Uma outra abordagem possível é a construção dos modelos em redes de Petri de um nível superior para utilizá-los como base para a sua decomposição, como proposto em [Paludetto, 1991].

4.4. Modelagem dos comportamentos dos objetos

Para a representação da parte híbrida utiliza-se as redes de Petri e das redes Predicado Transição Diferenciais como suporte para formalização dos modelos representados nos diagramas UML.

Assim, uma vez definidos os candidatos a objetos, procede-se ao detalhamento dos modelos. A modelagem dos comportamentos em redes de Petri é realizada considerando que:

- A estrutura da rede define as possíveis evoluções para o estado de um objeto pertencente a uma determinada classe, ou seja, a rede modela o comportamento de uma classe.
- Cada *marca* presente nesta rede corresponde a instanciação de um objeto daquela classe.
- As *marcas* possuem atributos que correspondem aos atributos da classe que a define. Portanto, os atributos podem ser variáveis discretas (alteradas pelo disparo das transições), variáveis contínuas (alteradas pelas equações associadas aos *lugares*) ou constantes.
- Do ponto de vista da rede de Petri, a identificação do objeto é um atributo adicional constante da marca.
- A chamada de um método corresponde ao *disparo* de uma *transição*.
- A interface oferecida por uma classe corresponde a um conjunto de *transições*, cada uma correspondente a um método oferecido.

Para os objetos do Nível 2 da Figura 2 que apresentam comportamento puramente discreto tem-se os modelos em redes de Petri ilustrados na Figura 7. Conforme já definido, cada uma das *marcas* presentes nas redes da Figura 7 apresentam como atributos os atributos definidos nos modelos do item 4.2, mais um atributo adicional que corresponde a identificação do objeto (nome).

Para os objetos híbridos os modelos são construídos usando redes Predicado-Transição Diferenciais (Figura 8).

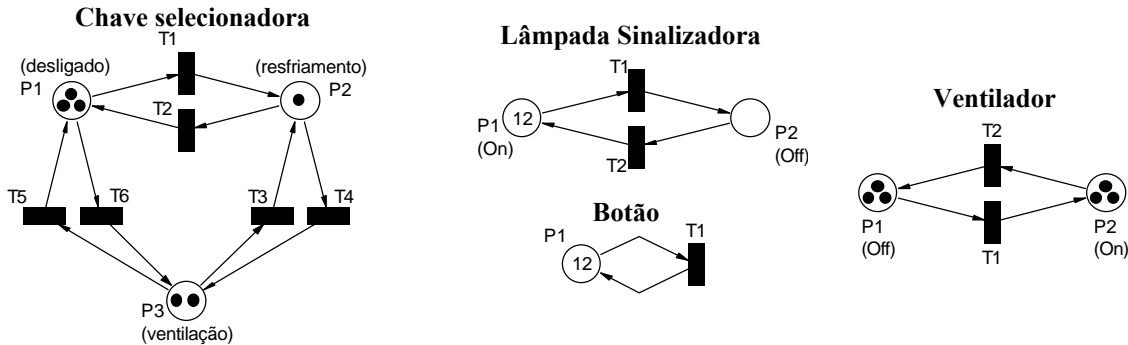


Figura 7. Modelo em redes de Petri dos objetos discretos.

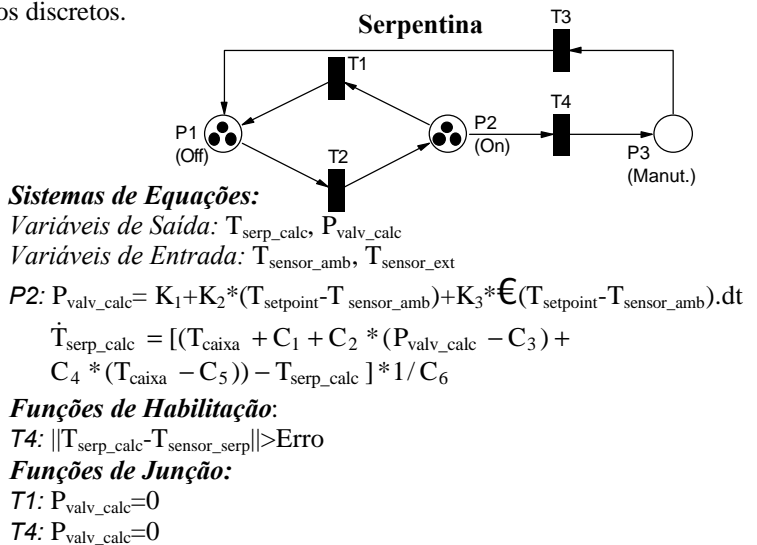
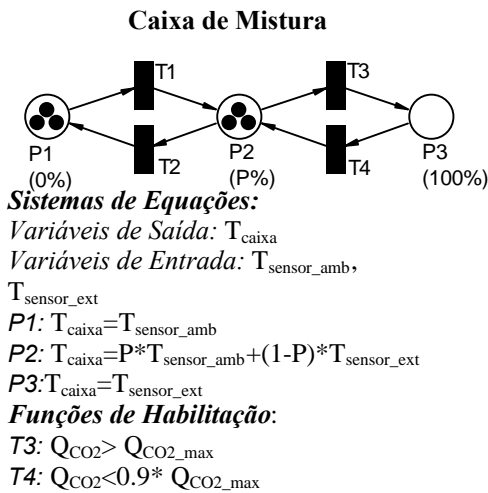


Figura 8. Modelo em redes de Petri dos objetos híbridos.

Observa-se que no modelo acima as variáveis utilizadas nos sistemas de equações correspondem a objetos de uma mesma zona. Por exemplo, no modelo da serpentina, quando uma *marca* estiver em P2, na resolução do sistema de equações será utilizada a variável T_{caixa} da marca correspondente a mesma zona. As variáveis T_{sensor_amb} , T_{sensor_serp} e T_{sensor_ext} correspondem às variáveis das diversas instâncias da classe “Sensor Temperatura” relacionadas aos diversos sensores de temperatura de uma mesma zona

Os sensores são objetos puramente passivos, cuja única variável tem seu valor atualizado constantemente através de comunicação com o dispositivo físico de detecção. Assim, não é necessário a especificação do seu comportamento.

Os modelos de comportamento acima propostos apresentam os possíveis estados dos objeto pertencentes a uma determinada classe e como estes estados podem ser modificados pela execução de métodos (associados a transições). No entanto não apresentam informações sobre como os objetos se relacionam entre si. Para tanto, deve acrescentar informações sobre as seqüências de atividades, ou sejam, sobre como os diversos processos são realizados.

4.5 Detalhamento dos Processos

Uma segunda etapa na modelagem do sistema consiste na modelagem dos processos envolvidos para um posterior detalhamento dos modelos dos objetos. Exemplos de processos (casos de utilização) são “ligar o resfriamento”, “detectar falha na serpentina”, “desligar o ar condicionado”, “detectar excesso de CO₂”, etc.

Para tanto, constrói-se inicialmente um modelo em redes de Petri de cada processo. Neste modelo, cada *transição* representa uma atividade desenvolvida. Os *lugares* representam estados e podem, eventualmente, estar relacionados a interações contínuas realizadas entre dois objetos. Estes modelos não incluem os recursos utilizados para as atividades, incluem apenas o sequenciamento de atividades. Como exemplo apresenta-se o modelo do processo “ligar o resfriamento”.

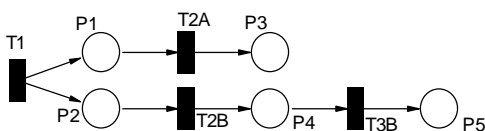


Figura 9. Modelo do processo “ligar o resfriamento”.

As *transições* acima representam os seguintes eventos:

- T1: usuário modifica a posição da chave seletora de “desligado” para “resfriamento”;
- T2A: modificação da posição da caixa de mistura para renovação P%;
- T2B: acionamento do ventilador;
- T3B: acionamento da serpentina.

Dos *lugares* acima, tem-se interações contínuas nos seguintes estados:

- P3: os dois objetos “Sensor Temperatura” correspondentes aos sensores de temperatura exterior e de temperatura ambiente devem fornecer as variáveis “ $T_{\text{sensor_amb}}$ ” e “ $T_{\text{sensor_ext}}$ ” para o objeto “caixa de mistura”, e o objeto “Sensor CO₂” deve fornecer a variável “ Q_{CO_2} ”;
- P5: o objeto “caixa de mistura” deve fornecer a variável “ T_{caixa} ” e o objeto “Sensor Temperatura” correspondente a temperatura ambiente deve fornecer a variável “ $T_{\text{sensor_amb}}$ ” para o objeto “serpentina”.

A partir da descrição do processo procede-se a construção de um diagrama de interação, onde associa-se uma transição ou um conjunto de transições à um método que é chamado por um objeto. Para o processo da Figura 9 tem-se o seguinte diagrama de colaboração:

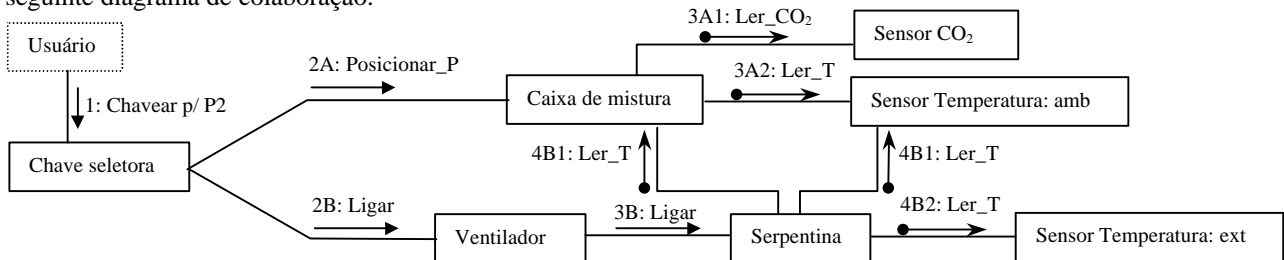


Figura 10. Diagrama de colaboração do processo “ligar o resfriamento”.

De acordo com a notação UML, neste diagrama utilizou-se o arco \longrightarrow para indicar uma interação discreta síncrona. Para indicar uma interação contínua, uma vez que a UML não dispõe de uma notação específica para este caso, define-se a utilização do arco $\bullet\longrightarrow$. Observa-se que o arco tem origem no objeto que solicita a informação do valor de uma variável e tem destino no objeto que fornece a informação.

Uma vez construído o diagrama de interações, detalha-se os diversos métodos do modelo em rede de Petri de cada objeto, associando à *transição* correspondente a respectiva parte do modelo em rede de Petri do processo. Como exemplo apresenta-se o detalhamento da *transição* T1 do modelo da chave seccionadora” (Figura 11). Neste modelo, a *transição* T2A representa a execução do método “Posicionar_P” e T2B a “Ligar” relativo ao ventilador:

O detalhamento apresentado acima deve ser realizado para todos os métodos correspondentes às interações discretas. Os métodos correspondentes às interações contínuas não necessitam de detalhamento, uma vez que sua execução está implícita nos sistemas de equações.

Para que os modelos sejam consistentes, o *disparo* de *transições* que adicionam *marcas* a um *lugar* do processo onde existe interação contínua (no exemplo da Figura 9, P3 e P5) deve corresponder a execução de um método da interface de um objeto híbrido (T2A – Posicionar_P, da caixa de mistura, e T3B – Ligar, da serpentina). O novo sistema de equações a ser resolvido do objeto híbrido deve ter como variáveis de entrada, variáveis de saída dos objetos com os quais realiza as interações contínuas representadas no diagrama de colaboração. Além disto, os objetos que fornecem as variáveis de entrada devem ter entre os seus métodos aqueles correspondentes a leitura de suas variáveis de saída.

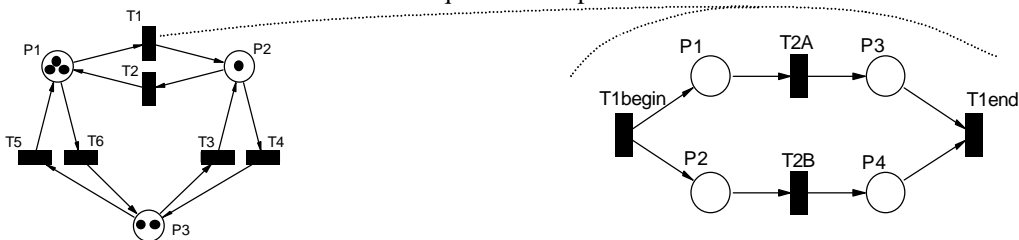


Figura 11. Adição de informações do processo aos modelos dos objetos.

4.6 Modelagem da comunicação entre objetos

Uma vez adicionada as informações sobre o processo, o passo seguinte refere-se a definir como, de fato, as diversas redes de Petri se relacionam entre si. Para tanto utiliza-se os modelos propostos em [Paludetto, 1991], onde a comunicação é realizada através de um fluxo de *marcas* entre objetos. Um exemplo é apresentado na Figura 12.

Uma outra opção para modelagem da comunicação entre os objetos é através de arcos habilitadores. Neste caso, a presença de uma *marca* em P3 habilitaria o *disparo* de T1 e a presença da *marca* correspondente em P2 habilitaria o *disparo* de T1end.

Quanto a comunicação contínua entre objetos, para que os modelos sejam consistentes, o *disparo* de *transições* que adicionam *marcas* a um *lugar* do modelo do processo onde existe interação contínua (no exemplo da Figura 9, P3 e P5) deve corresponder a execução de um método da interface de um objeto híbrido (T2A – Posicionar_P, da caixa de

mistura, e T3B – Ligar, da serpentina). O novo sistema de equações a ser resolvido do objeto híbrido deve ter como variáveis de entrada, variáveis de saída dos objetos com os quais realiza as interações contínuas representadas no diagrama de colaboração. Além disto, os objetos que fornecem as variáveis de entrada devem ter entre os seus métodos aqueles correspondentes a leitura de suas variáveis de saída.

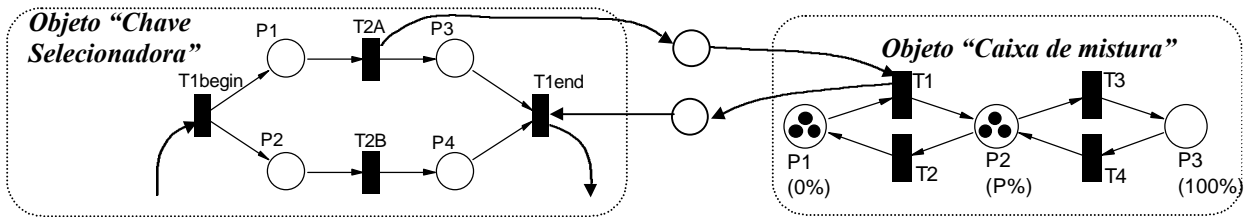


Figura 12. Modelagem da comunicação discreta entre os objetos através fluxo de marcas.

5. Considerações sobre a Análise do Sistema

No que se refere a análise dos modelos desenvolvidos, esta parte ainda está em desenvolvimento. No entanto, algumas conclusões já foram obtidas.

No que se refere a verificação da coerência dos modelos desenvolvidos, métodos podem ser propostos para verificar por exemplo, que não existam ciclos de compartilhamento de variáveis contínuas, ou que um objeto, em um determinado instante, utilize no seu sistema de equações uma variável externa que não está sendo calculada em nenhum outro objeto naquele instante.

Para a análise formal das propriedades da rede global (vivacidade, limitabilidade e reiniciabilidade), não existem métodos que garantam estas propriedades considerando a natureza híbrida do modelo. A determinação da habilitação ou não de uma transição depende da evolução no tempo das variáveis contínuas envolvidas na função de habilitação. Esta evolução é determinada pelos sistemas de equações. Como, para manter a flexibilidade do modelo, não foram impostas restrições quanto ao tipo de sistema de equação diferencial (se de 1ª ordem, 2ª ordem, não-linear, etc.), a resolução destes sistemas não pode ser garantida a não ser por simulação, o que inviabiliza uma solução teórica.

A análise deverá então ser realizada por etapas. Inicialmente será analisado o comportamento discreto de cada objeto isto é, da rede de Petri ordinária (ou predicado-transição) interna de cada objeto. Para tanto, cada objeto deve ser relativamente independente. Através da fusão sucessiva de objetos (observando, por exemplo a hierarquia de decomposição do sistema) deverá ser realizada a análise global do modelo. Observa-se no entanto, que a análise formal global só poderá ser realizada para a rede ordinária, o que dará uma indicação da coerência do modelo desenvolvido.

A parte continua será integrada na análise através a simulação. Poderá então verificar-se que uma determinada seqüência da rede de Petri ordinária global permanece possível tomando em conta a dinâmica continua dos modelos.

6. Conclusões

Este trabalho discute a aplicação dos conceitos de orientação a objetos para sistemas híbridos, propondo uma abordagem para construção de modelos baseados em redes de Petri e equações diferenciais. A notação UML é utilizada para auxiliar a representação de diferentes aspectos do sistema.

A utilização em conjunto de redes de Petri e orientação a objetos visa garantir a consistência dos modelos gerados para sistemas onde os processos e os recursos utilizados apresentam uma elevada complexidade. Neste sentido os modelos em redes de Petri são efetivos como elo de ligação entre os diferentes diagramas UML. Por sua vez a UML, através de suas visões complementares auxilia na representação da estrutura do sistema e na incorporação de informações sobre a seqüência de atividades, garantindo assim que os requisitos iniciais do sistema sejam obedecidos.

Em particular no que se refere a sistemas híbridos, através do exemplo do sistema de ar condicionado, verifica-se que a UML e as redes Predicado Transição Diferenciais se complementam, possibilitando o enfoque dos diferentes aspectos híbridos do sistema. De um lado as redes Predicado Transição Diferenciais explicitam os estados onde acontece uma evolução contínua do estado do objeto, de outro lado a UML, com pequenas modificações possibilita a representação, de uma forma explícita, da interação contínua entre os objetos.

Um resultado importante da aplicação dos conceitos de orientação a objetos é que deriva-se um sistema de inteligência distribuída, uma vez que cada objeto deve ter autonomia para gerenciar seus recursos e executar suas funções, ao contrário da abordagem funcional onde tem-se procedimentos complexos que controlam totalmente a execução de procedimentos de baixo nível.

Finalmente, observa-se que este trabalho é fortemente motivador para o desenvolvimento de uma metodologia para projeto de sistemas de controle híbridos que a partir dos modelos construídos possibilite a obtenção do software de controle através da implementação dos modelos em redes de Petri e em redes Predicado Transição Diferenciais.

7. Agradecimentos

Os autores agradecem o apoio das entidades FAPESP, CNPq e CAPES para o desenvolvimento deste trabalho.

5. Referências

- Alla, H. & David, R., 1998. "Continuous and Hybrid Petri Nets" Journal of Circuits, Systems and Computers, vol.8, n.1, pp 159-188.
- Antsaklis, P. J. & Nerode, A., 1998. "Hybrid Control Systems: An Introductory Discussion to the Special Issue" IEEE Transactions on Automatic Control, vol 43, n.4, pp 457-459.
- Baresi, L. & Pezzè, M., 1998. "On Formalizing UML with High-Level Petri Nets". Concurrent Object-Oriented Programming and Petri Nets - Lecture Notes in Computer Science - Springer Verlag (in press).
- Booch, G., 1994. "Object-Oriented Analysis and Design with Applications", 2nd ed. Addison-Wesley Longman, Inc. Harlow, England.
- Booch, G., Rumbaugh, J, Jacobson, I., 1998. "The Unified Modeling Language User Guide", Addison-Wesley Longman, Inc. Harlow, England.
- Champagnat, R., 1998. "Supervision des Systèmes Discontinus: Definition d'un Modèle Hybride et Pilotage en Temps-réel" Thèse de Doctorat, Université Paul Sabatier, Toulouse, France.
- Douglass, B. P., 1998. "Real-time UML: developing efficient objects for embedded systems", Addison-Wesley Longman, Inc. Harlow, England.
- Fraser, C.J. et al., 1993. "An Educational Perspective on Applied Mechatronics" Mechatronics, vol..3, n.1, pp.49-51.
- Garcia, H. E., 1997. "A hierarchical platform for implementing Hybrid Systems in Process Control", Control Eng. Practice, vol.5 n.6, pp 779-789.
- Gehrke, T., Goltz, U. & Wehrheim, H., 1998. "The Dynamic Models of UML: Towards a semantic and its application in the Development Process", Hildesheimer Informatik-Bericht 11/98, Institut für Informatik, Universität Hildesheim, Germany.
- Giese, H., Graf, J., Wirtz, G., 1999. "Closing the gap between object-oriented modeling of structure and behaviour", Proceedings of the Second International Conference on the Unified Modeling Language, Colorado, USA.
- Guéguen, H. & Lefebvre, M., 2000. "A comparison of mixed specification formalisms", Proc. ADPM 2000 - The 4th International Conference on Automation of Mixed Processes: Hybrid Dynamic Systems, Dortmund, Germany.
- Lemmon, M., He, K. X. & Markovsky, I., 1999. "Supervisory Hybrid Systems", IEEE Control System, vol.8 n.1, pp 42-55.
- Nimmo, I., 1999. "Future of Supervisory Systems in Process Industries: Lessons for Discrete Manufacturing", Annual Reviews in Control, vol.23, pp 45-52.
- Paludetto, M., 1991. "Sur la commande des procédés industriels: une méthodologie basée objects et réseaux de Petri" Thèse de Doctorat, Université Paul Sabatier, Toulouse, France.
- Paludetto, M. & Delatour, J., 1999. "UML et les réseaux de Petri – vers une sémantique des modèles dynamiques et une méthodologie de développement des systèmes temps réel", L'object, vol.5 n.3, pp 443-468.
- Valentin-Roubinet, C., 2000. "Hybrid Dynamic System verification with Mixed Petri Nets", Proc. ADPM 2000 - The 4th International Conference on Automation of Mixed Processes: Hybrid Dynamic Systems, Dortmund, Germany.

PETRI NETS AND OBJECT-ORIENTATION FOR HYBRID CONTROL SYSTEM DEVELOPMENT

Villani, Emilia

Dept. Eng. Mecatrônica e Sistemas Mecânicos, Escola Politécnica, USP
Av. Prof. Mello Moraes, 2001 São Paulo Brasil
emiliav@usp.br

Miyagi, Paulo Eigi

Dept. Eng. Mecatrônica e Sistemas Mecânicos, Escola Politécnica, USP
Av. Prof. Mello Moraes, 2001 São Paulo Brasil
pemiyagi@usp.br

Valette, Robert

LAAS – CNRS (Laboratoire d'Analyse et d'Architecture des Systèmes)
7, Avenue du Colonel Roche, 31077 Toulouse Cedex 4 FRANCE
robert@laas.fr

Abstract. This paper introduces a new approach for the development of hybrid control systems. The word "hybrid" means that the systems embody, simultaneously, characteristics of discrete event dynamic systems (where events are instantaneous and state variables are discrete numbers) and continuous systems (where events and state variables are continuous). Taking as a starting point a proposal of Petri Net associated with differential equations, we search a solution for the project of complex industrial systems by using a more structured approach. This new approach introduces the object-orientation concepts to the use of Petri Net associated with differential equations. The language UML is used in order to support the description of different aspects of the system. Through an example, we shown that UML and Petri Nets are complementary. By using both languages, it is possible to highlight different hybrid characteristics of the system.

Keywords. Hybrid systems, object orientation, Petri nets, UML

POLIEDROS DE CONFINAMENTO FINAL PARA SISTEMAS LINEARES DISCRETOS NO TEMPO COM ZONAS MORTAS

Basílio E. A. Milani

UNICAMP, Faculdade de Engenharia Elétrica e de Computação
Cx. P. 6101 - 13081-970 - Campinas, SP, Brasil

Alessandra D. Coelho

Escola de Engenharia Mauá, Depto. de Engenharia Elétrica
Praça Mauá 1, Bairro Mauá - 09580-900 - S. Caetano do Sul, SP, Brasil

Resumo. Este artigo trata de conjuntos de confinamento final uniforme positivamente invariantes para sistemas lineares discretos no tempo com controles em malha fechada com zonas mortas. O objetivo é delimitar um possível ciclo limite circundando a origem e estimar sua região de atração para sistemas instáveis em malha aberta. A delimitação do ciclo limite é realizada através da caracterização algébrica de conjuntos poliedrais positivamente invariantes e construção de uma estimativa poliedral convexa positivamente invariante do conjunto positivamente invariante mínimo contendo uma vizinhança arbitrariamente pequena da origem. Este conjunto convexo positivamente invariante delimita um ciclo limite se ele não contiver nenhum ponto de equilíbrio estável do sistema. A estimação da região de atração é realizada através da caracterização algébrica de funções de Lyapunov lineares por partes e construção de uma função de Lyapunov linear por partes garantindo confinamento final uniforme em um conjunto poliedral positivamente invariante que delimite o ciclo limite.

Palavras chaves. *Sistemas discretos no tempo, zonas mortas, ciclos limites, conjuntos de confinamento final, segundo método de Lyapunov.*

1. Introdução

Considere um sistema dinâmico autônomo \mathcal{S} . Grosso modo, um conjunto fechado $\Omega \subset \Omega_0$ é um conjunto de confinamento final uniforme local em Ω_0 se ele é o conjunto de confinamento final de todas as trajetórias de \mathcal{S} que iniciam em Ω_0 . Por sua vez, o conjunto Ω_0 é uma região de atração de Ω . Se Ω_0 é o espaço de estado de \mathcal{S} , tem-se confinamento final uniforme global. Também grosso modo, um conjunto Ω é positivamente invariante se todas as trajetórias de \mathcal{S} que iniciam em Ω permanecem em Ω para sempre. Estes conceitos associados ao segundo método de Lyapunov desempenham importante papel na análise e projeto de controladores de sistemas não lineares, permitindo a determinação e análise de estabilidade de conjuntos limite (La Salle, 1976) mais gerais que pontos de equilíbrio, tais como ciclos limite (movimentos periódicos).

Zonas mortas podem ocorrer em vários componentes de sistemas de controle, tais como sensores, amplificadores e atuadores. Possíveis efeitos indesejáveis das zonas mortas são perda de acuidade estática e instabilidade (Slotine e Li, 1981). A estabilidade de sistemas lineares com controles em malha fechada com zonas mortas tem sido muito estudada, considerando majoritariamente sistemas com uma entrada e uma saída e técnicas frequenciais (Zames e Falb, 1968; Slotine e Li, 1981; Khalil, 1992; Jönsson, 1997). Sistemas lineares com controles em malha fechada com zonas mortas são sistemas lineares por partes com possivelmente múltiplos pontos de equilíbrio e/ou ciclos limites. Em sistemas instáveis em malha aberta mas que são estáveis em malha fechada, suas trajetórias não convergem assintoticamente para a origem dentro da zona morta, mas sim para pontos de equilíbrio estáveis próximos à origem ou ciclos limite circundando a mesma. A determinação e verificação de estabilidade de pontos de equilíbrio são triviais. A determinação e análise de estabilidade de ciclos limite em sistemas multivariáveis em geral não são fáceis de serem feitas.

Este artigo trata de conjuntos convexos poliedrais de confinamento final uniforme, positivamente invariantes, para sistemas lineares discretos no tempo com controles em malha fechada com zonas mortas. O objetivo é a delimitação de ciclos limite circundando a origem e estimação de suas regiões de atração para sistemas instáveis em malha aberta, mas estáveis em malha fechada. O sistema em malha fechada é descrito por um modelo linear por partes nas regiões do espaço de estado definidas pela lei de realimentação com zonas mortas. Para delimitação de um ciclo limite são apresentados novos resultados sobre caracterização de conjuntos poliedrais positivamente invariantes e é proposto um procedimento para construção de uma estimativa poliedral compacta convexa, positivamente invariante, do mínimo conjunto positivamente invariante contendo uma vizinhança arbitrariamente pequena da origem. Este conjunto positivamente invariante convexo contém um ciclo limite se não contiver nenhum ponto de equilíbrio estável do sistema. Para estimação da região de atração de um ciclo limite são apresentados novos resultados sobre caracterização de funções de Lyapunov lineares por partes e é proposto um procedimento para construção de uma função de Lyapunov linear por partes garantindo confinamento final uniforme em um poliedro convexo positivamente invariante que delimite o ciclo limite.

2. Preliminares

Ao longo deste artigo: para duas matrizes reais $n \times m$ $A = (a_{ij})$ e $B = (b_{ij})$, $A \leq B$ é equivalente a $a_{ij} \leq b_{ij}$ para todo i, j tal que $1 \leq i \leq n$ e $1 \leq j \leq m$. Para qualquer $\epsilon \geq 0$, o conjunto $\epsilon\Omega$ é definido como $\{x = \epsilon y, y \in \Omega\}$.

Considere o sistema discreto no tempo:

$$x(k+1) = f(x(k)) \tag{1}$$

onde a função $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ é possivelmente não linear e $f(0) = 0$.

Definição 2.1: Um conjunto não vazio Ω definido no espaço de estado do sistema (1) é positivamente invariante com razão de contração $0 < \lambda \leq 1$ se $x(k) \in \Omega$ então $x(k+1) \in \lambda\Omega$ para todo $k \geq 0$.

Proposição 2.1: Um conjunto não vazio Ω definido no espaço de estado do sistema (1) é positivamente invariante com razão de contração $0 < \lambda \leq 1$ se e somente se a seguinte condição é satisfeita:

$$\mathcal{T} \subset \lambda\Omega$$

onde \mathcal{T} é o conjunto um passo à frente de Ω com respeito a (1):

$$\mathcal{T} = \{f(x) \in \mathbb{R}^n : x \in \Omega\}$$

Prova: Imediata da Definição 2.1. \square

Proposição 2.2: Seja \mathcal{S} a família de todos os conjuntos positivamente invariantes Ω , definidos no espaço de estado do sistema (1), com razão de contração λ , contendo um conjunto não vazio Γ . Se \mathcal{S} for não vazia, ela tem um elemento mínimo dado pela solução limite da recorrência:

$$\Omega_{i+1} = \Omega_i \cup \lambda^{-1}\mathcal{T}_i ; \quad \Omega_0 = \Gamma \tag{2}$$

onde \mathcal{T}_i é o conjunto um passo à frente de Ω_i .

Prova: Dados dois conjuntos positivamente invariantes $\Omega_1 \supset \Gamma$, $\Omega_2 \supset \Gamma$, aplicando a Definição 2.1, pode ser verificado que o conjunto $\Omega_1 \cap \Omega_2 \supset \Gamma$ é também positivamente invariante. Portanto \mathcal{S} é fechada sob a operação de intersecção e tem um elemento mínimo formado pela intersecção de todos os membros da família. Pode ser verificado que a solução limite da recorrência (2) é um elemento de \mathcal{S} : ela contém Γ e é positivamente invariante porque é obtida quando $\Omega_i = \Omega_i \cup \lambda^{-1}\mathcal{T}_i$, que corresponde a $\lambda^{-1}\mathcal{T}_i \subset \Omega_i$. Dado qualquer conjunto positivamente invariante $\Omega \supset \Gamma$, pode ser verificado que: $\Omega_1 \subset \Omega$ e se $\Omega_i \subset \Omega$ então $\Omega_{i+1} \subset \Omega$. Portanto, por indução, pode ser concluído que $\Omega_i \subset \Omega$ vale para todo i . Consequentemente o limite da recorrência (2) é um subconjunto de qualquer $\Omega \in \mathcal{S}$, sendo portanto o elemento mínimo de \mathcal{S} . \square

Definição 2.2: Um conjunto fechado $\Omega \subset \Omega_0$ definido no espaço de estado do sistema (1) é um conjunto de confinamento final uniforme local em Ω_0 se para toda condição inicial $x(0) \in \Omega_0$, existe um k_0 tal que $x(k) \in \Omega$ para todo $k \geq k_0$. O conjunto Ω_0 é a região de atração de Ω . Se Ω_0 é o espaço de estado completo do sistema (1) tem-se confinamento final uniforme global.

Proposição 2.3: Seja \mathcal{S} a família de todos os conjuntos de confinamento final uniforme no conjunto Ω_0 definido no espaço de estado do sistema (1). Se \mathcal{S} é não vazia, ela tem um elemento mínimo formado pela intersecção de todos os membros da família.

Prova: Aplicando a Definição 2.2, pode ser verificado que \mathcal{S} é fechada sob a operação de intersecção e portanto tem um elemento mínimo formado pela intersecção de todos os membros da família. \square

Definição 2.3: Uma função contínua $\Psi : \mathbb{R}^n \rightarrow \mathbb{R}$ é radialmente não limitada se

$$\Psi(x) \rightarrow \infty \text{ se } \|x\| \rightarrow \infty$$

em outras palavras, $\Psi(x)$ tende ao infinito quando x tende ao infinito em qualquer direção.

Considere o conjunto:

$$\mathcal{N}[\Psi, \check{b}, \hat{b}] = \{x \in \mathbb{R}^n : \check{b} \leq \Psi(x) \leq \hat{b}\} \tag{3}$$

onde $0 \leq \check{b} \leq \hat{b}$.

Definição 2.4: (Blanchini, 1999) Uma função definida positiva $\Psi : \mathbb{R}^n \rightarrow \mathbb{R}$ é uma função de Lyapunov, para o sistema (1) em $\mathcal{N}[\Psi, \check{b}, \hat{b}]$ (3), se para todo $x \in \mathcal{N}[\Psi, 0, \hat{b}]$,

$$\Psi(f(x)) \leq \max\{\lambda\Psi(x), \check{b}\} \tag{4}$$

vale para algum $0 \leq \lambda < 1$.

Os seguintes comentários sobre a Definição 2.4 são oportunos:

Os conjuntos de nível $\Psi(x) \leq b$ para $\check{b} \leq b \leq \hat{b}$ são conjuntos positivamente invariantes.

Para $\check{b} = 0$, $\mathcal{N}[\Psi, 0, \hat{b}]$ é uma região de estabilidade assintótica da origem com índice de convergência λ . A região de estabilidade assintótica da origem é local se \hat{b} é finito e é global se $\hat{b} \rightarrow \infty$ e $\Psi(x)$ é radialmente ilimitada.

Para $\tilde{b} > 0$, é garantido confinamento final uniforme do sistema (1) em $\mathcal{N}[\Psi, 0, \tilde{b}]$, com índice de convergência λ e região de atração $\mathcal{N}[\Psi, 0, \hat{b}]$. O confinamento final uniforme é local se \hat{b} é finito e é global se $\hat{b} \rightarrow \infty$ e $\Psi(x)$ é radialmente ilimitada.

Definição 2.5: O epígrafo da função $\Psi : \mathfrak{R}^n \rightarrow \mathfrak{R}$, restrito ao conjunto $\mathcal{Y} \subset \mathfrak{R}$, é o conjunto $\mathcal{E}_\Psi \subset \mathfrak{R}^{n+1}$:

$$\mathcal{E}_\Psi = \{(x, \zeta) : x \in \mathfrak{R}^n, \zeta \in \mathcal{Y}, \Psi(x) \leq \zeta\}$$

Lema 2.1: Sejam $\Psi_1 : \mathfrak{R}^n \rightarrow \mathfrak{R}$, $\Psi_2 : \mathfrak{R}^n \rightarrow \mathfrak{R}$, funções contínuas. Então $\Psi_2(x) \leq \max\{\Psi_1(x), \check{\rho}\}$ vale para todo $x \in \Psi_1(x) \leq \hat{\rho}$, $\hat{\rho} \geq \check{\rho}$ se e somente se seus epígrafos \mathcal{E}_{Ψ_1} , \mathcal{E}_{Ψ_2} , restritos ao conjunto $\mathcal{Y} = \{y \in \mathfrak{R} : \check{\rho} \leq y \leq \hat{\rho}\}$, satisfazem $\mathcal{E}_{\Psi_1} \subset \mathcal{E}_{\Psi_2}$.

Prova: Pode ser verificado que $\Psi_2(x) \leq \max\{\Psi_1(x), \check{\rho}\}$ para todo $x \in \Psi_1(x) \leq \hat{\rho}$ corresponde a:

$$\Psi_2(x) \leq \Psi_1(x) \text{ se } \check{\rho} \leq \Psi_1(x) \leq \hat{\rho} \quad (5)$$

$$\Psi_2(x) \leq \check{\rho} \text{ se } \Psi_1(x) \leq \check{\rho} \quad (6)$$

Suficiência: Da Definição 2.5, tem-se:

$$\mathcal{E}_{\Psi_1} = \{(x, \zeta) : x \in \mathfrak{R}^n, \Psi_1(x) \leq \zeta, \check{\rho} \leq \zeta \leq \hat{\rho}\} \quad (7)$$

$$\mathcal{E}_{\Psi_2} = \{(x, \zeta) : x \in \mathfrak{R}^n, \Psi_2(x) \leq \zeta, \check{\rho} \leq \zeta \leq \hat{\rho}\} \quad (8)$$

Inspecionando (7), (8), pode ser verificado que $\mathcal{E}_{\Psi_1} \subset \mathcal{E}_{\Psi_2}$ implica que $\Psi_2(x) \leq \zeta$ vale para todo x tal que $\Psi_1(x) \leq \zeta$, $\check{\rho} \leq \zeta \leq \hat{\rho}$. Consequentemente, tomando $\zeta = \check{\rho}$ tem-se $\Psi_2(x) \leq \check{\rho}$ para $\Psi_1(x) \leq \check{\rho}$ (6). Tomando $\zeta = \Psi_1(x)$ tem-se $\Psi_2(x) \leq \Psi_1(x)$ para $\check{\rho} \leq \Psi_1(x) \leq \hat{\rho}$ (5). Necessidade: Se $\mathcal{E}_{\Psi_1} \not\subset \mathcal{E}_{\Psi_2}$, em (7), (8) pode ser verificado que deve existir x_1 tal que $\Psi_1(x_1) \leq \check{\rho}$, $\check{\rho} \leq \zeta \leq \hat{\rho}$ e $\Psi_2(x_1) > \zeta$. Se $\check{\rho} \leq \Psi_1(x_1) \leq \hat{\rho}$, fazendo $\zeta = \Psi_1(x_1)$ tem-se $\Psi_2(x_1) > \Psi_1(x_1)$, que contradiz (5). Se $\Psi_1(x_1) \leq \check{\rho}$, fazendo $\zeta = \check{\rho}$ tem-se $\Psi_2(x_1) > \check{\rho}$, que contradiz (6). \square

Corolário 2.1: Uma função definida positiva $\Psi(x)$ é uma função de Lyapunov, para o sistema (1) em $\mathcal{N}[\Psi, \tilde{b}, \hat{b}]$, se e somente se existe um $0 \leq \lambda < 1$ tal que:

$$\mathcal{E}_\Psi \subset \mathcal{E}_{\Psi_f} \quad (9)$$

onde \mathcal{E}_Ψ e \mathcal{E}_{Ψ_f} são respectivamente epígrafos das funções $\Psi(x)$ e $\lambda^{-1}\Psi(f(x))$, restritos ao conjunto $\mathcal{Y} = \{y \in \mathfrak{R} : \lambda^{-1}\tilde{b} \leq y \leq \hat{b}\}$

Prova: Imediata aplicando a Definição 2.4 e Lema 2.1. \square

Considere o sistema linear discreto no tempo:

$$x(k+1) = Ax(k) + Bu(k) \quad (10)$$

onde: $x(k) \in \mathfrak{R}^n$ e $u(k) \in \mathfrak{R}^m$ são, respectivamente, variáveis de estado e controle; A , B , são matrizes reais de dimensões apropriadas.

Seja a lei de controle por realimentação com zonas mortas:

$$u(k) = dz(Fx(k)) \quad (11)$$

onde $F \in \mathfrak{R}^{m \times n}$ é constante. Os componentes de $dz(Fx)$ são dados por:

$$dz(Fx)_i = \begin{cases} f_i x + \tilde{u}_i & \text{se } f_i x < -\tilde{u}_i \\ 0 & \text{se } -\tilde{u}_i \leq f_i x \leq \hat{u}_i \\ f_i x - \hat{u}_i & \text{se } f_i x > \hat{u}_i \end{cases} \quad (12)$$

onde: f_i denota a i ésima linha da matriz F e \tilde{u}_i , \hat{u}_i denotam, respectivamente, as i ésimas linhas dos vetores limitante inferior $\tilde{u} \geq 0$, limitante superior $\hat{u} \geq 0$, da função zona morta.

Das Eqs. (10), (11) tem-se o sistema em malha fechada descrito pelo modelo não linear:

$$x(k+1) = Ax(k) + Bdz(Fx(k)) \quad (13)$$

Considerando todos os $x \in \mathfrak{R}^n$, cada uma das m componentes da lei de controle com zonas mortas (12) tem 3 possíveis situações: abaixo da zona morta, dentro da zona morta e acima da zona morta. Consequentemente \mathfrak{R}^n pode ser decomposto em $j = 1 : 3^m$ regiões $S(R_j, d_j) \subset \mathfrak{R}^n$, chamadas de regiões da zona morta, dadas pelos poliedros:

$$S(R_j, d_j) = \{x \in \mathfrak{R}^n ; R_j x \leq d_j\} \quad (14)$$

$$R_j = \begin{bmatrix} F_{iz}^j \\ -F_{iz}^j \\ -F_{az}^j \\ F_{bz}^j \end{bmatrix} ; d_j = \begin{bmatrix} \hat{u}_{iz}^j \\ \tilde{u}_{iz}^j \\ -\hat{u}_{az}^j \\ -\tilde{u}_{bz}^j \end{bmatrix} \quad (15)$$

onde F_{iz}^j , \hat{u}_{iz}^j , \check{u}_{iz}^j , F_{az}^j , \hat{u}_{az}^j , F_{bz}^j , \check{u}_{bz}^j , são matrizes e vetores apropriadamente formados pelas linhas de F , \hat{u} , \check{u} , relacionadas, respectivamente, aos componentes dentro, acima e abaixo da zona morta que caracterizam a j ésima região.

Dentro de cada região de zona morta $S(R_j, d_j)$, o sistema em malha fechada (13) pode ser representado por um modelo linear:

$$\begin{aligned} x(k+1) &= A_j x(k) + p_j \\ A_j &= [A + B_{az}^j F_{az}^j + B_{bz}^j F_{bz}^j] \\ p_j &= -B_{az}^j \hat{u}_{az}^j + B_{bz}^j \check{u}_{bz}^j \end{aligned} \quad (16)$$

onde B_{az}^j e B_{bz}^j são, respectivamente, matrizes apropriadamente formadas pelas colunas de B relacionadas à F_{az}^j e F_{bz}^j . Ao longo do artigo, será designado $j = 1$ para a região de comportamento linear de $dz(Fx)$, descrita por:

$$R_1 = \begin{bmatrix} F \\ -F \end{bmatrix} ; \quad d_1 = \begin{bmatrix} \hat{u} \\ \check{u} \end{bmatrix} ; \quad A_1 = A ; \quad p_1 = 0 \quad (17)$$

3. Poliedros positivamente invariantes

Considere o poliedro:

$$S(G, w) = \{x \in \mathfrak{R}^n : Gx \leq w\} \quad (18)$$

onde $G \in \mathfrak{R}^{r \times n}$ e $w > 0 \in \mathfrak{R}^r$.

Seja $T(G, w)$ o conjunto um passo à frente de $S(G, w)$ (18) com respeito ao sistema (13):

$$T(G, w) = \{Ax + Bdz(Fx) \in \mathfrak{R}^n ; x \in S(G, w)\} \quad (19)$$

A seguir, para simplificar a apresentação, as matrizes A_j são supostas não singular.

Proposição 3.1: Considerando as $j = 1 : 3^m$ regiões de zona morta (14), (15), (16), o conjunto $T(G, w)$ (19) é dado por:

$$T(G, w) = \bigcup_{j=1}^{j=3^m} T_j(G, w) \quad (20)$$

onde $T_j(G, w) \subset \mathfrak{R}^n$ é o conjunto um passo à frente de $S(G, w) \cap S(R_j, d_j)$ dado pelo poliedro:

$$\begin{bmatrix} G \\ R_j \end{bmatrix} A_j^{-1} x \leq \begin{bmatrix} w \\ d_j \end{bmatrix} + \begin{bmatrix} G \\ R_j \end{bmatrix} A_j^{-1} p_j \quad (21)$$

Prova: Notando que

$$\mathfrak{R}^n = \bigcup_{j=1}^{j=3^m} S(R_j, d_j) ; \quad S(G, w) = \bigcup_{j=1}^{j=3^m} S(G, w) \cap S(R_j, d_j)$$

da Eq. (19) tem-se:

$$T(G, w) = \{Ax + Bdz(Fx) \in \mathfrak{R}^n ; x \in \bigcup_{j=1}^{j=3^m} S(G, w) \cap S(R_j, d_j)\}$$

e portanto,

$$\begin{aligned} T(G, w) &= \bigcup_{j=1}^{j=3^m} T_j(G, w) \\ T_j(G, w) &= \{Ax + Bdz(Fx) \in \mathfrak{R}^n ; x \in S(G, w) \cap S(R_j, d_j)\} \end{aligned} \quad (22)$$

Das Eqs. (14)-(16) pode ser verificado que o conjunto (22) corresponde ao poliedro:

$$z = A_j x + p_j ; \quad \begin{bmatrix} G \\ R_j \end{bmatrix} x \leq \begin{bmatrix} w \\ d_j \end{bmatrix} \quad (23)$$

Notando que $x = A_j^{-1} z - A_j^{-1} p_j$, é fácil verificar que o poliedro (23) pode ser colocado na forma (21), completando a prova. \square .

Proposição 3.2: O poliedro $S(G, w)$ (18) definido no espaço de estado do sistema (13) é positivamente invariante com razão de contração $0 < \lambda \leq 1$ se e somente se para as $j = 1 : 3^m$ regiões da zona morta (14)-(16):

$$Gx \leq \lambda w \quad (24)$$

vale para todo x satisfazendo:

$$\begin{bmatrix} G \\ R_j \end{bmatrix} A_j^{-1} x \leq \begin{bmatrix} w \\ d_j \end{bmatrix} + \begin{bmatrix} G \\ R_j \end{bmatrix} A_j^{-1} p_j \quad (25)$$

Prova: Imediata aplicando as Proposições 2.1 e 3.1. \square

O Corolário seguinte dá uma formulação primal de programação linear (Bazaraa et al, 1990) para a Proposição 3.2.

Corolário 3.1: O poliedro $S(G, w)$ (18) definido no espaço de estado do sistema (13) é positivamente invariante com razão de contração $0 < \lambda \leq 1$ se e somente se para as $j = 1 : 3^m$ regiões de zona morta (14)-(16):

$$\max_{i,j} \{\sigma_j^i\} \leq 0 \quad \text{para : } 1 \leq i \leq r ; 1 \leq j \leq 3^m \quad (26)$$

onde os σ_j^i são obtidos resolvendo os seguintes problemas independentes de programação linear:

$$\sigma_j^i = \max_x g_i x - \lambda w_i \quad \text{sujeito a : } \begin{bmatrix} G \\ R_j \end{bmatrix} A_j^{-1} x \leq \begin{bmatrix} w \\ d_j \end{bmatrix} + \begin{bmatrix} G \\ R_j \end{bmatrix} A_j^{-1} p_j$$

onde: g_i, w_i são respectivamente, as i ésimas linhas de G, w ; R_j, d_j, A_j, p_j são relacionados às regiões de zona morta e modelo linear por partes (14)-(16). Além disso, seja x_j^i uma solução ótima relacionada com um $\sigma_j^i > 0$. Isto indica que $x_j^i \in T_j(G, w)$ (19) está fora de S_i , o i ésimo semi-espaço definindo $S(G, \lambda w)$:

$$S_i = \{x \in \mathbb{R}^n : g_i x \leq \lambda w_i\}$$

Prova: Imediata aplicando a Proposição 3.2. \square

Para um sistema (13) estável em malha fechada ($A + BF$ é Schur) mas com origem $x = 0$ instável, as trajetórias que iniciam na região dentro da zona morta convergem para ciclos limites ou pontos de equilíbrio estáveis nas regiões vizinhas. Neste caso, a Proposição 2.2 garante a existência de um conjunto positivamente invariante mínimo contendo uma vizinhança arbitrariamente pequena da origem. Se este conjunto mínimo não contiver nenhum ponto de equilíbrio estável de (13), pode-se concluir que ele contém um ciclo limite. Em outras palavras, ele delimita um ciclo limite.

Baseado na Proposição 2.2, o seguinte procedimento determina uma estimativa externa do mínimo poliedro convexo positivamente invariante contendo $x = 0$, definido no espaço de estado do sistema (13).

Procedimento 3.1: Estimação do mínimo poliedro convexo positivamente invariante com respeito a (13) contendo uma vizinhança poliedral compacta arbitrariamente pequena da origem $x = 0$.

1 - Inicialização

λ : razão de contração

$S(G, w)$: poliedro inicial convexo e compacto contendo a origem ($w > 0$)

itmax : número máximo de iterações

Fazer $it = 0$

2 - Iteração

2.1 - Calcular o conjunto um passo à frente $T_j(G, w)$ (Proposição 3.1)

2.2 - Verificar se $S(G, w)$ é positivamente invariante com razão de contração λ (Corolário 3.1):

Sim: **Pare** : $S(G, w)$ é o poliedro desejado

Não: Fazer $it = it + 1$ se $it > itmax$: **Pare** - convergência falhou.

2.3 - Calcular a aproximação poliedral convexa Cvp :

$$Cvp \approx \left[\bigcup_{j=1}^{j=3^m} T_j(G, w) \right] \cup S(G, w)$$

2.4 - Fazer $S(G, w) = Cvp$ Voltar para 2.1

A eficiência computacional e a precisão do Procedimento 3.1 são fortemente dependentes da aproximação poliedral Cvp usada no passo 2.3. A Cvp ideal é a casca convexa da união de poliedros, que é um problema com complexidade computacional não polinomial. Um bom compromisso entre eficiência computacional e precisão pode ser obtido tomando Cvp como o mínimo poliedro $S(G, \rho)$ tal que:

$$S(G, \rho) \supset \left[\bigcup_{j=1}^{j=3^m} T_j(G, w) \right] \cup S(G, w) \quad (27)$$

Neste caso, o Procedimento 3.1 expande progressivamente o poliedro inicial $S(G, w)$ mudando apenas seu vetor do lado direito w . A precisão de Cvp aumenta com o número de faces permitidas a $S(G, w)$.

4. Funções de Lyapunov lineares por partes

Considere a função linear por partes (FLP):

$$\Psi(x) = \max_{1 \leq i \leq r} w_i^{-1} \{g_i x + c_i\} \quad (28)$$

onde $x \in \mathfrak{R}^n$ e w_i, g_i, c_i são, respectivamente, iésimas linhas de $w > 0 \in \mathfrak{R}^r, G \in \mathfrak{R}^{r \times n}, c \leq 0 \in \mathfrak{R}^r$. É fácil verificar que a função $\Psi(x)$ (28) pode também ser definida como:

$$\Psi(x) = \min_{\epsilon \in \mathfrak{R}} \epsilon \quad s.a. \quad Gx + c \leq w\epsilon \quad (29)$$

Proposição 4.1: A FLP $\Psi(x)$ (28), (29) é definida positiva se e somente se existe uma matriz permutação P tal que:

$$P [G \quad c \quad w] = \begin{bmatrix} \tilde{G}_1 & \tilde{c}_1 & \tilde{w}_1 \\ \tilde{G}_2 & \tilde{c}_2 & \tilde{w}_2 \end{bmatrix} ; \quad \tilde{c}_1 = 0 ; \quad \tilde{c}_2 < 0 \quad (30)$$

e o poliedro $\tilde{G}_1 x \leq \tilde{w}_1$ é limitado.

Prova: Da Eq. (28) pode ser verificado que permutações de linhas de $[G \quad c \quad w]$ não mudam o valor de $\Psi(x)$ e também que $\Psi(0) = 0$ se e somente se $c \leq 0$ com pelo menos uma componente zero. Das Eqs. (29), (30) pode ser verificado que $\Psi(x) > 0$ para $x \neq 0$ vale se e somente se $\tilde{G}_1 x \leq \tilde{w}_1$ é limitado, concluindo a prova. \square

Proposição 4.2: A FLP $\Psi(x)$ (28), (29) é radialmente não limitada se e somente se o poliedro $Gx \leq w$ é limitado.

Prova: Necessidade: Se $Gx \leq w$ é não limitado, ele deve ter uma direção extrema x_0 (Bazaraa et all., 1990) tal que $G\mu x_0 \leq 0 \forall \mu \geq 0$. Portanto, das Eqs. (28), (29) pode ser verificado que $\Psi(\mu x_0)$ não tende ao infinito quando $\|\mu x_0\| \rightarrow \infty$ e portanto $\Psi(x)$ não é radialmente não limitada. Suficiência: Se $Gx \leq w$ é limitado, não existe x_0 tal que $G\mu x_0 \leq 0 \forall \mu \geq 0$. Portanto, na Eq. (29) pode ser verificado que para qualquer x, Gx tem sempre pelo menos uma componente positiva que causa $\Psi(\mu x) \rightarrow \infty$ quando $\|\mu x\| \rightarrow \infty$, concluindo a prova. \square

Corolário 4.1: Qualquer FLP definida positiva $\Psi(x)$ (28), (29) é radialmente não limitada.

Prova: Imediata aplicando as Proposições 4.1 e 4.2. \square

Aplicando a Definição 2.5, pode ser verificado que o epígrafo da FLP $\Psi(x)$ (28) restrito ao conjunto

$$\mathcal{Y} = \{y \in \mathfrak{R} : \lambda^{-1} \check{b} \leq y \leq \hat{b}\} \quad (31)$$

é dado por:

$$\mathcal{E}_{\Psi} = \{(x, \epsilon) \in \mathfrak{R}^{n+1} : Gx + c \leq w\epsilon ; \lambda^{-1} \check{b} \leq \epsilon \leq \hat{b}\} \quad (32)$$

Similarmente, pode também ser verificado que para o sistema (13), o epígrafo de $\lambda^{-1} \Psi(f(x))$ restrito ao conjunto \mathcal{Y} (31) é dado por:

$$\mathcal{E}_{\Psi_f} = \{(x, \epsilon) \in \mathfrak{R}^{n+1} : G[Ax + B \text{sat}(Fx)] + c \leq \lambda w\epsilon ; \lambda^{-1} \check{b} \leq \epsilon \leq \hat{b}\} \quad (33)$$

Proposição 4.3: Uma FLP definida positiva $\Psi(x)$ (28), (29), é uma função de Lyapunov para o sistema (13) no conjunto $\mathcal{N}[\Psi, \check{b}, \hat{b}]$ se e somente se existe $0 \leq \lambda < 1$ tal que os epígrafos (32), (33) satisfazem a relação de inclusão $\mathcal{E}_{\Psi} \subset \mathcal{E}_{\Psi_f}$.

Prova: Imediata aplicando o Corolário 2.1. \square

Proposição 4.4: Uma FLP definida positiva $\Psi(x)$ (28), (29) é uma função de Lyapunov para o sistema (13) no conjunto $\mathcal{N}[\Psi, \check{b}, \hat{b}]$ se e somente se para as $j = 1 : 3^m$ regiões da zona morta (14)-(16) existe $0 \leq \lambda < 1$ tal que

$$\begin{bmatrix} GA_j & -\lambda w \end{bmatrix} \begin{bmatrix} x \\ \epsilon \end{bmatrix} \leq -Gp_j - c \quad (34)$$

vale para qualquer (ϵ, x) satisfazendo:

$$\begin{bmatrix} G & -w \\ R_j & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ \epsilon \end{bmatrix} \leq \begin{bmatrix} -c \\ d_j \\ \check{b} \\ -\lambda^{-1} \check{b} \end{bmatrix} \quad (35)$$

ou, em outras palavras, se e somente se existe $0 \leq \lambda < 1$ tal que o poliedro (35) é um subconjunto do poliedro (34).

Prova: A Proposição 4.3 fornece:

$$\mathcal{E}_{\Psi} \subset \mathcal{E}_{\Psi_f} \quad (36)$$

Notando que

$$\mathfrak{R}^n = \bigcup_{j=1}^{j=3^m} S(R_j, d_j) \quad ; \quad \mathcal{E}_\Psi = \mathcal{E}_\Psi \cap (\mathfrak{R}^n \times \mathfrak{R})$$

tem-se

$$\mathcal{E}_\Psi = \bigcup_{j=1}^{j=3^m} \mathcal{E}_\Psi \cap (S(R_j, d_j) \times \mathfrak{R})$$

Portanto, Eq. (36) vale se e somente se para $j = 1 : 3^m$ valer:

$$\mathcal{E}_\Psi \cap (S(R_j, d_j) \times \mathfrak{R}) \subset \mathcal{E}_{\Psi_f} \quad (37)$$

Para $x \in S(R_j, d_j)$, nas Eqs. (16), (33) pode ser verificado que \mathcal{E}_{Ψ_f} corresponde ao poliedro:

$$\begin{bmatrix} GA_j & -\lambda w \end{bmatrix} \begin{bmatrix} x \\ \epsilon \end{bmatrix} \leq -Gp_j - c \quad (38)$$

De (14), (15), (32) pode ser verificado que $\mathcal{E}_\Psi \cap (S(R_j, d_j), \mathfrak{R})$ corresponde ao poliedro:

$$\begin{bmatrix} G & -w \\ R_j & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x \\ \epsilon \end{bmatrix} \leq \begin{bmatrix} -c \\ d_j \\ \hat{b} \\ \lambda^{-1}\hat{b} \end{bmatrix}$$

concluindo a prova. \square

O Corolário seguinte dá uma formulação primal de programação linear à Proposição 4.4.

Corolário 4.2: Uma FLP definida positiva $\Psi(x)$ (28), (29), é uma função de Lyapunov para o sistema (13) no conjunto $\mathcal{N}[\Psi, \check{b}, \hat{b}]$ se e somente se existe $0 \leq \lambda < 1$ tal que:

$$\max_{i,j} \{\sigma_j^i\} \leq 0 \quad \text{para : } 1 \leq i \leq r \quad ; \quad 1 \leq j \leq 3^m \quad (39)$$

onde σ_j^i são obtidos resolvendo os seguintes problemas independentes de programação linear:

$$\sigma_j^i = \max_{x,\epsilon} g_i A_j x - \lambda w_i \epsilon + g_i p_j + c_i \quad \text{sujeito a : } \begin{array}{l} Gx - w\epsilon \leq -c \\ R_j x \leq d_j \\ \lambda^{-1}\hat{b} \leq \epsilon \leq \hat{b} \end{array} \quad (40)$$

onde g_i, c_i, w_i são, respectivamente, as i ésimas linhas de G, c, w e R_j, d_j, A_j, p_j são relacionados às regiões de zona morta e modelo linear por partes (14)-(16). Além disso, seja (x_j^i, ϵ_j^i) uma solução ótima relacionada a um $\sigma_j^i > 0$. Isto indica que $(x_j^i, \epsilon_j^i) \in \mathcal{E}_\Psi$ (32) está fora do i ésimo semi-espaco definindo \mathcal{E}_{Ψ_f} (33) na j ésima região de zona morta:

$$\mathcal{H}_j^i = \{(x, \epsilon) \in \mathfrak{R}^{n+1} : g_i A_j x - \lambda w_i \epsilon \leq -g_i p_j - c_i\} \quad (41)$$

Prova: Imediata aplicando a Proposição 4.4. \square

O Corolário 4.2 é computacionalmente muito eficiente e pode efetivamente tratar problemas de grande porte. Ele corresponde a um conjunto de problemas de programação linear independentes, que podem ser resolvidos usando ferramentas computacionais poderosas. O procedimento seguinte usa a Proposição 4.3 e Corolário 4.2, para construção de uma FLP de Lyapunov garantindo confinamento final uniforme do sistema (13) em um poliedro compacto positivamente invariante dado.

A seguir, a FLP $\Psi(x)$ (28), (29), o epígrafo \mathcal{E}_Ψ (32) e os semi-espacos \mathcal{H}_j^i (41), serão, respectivamente, representados em forma compacta como $\Psi[G, c, w]$, $\mathcal{E}_\Psi[G, c, w]$ e $\mathcal{H}[g_i A_j, c_i + g_i p_j, \lambda w_i]$.

Procedimento 4.1: Construção de uma FLP de Lyapunov garantindo confinamento final uniforme do sistema (13) no poliedro positivamente invariante $Gx \leq w$.

1 - Inicialização

$\Psi[G, c = 0, w]$ - FLP candidata inicial a função de Lyapunov em $\mathcal{N}[\Psi, \check{b} = 1, \hat{b} = \infty]$, correspondente a um poliedro positivamente invariante $Gx \leq w$ contendo a origem ($w > 0$) com razão de contração λ .

2 - Verificar se a $\Psi[G, c, w]$ inicial é função de Lyapunov (Corolário 4.2) :

- Sim : **Pare** :

$\Psi[G, c = 0, w]$ é uma função de Lyapunov em $\mathcal{N}[\Psi, \check{b} = 1, \hat{b} = \infty]$.

- Não :

Identificar :

$\mathcal{H}[g_i A_j, c_i + g_i p_j, \lambda w_i]$ - o semi-espaço de \mathcal{E}_{Ψ_j} com mínimo ϵ_j^i que não satisfaz o Corolário 4.2

Atualizar FLP :

$$\mathcal{E}_{\Psi}[G, c, w] = \mathcal{E}_{\Psi}[G, c, w] \cap \mathcal{H}[g_i A_j, c_i + g_i p_j, \lambda w_i]$$

Fazer $\hat{b}_a = \epsilon_j^i$

3 - Iteração

3.1 - Verificar se $\Psi[G, c, w]$ é uma função de Lyapunov (Corolário 4.2) :

- Sim: **Pare** :

$\Psi[G, c, w]$ é uma função de Lyapunov em $\mathcal{N}[\Psi, \check{b} = 1, \hat{b} = \infty]$.

- Não: Identificar :

$\mathcal{H}[g_i A_j, c_i + g_i p_j, \lambda w_i]$ - o semi-espaço de \mathcal{E}_{Ψ_j} com mínimo ϵ_j^i que não satisfaz o Corolário 4.2

3.2 Verificar se $\epsilon_j^i < \hat{b}_a$

- Sim : **Pare** :

$\Psi[G, c, w]$ é uma função de Lyapunov em $\mathcal{N}[\Psi, \check{b} = 1, \hat{b} = \hat{b}_a]$.

- Não :

Atualizar FLP :

$$\mathcal{E}_{\Psi}[G, c, w] = \mathcal{E}_{\Psi}[G, c, w] \cap \mathcal{H}[g_i A_j, c_i + g_i p_j, \lambda w_i]$$

Fazer : $\hat{b}_a = \epsilon_j^i$ Retornar para 3.1

O resultado principal do procedimento 4.1 é a região de atração $\mathcal{N}[\Psi, 0, \hat{b}]$ com razão de convergência λ do conjunto de confinamento final positivamente invariante $Gx \leq w$. Utilizando $Gx \leq w$ dado pelo Procedimento 3.1, contendo um ciclo limite, o Procedimento 4.1 fornece uma boa estimativa da região de atração do referido ciclo limite.

5. Exemplo numérico

Considere o sistema com controle em malha fechada com zona morta:

$$x(k+1) = Ax(k) + Bu(k) ; \quad u(k) = dz(Fx(k))$$

$$A = \begin{bmatrix} 1.00 & 0.05 \\ -0.25 & 1.10 \end{bmatrix} ; \quad B = \begin{bmatrix} 0 \\ 0.05 \end{bmatrix} ; \quad \tilde{u} = \hat{u} = 2.0 ; \quad F = [-1.00 \quad -6.00] \quad (42)$$

Pode ser verificado que o sistema (42) tem um ponto de equilíbrio instável (0,0) na região dentro da zona morta, não tem pontos de equilíbrio em outras regiões e a matriz $A + BF$ é Schur. Neste caso, as trajetórias de (42) iniciando em $\{\mathbb{R}^2 - (0,0)\}$ devem convergir para um ciclo limite. Neste caso, o conjunto de confinamento final uniforme global mínimo, de acordo com a Definição 2.2, é o conjunto formado pela união de (0,0) com o ciclo limite. Usando a aproximação convexa poliedral (27) e razão de contração $\lambda = .998$, o poliedro positivamente invariante mínimo obtido pelo Procedimento 3.1 é:

$$S(G, w) = \{x \in \mathbb{R}^2 : -w \leq Gx \leq w\} ; \quad G = \begin{bmatrix} 1.0000 & 0 \\ 0.9877 & 0.1564 \\ 0.9511 & 0.3090 \\ 0.8910 & 0.4540 \\ 0.8090 & 0.5878 \\ 0.7071 & 0.7071 \\ 0.5878 & 0.8090 \\ 0.4540 & 0.8910 \\ 0.3090 & 0.9511 \\ 0.1564 & 0.9877 \\ 0.0000 & 1.0000 \\ -0.1564 & 0.9877 \\ -0.3090 & 0.9511 \\ -0.4540 & 0.8910 \\ -0.5878 & 0.8090 \\ -0.7071 & 0.7071 \\ -0.8090 & 0.5878 \\ -0.8910 & 0.4540 \\ -0.9511 & 0.3090 \\ -0.9877 & 0.1564 \end{bmatrix} ; \quad w = \begin{bmatrix} 0.3526 \\ 0.3438 \\ 0.3582 \\ 0.3794 \\ 0.4277 \\ 0.4977 \\ 0.5733 \\ 0.6453 \\ 0.7069 \\ 0.7554 \\ 0.7889 \\ 0.8057 \\ 0.8054 \\ 0.7878 \\ 0.7537 \\ 0.7044 \\ 0.6420 \\ 0.5683 \\ 0.4901 \\ 0.4139 \end{bmatrix} \quad (43)$$

A FLP de Lyapunov obtida pelo Procedimento 4.1 garante confinamento final uniforme global do sistema (42) em $S(G, w)$ (43). Pode ser verificado que o único ponto de equilíbrio do sistema (42) em $S(G, w)$ é a origem $(0, 0)$ que é instável. Portanto, pode-se concluir que $S(G, w)$ delimita um ciclo limite do sistema (42) com região de atração estimada $\{\mathbb{R}^2 - (0, 0)\}$, igual à região de atração verdadeira. O conjunto de confinamento final uniforme global $S(G, w)$ e algumas trajetórias do sistema (42) são apresentadas na Figura 1. Pode ser verificado nessa figura que $S(G, w)$ delimita com boa aderência o ciclo limite, sendo também uma boa aproximação poliedral convexa externa do mínimo conjunto de confinamento final uniforme global do sistema (42).

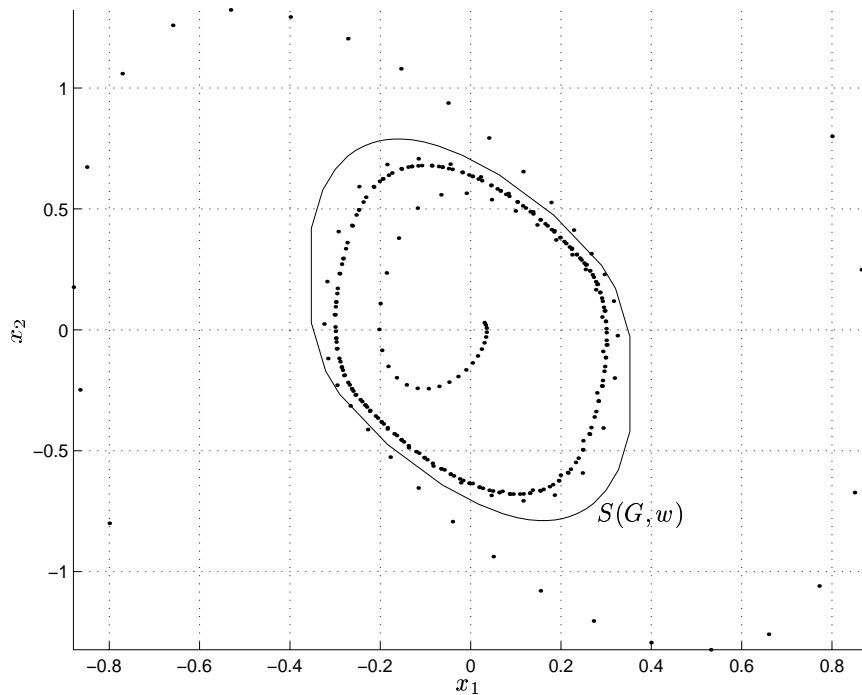


Figura 1. : Conjunto de confinamento final uniforme $S(G, w)$

6. Conclusão

Este artigo tratou de conjuntos de confinamento final uniforme positivamente invariantes para sistemas lineares discretos no tempo com controles em malha fechada com zonas mortas. O objetivo principal foi delimitar um possível ciclo limite circundando a origem e estimar sua região de atração para sistemas multivariáveis instáveis em malha aberta. Usando um modelo linear por partes do sistema controlado em malha fechada, os seguintes resultados foram apresentados. Delimitação de ciclos limites através da caracterização e construção de poliedros positivamente invariantes: novas condições necessárias e suficientes para um poliedro convexo fechado ser positivamente invariante; baseado em formulação de programação linear dessas condições, um procedimento foi proposto para construção de uma estimativa poliedral convexa positivamente invariante do conjunto positivamente invariante mínimo contendo uma vizinhança arbitrariamente pequena da origem; este conjunto convexo positivamente invariante contém um ciclo limite se ele não contiver nenhum ponto de equilíbrio estável do sistema. Estimação da região de atração de ciclos limites através da caracterização e construção de FLPs de Lyapunov: novas condições necessárias e suficientes para uma FLP ser função de Lyapunov; baseado em formulação de programação linear dessas condições, um procedimento foi proposto para construção de uma FLP de Lyapunov garantindo confinamento final uniforme em um conjunto poliedral positivamente invariante que delimite o ciclo limite.

7. Referências

- Bazaraa, M.S., Jarvis J.J. and Sherali, H.D., 1990, "Linear Programming and Network Flows", John Wiley & Sons, New York.
- Blanchini, F., 1999, "Set Invariance in Control: A Survey", *Automatica*, Vol.35 pp. 1747-1746.
- Jönsson, U., 1997, "Stability Criterion for Systems with Neutrally Stable Modes and Deadzone Nonlinearities", CALTECH Report CDS97-007.

Khalil, H.K., 1992, "Nonlinear Systems", Macmillan, New York.

La Salle, J.P., 1976, "The Stability of Dynamical Systems", SIAM, Philadelphia.

Slotine, J. E. and Li W., 1981, "Applied Nonlinear Control", Prentice-Hall, Englewood Cliffs NJ.

Zames, G. and Falb, P.L., 1968, "Stability Conditions for Systems with Monotone and Slope-restricted Nonlinearities", *SIAM Journal of Control*, Vol. 6, No. 1, pp. 89-108.

ULTIMATE BOUNDEDNESS POLYHEDRA FOR DISCRETE-TIME LINEAR SYSTEMS WITH DEADZONES

Abstract. This paper is concerned with positively invariant convex polyhedral uniform ultimate boundedness sets for linear discrete-time systems with deadzone feedback control laws. The objective is delimitation of a possible limit cycle encircling origin and estimation of its region of attraction for open-loop unstable systems. Limit cycle delimitation is performed via algebraic characterization of polyhedral positively invariant sets and construction of a positively invariant convex compact polyhedral estimate of the minimal positively invariant set containing an arbitrarily small neighborhood of origin. This convex positively invariant set delimits a limit cycle if it does not contain any stable equilibrium point. The region of attraction estimation is performed via algebraic characterization of piecewise-linear Lyapunov functions and construction of a piecewise-linear Lyapunov function assuring uniform ultimate boundedness in a convex positively invariant polyhedral set delimiting the limit cycle.

Key Words. *Discrete-time systems, deadzones, limit cycles, ultimate boundedness sets, Lyapunov second method.*

MODELAGEM E SIMULAÇÃO DINÂMICA DE UM ROBÔ BÍPEDE

Daniel Carmona de Campos.*Universidade Estadual de Campinas. Departamento de Mecânica Computacional***Carlos André Dias Bezerra***Universidade Federal do Ceará. Departamento de Engenharia Mecânica e Produção***Douglas Eduardo Zampieri***Universidade Estadual de Campinas. Departamento de Mecânica Computacional***André Mendeleck***Universidade Estadual de Campinas. Departamento de Mecânica Computacional*

Resumo. Muitos estudos têm sido feitos na área de robôs bípedes nas últimas três décadas, devido ao potencial de mobilidade em diferentes tipos de superfície e ambientes, comparados com os robôs com rodas. Como consequência desta característica, eles são mais complexos e a determinação de um modelo dinâmico eficiente torna-se extremamente importante para simplificar a obtenção das equações sem comprometer o equilíbrio. Este trabalho consiste na modelagem dinâmica e simulação de um robô bípede formado por sete corpos rígidos conectados por juntas de rotação, com um atuador independente cada. Para a obtenção das 6 equações de movimento que representam o sistema mecânico foi adotada a metodologia de Newton-Euler-Jourdain. As equações de movimento foram implementadas em software numérico para a simulação do modelo dinâmico. Através destas equações os torques foram obtidos em função dos parâmetros geométricos, de massa, ângulos entre os corpos e suas derivadas, as quais foram previamente obtidas através da simulação cinemática ao se estabelecer as trajetórias para o pé e pelve. Um estudo comparativo entre dois tipos de trajetória para o pé é apresentado, mostrando as diferentes características.

Palavras chave: Robô bípede, Modelagem dinâmica, Simulação.

1. Introdução

Ao longo dos últimos anos muitos pesquisadores têm sido atraídos para a área de sistemas robóticos bípedes (Raibert, 1996) devido à sua semelhança antropomórfica e devido ao potencial de mobilidade similar ao ser humano, que os permite subir escadas, andar em áreas contendo obstáculos, superfícies inclinadas e rugosas e até mesmo andar sem atuadores (McGeer, 1990). Além disso, os robôs bípedes apresentam potencial de substituição do ser humano em áreas perigosas, para uso militar, reabilitação de deficientes físicos, entre outros e possuem mobilidade muito melhor que os robôs com rodas em geral. Todavia, nenhum dos robôs bípedes até agora estudados apresenta a mobilidade e destreza humana (Lum et al, 1999), pois os modelos usados são simplificados através da redução do número de graus de liberdade, da substituição das juntas por uniões de rotação e ao se considerar os segmentos como corpos rígidos (Popovic et al, 1999).

Desde 1968 várias técnicas de modelagem e controle têm sido propostas (Frank, 1968; Hemani and Wyman, 1979; Furusho and Masubuchi, 1986; Vukobratovic et al, 1990 e Raibert, 1996) no intuito de se entender e solucionar o principal problema inerente ao robô bípede, que é a sua instabilidade durante a fase em que somente um pé está em contato com o chão. Porém, foi somente em 1973 que o primeiro robô bípede, o WL-5, foi projetado, construído e citado na literatura científica por Kato e seus colegas (Kato et al, 1974).

O modo de andar de um robô bípede é obtido através do movimento alternado de suas pernas, excetuando-se os robôs que pulam (Raibert, 1984 e Hodings, 1988). Pode-se dizer que o modo de andar (*gait*) é constituído por duas fases distintas: biapoiada (*stance fase*) quando os pés do robô bípede estão em contato com o chão e a fase de balanço (*balance fase*) quando somente um pé está apoiado no chão. Nesta última fase a perna que não se encontra apoiada é denominada de perna em balanço (*swinging leg*).

Na fase de balanço a estrutura é caracterizada como sendo uma cadeia cinemática aberta e a fase biapoiada como uma cadeia cinemática fechada, pois se conectam ambas as extremidades com o solo. Desta maneira o robô bípede é classificado como uma estrutura híbrida, pois possui os dois tipos de cadeias, dependendo do instante do movimento.

Existem basicamente 2 modos de andar de um robô bípede. O andar estático, onde a projeção do centro de gravidade do sistema mecânico sempre permanece dentro da área de apoio do pé no chão e o andar dinâmico onde a projeção em determinados instantes permanece fora desta região.

O estudo do robô bípede consiste na modelagem cinemática e dinâmica de um sistema mecânico variante no tempo, ou seja, um sistema em que os parâmetros das equações de movimento mudam dependendo do instante de tempo considerado, tanto em quantidade, como na forma. A partir da modelagem dinâmica realizada neste trabalho, torna-se possível o desenvolvimento do sistema de controle do robô bípede, que o permitirá se movimentar no modo estático. As

simulações cinemática e dinâmica darão subsídios para a determinação dos torques aplicados nas juntas, permitindo assim selecionar os atuadores adequados à movimentação do robô bípede. Assim o objetivo deste trabalho é a concepção, modelagem e simulação dinâmica de um robô móvel do tipo bípede, semelhante a uma estrutura antropomórfica com três rotações por perna.

2. Modelagem

O robô bípede a ser simulado computacionalmente, é constituído por 7 corpos, denominados de pé direito - segmento 1, perna (popular canela) direita - segmento 2, coxa direita - segmento 3, pelve (popular bacia) - segmento 4, coxa esquerda - segmento 5, perna esquerda - segmento 6, pé esquerdo - segmento 7, conforme Figura 1(b). Todas as juntas de união entre os segmentos são de rotação, com 1 grau de liberdade.

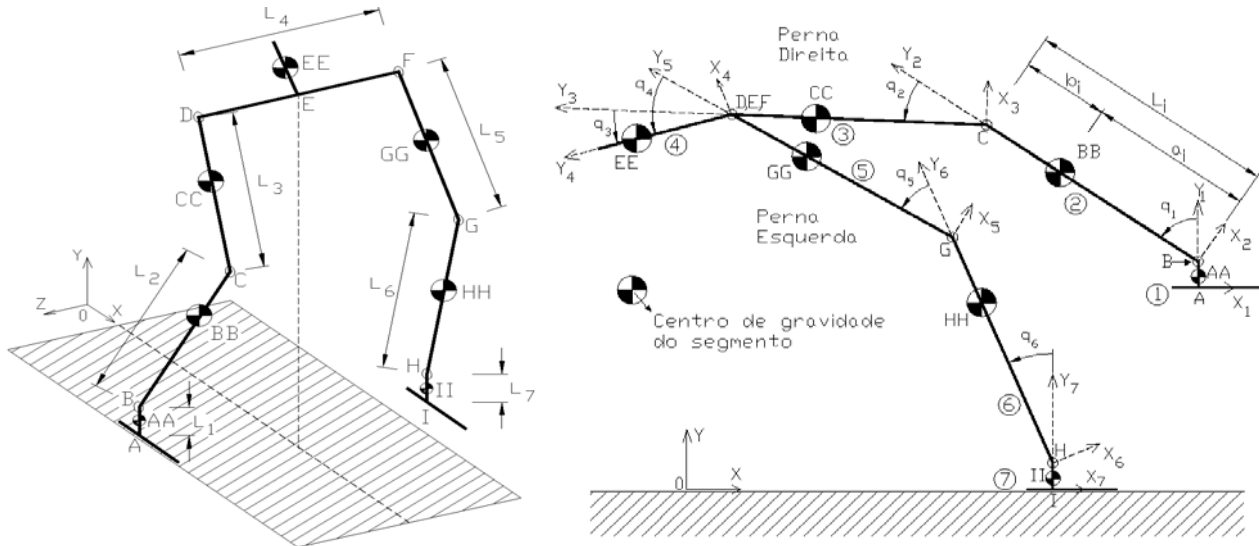


Figura 1. (a) Vista 3D do Robô Bípede. (b) Bases Móveis (Mita et al, 1984)

Os centros de gravidade (CG) de cada segmento estão localizados a uma distância a_i da junta inferior e b_i da junta superior ($i=1,...,7$), sendo L_i é o comprimento de cada segmento, conforme a Figura 1b. As juntas de rotação são designadas pelas letras B, C, D, F, G e H, enquanto que os pontos dos pés que tocam o chão são designados A e I, e os CG dos diversos corpos são designados como AA, BB, CC, EE, GG, HH e II, sendo o ponto E central a pelve. A pelve, representada pelo segmento D, E e F, é articulada em relação às duas pernas, servindo de suporte às mesmas. Considerou-se também que durante o movimento os pés se mantêm paralelos ao solo.

As coordenadas generalizadas neste modelo são representadas pelos ângulos relativos q_i ($i=1,...,6$), todos definidos como positivos na direção do eixo Z (Figura 1b). Estes ângulos relativos são obtidos através de rotações das bases móveis B_i ($i=1,...,7$), uma em relação a outra (q_1 é a rotação da base B2 em Z em relação a B1, por exemplo). Cada base é solidária a seu respectivo corpo, assim a base B3 é solidária ao corpo 3 (coxa direita) e assim por diante. Considerou-se na simulação que o robô bípede se locomove em uma superfície plana e horizontal, que não há atrito entre as juntas, que cada junta é movida por um atuador independente, que o impacto entre o pé e o chão é perfeitamente inelástico e que não existe deslizamento entre os pés e a superfície de locomoção. Além disso, o movimento é somente no plano sagital, que é o plano perpendicular ao eixo Z formado pelos eixos XY.

A partir da transformação de coordenadas entre as bases fixa e móveis, utilizando matrizes de transformação de coordenadas (Paul, 1986), obtém-se a equação da posição das juntas e centros de gravidade. As posições, velocidades e acelerações dos pontos do robô bípede (juntas e CG) foram obtidas através das equações de posição, velocidade e aceleração para um sistema com bases móveis (Campos et al, 2000). Por exemplo, para o CG do corpo 3 (perna esquerda em balanço) tem-se a posição, velocidade e aceleração mais facilmente representadas na base móvel B3:

$$\begin{aligned}
 {}_{B3}r_{CC} &= \begin{Bmatrix} L_1 \text{sen}(q_1 + q_2) + L_2 \text{sen } q_2 \\ L_1 \text{cos}(q_1 + q_2) + L_2 \text{cos } q_2 + a_3 \\ 0 \end{Bmatrix} \\
 {}_{B3}v_{CC} &= {}_{B3}v_C + {}_{B3}\omega_3 \times {}_{B3}r_{CC} + \frac{d}{dt}({}_{B3}r_{CC}) = \begin{Bmatrix} -L_2 \dot{q}_1 \text{cos } q_2 - a_3 (\dot{q}_1 + \dot{q}_2) \\ L_2 \dot{q}_1 \text{sen } q_2 \\ 0 \end{Bmatrix} \quad (1)
 \end{aligned}$$

$${}_{B_3}a_{CC} = {}_{B_3}a_C + {}_{B_3}\omega_3 \times {}_{B_3}\omega_3 \times {}_{B_3}r_{CCC} + {}_{B_3}\dot{\omega}_3 \times {}_{B_3}r_{CCC} + \underbrace{2 {}_{B_3}\omega_3 \times \frac{d}{dt}({}_{B_3}r_{CCC})}_{=0} + \underbrace{\frac{d^2}{dt^2}({}_{B_3}r_{CCC})}_{=0} \Rightarrow$$

$${}_{B_3}a_{CC} = \begin{Bmatrix} l_2(-\ddot{q}_1 \cos q_2 - \dot{q}_1^2 \sin q_2) - a_3(\ddot{q}_1 + \ddot{q}_2) \\ l_2(+\ddot{q}_1 \sin q_2 - \dot{q}_1^2 \cos q_2) - a_3(\dot{q}_1 + \dot{q}_2)^2 \\ 0 \end{Bmatrix}$$

No diagrama de corpo livre (DCL) da Figura 2 as reações foram desprezadas, pois o dimensionamento e a determinação dos esforços sobre a estrutura não fazem parte do escopo deste trabalho. Portanto, para o desenvolvimento da modelagem dinâmica do robô bípede, somente as equações de movimento são necessárias.

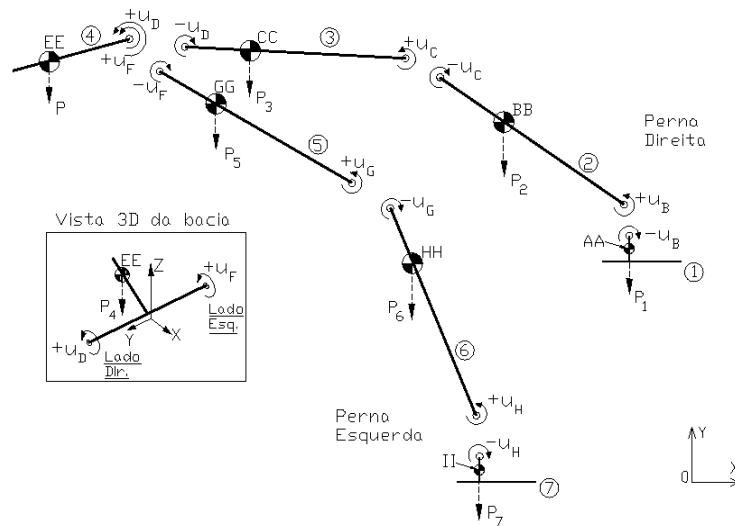


Figura 2. Diagrama de Corpo Livre simplificado do robô bípede

De acordo com a Figura 2, tem-se a representação das forças e momentos externos atuantes no sistema: os torques $u_B, u_C, u_D, u_F, u_G, u_H$ e as forças gravitacionais de cada corpo P_i ($i=1, \dots, 7$) agindo no respectivo CG. Cada corpo é formado pela massa da estrutura e do atuador solidário a este corpo. Na Tabela 1 verifica-se onde os vários atuadores estão solidários e em quais corpos os mesmos atuam. Vale lembrar que ao atuar em um outro corpo, uma reação de mesmo módulo e de sentido oposto atua no corpo solidário ao atuador. Todos os eixos dos atuadores estão localizados na direção das juntas de rotação (eixo Z).

Tabela 1. Atuadores

Atuador	Solidário ao Corpo	Atua no sentido de Z Positivo no Corpo
B	1 (pé direito)	2 (perna direita)
C	2 (perna direita)	3 (coxa direita)
D	3 (coxa direita)	4 (pelve)
F	5 (coxa esquerda)	4 (pelve)
G	6 (perna esquerda)	5 (coxa esquerda)
H	7 (pé esquerdo)	6 (perna esquerda)

O método a ser utilizado para a obtenção das equações de movimento, sem a obtenção das reações internas, é o Método de Newton-Euler-Jourdain (Santos, 2001). A equação vetorial que descreve este método é apresentada:

$$\sum_{i=1}^p \left[J^T v_i \left(\sum_{B_m} F_E - m_i a_i \right) + J^T \omega_i \left(\sum_{B_m} M_E - I_{O_i} \frac{d}{dt}({}_{B_m}\omega_i) - \underbrace{{}_{B_m}\Omega_i \times (I_{O_i} \omega_i)}_{=0} - \underbrace{m_i r_{O_i}^{CG} \times a_i}_{=0} \right) \right] = 0 \quad (2)$$

onde,

$\sum_{i=1}^p$ é a somatória em todos os corpos p

$\sum_{B_m} M_E$ é a somatória dos momentos externos representados na base B_m em relação ao ponto O do corpo i

$J_{v_i}^T$ é o Jacobiano de velocidade linear (Jacobiano de translação) do corpo i

$\sum_{Bn} F_E$ é a somatória das forças externas na Base Bn agindo no corpo i

m_i é a massa do corpo i

${}_{Bn}a_i$ é a aceleração linear do CG do corpo i

$J_{\omega_i}^T$ é o Jacobiano de velocidade angular (Jacobiano de rotação) do corpo i

I_{O_i} é o momento de inércia de massa do corpo i em relação ao ponto O

$\frac{d}{dt}({}_{Bm}\omega_i)$ é a derivada temporal do vetor de velocidade angular do corpo i na própria base Bm

${}_{Bm}\Omega_i$ é o vetor de velocidade angular da base Bm representada na própria base Bm

${}_{Bm}\omega_i$ é o vetor de velocidade angular do corpo i na base Bm

${}_{Bm}r_{O_i}^{CG}$ é o vetor de deslocamento do ponto O, onde ocorre a somatória de momentos, até o ponto CG, ou seja, o centro de gravidade do corpo i

A primeira parte da somatória está associada à equação de Newton (variação no tempo do momento linear) e a segunda parte está associada à equação de Euler (variação no tempo do momento angular). As forças e momentos internos (reações internas) não aparecem na Equação 2, pois a multiplicação do Jacobiano de velocidade linear do corpo i com as forças internas atuantes neste corpo é identicamente nulo, assim como a multiplicação do Jacobiano de velocidade angular do corpo i com os momentos internos deste corpo. As reações internas (forças e momentos) estão em direções perpendiculares aos graus de liberdade do sistema e não realizam trabalho porque surgem aos pares, com mesmo módulo e direções opostas.

O termo ${}_{Bm}\Omega_i \times (I_{O_i} {}_{Bm}\omega_i)$ é nulo pois como as bases móveis são solidárias aos vários corpos, então a rotação das bases é igual à rotação dos corpos, e portanto o produto vetorial é nulo. Já o termo $m_i {}_{Bm}r_{O_i}^{CG} \times {}_{Bm}a_i$ é identicamente nulo pois a somatória dos momentos é realizada em relação ao CG de cada corpo.

O Jacobiano de um vetor genérico $s = \{s_x \ s_y \ s_z\}^T$ em relação às coordenadas mínimas de velocidade é definido como $J_s = \partial s / \partial \dot{q}$. Para o robô bípede, o Jacobiano é formado pelas velocidades angulares relativas $\dot{q} = \{\dot{q}_1 \ \dot{q}_2 \ \dots \ \dot{q}_6\}$. Por exemplo, para o corpo 3, os termos da equação são:

$$\begin{aligned}
 J_{v_{cc}} &= \frac{\partial}{\partial \dot{q}} ({}_{B3}v_{cc}) = \begin{bmatrix} -L_2 \cos q_2 - a_3 & -a_3 & 0 & 0 & 0 & 0 \\ L_2 \sin q_2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} & I_{CC_3} &= \begin{bmatrix} I_{xx_3} & 0 & 0 \\ 0 & I_{yy_3} & 0 \\ 0 & 0 & I_{zz_3} \end{bmatrix} \\
 J_{\omega_i} &= \frac{\partial}{\partial \dot{q}} ({}_{B3}\omega_3) = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, \text{ onde } {}_{B3}\omega_3 = \begin{bmatrix} 0 \\ 0 \\ \dot{q}_1 + \dot{q}_2 \end{bmatrix} & & (3) \\
 \sum {}_{B3}F_E &= [R_2][R_1] {}_{B3}P_3 = \begin{bmatrix} -m_3 g \sin(q_1 + q_2) \\ -m_3 g \cos(q_1 + q_2) \\ 0 \end{bmatrix} & \sum {}_{B3}M_E &= \begin{bmatrix} 0 \\ 0 \\ u_C - u_D \end{bmatrix}
 \end{aligned}$$

Desenvolvendo a equação de movimento, Eq. 2, para o sistema mecânico como um todo, obtêm-se um sistema de 6 equações que podem ser escritas de forma matricial:

$$[A]\{\ddot{q}\} = \{bb\} + \{u\} \tag{4}$$

A equação acima é função dos parâmetros geométricos (L_i , a_i e b_i), dos parâmetros de massa (m_i e I_{zz_i}) para $i=1, \dots, 7$, das variáveis q_i , \dot{q}_i e \ddot{q}_i , ($i=1, \dots, 6$) e dos torques de acionamento (u_B , u_C , u_D , u_F , u_G e u_H). Para a obtenção das 6 equações não lineares de movimento e sua manipulação simbólica, utilizou-se o software MATHEMATICA[®], que permitiu simplificações e agrupamento de termos semelhantes (Campos e Zampieri, 2000).

3. Simulação

Para a determinação da melhor trajetória a ser seguida em função do acionamento das juntas, Vukobratovic e seus colegas (Vukobratovic et al, 1990) adotaram o modelo humano, ou seja, foram obtidos dados através de medições do modo de andar de seres humanos. Bay e Hemani (1987) utilizaram o Gerador Central de Padrões para gerar movimentos rítmicos das pernas. Katoh e Mori (1984) usaram um par de osciladores de van der Pol para gerar padrões de passos para o robô bípede BIPMAN. McClintick e Moskowitz (1998) utilizaram uma função quadrática para definir a posição inicial e final da perna em balanço. Cheng e Lin (1997) propuseram um algoritmo genético para a determinação da trajetória ótima.

Neste trabalho foi estabelecida para a trajetória dos pés em cada passo como meia senóide ou meia elipse. Especificando-se o ponto de início, a amplitude (altura do chão) e a distância em X de cada passo, a trajetória do pé se torna totalmente conhecida, tanto para a função senoidal, quanto para função elíptica. Considerou-se também que tanto a pelve quanto os pés possuem velocidades lineares constantes na direção X e que a fase biapoiada é instantânea. Em regime permanente a pelve percorre uma trajetória senoidal de pequena amplitude e, portanto, mantém uma altura média constante em relação ao solo. O mínimo desta função senoidal ocorre quando os dois pés estão apoiados no solo (fase biapoiada) e seu máximo ocorre durante a fase balanço, quando a pelve está exatamente na mesma linha vertical do centro do pé de apoio, semelhante ao modo de andar humano. A simulação cinemática foi realizada em código MATLAB e apresenta as trajetórias dos centros de gravidade de cada segmento em função do tempo do passo. Em seguida calcula-se a posição dos joelhos e os ângulos q_i em cada instante de tempo e como consequência suas derivadas \dot{q}_i e \ddot{q}_i . Em todas simulações o robô bípede realiza o primeiro passo em regime transitório com a perna esquerda e em seguida um determinado número de passos em regime permanente, começando com a perna direita.

Os dados gerados na simulação cinemática (Campos et al, 2000) são dados de entrada para a simulação dinâmica quando as equações de movimento estão na forma (modelo dinâmico inverso):

$$\{u\} = [A]\{\ddot{q}\} - \{bb\} \tag{5}$$

que os utiliza para calcular a variação dos torques (u_B, u_C, u_D, u_F, u_G e u_H) ao longo do tempo.

4. Resultados

Nas simulações utilizou-se os seguintes parâmetros físicos e de simulação:

Pernas e Pés.....	$L_2=L_3=L_5=L_6 = 0,15$ m e $L_1=L_7 = 0,02$ m
Pelve.....	altura mínima da pelve em relação ao solo = 0,31 m amplitude de oscilação da pelve (pico a pico) = 0,005 m
Passos.....	amplitude dos passos (altura máxima do solo) = 0,01 m distância de um passo na direção X (regime permanente) = 0,1 m (transitório: 0,05 m) período de um passo = 1 s
Massas.....	$m_2=m_3=m_5=m_6 = 0,126$ Kg, $m_1 = m_7 = 0,233$ Kg e $m_4 = 0,300$ Kg
Momentos de Inércia.	$I_{zz2} = I_{zz3} = I_{zz5} = I_{zz6} = 2,8261 \cdot 10^{-4}$ Kg m ² $I_{zz1}=I_{zz7} = 2,5949 \cdot 10^{-4}$ Kg m ² e $I_{zz4} = 1,2250 \cdot 10^{-4}$ Kg m ²
Centros de gravidade.	$a_2=a_3=a_5=a_6 = 0,126$ m $a_1 = a_7 = 0,009$ m e $a_4 = 0,02$ m
Iterações	100 por passo ($\Delta t = 1$ s/100 = 0,01 s)

Os segmentos das coxas e pernas foram confeccionados em alumínio com 15 cm de comprimento, a pelve e os pés como chapas de 10 x 15 cm por 0,3 cm de espessura. O CG de cada coxa e perna (conjunto solidário atuador mais perfil de alumínio) está localizado a 12,6 cm do ponto mais inferior do respectivo segmento (como pode ser visto na Figura 1). Para os pés o CG está localizado a 0,9 cm do ponto mais inferior. No caso da pelve estabeleceu-se uma distância de 2 cm para o CG em relação ao ponto E e uma massa de 0,300 Kg. Cada atuador tem massa de 0,1Kg.

A solução numérica da Equação 1 (Campos et al, 2000) permitiu simular a trajetória do robô bípede, conforme mostrado na Figura 3. Para ambas as curvas a amplitude em Y e a distância de cada passo em X são as mesmas.

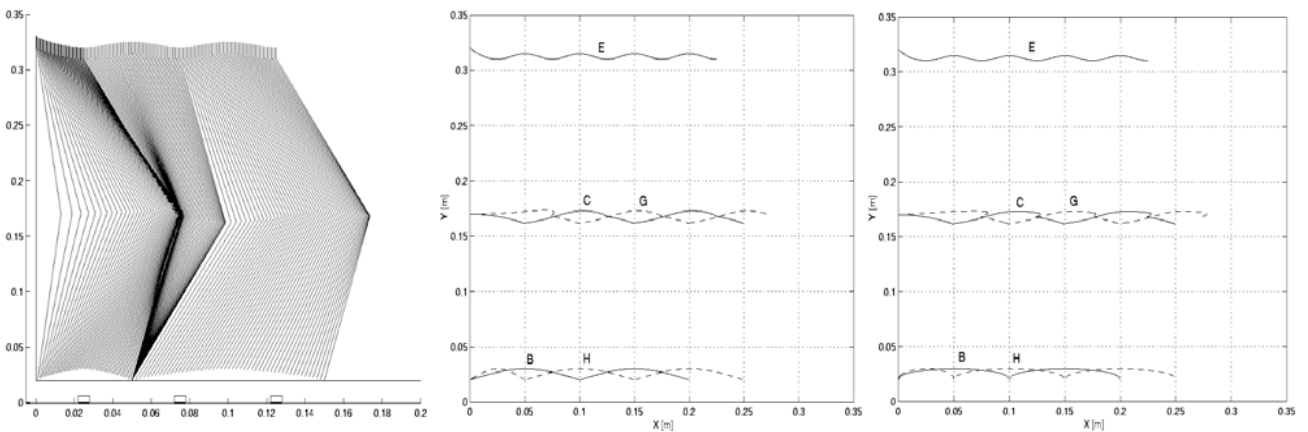
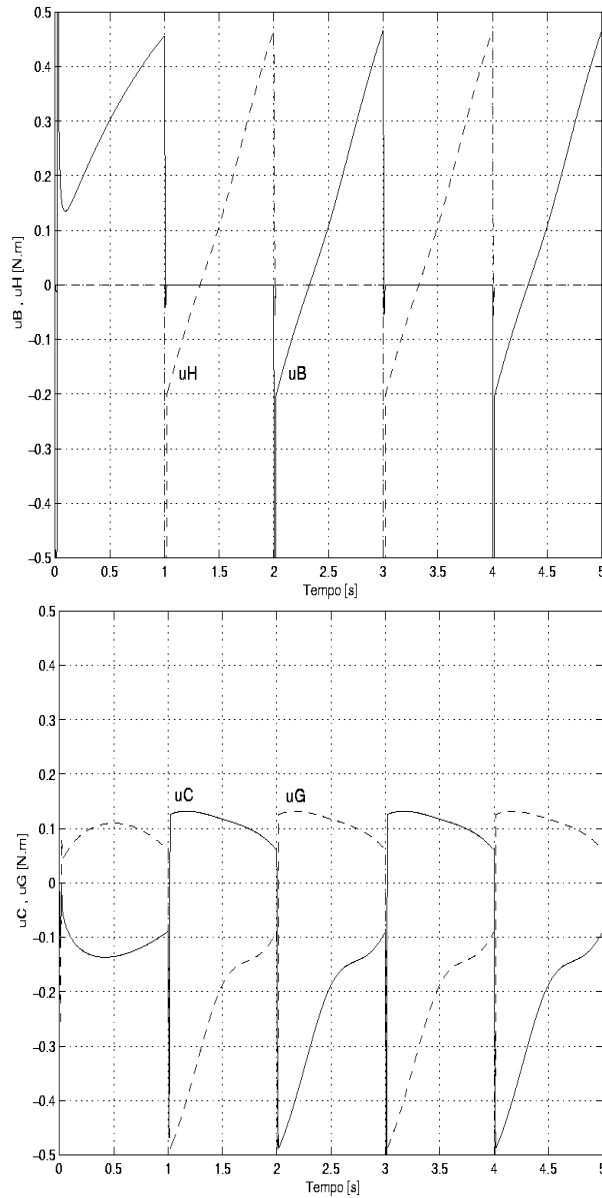


Figura 3: (a) Trajetória da Perna esquerda (b) Trajetória senoidal (juntas) (c) Trajetória elíptica (juntas)

As simulações apresentadas consistem em 1 passo inicial em regime transitório com a perna esquerda e 4 passos no regime permanente. Nota-se nas figuras acima que as curvas das trajetórias Senoidal e Elíptica, variam ligeiramente umas das outras. Os dados de saída do programa de simulação cinemática (Campos et al, 2000) são os ângulos q_i , a velocidade \dot{q}_i e acelerações \ddot{q}_i ao longo do tempo. Estes arquivos são dados de entrada para o programa de simulação dinâmica. Assim os torques ao longo do tempo para o pé com trajetória Senoidal e Elíptica foram determinados e podem ser vistos nas Figuras 4 e 5.



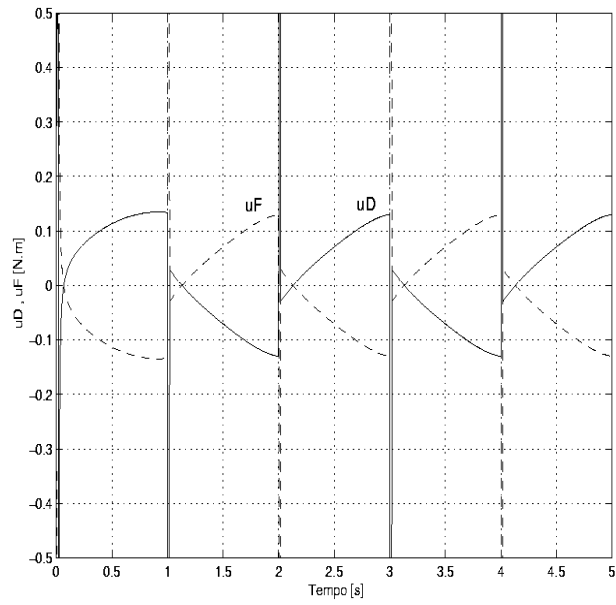
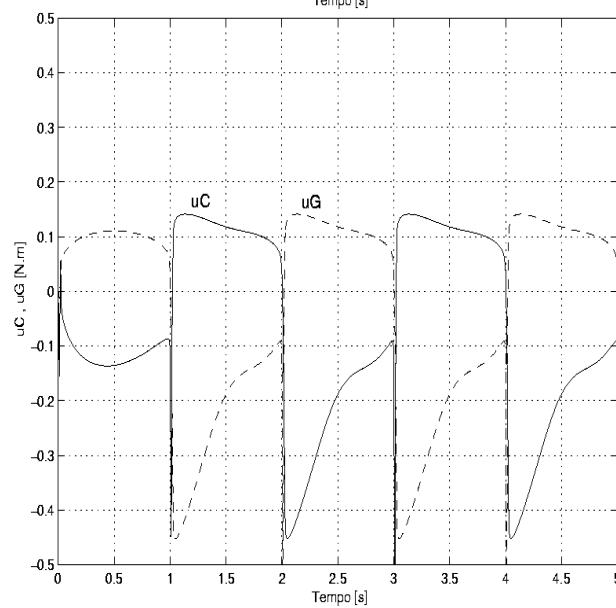
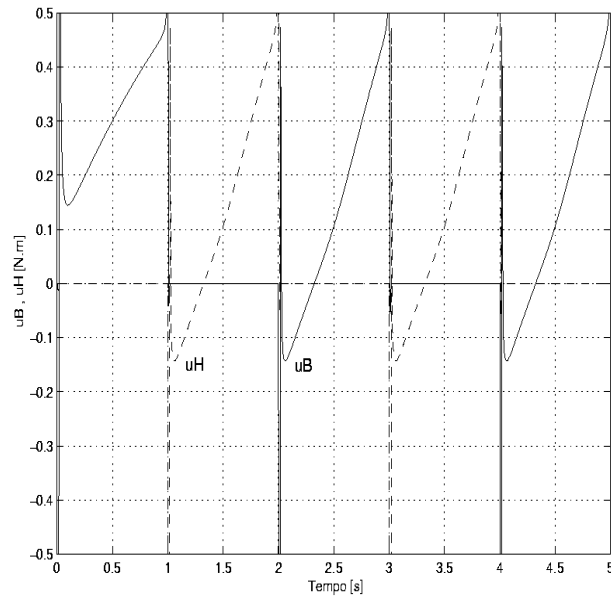


Figura 4. Torques nas juntas para a trajetória Senoidal dos pés.



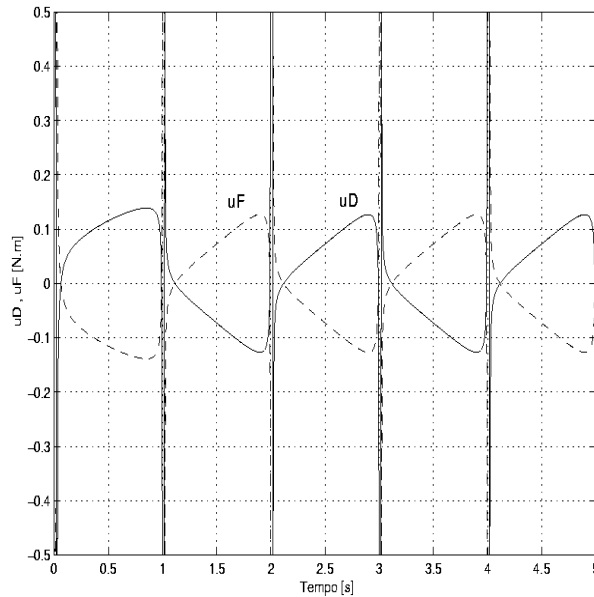


Figura 5. Torques nas juntas para a trajetória Elíptica dos pés.

No regime permanente, os torques nos tornozelos (u_B e u_H) são negativos no início do passo e positivos no restante do mesmo. Isto mostra que ao iniciar o passo, o torque no tornozelo impulsiona o robô bípede para frente, mas posteriormente, o mesmo torque tende a forçá-lo para trás, freando o movimento para que o impacto com o solo não seja muito forte. Pode-se verificar este comportamento, por exemplo, no instante de $2 < t < 3$ s, em que a perna esquerda está em balanço e a perna direita está apoiada no solo.

Nas Figuras 4 e 5, os valores dos torques na pelve (u_D e u_F) são complementares, $u_D(t) + u_F(t) = 0$, para todo instante t , pois na simulação cinemática, a pelve permanece sem girar em relação ao sistema inercial: a somatória dos torques sobre a pelve em relação ao eixo Z deve ser igual a zero. A finalidade de manter a pelve sempre paralela ao solo é evitar movimentos bruscos. Verifica-se ainda, nas Figuras 4 e 5 que os torques de acionamento podem ser analisados aos pares: $u_B \Leftrightarrow u_H$ (torques nos tornozelos), $u_C \Leftrightarrow u_G$ (torques nos joelhos) e $u_D \Leftrightarrow u_F$ (torques na pelve). As curvas dos pares são idênticas, defasadas do tempo de um passo, neste caso 1s. Esta característica decorre da simetria adotada na escolha das bases móveis do modelo e sentido adotado dos torques positivos. Foram observadas pequenas diferenças entre os resultados do torque para a trajetória Senoidal ou Elíptica dos pés.

Os picos de torque nos instantes $i=1, \dots, 5$ se devem aos dados de entrada do programa de simulação cinemática. São exatamente os instantes em que os pés tocam o solo, provocando discontinuidades, já que o equacionamento muda ao trocar o pé de apoio e esta mudança é instantânea na simulação cinemática.

No intuito de validar os dados obtidos na simulação computacional feita, aplicou-se a trajetória gerada pela simulação cinemática (coordenadas generalizadas q_i) no robô bípede RB-1. O robô realizou uma locomoção estável no modo estático, validando a metodologia e simulação apresentada.



Figura 6: Robô Bípede RB-1

5. Conclusão

A modelagem permitiu verificar a importância da cinemática inversa, como ferramenta para a síntese de sistemas mecânicos robóticos na medida que, através da pré-definição de trajetórias foi possível determinar as curvas dos ângulos das juntas, indispensáveis para a obtenção dos torques.

A utilização do Método de Newton-Euler-Jourdain se mostrou válida, já que permitiu a obtenção das equações de movimento diretamente, sem se obter as reações internas que neste caso não eram necessárias.

Assim, a modelagem e simulação Dinâmica do robô bípede formado por 7 corpos rígidos foi realizada, obtendo-se as curvas de torques ao longo do tempo. Através de animação computacional, pôde-se verificar que o sistema respeitou as restrições geométricas impostas.

6. Agradecimentos

À FAPESP pelo financiamento da bolsa (FAPESP Proc. 99/09602-4) e auxílio a pesquisa (Proc.99/07715-6).

7. Referências

- Bay, J.S. and Hemami, H., 1987, "Modeling of a neural pattern generator with coupled nonlinear oscillators". IEEE Transactions on Biomedical Engineering, Vol. 34, no. 4.
- Campos, D.C. e Zampieri, D.E. "Modelagem e Simulação Dinâmica de um Robô Bípede", FAPESP Proc. 99/09602-4
- Campos, D.C, Bezerra, C.A.D, Mendeleck, A. e Zampieri, D.E. "Modelagem e Simulação de um Robô Bípede", CONEM 2000 (Congresso Nacional de Engenharia Mecânica), Natal – RN, Brazil.
- Cheng, M. Y. and Lin, C. S., 1997, "Genetic algorithm for control design of biped locomotion", Journal of Robotics Systems, 14(5), pp. 365-373.
- Frank, A. A., 1968, "Automatic control systems for legged locomotion machines", Ph.D. Thesis, University of Southern California.
- Furusho, J. and Masubuchi, M., 1986, "Control of a dynamical biped locomotion system for a steady walking", J. of Dyn. Systems Meas. And Control. Transaction of ASME, 108,111-118.
- Hemani, H. and Wyman, B.F., 1979, "Modeling and Control of Constrained Dynamic Systems with Application to Biped Locomotion in the Frontal Plane", IEEE Transactions on Automatic Control, vol. AC-24, no. 4.
- Lum, H.K., Zribi, M. And Soh, Y.C., 1999, "Planning and control of a biped robot". International Journal of Engineering Science. Vol. 37, pp. 1319-1349.
- Kato, I., Ohteru, S., Kobayashi, H., Shirai, K. and Uchiyama, A., 1974, "Information-power machine with senses and limbs" in First CISM-IFTOMM Symp. On Theory and Practice of Robots and Manipulators: Springer - Verlag.
- Kato, R. and Mori, M., 1984, "Control method of biped locomotion giving asymptotic stability of trajectory". Automatica, vol. 20, no. 4, pp. 405-414.
- McClintick, B. E. and Moskowitz, G. D., 1998, "The behavior of a biped walking gait on irregular terrain". International Journal of Robotics Research, Vol. 17, no. 1, pp. 43-55.
- McGeer, T., 1990, "Passive walking with knees". Proceedings of the IEEE Conference on Robotics and Automation, vol. 2, pp. 1640-1645.
- Mita, T., Yamaguchi, T., Kashiwase, T., Kawase, T., 1984, "Realization of high speed biped using modern control theory". International J. Control, vol. 40, no. 1, 107-119
- Paul, R.P. "Robot Manipulators". MIT Press. Cambridge. Massachussets. 1986.
- Popovic, D., Stein, R.B., Oguztoreli, M. N., Lebedowska, M. and Jonic, S., 1999, "Optimal control of walking with functional electrical stimulation study". IEEE Transactions on Rehabilitation Engineering, Vol. 7, no. 1, March.
- Raibert, M.H., 1996, "Legged Robots that balance". MIT Press, Cambridge, MA.
- Santos, I.F. "Cinemática e Dinâmica de Sistemas Mecânicos". Makron. São Paulo. 2001
- Vukobratovic, M., Borovac, B., Surla, D. and Stokic, D., 1990, "Scientific fundamentals of robotics 7. Biped locomotion: Dynamics stability, control and application", Springer - Verlag.

DYNAMIC MODELING AND SIMULATION OF A BIPED ROBOT

Abstract. *In the last three decades, many studies have been made in the field of biped robots, because of their mobility in different types of surfaces and environments, compared with wheeled robots. As a consequence of this characteristic, they are more complex and the determination of a correct dynamic model becomes extremely important to simplify the obtaining of the equations without compromising his equilibrium through control. This work consists in the dynamic modeling and simulation of a biped robot formed by 7 rigid bodies connected through rotation hinges, each one having an independent rotational actuator. The modeling was developed using the Newton-Euler-Jourdain methodology for obtaining the 6 equations of motion that represent this mechanical system. These were achieved by means of symbolical manipulation software and then, implemented in numerical software for the simulation of the dynamics. Using these equations, the torques were obtained in function of the geometric and mass parameters and the generalized coordinates relative angles between the bodies and their derivatives, which were previously obtained with a kinematic simulation by establishing trajectories for the feet and the hip.*

Keywords. *biped robot, dynamic modeling, dynamic simulation*

POSITION CONTROL OF A FLEXIBLE STRUCTURE WITH PIEZOELECTRIC ACTUATORS

Edilson Hiroshi Tamai

Laboratory of Dynamic Systems and Control (LDC)

Dept. of Mechanical Engineering, Polytechnic School, University of São Paulo

Av. Prof. Mello Moraes, 2231, 05508-900, São Paulo, SP, Brazil

edilson.tamai@poli.usp.br

Abstract. Distributed actuators and sensors made of piezoelectric materials have increased the attention to the so called smart structures, where active control is used to damp their elastic vibrations and control their position or shape. This study presents the modeling and control of a highly flexible aluminum structure with two pairs of thin PZT (lead zirconate titanate) patches bonded to its surface. The controller system objective is the displacement of the free tip of a slender cantilevered aluminum beam, with simultaneous control of the tip orientation, while damping out the elastic vibrations. This action is performed by the PZT patches that impose an elastic deformation to the structure. The first two elastic modes are considered in control design, and in simulation studies the first five modes are used. The damping of the structure is increased by feeding back the modal position signal through a first order filter, a strategy that is known as Positive Position Feedback (PPF). The performance of controller is studied in time and frequency domains, and the residual vibrations are also quantified by the integration over simulation time of the modal energy.

Keywords: flexible beam, piezoelectric actuators, vibration control, flexible structures.

1. Introduction

Flexible structures have received considerable attention due to their applications in smart structures. In particular, piezoelectric materials have been studied as actuators and sensors in vibration control of flexible structures (Choi, Cho, Park, 1999; Pota, Alberts, 1995; Baz, Poh, Fedor, 1992; Liu, Yang, 1995). This study presents the modeling and control of a highly flexible cantilevered aluminum beam with two pairs of thin PZT (lead zirconate titanate) patches bonded to its surface. The piezoelectric actuator has been studied as a method of vibration damping of a slewing beam (Tamai, 2000), and to control the free tip displacement of a cantilevered beam (Tamai, 2001). Several control approaches for the vibration damping have been studied, like robust control (Warren, et al, 1995), H_∞ , LQR (Linear Quadratic Regulator), LQG-LTR (Linear Quadratic Gaussian – Loop Transfer Recovery), studied by Lim, Maghami, Joshi (1992), but these studies do not addressed the position control problem. The control problem of vibration damping and position tracking is studied by Choi, Cho and Park (1999), who used Quantitative Feedback Theory, but limited to the position control of the free tip of a cantilevered beam. PPF (Positive Position Feedback) are suggested by Baz, Poh, Fedor (1992), although only the vibration damping results were presented. In this work the controller system objective is the controlled displacement of the free tip of the beam, with simultaneous control of the tip orientation, by elastic deformation of the structure, while damping out the elastic vibrations. Strain gages are used to measure the beam deformation. In order to use only modal position signals, the control algorithm is based on PPF strategy. The modal position signal is positively fed back through a first order filter, achieving the increase in damping. The performance of controller is studied in time and frequency domains, and the residual vibrations are also quantified by the integration over simulation time of the modal energy. Although only simulation results are presented, the construction of the experimental setup is under way.

2. Modeling of the system

Figure (1) shows the system scheme. The system is composed by a thin rectangular cantilevered aluminum beam with two pairs of PZT (lead zirconate titanate) patches bonded to its surface, near the clamped end and at the middle of the beam. Strain gages are used to measure the beam deformation, and their position are selected to assure observability of the system. The aluminum beam is 0.9 m long, and its transversal section is 1 mm by 25 mm. As the dimensions of transversal section of the beam are very small compared to its length, the Euler-Bernoulli approach is used to obtain the dynamic model. The PZT patches are 25mm by 70 mm and 0.19 mm thick. The coordinate through the beam length is x , and $y(x,t)$ is the elastic displacement of the beam. In Fig. 1 θ is the angular displacement of the free tip of the cantilevered beam.

The system model includes the first five elastic modes, but only the first two are used in the controller design.

The elastic displacement $y(x,t)$ is expressed by Eq. (1):

$$y(x,t) = \sum_j \phi_j(x) \cdot q_j(t) \quad (1)$$

In Eq. (1) $\phi_j(x)$ is the mode shape function, and $q_j(t)$ is the generalized modal coordinate.

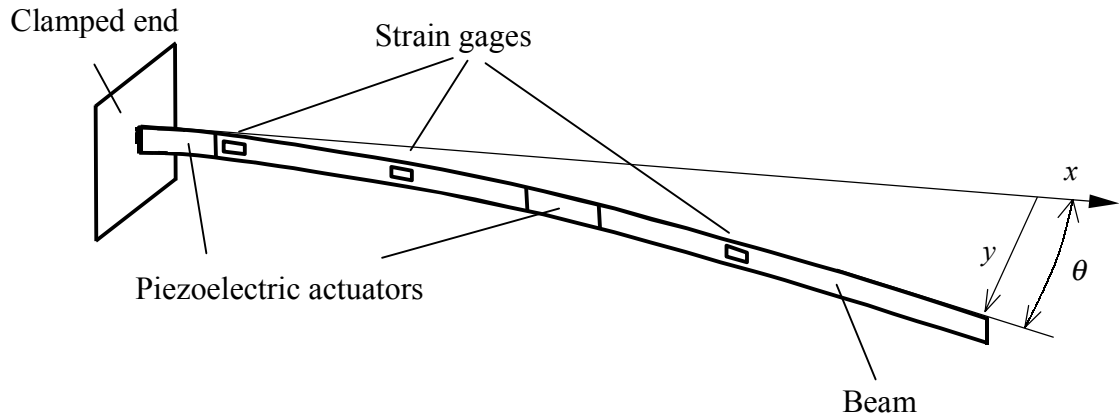


Figure 1. System scheme (the actual position and quantity of the strain gages are not shown).

The angular displacement of the beam at the \$x\$ coordinate is expressed by:

$$\theta(x, t) = \sum_j \phi_j'(x) \cdot q_j(t) \quad (2)$$

The \$j\$th mode shape \$\phi_j(x)\$ is attained using clamped – free boundary conditions, and the result is:

$$\phi_j(x) = C_j [\sin \beta_j x - \sinh \beta_j x + \alpha_j (\cos \beta_j x - \cosh \beta_j x)] \quad (3)$$

The following equations define \$C_j\$, \$\alpha_j\$ and \$\beta_j\$:

$$\alpha_j = \frac{\cos \beta_j L + \cosh \beta_j L}{\sin \beta_j L - \sinh \beta_j L} \quad (4)$$

$$\cos \beta_j L \cdot \cosh \beta_j L + 1 = 0 \quad (5)$$

$$\int_0^L \phi_j^2(x) dx = 1 \quad (6)$$

The constant \$C_j\$ is determined by the condition expressed in Eq. (6). In the above equations, \$L\$ is the beam length. The generalized modal coordinate \$q_j(t)\$ is defined by:

$$\ddot{q}_j(t) + 2\zeta_j \omega_{nj} \dot{q}_j(t) + \omega_{nj}^2 q_j(t) = \frac{k_{aj1}}{\mu_j} V_1 + \frac{k_{aj2}}{\mu_j} V_2 \quad (7)$$

In Eq. (7) \$V_1\$ and \$V_2\$ are the voltages applied in the two pairs of PZT patches.

Also in Eq. (7) \$\zeta_j\$ is a damping factor proportional to mass and stiffness (proportional factors \$a_{f0}\$ and \$a_{f1}\$):

$$\zeta_j = \frac{a_{f0}}{2\omega_{nj}} + \frac{a_{f1}\omega_{nj}}{2} \quad (8)$$

The natural frequency \$\omega_{nj}\$ is determined by:

$$\omega_{nj} = \left(\frac{\beta_j^4 EI}{\rho A} \right)^{1/2} \quad (9)$$

In Eq. (9) \$E\$ is the Young's modulus of beam material, \$I\$ is its area moment of inertia, \$A\$ is its cross-sectional area, \$\rho\$ is its density. The modal mass \$\mu_{nj}\$ is calculated by:

$$\mu_{nj} = \rho A \int_0^L \phi_j^2(x) dx \quad (10)$$

The parameter k_{aji} is defined by:

$$k_{aji} = E_a d_{31} b_a (t_a + t_b) [\phi_j'(l_{pi}) - \phi_j'(l_{di})] \quad (11)$$

In Eq. (11) E_a is the Young's modulus of piezoelectric material, d_{31} is its electric charge constant, b_a is the width of PZT patch, t_a is the thickness of the PZT patch, and t_b is the beam thickness. The extremities of the i_{th} PZT patch are located at distances l_{di} and l_{pi} of clamped end of the beam.

Equations (1) to (11), extended to the fifth mode, describe the model used in the simulations. The controller is designed using a nominal reduced model that includes only the first two elastic modes and does not consider the damping of the structure.

3. Controller design

The objective of the positive position feedback (PPF) controller is to increase the damping by feeding back the modal displacement through a first order filter. Using a nominal model without damping, the nominal system model is shown in Eq. (12), where j stands for 1 or 2 (the first and second elastic modes):

$$\ddot{q}_j(t) + \omega_{nj}^2 q_j(t) = u_j(t) \quad (12)$$

The input of the first order filter is the time function $q_j(t)$:

$$T_j \dot{z}_j(t) + z_j(t) = q_j(t) \quad (13)$$

The controller signal is determined by:

$$u_j(t) = \gamma_j \omega_{nj}^2 (z_j + q_{jr}) \quad (14)$$

In Eq. (14) q_{jr} is the reference input which gives the desired tip displacement and orientation. The transfer function of the controlled system is:

$$\frac{\gamma_j \omega_{nj}^2 (T_j s + 1)}{T_j s^3 + s^2 + T_j \omega_{nj}^2 s + \omega_{nj}^2 (1 - \gamma_j)} \quad (15)$$

In order to achieve a null steady state error, γ_j must be 0.5, so only the time constant T_j of the j_{th} filter needs to be calculated. The damping is affected by the time constant, but there is a limit, so the damping cannot be arbitrarily increased. Using the root locus, the time constant T_j is determined so that the damping of the controlled mode is maximum.

The reference inputs q_{jr} are the results of the following transformation:

$$\begin{bmatrix} \phi_1 & \phi_2 \\ \phi_1' & \phi_2' \end{bmatrix} \cdot \begin{bmatrix} q_{1r} \\ q_{2r} \end{bmatrix} = \begin{bmatrix} k_{11} & k_{12} \\ k_{21} & k_{22} \end{bmatrix} \begin{bmatrix} y_r \\ \theta_r \end{bmatrix} \quad (16)$$

In Eq. (16) y_r is the desired displacement of the free tip of the beam, and θ_r is the desired angular displacement of the beam at the free end. The gains k_{11} , k_{12} , k_{21} , and k_{22} are calculated in order to compensate the steady state errors due to the modes not considered in the design of the control system. Once the control signal u_j is determined, the voltage V_i applied in the i_{th} piezoelectric actuator is calculated by:

$$\begin{bmatrix} \frac{k_{a11}}{\mu_1} & \frac{k_{a12}}{\mu_1} \\ \frac{k_{a21}}{\mu_2} & \frac{k_{a22}}{\mu_2} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (17)$$

The controller based on positive position feedback is compared with an open loop controller where the input signals are adjusted to impose the desired tip deflection and orientation, but are not able to suppress the elastic vibrations.

4. Simulation results

The simulations are based on the following parameters:

Aluminum Beam: Young's modulus $E = 7,1 \cdot 10^{10}$ Pa, density $\rho = 2710$ kg/m³, width $b_b = 0,025$ m, thickness $t_b = 0,001$ m, length $L = 0,9$ m, coefficients of damping, $a_{r0} = 0.15544$ s⁻¹, $a_{r1} = 8.9894 \cdot 10^{-4}$ s. With that values, the beam natural frequencies are: $\omega_{n1} = 6.414$ rad/s, $\omega_{n2} = 40.20$ rad/s, $\omega_{n3} = 112.6$ rad/s, $\omega_{n4} = 220.6$ rad/s, $\omega_{n5} = 364.6$ rad/s.

PZT patch: Young's modulus $E_a = 6,2 \cdot 10^{10}$ Pa, width $b_a = 0,025$ m, thickness $t_a = 0,19 \cdot 10^{-3}$ m, charge constant $d_{31} = -1,79 \cdot 10^{-10}$ m/V, length $l_a = 0,070$ m, position of patches extremities (from the clamped end of the beam) $l_{p1} = 0,000$ m, $l_{d1} = 0,070$ m, $l_{p2} = 0,415$ m, $l_{d2} = 0,485$ m.

The gains for the PPF controller are: $k_{11} = 7.7293 \cdot 10^{-1}$, $k_{12} = 1.1989 \cdot 10^{-1}$, $k_{21} = -2.1388$, $k_{22} = 2.1909$. The gains for the open loop controller are: $k_{m11} = 1.6138 \cdot 10^4$, $k_{m12} = -7.1563 \cdot 10^3$, $k_{m21} = -1.6860 \cdot 10^4$, $k_{m22} = 1.3724 \cdot 10^4$.

The controller has two parameters to be calculated, the time constants T_1 and T_2 of the two first order filters. They will be chosen so that the damping is maximum.

The Fig. (2) shows the root locus of PPF controller (first elastic mode), used to calculated the time constant T_1 of the first filter (left plot) and the root locus of PPF controller (second elastic mode), used to calculated the time constant T_2 of the second filter (right plot):

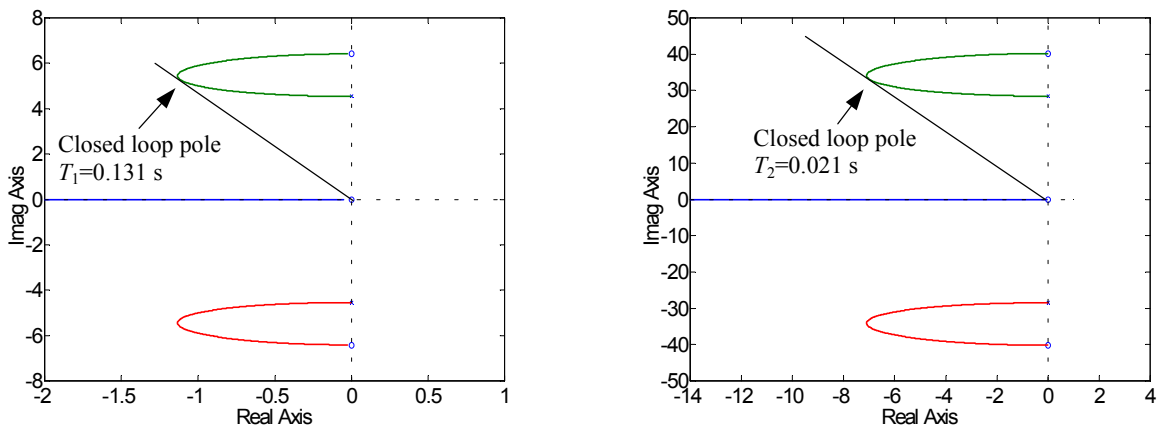


Figure 2. Root loci of the controlled modes as function of time constants of the filters. Left plot: first elastic mode. Right plot: second elastic mode.

The resulting time constants are $T_1 = 0.131$ s and $T_2 = 0.021$ s.

The frequency response of the controlled system is shown in Fig. (3) and Fig. (4). The controlled system is compared with a open loop control system.

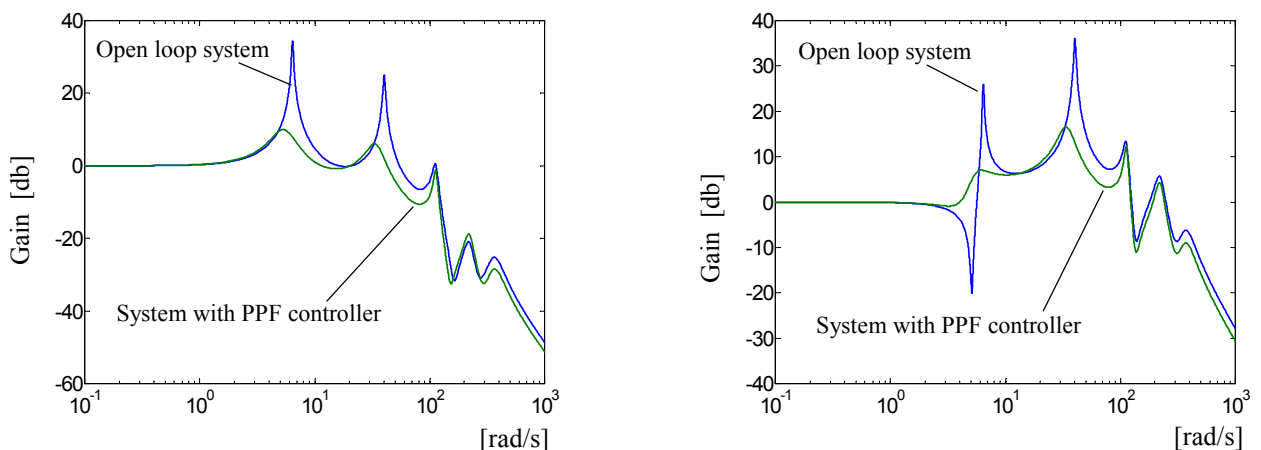


Figure 3. Frequency response of the system. Left plot: input y_r and output y . Right plot: input r and output y .

Figure (3) shows that the controlled structure can effectively follow the reference inputs y_r and r .

It can be noted that the usable bandwidth of the closed loop system is rather low (1 rad/s). This control scheme does not address this bandwidth specification, however, in this particular structure, the limits in the deformation of the PZT patches and the maximum voltage these patches can tolerate will not allow any significant increase in the bandwidth of the controlled system.

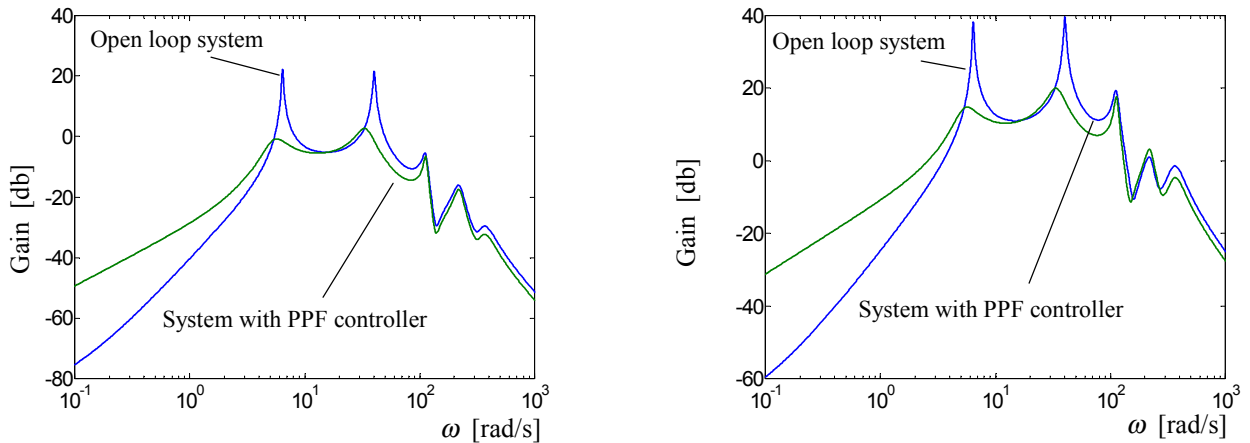


Figure 4. Frequency response of the system. Left plot: input y_r and output θ . Right plot: input θ_r and output y .

Figure (4) shows that, within the bandwidth of the controlled system, the reference input y_r does not affect the output θ in a significant way and the reference input θ_r also does not significantly affect the output y . The Bode plots show that the first two peaks are highly attenuated, without a appreciable change in the non controlled modes.

The controlled system was also simulated using step inputs, where the structure is forced to stay at a prescribed shape. The following figures show the simulation results using input $y_r = 0.005$ m and input $\theta_r = 0.000$ rad. Figure (5) shows the displacement of the free tip (left plot) and the angular displacement of the free tip (right plot).

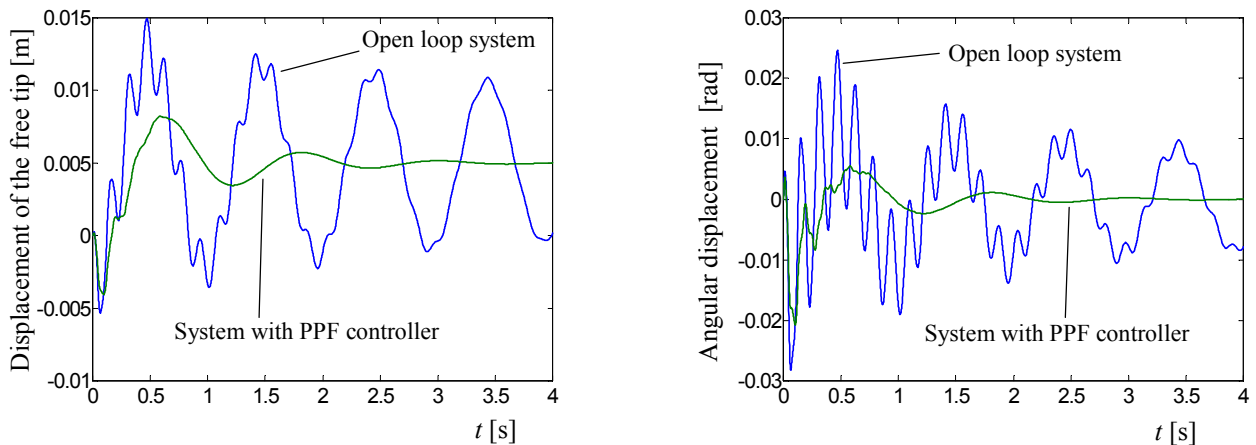


Figure 5. Displacement of the free tip (left plot) and angular displacement (orientation) of the free tip (right plot).

In the plots of Fig. (5) only the oscillations of the first and second modes are clearly visible in the open loop controller. The damping in the system with PPF controller is clearly higher than the damping in the open loop system.

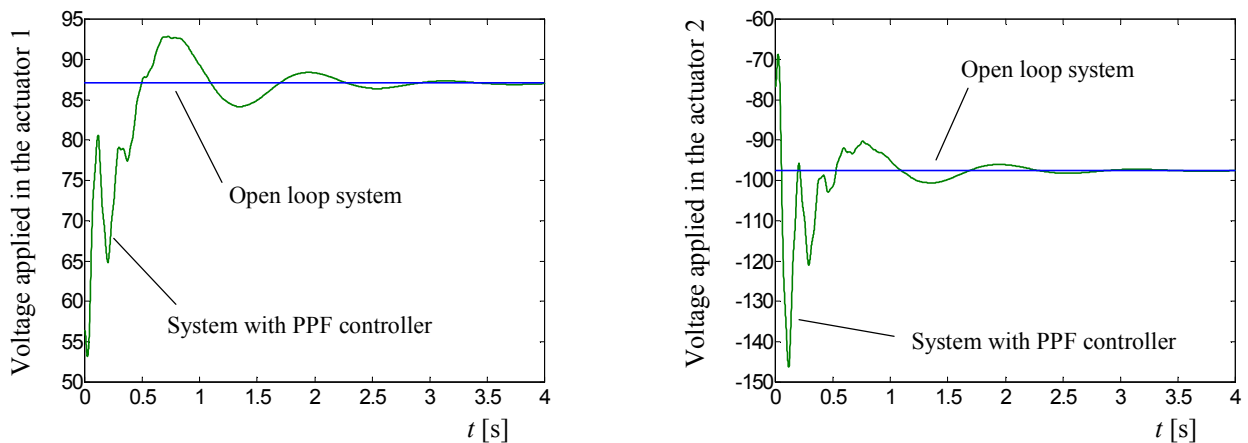


Figure 6. Left plot: voltage applied in the piezoelectric actuator 1 (near the clamped end of the beam). Right plot: voltage applied in the piezoelectric actuator 2 (in the middle of the beam).

The electric voltages applied in the piezoelectric actuators are shown in Fig. (6). These voltages are relatively high, since the manufacturer of the PZT patches recommends maximum voltages about ± 90 V (it is a conservative recommendation). In the experimental tests (not performed yet) some modifications in the control system may become necessary, if these high voltages had to be avoided.

The modal energy E_j is also used to evaluate the controllers:

$$E_j(t) = \omega_j^2 q_j^2(t) + \dot{q}_j^2(t) \tag{18}$$

$$E_m = E \left[\sum_j E_j(t) \right] \tag{19}$$

Using Eq. (19), the mean value of the modal energy is calculated (from 0 to 4 seconds) and the results are:

Open loop system: $E_m = 7.45 \times 10^{-3}$.

PPF controller: $E_m = 5.61 \times 10^{-3}$.

The reduction in the mean value of the modal energy is about 25%. If the steady state energy (5.51×10^{-3}) is subtracted from the above values, the reduction in the mean value of the modal energy is about 95%.

The next two figures show the elastic deformation y of the beam versus coordinate x and time t , giving a better visualization of the beam vibrations.

Figure (7) shows the elastic deformation y of the beam along the x coordinate as a function of time t , considering the response of the open loop system to the step inputs.

The mathematical model of the structure has positive transmission zeros, in other words, it is a non minimum phase system, and the consequences in the transient response of the displacements of the free tip can be seen in Fig. (5), left, and Fig. (7), as the free tip starts to go to the opposite direction of the reference input.

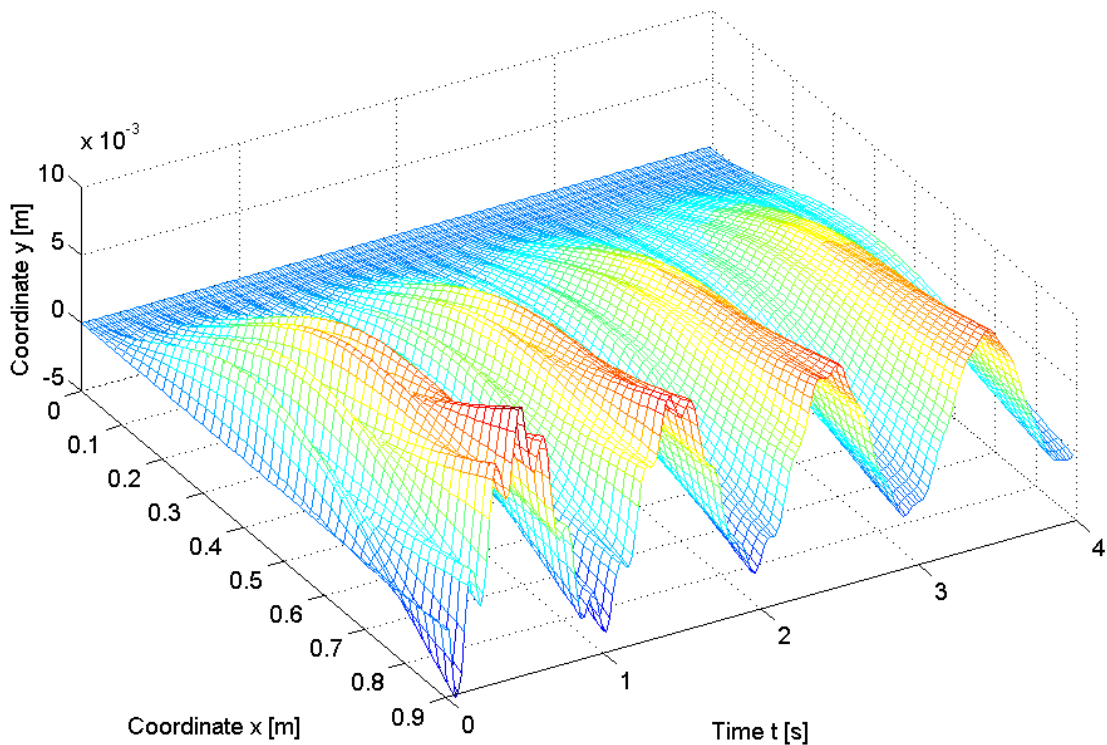


Figure 7. Open loop system: elastic deformation y of the beam versus coordinate x and time t .

The elastic deformation y of the beam along the x coordinate as a function of time t , considering the step response of the closed loop system with PPF controller is presented in Fig. (8).

Figure (8) shows that the non minimum phase behavior at the free tip of the beam is also present in the system with PPF controller.

The two piezoelectric actuators effectively suppress the elastic vibration and they also control the free tip displacement and orientation, which allow a limited control of the beam shape. Of course, only with distributed actuation over the entire beam length it will be possible the total control of the beam shape.

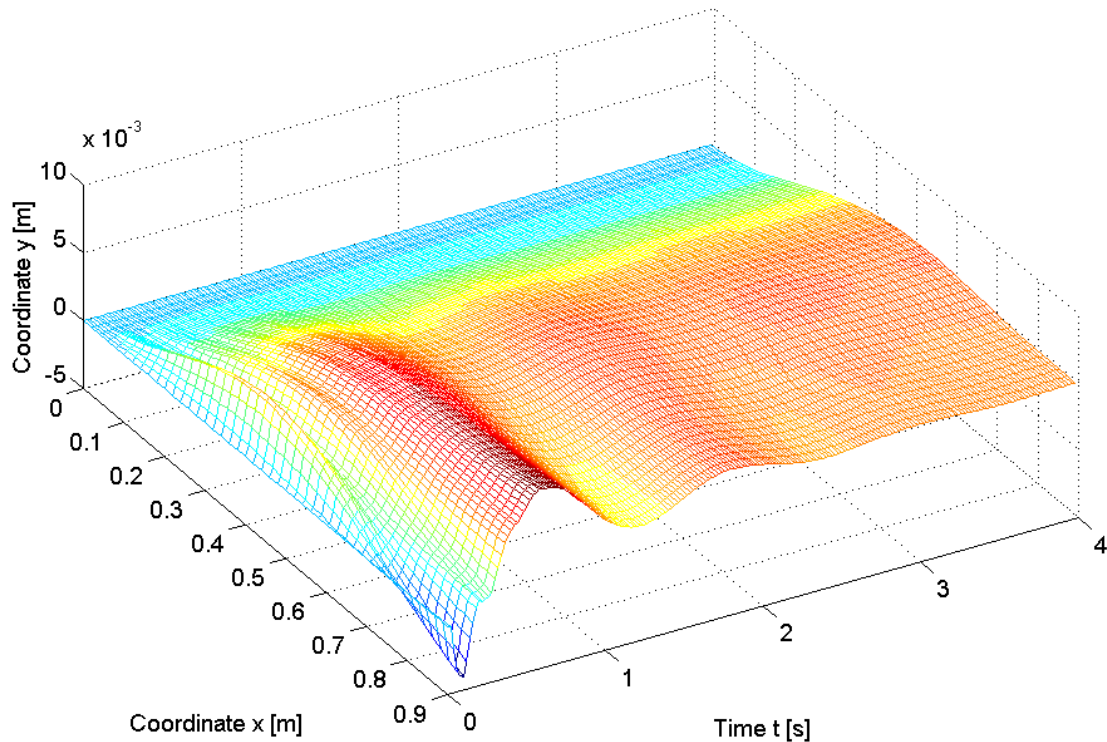


Figure 8. System with PPF controller: elastic deformation y of the beam versus coordinate x and time t .

5. Remarks

The PPF controller was designed using a reduced order model, and applied to the full order model, achieving the free tip positioning objectives. The resonance peaks of the controlled modes were effectively attenuated, without a appreciable change in the peaks of the non controlled modes. The PPF controller has also the advantage of requiring only the displacements y and θ at the free tip of the beam. However there are limits to the increase in the damping, as shown in the root loci, and the speed of the response is not directly addressed in the controller design.

The sequence of this research is the study of the sensor allocation and the construction of the experimental setup.

6. References

- Baz, A., Poh, S. e Fedor, J., 1992, "Independent Modal Space Control With Positive Position Feedback", Transactions of ASME, Journal of Dynamic Systems, Measurement, and Control, Vol.114, No.3, pp. 96–103.
- Choi, S. B., Cho, S. S. e Park, Y. P., 1999, "Vibration and Position Tracking Control of Piezoceramic-Based Smart Structures Via QFT", Transactions of ASME, Journal of Dynamic Systems, Measurement, and Control, Vol.121, No.3, pp. 27–33.
- Lim, K.B., Maghami, P.G. e Joshi, S.M., 1992, "Comparison of Controller Designs for an Experimental Flexible Structure", IEE Control Systems, Vol.12, No.3, pp. 108-118.
- Liu, Y. C. e Yang, S. M., 1995, "Vibration Control Experiment of a Slewing Flexible Beam", Transactions of ASME, Journal of Dynamic Systems, Measurement, and Control, Vol.117, No.9, pp. 432–435.
- Pota, H. R., Alberts, T. E., 1995, "Multivariable Transfer Functions for a Slewing Piezoelectric Laminate Beam", Transactions of ASME, Journal of Dynamic Systems, Measurement, and Control, Vol.117, No.9, pp. 352–359.
- Tamai, E. H., 2000, "Control of a Flexible Slewing Beam", Proceedings of the 13th Brazilian Congress of Automatica. Florianopolis, Brazil, pp. 1863-1868. In Portuguese.
- Tamai, E. H., 2001, "Vibration Control of Flexible Beam With Positive Position Feedback", Proceedings of the 9th International Symposium on Dynamic Problems of Mechanics, Florianopolis, Brazil, pp. 405-410.
- Warren, S. R. et al., 1995, "Robust Control of a Slewing Beam System", Journal of Vibration and Control, n.1, pp. 251-271, Aug. 1995.

7. Copyright Notice

The author is the only responsible for the printed material included in his paper.

OPTIMUM DESIGN OF GENERAL 3R MANIPULATORS BY USING TRADITIONAL AND RANDOM SEARCH OPTIMIZATION TECHNIQUES

Chiara Lanni

Laboratory of Robotics and Mechatronics

DiMSAT, University of Cassino, Via Di Biasio 43, 03043 Cassino (Fr), Italy- e-mail: lanni@ing.unicas.it

Sezimaria F. Pereira Saramago

College of Mathematics -Federal University of Uberlandia

Campus Santa Monica, CEP 38400-902, Uberlandia - MG, Brazil- e-mail: saramago@ufu.br

Marco Ceccarelli

Laboratory of Robotics and Mechatronics

DiMSAT, University of Cassino, Via Di Biasio 43, 03043 Cassino (Fr), Italy- e-mail: ceccarelli@ing.unicas.it

Abstract. The manipulator design can be conveniently attached as a problem of optimization, whose objective functions can be the size of the manipulator and workspace volume; and the constraints can be given as a prescribed workspace volume. In this paper, the optimum design of 3R manipulators has been formulated and solved by using an algebraic formulation of workspace boundary. The numerical solution of the optimization problem has been investigated by using two different numerical techniques, namely Sequential Quadratic Programming and Simulated Annealing. Basic computational characteristics are outlined and numerical examples have been reported to discuss the results and efficiency of the proposed algorithms, even by a comparison of the used two procedures.

Keywords: Robotics, Computational Kinematics, Manipulator Design, Workspace, Optimization

1. Introduction

The workspace of a robot arm is recognized of great interest from theoretical and practical viewpoints as a fundamental characteristic for kinematic evaluation and dimensional design. Thus, the workspace study, for both analysis and synthesis purposes, still deserves great interest, although several procedures were proposed in the last decade. Manipulators design can be conveniently expressed as a function of workspace requirements, since a fundamental feature of manipulators is recognized in workspace capabilities.

The manipulator workspace is defined as the region of reachable points by a reference point H on the extremity of a manipulator chain (Kumar and Waldron, 1981). Several design procedures have been proposed for dimensional synthesis of manipulators. Solutions of the optimization problem has been approached by means of a heuristic technique (Yang and Lee, 1984), a grid search method (Tsai and Sony, 1985), the Monte Carlo method (Rastegar and Fardanesh, 1990) a sequential quadratic programming (Ceccarelli et al. 1994; Ceccarelli and Lanni, 1999; Mata and Ceccarelli, 1993). Other techniques are based on the kinematic geometry of the chain (Vijaykumar et al., 1986) on a suitable description of workspace boundary (Gosselin and Guillot, 1991), or on systematic search procedure developed on algebraic criteria (Lin and Freudenstein, 1986), that have been deduced from an algebraic workspace formulation (Freudenstein and Primerose, 1984). The advantages of the algebraic approach has been pointed out by Freudenstein and Primerose (Freudenstein and Primerose, 1984) in the form of the theorems useful both in analysis and synthesis.

A suitable formulation for the workspace has been used in the proposed manipulator design algorithms, which have been formulated as an optimization problem by using the workspace volume and total dimension of the manipulator as the multi-objective functions, and given workspace limits as the constraints. Additional constraints have been included to obtain manipulator sizes within practical values.

Taking into account an algebraic formulation for the manipulator workspace given in (Ceccarelli, 1989 and 1996) and following previous experiences on optimization procedure presented in (Saramago and Steffen, 2000; Ceccarelli, 1993; Ceccarelli et al., 1994; Ceccarelli and Lanni, 1999), in this paper the optimization problem has been investigated by using two different numerical techniques. The first one makes use of a Sequential Quadratic Programming SQP technique (Grace, 1992); the second technique uses a random search technique. In the case of SQP two different codes have been used: DOT (Design Optimization Tools), (Vanderplaats, 1995) and minimax of Matlab Toolbox (Grace, 1992). Referring to the second numerical technique, the results have been obtained by using Simulated Annealing SA technique (Kirkpatrick et al., 1983).

The optimum design has been successfully tested through numerical examples, which have also proved the efficiency of using the algebraic formulation for the workspace. Numerical results and computational efforts are discussed to show the evolution of the design process. In addition, a comparison of results obtained with different procedures are shown in order to characterize both the design results and numerical procedures.

2. The design problem

A general open chain 3R manipulator with three revolute joint is sketched in Fig.(1), in which the design parameters are represented using Hartenberg and Denavit H-D notation for the link size as $a_1, a_2, a_3, d_2, d_3, \alpha_1, \alpha_2$. (d_1 is

not meaningful since it shifts up and down the workspace only).

The design parameters of a 3R manipulator represent:

- a_i ($i=1, 2, 3$) the link length measured on the common normal between to consecutive joint axes;
- d_i ($i=2, 3$) the link offset measured as the distance between two consecutive joint along latter of the axes;
- α_i ($i=1, 2$) the twist angle measured as the angle between two consecutive joint axes.

The ring workspace geometry of a 3R manipulator can be described by considering a generation process of the workspace boundary through a suitable torus family envelope, (Ceccarelli, 1989 and 1996). In Fig.(1) the geometry of a generation of a ring workspace boundary for 3R manipulators is illustrated by using the envelope of toroidal surfaces, which are traced by a reference extremal point H for different values of angle θ_3 because of the revolution about Z_1 and Z_2 axes. Particularly, in Fig.(1) one representative toroidal surface of the envelope is drawn and the cross-section contour of the ring workspace boundary is represented together with a cross-section contour of an enveloping torus.

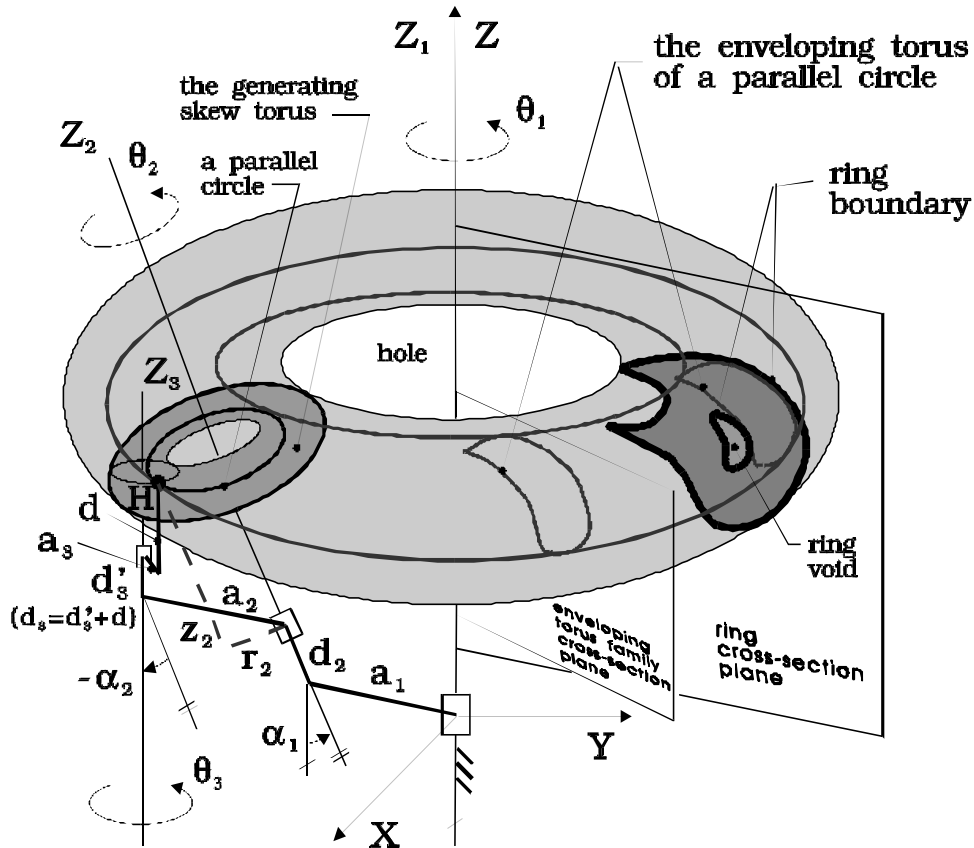


Figure 1. Design parameters and ring workspace for 3R manipulators.

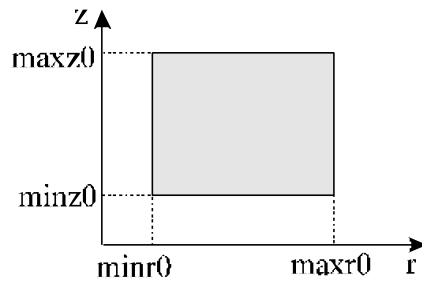


Figure 2. Axial and radial reaches of a given manipulator workspace.

Thus, an algebraic formulation can be deduced, with the hypotheses $C_1 \neq 0$ and $E_1 \neq 0$, and expressed through the radial reach r and the axial reach z of the workspace boundary points with respect to the base frame in the form, (Ceccarelli, 1989 and 1996),

$$r = \left[A_1 - z^2 + \frac{(C_1 z + D_1)G_1 + F_1}{E_1} \right]^{1/2} \quad z = \frac{-L_1 \pm Q_1^{1/2}}{K_1 C_1} - \frac{D_1}{C_1} \quad (1)$$

where the so-called structural coefficients can be computed, after algebraic manipulations in deriving Eqs.(1), as

$$\begin{aligned}
 A_1 &= a_1^2 + r_2^2 + (z_2 + d_2)^2 & B_1 &= -4 a_1^2 r_2^2 & C_1 &= 2 a_1 / s\alpha_1 \\
 D_1 &= -2 a_1 (z_2 + d_2) c\alpha_1 / s\alpha_1 & E_1 &= -2 a_3 (d_2 s\alpha_2 c\theta_3 + a_2 s\theta_3) \\
 F_1 &= 4 a_1^2 a_3 (a_3 s\alpha_2^2 s\theta_3 c\theta_3 + a_2 s\theta_3 - d_3 s\alpha_2 c\alpha_2 c\theta_3) & G_1 &= 2 a_1 a_3 c\alpha_1 s\alpha_2 c\theta_3 / s\alpha_1 \\
 K_1 &= G_1^2 + E_1^2 & L_1 &= F_1 G_1 & Q_1 &= L_1^2 - K_1 (F_1^2 + B_1 E_1^2)
 \end{aligned} \tag{2}$$

in which $c\alpha_1 = \cos\alpha_1$, $s\alpha_1 = \sin\alpha_1$ and so on for other angles.

The reach distances r_2 and z_2 can be expressed from the link geometry as

$$r_2 = [(a_3 c\theta_3 + a_2)^2 + (a_3 s\theta_3 c\alpha_2 + d_3 s\alpha_2)^2]^{1/2} \quad z_2 = d_3 c\alpha_2 - a_3 s\theta_3 s\alpha_2 \tag{3}$$

The angle θ_3 is the joint angle of the extreme pair in the chain and it is the kinematic variable for the workspace determination. In fact, the workspace boundary can be obtained from Eqs.(1)-(3) by scanning θ_3 from 0 to 2π .

Referring to Fig.(2) workspace design data can be prescribed by means of workspace limits in term of radial reach r and axial reach z so that the design problem can be formulated as to find the dimensions of a 3R manipulator arm whose workspace cross-section is within or is delimited by the given axial and radial reaches $\min r_0, \max r_0, \min z_0, \max z_0$.

3. Formulation of an optimum design procedure

The objective of the proposed manipulator design procedure is the dimensional synthesis of 3R manipulator, which will have a minimum total size and a maximum workspace volume. Thus, in the optimization problem the workspace volume and total dimension of the manipulator can be considered together to build a multi-objective function. The optimization problem is subject to given workspace limits as constraints, and additional constraints can be included to obtain manipulator sizes within practical values.

Thus, the optimum design can be formulated as a multi-objective optimization problem in the form

$$\begin{aligned}
 \min f_1 &= -V \\
 \min f_2 &= L
 \end{aligned} \tag{4}$$

subject to

$$\min z \geq \min z_0 \quad ; \quad \max z \leq \max z_0 \quad ; \quad \min r \geq \min r_0 \quad ; \quad \max r \leq \max r_0 \quad ; \tag{5}$$

and

$$\frac{a_3}{a_1} \geq K_0 \quad ; \quad \text{Area} \geq \text{Area}_0 \tag{6}$$

where V indicates a measure of the workspace volume and L is the total dimension of the manipulator. Equation (4) has been used to express the maximization of manipulator workspace in term of volume V and the minimization of the design dimension L of the manipulator.

The workspace volume V can be evaluated according to Pappus-Guldin Theorem by multiplying the cross-section area Area by the radial reach r_g of center of cross-section, Fig. (3a), by means of the formula

$$V = 2\pi \text{Area} r_g \tag{7}$$

in which the Area can be evaluated by summing up the triangle and rectangle areas A_{t_j} and A_{r_j} respectively, as shown in Fig. (3b), in the form as

$$\text{Area} = \sum_{j=1}^N (A_{t_j} + A_{r_j}) \tag{8}$$

with

$$A_{t_j} = \frac{[r(i+1) - r(i)] \cdot [z(i) - z(i+1)]}{2} \quad ; \quad A_{r_j} = [z(i) - z(i+1)] \cdot r(i) \tag{9}$$

in which N is the total number of the triangles and rectangles and i the current number of computed position of a boundary point. N is equal to the number of the boundary points.

By considering the cross-section area, the co-ordinater r_g of the center of the mass can be given as

$$rg = \frac{\sum_{j=1}^N (rgt_j At_j) + \sum_{j=1}^N (rgr_j Ar_j)}{\text{Area}} \tag{10}$$

with

$$rgt_j = r(i) + \frac{r(i+1) - r(i)}{3} \qquad rgr_j = \frac{r(i)}{2} \tag{11}$$

The total dimension L of the manipulator workspace can be evaluated using design parameters to give

$$L = a_1^2 + d_2^2 + a_2^2 + d_3^2 + a_3^2 \tag{12}$$

The Eqs. (5) express the available working area of the robot. In Eqs.(6) the constraint Area-Area0 on the cross-section area of 3R manipulator ensures that the value of the workspace volume will not converge to zero. The first constraint in Eqs.(6) has been used to formulate the necessary condition for void existence, (Gupta and Roth, 1982), by using a proper value of K₀ coefficient in a simplified expression.

A way to perform numerically an efficient design procedure requires a suitable analytical formulation for the involved quantities of the manipulator workspace. Because of the formulation for the manipulator design, workspace volume can be conveniently expressed in term of the workspace boundary. Thus, the algebraic formulation for the ring workspace has been used in the form which has been deduced in (Ceccarelli, 1989 and 1996).

In the case of the numerical code SQP of DOT and SA for the solution of the optimization problem given by Eqs.(4), a multi-objective function has been used using weighting coefficients w₁ and w₂ in the form

$$\min f = w_1 L + w_2 (-V) \tag{13}$$

subject to the same constraints listed in Eqs.(5) and (6). Equation (13) has been formulated in order to achieve the best fit to the numerical performance of the available codes SQP of DOT and SA. In this way, besides working with different optimization techniques, we have experienced the use of a different traditional formulation for a multi-objective function too.

4. Numerical techniques for optimization problem

In this section a review of optimization procedures using traditional and random search techniques is presented.

In the first test, a minimization is achieved by means of a Sequential Quadratic Programming technique, using two different codes, namely: Design Optimization Tools (DOT), and minimax of Matlab Optimization Toolbox (Grace, 1992). Both codes perform the sequential optimization method and a pseudo-objective function is written using the Augmented Lagrange Multiplier Method. The unconstrained minimization is performed by the Broydon–Fletcher–Goldfarb–Shanno (BFGS) method and one-dimensional search uses polynomial interpolation technique. The second numerical procedure is based on the random algorithm Simulated Annealing (SA).

4.1. On overview on Sequential Quadratic Programming

Sequential Quadratic Programming SQP represent a nonlinear programming method. The main idea is the formulation of a sub-problem based on a quadratic approximation of the Lagrangian function.

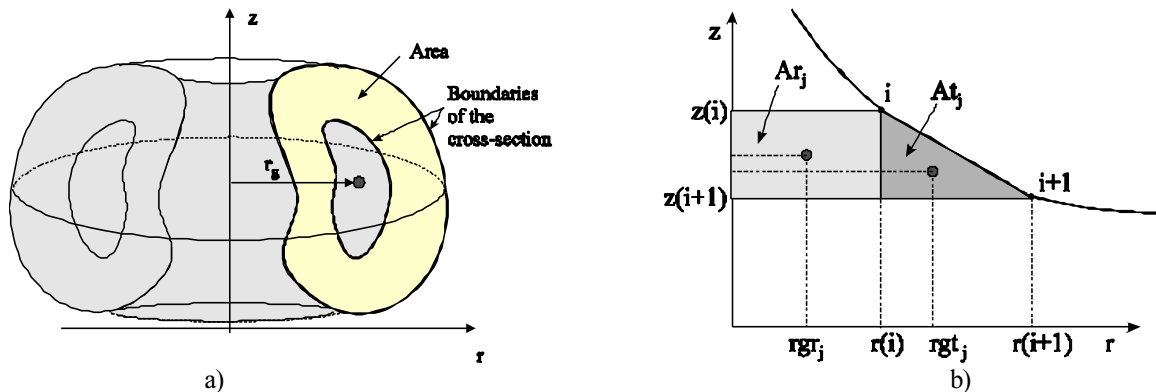


Figure 3. A scheme for evaluation of the workspace of 3R manipulators: a) volume computation; b) area computation from boundary points.

Considering \underline{x}_k the vector containing the design parameters at the step k, the sub-problem is obtained by linearizing the nonlinear constrains, in the form as (Grace, 1992)

$$\min \left\{ \frac{1}{2} \underline{d}^T H_k \underline{d} + \nabla f(\underline{x}_k)^T \underline{d} \right\} \tag{14}$$

subject to

$$\begin{aligned} \nabla g_i(\underline{x})^T \underline{d} + g_i(\underline{x}) &= 0 & i=1, \dots, m_e \\ \nabla g_i(\underline{x})^T \underline{d} + g_i(\underline{x}) &\leq 0 & i=m_e+1, \dots, m \end{aligned} \tag{15}$$

in which H_k is the Hessian matrix of the Lagrangian function Λ

$$\Lambda(\underline{x}, \lambda) = f(\underline{x}) + \sum_{i=1}^m \lambda_i g_i(\underline{x}) \tag{16}$$

in which λ is Lagrangian multipliers; \underline{d} is the search direction; m_e is the number of the equality constraints; m is the total number of the constraints; $\nabla g_i(\underline{x})^T$ is the Jacobian matrix of the constraints; \underline{x} is the vector containing the design parameters $a_1, a_2, a_3, d_2, d_3, \alpha_1, \alpha_2$.

The numerical procedure starts with an initial guess of the manipulator chain solution and, during each iteration k, a quadratic programming problem is solved to provide a search direction \underline{d}_k so that the solution is updated as

$$\underline{x}_{k+1} = \underline{x}_k + \psi_k \underline{d}_k \tag{17}$$

in which ψ_k is a step size obtained minimizing a Lagrangian function.

4.2. On overview on Simulated Annealing

Simulated Annealing is an iterative technique, that is based on random search with adaptive minimum searching along coordinate directions. It can be useful to overpass small variations of the objective function by using a probabilistic criterion in order to avoid to stop the procedure at a local minimum.

This optimization technique is based on an analogy with the annealing process from Metallurgy. Annealing means that the metal is heated to a high temperature, so that the atoms are shaken violently. If the metal is cooled rapidly, the microstructure would be locked into a random unstable state; if the temperature is cooled very slowly, the atoms will fall into patterns that are relatively stable for the temperature. The SA technique provides to optimize results by randomly perturbing the design variables and keeping track the best objective function value from each randomized set of variables. After many tries, the set that produced the best objective function value is designated to be the center about which perturbation will take place for the next temperature change. The temperature change that in this technique is given as the standard deviation of the random number generator, is then reduced, and new tries are performed .

Let each configuration be defined by the set of atom positions where E represents the energy of the configuration and T is the temperature. At each step of the algorithm, a small random displacement is given to an atom and the resulting change ΔE in the energy of the system is computed. When the process generates new states it is either accepted or rejected according to the Metropolis criterion (Metropolis et al, 1953): if $\Delta E \leq 0$ the displacement is accepted, and this configuration is used as the starting point of the next step; if $\Delta E > 0$ the probability P that the configuration is accepted is given by the equation

$$P(\Delta E) = e^{\left(\frac{-\Delta E}{K_b T} \right)} \tag{18}$$

where K_b is the Boltzmann's constant. The choice of the probability function given by Eq. (18) has the consequence that the system evolves according to a Boltzmann distribution.

The random numbers 'r' can be uniformly distributed in the interval [0, 1]. If $r < P(\Delta E)$ the new configuration is accepted, otherwise the original configuration is used to start the next step. The temperature T can be considered a control parameter. The initial temperature T_i is related with the standard deviation of the random perturbation and the final T_f , with the order of magnitude of the desired accuracy, will give the location of the optimum solution.

The annealing schedule starts at a high temperature and it discretely lowers the temperature, using a factor r_t ($0 < r_t < 1$) given by

$$r_t = e^{\frac{\ln\left(\frac{T_f}{T_i}\right)}{n_{temp}-1}} \tag{19}$$

until the system is “frozen” after a determined number n_{temp} of temperatures that can be tried to reach hopefully a global minimum.

In the case of constrained optimization problems the procedures with constrains are transformed in unconstrained problems, by adding a penalty function to the function original objective in order to limit constraint violations (Samarago and Steffen, 2000).

5. Numerical example

In order to prove the soundness of the proposed optimization design procedures numerical examples have been reported. In particular, Fig. (4) shows the initial solution for the proposed test of the optimum design of a 3R manipulator, in which dot lines indicate the cross-section of the initial solution and the grey rectangle indicates the prescribed working area. The prescribed value $Area_0$ of cross-section area has been assumed as 60% of the computed area $Area$; $K_0=2$ is assumed to ensure void absence; and the weights are assumed as $w_1 = w_2 = 0.5$.

Figures (5) to (9) show the obtained results as a comparison among the proposed procedures. Tables (1), (2), (3) show the numerical results for the multi-objective function, the constraints and the design parameters for the design case, respectively.

Figure (5) shows the results of a 3R manipulator in term of shape of cross-section, and objective functions Volume V and manipulator size L , using the numerical technique SQP of Matlab, Fig. (5a), SQP of DOT, Fig. (5b) and SA, Fig. (5c). One can note that the shape of the cross-section is similar in the three cases; the location of the cross-section at the internal assigned area is quite similar too.

The iteration number for the convergence to the optimum solution is quite similar in the case of SQP of Matlab and SQP of DOT, while in the case of SA is very big, as expected. In fact, we have reported a zoom-evolution of objective function V and L in order to show the evolution during the latest 200 iteration number, Fig. (5c) in which one can observe still large variation in searching the optimum value.

One can observe that among the proposed procedure there is a not great difference for the optimum design in the numerical value of the volume V but a considerable difference in term of manipulator size L , as stressed in Table (1).

Figure (6) shows the evolution of two significant constraints ($minz_0-minz$) and ($maxr-maxr_0$) during the optimization process to illustrate how fast and accurately the constraints have been satisfied. SQP of Matlab has converged very rapidly, Fig. (6a), since the variation of the constraints from iteration 50 to 232 changes very little. Similarly, using SQP of DOT, Fig. (6b), the variation of the constraints shows some change with respect to the case of Fig. (6a). But, using SA, Fig. (6c), the variation of the constraints shows waving large changes before to converge. Final numerical results for all constraints are listed in Table (2).

Figures (7), (8) and (9) show the evolution of the design parameters a_1, a_2 and a_3, d_2 and d_3 and finally α_1 and α_2 respectively. Using SQP of Matlab, Figs. (7a), (8a) and (9a) show the variation of the design parameters during the optimization process, and one can observe that from iteration 50 to 232 they show little changes. Using SQP of DOT, Figs. (7b), (8b) and (9b) show the variation of the design parameters, but they change considerably more than in the case of Figs. (7a), (8a) and (9a). Using SA, Figs. (7c), (8c) and (9c) show the variation of the design parameters, and they show large changes before they converge to an optimum value.

Optimum numerical results for all design parameters are listed in Table (3). A considerable difference among the results of the three procedures can be observed mainly in d_2 and d_3 , that are the most sensitive parameters in the proposed optimization problem for designing optimum general 3R manipulators.

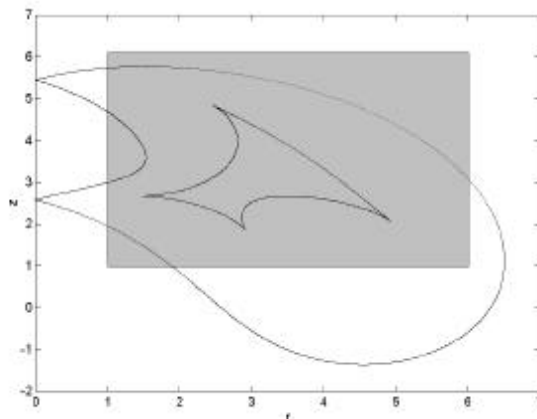


Figure 4. Initial guess for the test of the optimum design of a 3R manipulator.

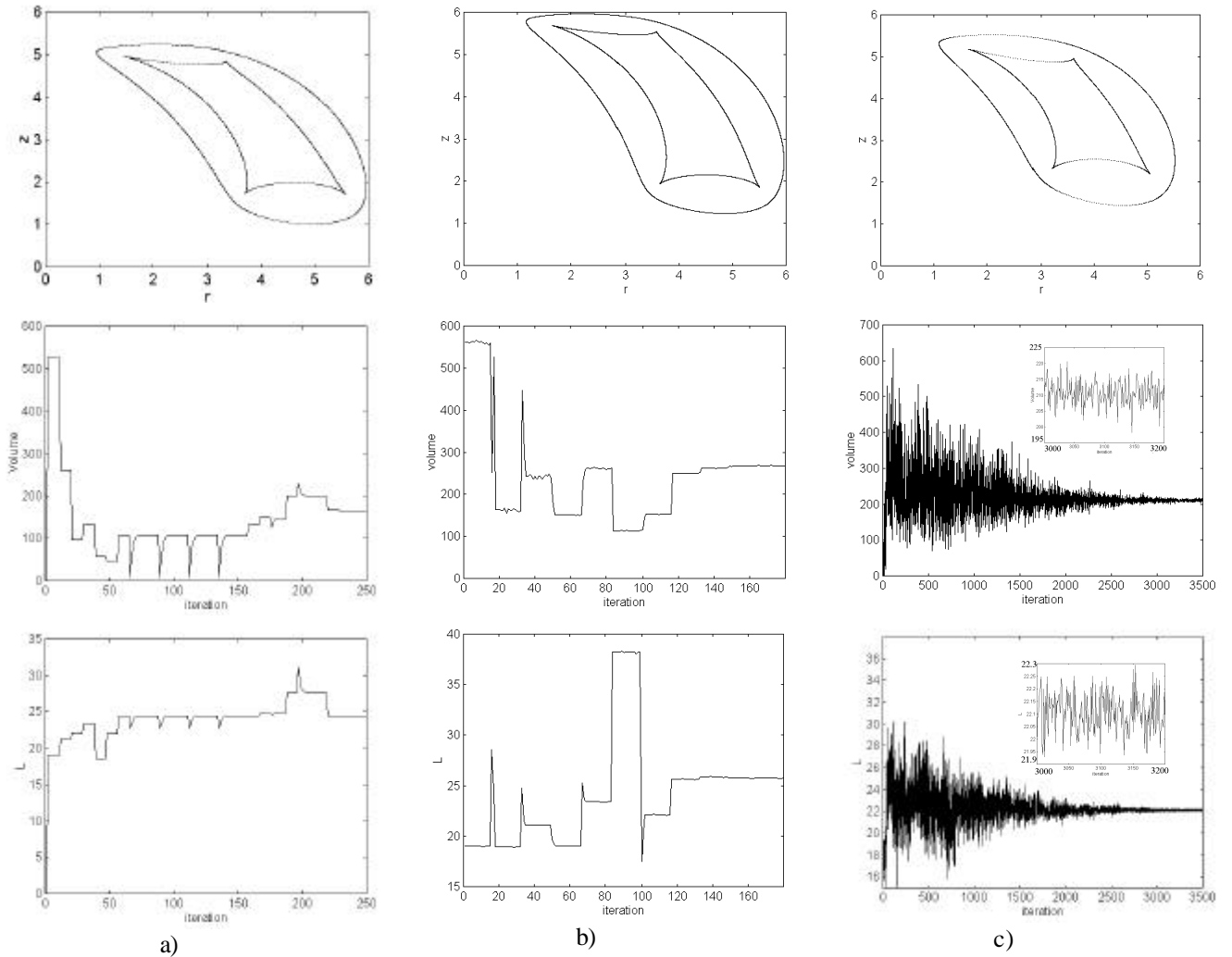


Figure 5. Results of the optimum design of a 3R manipulator in term of obtained workspace cross-section (first row), evolution of objective functions volume V (second row) and manipulator size L (third row) as a functions of the numerical technique: a): SQP of Matlab; b) SQP of DOT; c) Simulated Annealing.

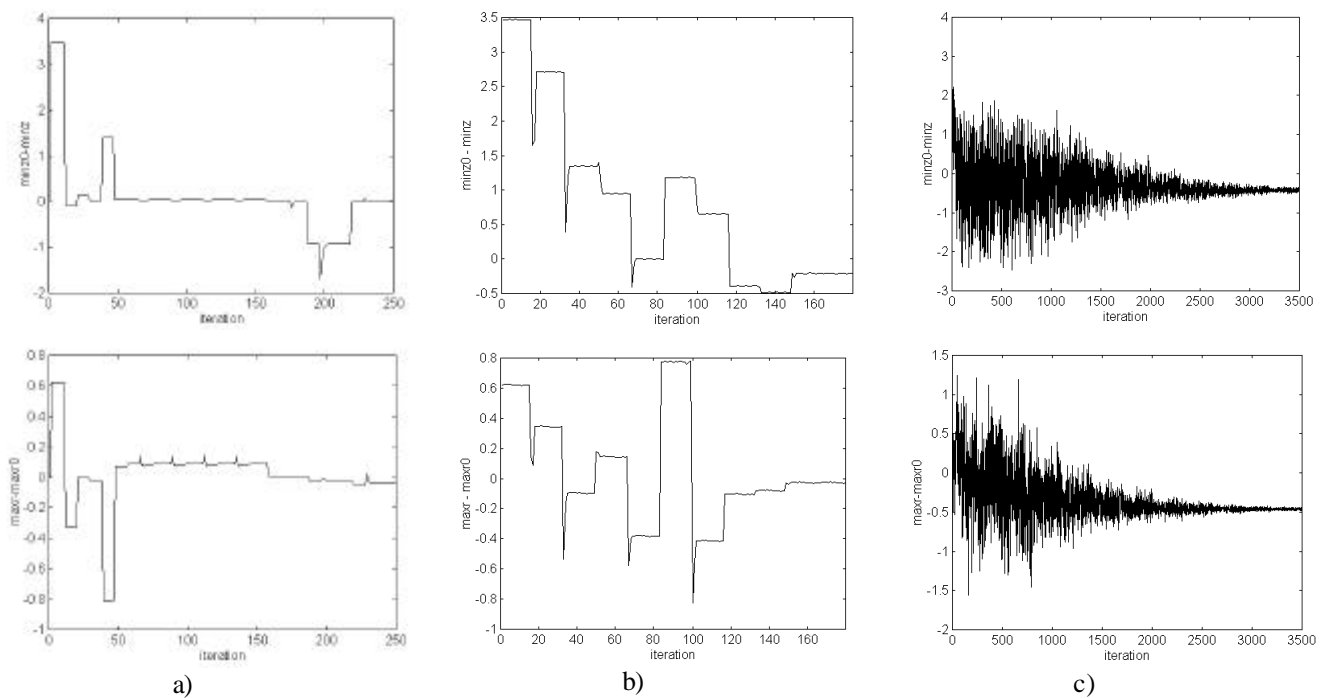


Figure 6. Results of the evolution of constrains ($minz_0 - minz$) and ($maxr - maxr_0$) during the optimization process for the case of Fig. 5, by using: a): SQP of Matlab ; b) SQP of DOT; c) Simulated Annealing.

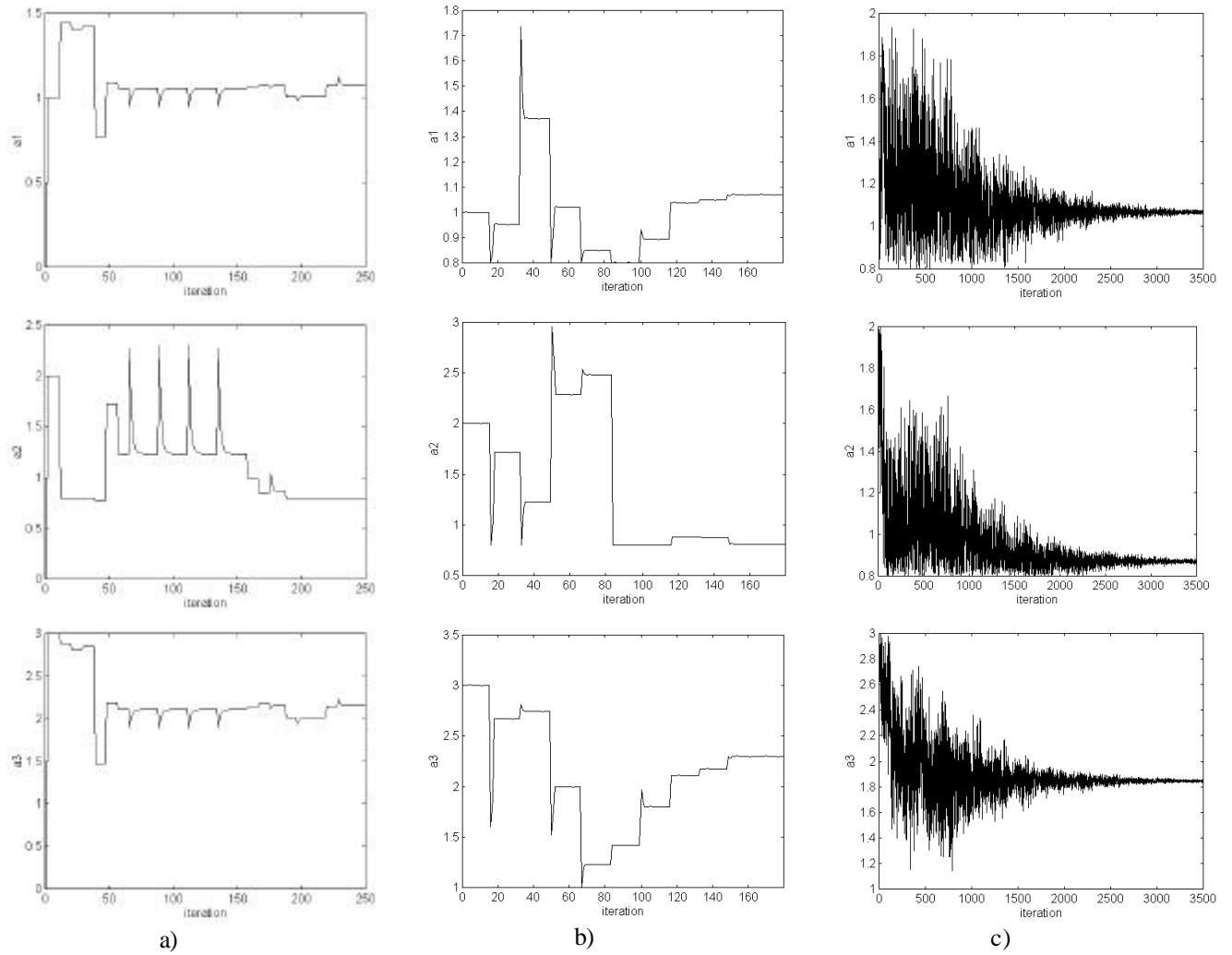


Figure 8. Results of the evolution of the design parameters a_1 , a_2 and a_3 during the optimization process for the case of Fig.5, by using: a): SQP of Matlab; b) SQP of DOT; c) Simulated Annealing.

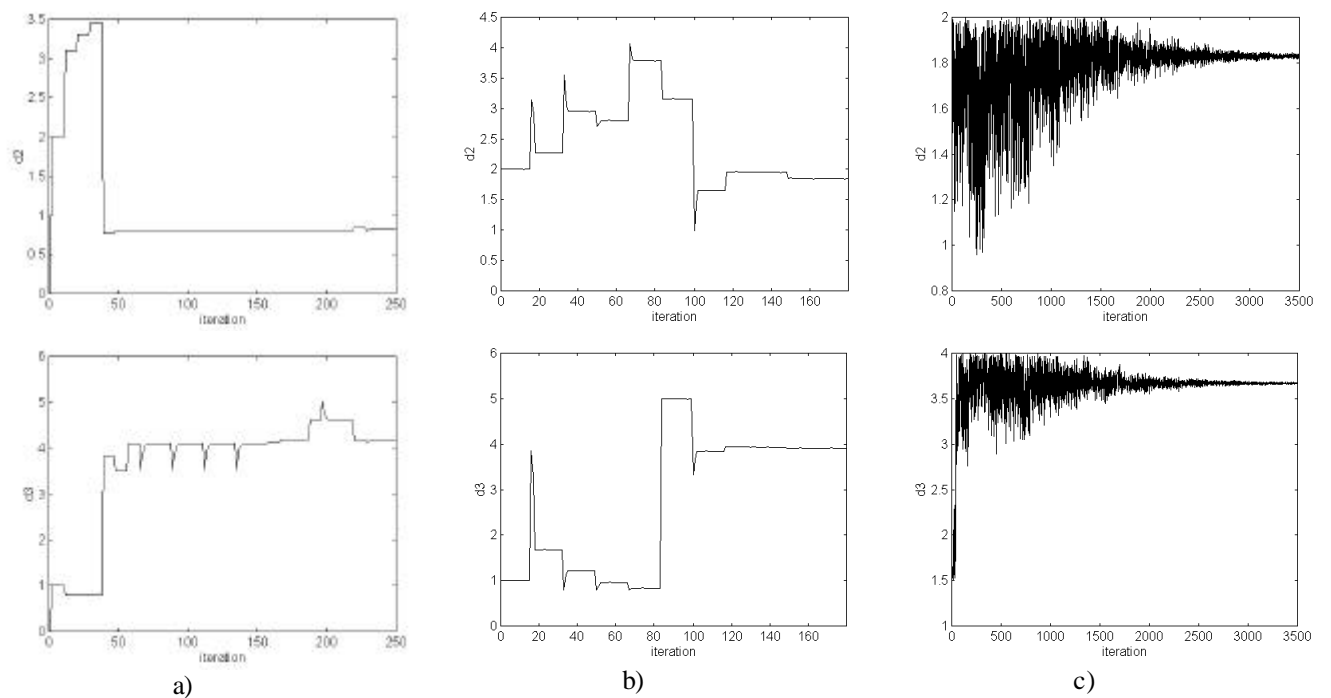


Figure 8. Results of the evolution of the design parameters d_2 and d_3 during the optimization process for the case of Fig. 5, by using: a): SQP of Matlab; b) SQP of DOT; c) Simulated Annealing.

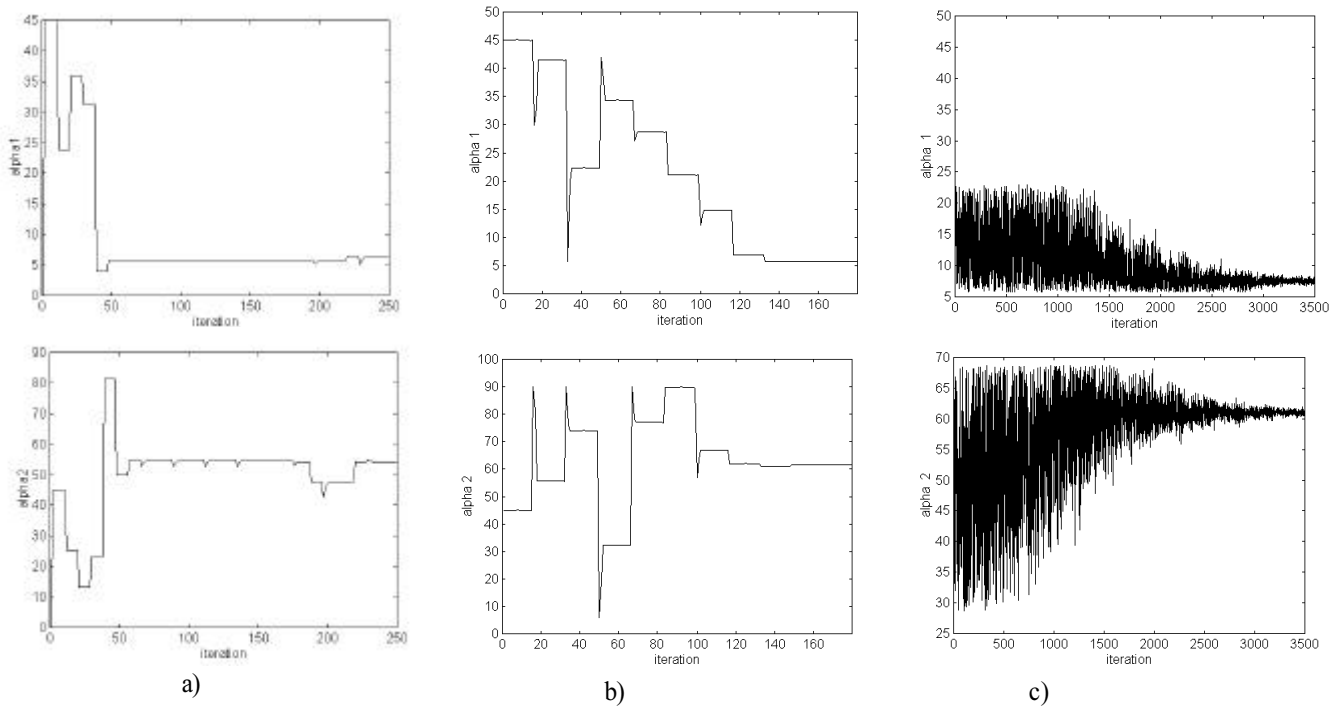


Figure 9. Results of the evolution of the design parameters α_1 and α_2 during the optimization process for the case of Fig. 5, by using: a) SQP of Matlab; b) SQP of DOT; c) Simulated Annealing (Angles are in degrees).

Table 1. Numerical results for the multi-objective functions for the design case of Fig. 5. (Dimensions in u unit length).

Technique / Objective Function	V (u ³)	L (u)	Iteration number
initial	555.750	19.000	-
SQP of Matlab	164.508	24.454	232
SQP of DOT	268.440	25.705	181
Simulated Annealing	213.987	22.222	3,496

Table 2. Numerical results for the constraints in Fig. 6 and 7 (Dimensions are expressed in u unit length).

Technique / Constraints	Minz0-minz (u)	Minr0-minr (u)	Maxr-maxr0 (u)	Maxz-maz0 (u)	Area0-area (u ²)	2a ₁ -a ₃ (u)
initial	3.466	0.999	0.622	-0.047	-11.379	-1.000
SQP of Matlab	-0.005	0.039	-0.041	-0.767	1.853	0.003
SQP of DOT	-0.212	-0.178	-0.026	-0.057	-2.130	-0.159
Simulated Annealing	-0.426	0.053	-0.460	-0.494	-0.120	0.187

Table 3. Numerical results for the design parameters of Figs. 8 to 9 (Dimensions are expressed in u unit length).

Technique / Design Parameters	a ₁ (u)	a ₂ (u)	a ₃ (u)	d ₂ (u)	d ₃ (u)	α_1 (deg)	α_2 (deg)
Initial	1.000	2.000	3.000	2.000	1.000	0.785	0.785
SQP of Matlab	1.080	0.798	2.156	0.836	4.159	0.107	0.946
SQP of DOT	1.069	0.808	2.298	1.841	3.909	0.100	1.075
Simulated Annealing	1.066	0.871	1.846	1.830	3.669	0.132	1.065

6. Conclusions

The optimum design of a general 3R manipulator has been formulated by using workspace characteristics and manipulator size. A suitable formulation for manipulator workspace has been used to obtain efficient numerical procedure for solving the optimization problem. The design problem has been formulated as a multi-objective optimization problem. Three different numerical techniques proposed have been used to solve the design problem.

In the paper, a numerical example is illustrated in detail to show the efficiency of the design process independently of the used numerical technique. Indeed comparison of results obtained with three different technique, namely SQP of Matlab, SQP of DOT and SA, has shown similarly in the results for the obtained optimum solution.

7. Acknowledgments

The second author is thankful to the Brazilian Ministry of Education (Capes) for the grant, which permitted her to spend a sabbatical year in 2001 at the Laboratory of Robotics and Mechatronics in Cassino, Italy.

8. References

- Ceccarelli M., 1989, "On the Workspace of 3R Robot Arms", Fifth IFToMM International Symposium on Theory and Practice of Mechanisms, Bucharest, Vol. II-1, pp.37-46.
- Ceccarelli M., 1993, "Optimal Design and Location of Manipulators", NATO Advanced Study Institute on Computer Aided Analysis of Rigid and Flexible Mechanical Systems, Troia, Vol.II, pp.299-310.
- Ceccarelli M., 1996, "A Formulation for the Workspace Boundary of General N-Revolute Manipulators", IFToMM Jnl Mechanism and Machine Theory, Vol.31,pp.637-646.
- Ceccarelli M., Lanni C., 1999, 'Sintesis Optima de Brazos Manipuladores Considerando las Caracteristicas de su Espacio de Trabajo', Revista Iberoamericana de Ingenieria Mecanica, Vol.3, n.1, pp.49-59.
- Ceccarelli M., Mata V., Valero F., 1994, "Optimal Synthesis of Three-Revolute Manipulators", AIMETA International Jnl Mecanica, Vol. 29, N. 1, pp.95-103.
- Freudenstein F., Primerose E.J.F.,1984, "On the Analysis and Synthesis of the Workspace of a Three-Link, Turning Pair Connected Robot Arm", ASME Jnl of Mechanisms, Transmissions, and Automation in Design, Vol. 106, pp.365-370.
- Gosselin C.D., Guillot M.,1991, "The Synthesis of Manipulators with Prescribed Workspace", ASME Jnl of Mechanical, Design, 1991, vol. 113, pp.451-455.
- Grace A.,1992,"Optimization Toolbox- For use with Matlab", The Math Works Inc., Natick.
- Gupta K.C., Roth B.,1982, " Design Considerations for Manipulator Workspace", ASME Jnl of Mechanical Design, Vol. 104, pp.704-711.
- Kirkpatrick S., Gelatt C.D. Jr, Vecchi M.P., 1983, "Optimization by Simulated Annealing", Science, 220, N. 4598, 1983, pp. 671-680.
- Kumar A., Waldron K.J., 1981,"The Workspaces of a Mechanical Manipulator", ASME Jnl of Mechanical Design, Vol. 103, pp. 665-672.
- Lin C.D., Freudenstein F., 1986, "Optimization of the Workspace of a Three-Link Turning-Pair Connected Robot Arm", The International Journal of Robotics Research, Vol.5, N.2, pp.104-111.
- Mata A.V., Ceccarelli M., 1993, "Funciones Objetivo para la Optimizacion de la Cadena Cinematica de Robots", 1st Iberoamerican Conference on Mechanical Engineering, Madrid, Vol.3, pp.47-54.
- Metropolis N., Rosenbluth A. W., Rosenbluth M. N., Teller A.H., 1953, "Equations of State Calculations by Fast Computing Machines", Jnl of Chemical Physics, 21, pp. 1087-1092.
- Rastegar J., Fardanesh B, 1990, "Geometric Synthesis of Manipulators using the Monte Carlo Methods", ASME Jnl of Mechanical Design, Vol.112, pp.450-452.
- Saramago S.F.P., Steffen Jr V., 2000, "Optimal Trajectory Planning of Robot Manipulators Using Classical Techniques and Simulated Annealing", in Nonlinear Dynamics, Chaos, Control and Their Applications to Engineering Sciences, Vol.6, Applications of Nonlinear Phenomena, 2000, pp. 148-156.
- Tsai Y.C., Soni A.H., 1985, " Workspace Synthesis of 3R, 4R, 5R and 6R Robots", IFToMM Jnl Mechanism and Machine Theory, Vol.20, n.6, pp.555-563.
- Vanderplaats, G., 1995, "DOT - Design Optimization Tools Program – Users Manual", Vanderplaats Research & Development, Inc, Colorado Springs.
- Vijaykumar R., Waldron K.J., Tsai M.J.,1986, "Geometric Optimization of Serial Chain Manipulator Structures for Working Volukme and Dexterity", The International Jnl of Robotics Research, Vol.5, n.2, pp.91-103.
- Yang D.C.H., Lee T.W.,1984, "Heuristic Combinatorial Optimization in the Design of Manipulator Workspace", IEEE Transactions on Systems, Man, and Cybernetics, 1984, Vol.SMC-14, n.4, pp.571-580.

NUMERICAL SIMULATION AND EXPERIMENTAL VALIDATION OF DYNAMIC CHARACTERISTICS OF CAPAMAN (CASSINO PARALLEL MANIPULATOR)

João Carlos Mendes Carvalho

Universidade Federal de Uberlândia, Faculdade de Engenharia Mecânica, Campus Santa Mônica, 38400-902 Uberlândia/MG, Brasil
jcmendes@mecanica.ufu.br

Marco Ceccarelli

University of Cassino, DiMSAT, Laboratory of Robotics and Mechatronics, Via di Biasio 43, 03043 Cassino (FR), Italy
ceccarelli@ing.unicas.it

Abstract. In this paper we have presented a numerical simulation and experimental validation for the inverse dynamics of CaPaMan – Cassino Parallel Manipulator in order to compute the input torque on mechanism legs for a known trajectory of the mobile platform. The inertia effects of the movable platform has been obtained using the Newton-Euler formulation and the kinetostatic analysis of mechanisms for computing the inertial effects of the parallelogram legs. Numerical simulations are reported and compared with experimental results obtained with a built prototype and show the soundness of the proposed formulation.

Keywords: *Robot Dynamics, Parallel Manipulators, Numerical Simulation*

1. Introduction

In last years parallel architectures have been extensively studied because they show advantages such as higher stiffness and accuracy positioning with respect to the serial architectures, and can operate at high velocities and accelerations. These characteristics permit their use for many applications: assembly and disassembly processes, packing, manipulation, milling machines and motion simulation.

Thus, several new parallel mechanisms have been conceived and designed and built together with the development of theoretical and practical investigations such as presented in Clavel (1988), Merlet and Gosselin (1991), and Miller (1995).

Cassino Parallel Manipulator – CaPaMan is a parallel manipulator, having three degrees of freedom, which was conceived at the Laboratory of Robotics and Mechatronics at Cassino, Italy (Ceccarelli, 1997). Performances and suitable formulation for kinematics have been investigated and are reported in Ceccarelli and Figliolini (1997) and Ceccarelli and Ottaviano (2000). A prototype has been built and an application of CaPaMan as an earthquake simulator, which can reproduce a really happened earthquake is reported in Ceccarelli et al (1999b) and Carvalho and Ceccarelli, (1999b and c). Using the built prototype some experimental validations was carried out and are reported in Ceccarelli et al (1999a).

When the robot operates at high speed, the inertial effects are important and cannot be neglected. Then, the robot dynamic behavior must be known to improve the accuracy and control of the robot motion and, the forces and couples that may be exerted by the end-effector.

In general on parallel architectures the legs are lighter than the mobile platform. Then, its inertial effects are neglected like is reported in Carvalho and Ceccarelli (1999a and 2001a).

In this paper we will present the experimental activity on CaPaMan prototype using accelerometers and a torsionmeter to measure CaPaMan characteristics and validate the dynamic model of CaPaMan using two analytical model. The first consider only the inertial effects of the movable platform, neglecting the inertial effects of the legs. The second by considering both the inertial effects of the mobile platform and the legs.

The inverse analytical model for the dynamics of CaPaMan compute the input torques, which are necessary for a given trajectory of movable plate. The dynamic behavior of movable plate has been analyzed by means of Newton-Euler formulation in a first dynamic model (Carvalho and Ceccarelli, 2001a) and after it has been superposed to the dynamic behavior of articulated parallelograms of the legs, which has been formulated using the kinetostatic analysis of mechanisms (Carvalho and Ceccarelli, 2001b).

Results of numerical simulations and experimental tests are presented to show the validity of the proposed approach and the performance of CaPaMan.

2. CaPaMan architecture and kinematics

CaPaMan – Cassino Parallel Manipulator is a 3 d.o.f. parallel manipulator, which is characterised by three four-bar mechanism legs AP connecting the fixed base FP and the movable platform MP as shown in Fig. 1. The 4-bar mechanisms are assembled in an equilateral triangle way with respect to the fixed base giving symmetry properties to the manipulator. On the coupler link of each 4-bar mechanism is installed a passive prismatic joint SJ which is connected to the movable platform by a rod CB and a spherical joint BJ. This configuration enables the rod CB translate

along the passive prismatic joint SJ keeping its vertical posture while the spherical joint BJ allows the movable platform to rotate in space. Like the constructed prototype, in this paper all the three 4-bar mechanism legs are considered as articulated parallelograms. Then, the kinematic variables are the input crank angles a_k ($k=1,2,3$).

In order to describe the kinematic and dynamic behaviour of CaPaMan, a base reference frame $OXYZ$ has been located in center of the fixed plate FP and a frame $HX_pY_pZ_p$ has been fixed to the center of the movable platform MP. A reference frame $O_kX_kY_kZ_k$ has been assumed fixed on the plane of each k articulated parallelogram where O_k coincides with the center point of frame link a_k .

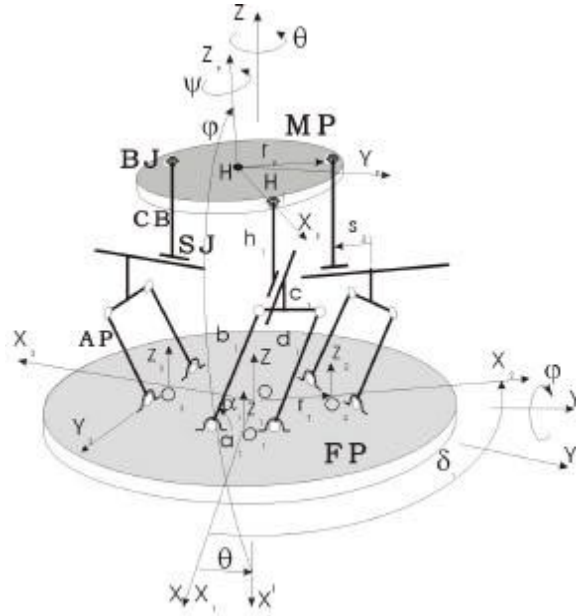


Figure 1. Architecture and kinematic parameters of CaPaMan.

The kinematic formulation can be obtained by using a vectors and matrix formulation. In fact, the position coordinates x , y and z of the center point H and the orientation angles α , j and γ of the movable platform MP can be written as function of the coordinates y_k and z_k of the points H_k . And then, these coordinates are easily expressed as function of the input crank angles a_k ($k=1,2,3$).

Thus, the forward displacement and orientation analysis can be given by (Ceccarelli, 1997):

$$\begin{aligned}
 x &= \frac{y_3 - y_2}{\sqrt{3}} - \frac{r_p}{2} (1 - \sin j) \cos(\gamma - \alpha); & y &= y_1 - r_p (\sin \gamma \cos \alpha + \cos \gamma \sin j \sin \alpha) \\
 z &= \frac{z_1 + z_2 + z_3}{3}; & \alpha &= \sin^{-1} \left[2 \frac{y_1 + y_2 + y_3}{3r_p (1 + \sin j)} \right] - \gamma; & \gamma &= \tan^{-1} \left(\sqrt{3} \frac{z_3 - z_2}{2z_1 - z_2 - z_3} \right) \\
 j &= \cos^{-1} \left(\pm \frac{2}{3r_p} \sqrt{z_1^2 + z_2^2 + z_3^2 - z_1 z_2 - z_2 z_3 - z_1 z_3} \right) & & \text{with } (z \geq z_1 \Rightarrow "+"; z < z_1 \Rightarrow "-") & (1)
 \end{aligned}$$

Where the coordinates y_k and z_k are given by:

$$\begin{aligned}
 y_k &= b_k \cos a_k \\
 z_k &= b_k \sin a_k + h_k
 \end{aligned} \quad (2)$$

Since a PID control of the actuators is available for commercial motors, the input motion of the input crank b_k can be given by a cubic function of time t between given initial a_{ki} and final a_{kf} angles at initial t_{ki} and final t_{kf} times, respectively, as

$$a_k = a_{ki} + \frac{3(a_{kf} - a_{ki})}{(t_{kf} - t_{ki})^2} t^2 - \frac{2(a_{kf} - a_{ki})}{(t_{kf} - t_{ki})^3} t^3 \quad (3)$$

The inverse kinematic formulation can be obtained by inverting the above Eqs. (1). This formulation and analysis of workspace, orientation capabilities and displacement generation for CaPaMan are presented and discussed in Ceccarelli (1997) and Ceccarelli and Figliolini (1997).

3. The inverse dynamics of CaPaMan

The symmetry characteristics of CaPaMan architecture have been useful to formulate the analytical dynamic equations to compute the input torques which are necessary for a given motion trajectory of the movable platform. Considerations have been made in order to simplify the equations: the effects of link elasticity and viscous damping of the joints have been neglected; links are assumed to be rigid bodies and the joints are frictionless and have no clearance.

Due to the legs of parallel architectures are lighter than the movable plate, it has been considered, in a first way, only the inertial effects of the movable platform. After, it has been superposed with the inertial effects of the legs. The both dynamic analysis are presented in the following items.

3.1. Dynamics of the movable platform

The analytical formulation for the dynamics of the mobile platform of CaPaMan can be obtained by using the Newton-Euler approach. The position, velocity and acceleration of the movable platform are given by the direct kinematic model.

Assuming F_{ext} and N_{ext} as the external force and torque and G the platform weight, from Fig. 2 the dynamic equilibrium for the mobile platform can be expressed as:

$$F + F_{ext} + G = F_{in} \tag{4}$$

$$N + N_{ext} = N_{in} \tag{5}$$

Where F is the sum of the reaction force F_i acting at point H_k ($k=1,2,3$) of the MP platform and N the consequent torque.

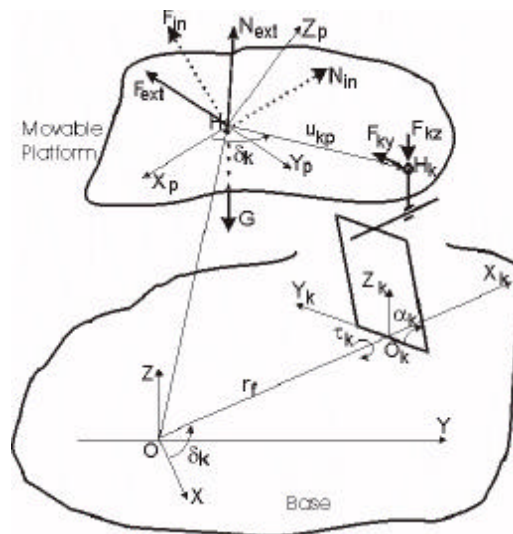


Figure 2. Forces acting on the mobile platform.

Being m the mass of the platform; a_H the acceleration of center point H ; \dot{w} and w the angular acceleration and velocity of the platform; and the inertia matrix I that can be determined as $I = R I_c R^T$, by using the rotation matrix R between the $OXYZ$ and $HX_pY_pZ_p$ reference frames, and the inertia matrix I_c of the platform with respect to its center of mass, we can consider the expressions:

$$\begin{aligned}
 F_{in} &= m a_H \\
 N_{in} &= I \dot{w} + w \times (I w) \\
 F &= \sum_{k=1}^3 F_k \\
 N &= \sum_{k=1}^3 (r_p R u_{kp}) \times F_k \quad (k=1,2,3)
 \end{aligned} \tag{6}$$

By neglecting the friction on prismatic and spherical joints, the only forces applied to the rods CB by the mobile platform are those which are contained in the plane of the articulated parallelogram i.e. F_{ky} and F_{kz} as shown in Fig. 2. The F_{kx} component, when it is not zero, determines the sliding of the prismatic joint to an equilibrium position. Thus, the components of the resultant force F and torque N , given by Eqs. (6), can be written as:

$$\begin{Bmatrix} F_x \\ F_y \\ F_z \\ N_x \\ N_y \\ N_z \end{Bmatrix} = \begin{Bmatrix} -\frac{\sqrt{3}}{2}F_{2y} + \frac{\sqrt{3}}{2}F_{3y} \\ F_{1y} - \frac{1}{2}F_{2y} - \frac{1}{2}F_{3y} \\ F_{1z} + F_{2z} + F_{3z} \\ -u_{1z}F_{1y} + \frac{1}{2}u_{2z}F_{2y} + \frac{1}{2}u_{3z}F_{3y} + u_{1y}F_{1z} + u_{2y}F_{2z} + u_{3y}F_{3z} \\ -\frac{\sqrt{3}}{2}u_{2z}F_{2y} + \frac{\sqrt{3}}{2}u_{3z}F_{3y} - u_{1x}F_{1z} - u_{2x}F_{2z} - u_{3x}F_{3z} \\ \frac{1}{2}(\sqrt{3}u_{2y} - u_{2x})F_{2y} - \frac{1}{2}(\sqrt{3}u_{3y} + u_{3x})F_{3y} + u_{1x}F_{1y} \end{Bmatrix} \quad (7)$$

with

$$\begin{Bmatrix} u_{kx} \\ u_{ky} \\ u_{kz} \end{Bmatrix} = r_p R \begin{Bmatrix} \cos d_k \\ \sin d_k \\ 0 \end{Bmatrix}; \quad (k=1,2,3) \quad \text{and} \quad d_1 = 0; d_2 = 2p/3; d_3 = 4p/3. \quad (8)$$

Equations (4), (5) and (6) can be solved in a closed form to obtain an explicit expression for forces F_{ky} and F_{kz} .

Referring to Fig. 2 and once the reaction forces in the spherical joints H_k are computed, the torque τ_{pk} ($k=1,2,3$) on the input crank shaft of each articulated parallelogram can be obtained by considering only the inertial effects of the movable platform as

$$\tau_{pk} = F_{kz} \left(b_k \cos a_k + \frac{c_k}{2} \right) - F_{ky} (b_k \sin a_k + h_k) \quad , \quad (k=1,2,3) \quad (9)$$

3.2. Dynamics of the articulated parallelograms

For the analytical analysis of the dynamics of the CaPaMan legs the linear accelerations of the mass center and the angular accelerations of the moving links has been determined by the kinematic analysis of the articulated parallelograms. Based on built prototype used on experiments the mass center of links are coinciding with its figure centers.

From the kinetostatic analysis of mechanisms one can consider the dynamic equilibrium of a rigid body in the presence of an inertia force F_{in} which application point is determined by an offset e from its mass center, i.e.

$$F_{in} = -m a_G \quad \text{and} \quad e = \frac{I_G \ddot{\alpha}}{F_{in}} \quad (10)$$

Where m is the mass of the rigid body; a_G the linear acceleration of its center mass; I_G its inertia matrix with respect to its center of mass and $\ddot{\alpha}$ its angular acceleration. Thus, the kinetostatic analysis of the each k articulated parallelogram gives the inertia forces F_{inbk} , F_{inck} and F_{indk} , with the offset e_{bk} , e_{ck} and e_{dk} from the mass center of links b , c and d , respectively, as shown in Fig. 3, are given as

$$F_{inbk} = -m_{bk} a_{Gbk}; \quad F_{inck} = -m_{ck} a_{Gck}; \quad F_{indk} = -m_{dk} a_{Gdk}; \quad (11)$$

$$e_{bk} = \frac{I_{Gbk} \dot{w}_{bk}}{F_{inbk}} = \frac{I_{Gbk} \ddot{\alpha}_k}{F_{inbk}}; \quad e_{ck} = \frac{I_{Gck} \dot{w}_{ck}}{F_{inck}} = 0; \quad e_{dk} = \frac{I_{Gdk} \dot{w}_{dk}}{F_{indk}} = \frac{I_{Gdk} \ddot{\alpha}_k}{F_{indk}} \quad (12)$$

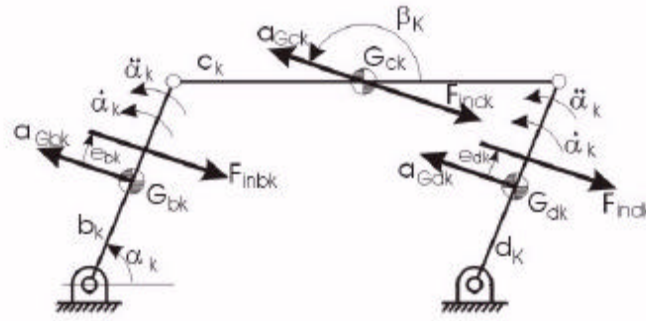


Figure 3. Inertia forces arising on each articulated parallelogram.

The superposition principle allows to obtain separately the effects of each inertia force of links and then superposed to determine their combined effect. The input torque τ_{Mk} on each articulated parallelogram k obtained from the inertia effects of the links b , c , d and its gravitational effects can be written as

$$\tau_{Mk} = 2l_{bk} F_{inbk} \sin(a_k - b_k + p) + F_{23k} b \sin(a_k + p - g_k) + [(m_{bk} + m_{ck})b \cos a_k + m_{ck} c/2]g \quad (13)$$

Where the angle b_k defines the direction of the acceleration of the mass center of the link k with respect to the horizontal axis, and g_k defines the direction of the reaction force vector acting on the ground pivot of link d_k , both assumed to be positive counter-clockwise, and

$$l_{bk} = \frac{b}{2} + \frac{I_{Gbk} \ddot{\alpha}_k}{F_{inbk} \sin(a_k - b_k + p)}$$

$$F_{23k} = \sqrt{\left[F_{inck} \left[\cos(b_k + p) + \frac{\sin(p - b_k)}{2 \tan a_k} \right] \right]^2 + \left[\frac{F_{inck} \sin(b_k + p)}{2} \right]^2}$$

$$g_k = \operatorname{tg}^{-1} \left\{ \frac{F_{inck} \left[\cos(b_k + p) + \frac{\sin(p - b_k)}{2 \tan a_k} \right]}{\left[\frac{F_{inck} \sin(b_k + p)}{2} \right]} \right\} \quad (14)$$

3.2. The total input torque on the articulated parallelogram

Using the principle of superposition, the dynamics effects of movable platform can be superposed to the dynamics effects of the links of the articulated parallelograms. Thus, the total torque τ_k on the input crank shaft of the articulated parallelogram k is obtained by summing the torques τ_{Pk} and τ_{Mk} from Eqs. (9) and (13): $\tau_k = \tau_{Pk} + \tau_{Mk}$.

4. Experimental system

An experimental system with a built prototype has been designed and settled up at the Laboratory of Robotics and Mechatronics in Cassino, Italy. The prototype of CaPaMan has built using commercial measuring sensors and equipment, having the following data: $a_k=c_k=200\text{mm}$; $b_k=d_k=80\text{mm}$; $h_k=116\text{mm}$; $r_p=r_b=109.5\text{mm}$; $a_k=45$ to 135 deg; $s_k=-50$ to $+50\text{mm}$; $m=2.912\text{kg}$; $m_{dk}=m_{bk}=0.103\text{kg}$; $m_{ck}=0.547\text{kg}$; $I_{Gck}=0.0027\text{kgm}^2$; and $I_{Gbk}=I_{Gdk}=0.00013\text{kgm}^2$, ($k=1,2,3$). Three accelerometers have been fixed on the three perpendicular faces of a small block and it has been installed on the mobile platform along the axis directions of the mobile frame as shown in Fig. 4. A dynamic torsionmeter has been installed on the actuator shaft of the articulated parallelogram $k=1$ to measure the torque of the motor. The passive prismatic joint SJ has been selected from low-cost commercial linear ball guide. The spherical joint BJ has been designed as a spherical head of the connecting bar CB and it has been constructed by a well turning as well as a spherical hole on the mobile plate has been obtained by milling the plate yet. Then a simple cover has been built to keep the spherical head of CB into the spherical hole on MP and it has been fixed to MP by means of two small screws. The four-bar linkages has been obtained by usual machining and small commercial ball bearings have been used for it. Trajectories of the mobile platform are defined by a suitable motion of the motors by using a PID control as given by

Eq. (3). Data from the sensors are acquired with an acquisition card installed in a PC using a suitable virtual instrument with LabVIEW®.

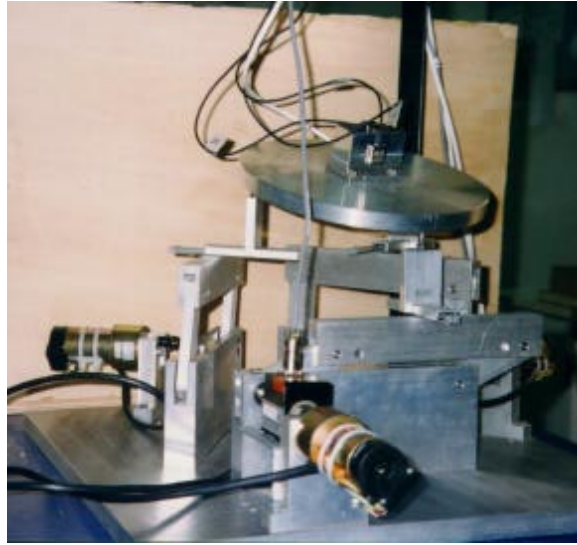


Figure 4. CaPaMan prototype with measured sensors on the mobile platform and a torsionmeter on an input shaft.

5. Experimental results and numerical simulations

Several numerical simulations and experimental activities have been carried out obtaining results like these shown in this paper, which can be useful to identify mechanical characteristics of CaPaMan, also for practical applications, and to validate models for kinematic and dynamic behaviour.

In this section we have been presented some numerical simulations and experimental results for the three cases: a) considering only the inertial effects of the mobile platform; b) superposing the inertial effects of the mobile platform and the articulated parallelograms and, c) the experimental results. Always it has been assumed that no external force and torque are applied to the mobile platform i.e. $F_{ext}=0$ and $N_{ext}=0$.

5.1. A spatial circular arc motion

In this example the mobile platform moves along a circular arc in the space. The motion is obtained by moving only one leg mechanism and means that the crank starts at the initial input angles α_{ki} and goes to the final angles α_{kf} . The other two leg mechanisms, in this case, remains stopped at $\alpha_k=90\text{deg}$. The torque is measured on the crank of the moved leg mechanism ($k=1$).

Experimental and numerical simulations have shown that the torques on the fixed crank are negligible vis-a-vis to the moved one. Two examples are presented.

Example 1.

In this example the crank of the mechanism leg "1" starts at the initial input angles $\alpha_{1i} = 50\text{deg}$ and goes to the final angles $\alpha_{1f}=130\text{deg}$ with duration of 1 second.

In Figure 5 is shown the simulated torque on crank of the leg "1" by considering: a) only the inertial effects of the mobile platform; b) only the inertial effects of the articulated parallelograms; and c) superposing the inertial effects of the mobile platform and the articulated parallelograms.

It can be observed that the inertial effects of the articulated parallelogram is smaller than the mobile platform one as previously commented.

In Figure 6 the numerical simulation and the experimental results are superposed. It can be noted that the difference between the minimum and the maximum torques are quite similar and the similarity of torque curves. In the experimental results there exists an offset in the values not shown in the figure.

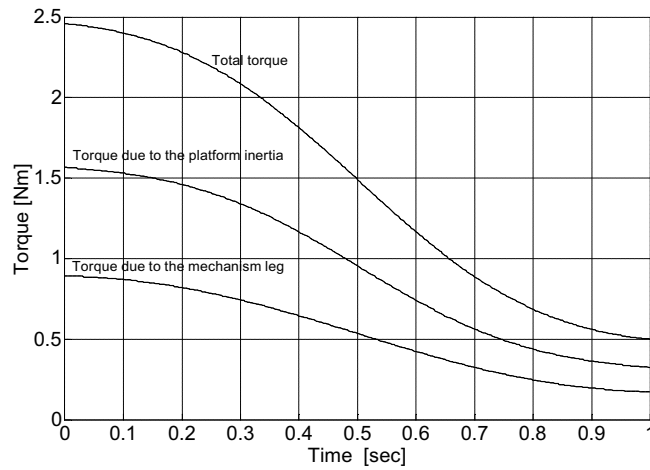


Figure 5. Numerical results of a dynamic simulation of CaPaMan for a circular arc motion.

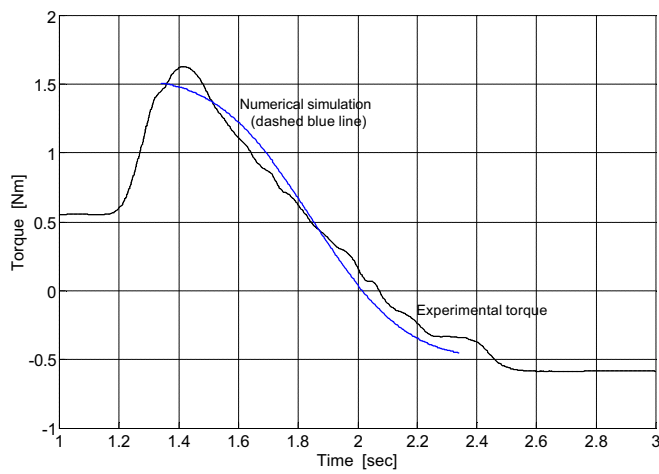


Figure 6. Superposing the numerical and experimental results of the torque on moved crank for a circular arc motion.

Example 2.

For this example the circular arc is prescribed within 0.5 seconds and the motion starts at the initial input angles $\alpha_{Ii} = 60\text{deg}$ and goes to the final angles $\alpha_{If} = 120\text{deg}$.

In the same way of the example 1, there exists an offset between of the numerical simulation and the experimental results. Superposing the results it can be observed that the curves are quite similar as shown in Fig. 7.

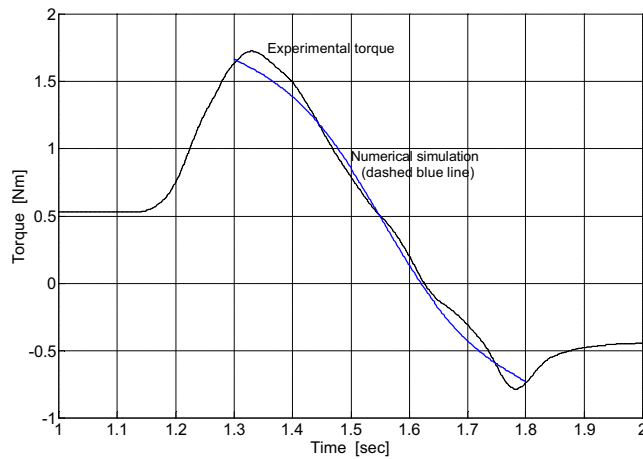


Figure 7. Numerical simulation superposed to the experimental results of the torque on moved crank for a circular arc motion.

5.2. A straight-line motion

This simulation is obtained by considering the mobile platform to move along a vertical straight-line from point (0, 0, 177.3)mm to end point (0, 0, 196)mm and returning at initial point (0, 0, 177.3)mm, with duration of 0.5 seconds. To obtain this motion all the three articulated mechanism legs are the same i.e., all the cranks motion starts at the same initial input angles α_{ki} and goes to the final angles α_{kf} . For the presented example the initial angles are $\alpha_{ki}=50\text{deg}$ and the final angles $\alpha_{kf}=130\text{deg}$. The vertical linear motion is accomplished with a small rotation $\theta+\psi$ about the axis Z, since φ is constant and, the higher position is attained when the crank angles are 90deg. In this motion, the torques on all the three cranks are equals.

In Figure 8 are shown the numerical results of torques by considering: a) only the inertial effects of the mobile platform; b) only the inertial effects of the articulated parallelograms; and c) superposing the inertial effects of the mobile platform and the articulated parallelograms. Like the other examples, the inertial effects of the articulated parallelogram is smaller than the mobile platform one.

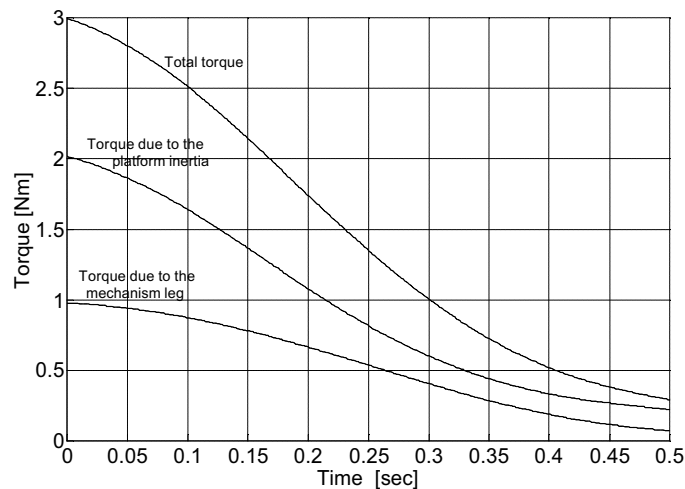


Figure 8. Numerical results of a dynamic simulation of CaPaMan for a vertical straight-line motion.

Figure 9 shows the superposed results of the numerical and experimental simulations. In this case still exists an offset between the values not shown in the figure. The difference between the minimum and the maximum torques for both results are quite similar.

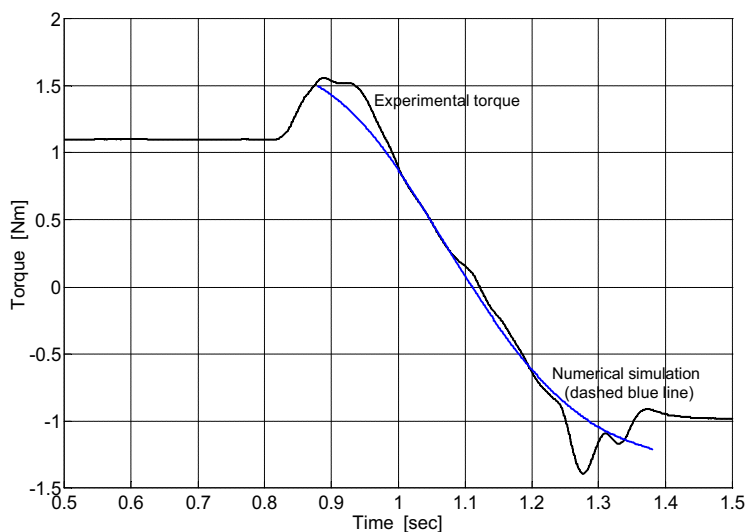


Figure 9. Numerical simulation superposed to the experimental results of the torque for a vertical straight-line motion.

6. Conclusions

In this paper we have presented a numerical simulation and experimental validation model for the inverse dynamics of CaPaMan – Cassino Parallel Manipulator by considering the inertia effects of the movable platform and the mechanism legs. The symmetry characteristics of CaPaMan has been useful to analyse the dynamic behaviour of the mobile platform using the Newton-Euler formulation and the kinetostatic analysis of mechanisms for computing the inertial effects of the parallelogram legs.

Numerical simulations have been reported with the aim to compare to the experimental results obtained with a prototype to illustrate the soundness of the proposed formulation.

Based on Figures 5 and 6 and others simulations not presented in this paper we can conclude that the dynamic behaviour of the legs can not be neglected for all parallel structures as considered by many authors.

In the experimental results have been observed an offset related to the numerical ones for about 1Nm. A new structure has been constructed using precision joints and new experiments will be made to verify the origin of the offset between the numerical and experimental results.

7. References

- Carvalho, J.C.M. and Ceccarelli, M., 1999a, "A Dynamic Analysis for Cassino Parallel Manipulator", Proceedings of the 10th IFToMM World Congress, Vol. 3, Oulu, Finland, pp.1202-1207.
- Carvalho, J.C.M. and Ceccarelli, M., 1999b, "Earthquake Simulation by Means of CaPaMan", Proceedings of the 4^o Congreso Iberoamericano de Ingenieria Mecanica, Vol. 2, Santiago del Chile, Chile.
- Carvalho, J.C.M. and Ceccarelli, M., 1999c, "Seismic Motion Simulation Based on Cassino Parallel Manipulator", Proceedings of the XVth Brazilian Congress on Mechanical Engineering – COBEM99, Campinas, Brazil.
- Carvalho, J.C.M., Ceccarelli, M., 2001a, "A Closed Form Formulation for the Inverse Dynamics of Cassino Parallel Manipulator", J. Multibody System Dynamics, Vol.5, No.2.
- Carvalho, J.C.M., Ceccarelli, M., 2001b, "The Inverse Dynamics of Cassino Parallel Manipulator", 2nd Workshop on Computational Kinematics, Seoul, Korea, May 20-22 (Accepted to be presented).
- Ceccarelli, M. and Figliolini G., 1997, "Mechanical Characteristics of CaPaMan (Cassino Parallel Manipulator)", Proceedings of the 3rd Asian Conference on Robotics and its Application, Tokyo, Japan, pp. 301-308.
- Ceccarelli, M., 1997, "A New 3 D.O.F. Parallel Spatial Mechanism", IFToMM Journal Mechanism and Machine Theory, Vol. 32, No. 8, pp. 895-902.
- Ceccarelli, M., Pugliese, F. and Carvalho, J.C.M., 1999a, "An Experimental System for Measuring CaPaMan Characteristics", Proceedings of the 8th International Workshop on Robotics in Alpe-Adria-Danube Region – RAAD99, Munich, Germany, pp. 31-36.
- Ceccarelli, M., Carvalho, J.C.M., Pugliese, F. and Lanni, C., 1999b, "CaPaMan as Sensored earthquake Simulator", Proceedings of the International Conference on Intelligent Robots and Systems – IROS99, Vol. 3, Kyongju, Korea, pp. 1501-1506.
- Ceccarelli, M., Ottaviano, E., 2000, "An Analytical Design for CaPaMan with Prescribed Position and Orientation", Proceedings of the 2000 ASME Biennial Mechanisms and Robotics Conference, Baltimore, USA.
- Clavel, R. 1988, "DELTA: A Fast Robot with Parallel Geometry", Proceedings of the 18th International Symposium on Industrial Robots, Lausanne, Swiss, pp. 91-100.
- Merlet, J.P. and Gosselin, C., 1991, "Nouvelle Architecture pour Manipulateur Parallèle a Six Degrés de Liberté", Mechanism and Machine Theory, Vol. 26, No. 1, pp. 77-90.
- Miller, K., 1995, "Experimental Verification of Modelling of Delta Robot Dynamics by Direct Application of Hamilton's Principle", Proceedings of the 1995 IEEE Int. Conf. On Robotics and Automation, Nagoya, Japan, pp. 532-537.

OBSTACLE AVOIDANCE PROCEDURES FOR MOBILE ROBOTS

Marcelo Becker

Pontifical Catholic University of Minas Gerais – Polytechnic Institute, Av. Dom José Gaspar, 500, CEP 30535-610, Belo Horizonte, MG, Brazil

e-mail: marcelo.becker@pucminas.br

Abstract. *This article relates researches on Obstacle Avoidance techniques for autonomous and assisted navigation in dynamically changing environments. Simulations, using the software MatLab™, were done for three different techniques: Inhibition Rules, Proportional Behavior Selector, and Velocity Obstacle. The results shown which technique is more indicated for real environments, where unknown obstacles may be located on a priori planned path. The results obtained were encouraged and at this moment the real implementation of all techniques are in progress.*

Keywords. *Obstacle Avoidance, Mobile Robot, Sensor, Simulation, Dynamically Changing Environments.*

1. Introduction

Autonomous navigation of mobile robots is essentially a trajectory control problem. In general terms, the control system must execute a given task, such as reaching a target, while avoiding obstacles. The intrinsic difficulties of the autonomous navigation problem have captivated many artificial intelligent researchers who have found it to be of considerable challenge (Fabro and Gomide, 1996). Navigation control of autonomous mobile robots is a research area that can be divided into two main approaches: global path planning, based on *a priori* complete information about the environment, and local path planning, based on sensory information in uncertain environment where the size, shape and location of obstacles are unknown (Beom and Cho, 1995). The last option is a common situation in real environments.

Global path planning methods can solve the path planning problems for completely known environments. But, they can not be used for navigation in complex and dynamically changing environments, where unknown obstacles may be located on *a priori* planned path. To overcome these difficulties, methods considering real-time environment information from sensors must be considered. Local path planning methods uses information of the sensors (optical sensors, ultrasonic, sonar, laser, etc.) to provide environmental information for the navigator of the robot.

The navigator is a navigation and obstacle avoidance controller that generates the steering and velocity commands for the autonomous robot. Based on sensor readings, the robot should be able to perform local path planning and to take appropriate control actions. Path planning and obstacle avoidance approaches consist, respectively, on an algorithm that plans a path that results in achieving a given goal, and another one that allows the controller to detect and, if possible, to track obstacles, generating a new path. Actually, these techniques are being applied in real applications like: robot manipulators and mobile robots avoiding moving obstacles, intelligent vehicles negotiating freeway traffic, etc.

In the case of autonomous navigation of a robot, the obstacle avoidance algorithm generates the steering and velocity commands for the robot, guiding it to its goal, taking any necessary action to avoid collisions during the path. In the case of assisted navigation, the final decision is always took by the user and the controller should aid the user, providing suggestions of alternative paths to avoid crashes with obstacles. This kind of control is found mainly in applications where the interaction between man-machine is very close and the independence feeling of the user is very important (e.g. intelligent wheelchairs, assisted navigation systems for automobiles, etc.).

But, conflicts can appear between these two algorithms, e.g.: go to goal position vs. avoid an obstacle. There are many approaches used to solve the path planning problem: Visibility Graph (Fu and Liu, 1990); Pattern Recognition (Wang and Tsai, 1991); Feedback Solutions (Feng and Krogh, 1991); Deformable Virtual Zones (Zapata *et al.*, 1994); Fuzzy Sets and Neural Networks (Beom and Cho, 1995; Baxter and Bumby, 1995; and, Fabro and Gomide, 1996); Step-by-Step Planning (Boby and Lumelsky, 1999), and many others.

In order to choose the better procedure to obtain the obstacle avoidance behavior, three different techniques were tested: inhibition rules, proportional behavior selector, and velocity obstacle. This paper focuses these techniques and the results obtained simulating them in environments with and without moving obstacles. The three techniques were developed and simulated using the software MatLab™ and, in all cases, a fuzzy controller was used to generate the path planning procedure. The first approach, inhibition rules, was based on the previous works of Baxter and Bumby (1995) and Becker (1997), the second one, proportional behavior selector, was based on the previous works of Fabro and Gomide (1996) and Becker (1999), and the third one, velocity obstacle, was based on the previous works of Fiorini and Shiller (1998), Prassler *et al.* (1999a and 1999b), and Becker *et al.* (2001).

2. Navigator

The development of the navigator is conveniently divided into two parts: navigation control, and obstacle avoidance control. An important principle is adopted at the outset: the obstacle avoidance structure should always be active nevertheless, should not affect the normal operation of the mobile robot unnecessarily. This principle requires that the

navigation and obstacle avoidance controls be a carefully integrated control structure so as to prevent conflicts between them. The navigation controller, used in all cases to plan a path that results in achieving a given goal, uses the fuzzy control approach.

2.1 Fuzzy Navigation Controller

The use of fuzzy logic in control is well documented in literature. Initially the input variables are fuzzified and linked, via a set of linguistic rules, to the fuzzy output during the composition and encoding stages. The fuzzy output is then defuzzified to produce a crisp output value (Gomide and Gudwin, 1994). The approach adopted here is to have guidance plans for the vehicle expressed as fuzzy rules and continually apply these rules so as to be able to react appropriately in face of changing conditions. By adopting such a fuzzy logic based controller, only the initial and the final positions of the vehicle are specified and a set of linguistic rules is used to guide the vehicle to its goal, taking any necessary action to avoid obstacles on its way. The controller uses max-min inference and the correlation minimum encoding technique to give a wide range of output values.

The fuzzy navigation controller is responsible for moving the mobile robot from its start position to its goal position defined in terms of (x_f, y_f, θ_f) , see Fig. (1). To achieve this control, five principles were adopted in the development of the navigation control sets (Becker, 2000):

1. If the vehicle is *long way* from the goal position, steers so as to head straight towards it, that is: $\theta_{hc} \rightarrow 0$;
2. If the vehicle is a *medium distance* from the goal attempt to move the vehicle to line up behind the goal heading towards it, that is $\theta_{ge} \rightarrow 0$;
3. If the vehicle is a *small distance* from the goal position, try to line up directly with the center-line, that is: θ_{hc} e $\theta_{ge} \rightarrow 0$;
4. If the vehicle is *not orientated to succeed* steer away from the goal for a new approach;
5. If the vehicle is *almost on top of the goal position* try to achieve the demanded final orientation, that is: θ_{oc} or θ_{hc} and $\theta_{ge} \rightarrow 0$.

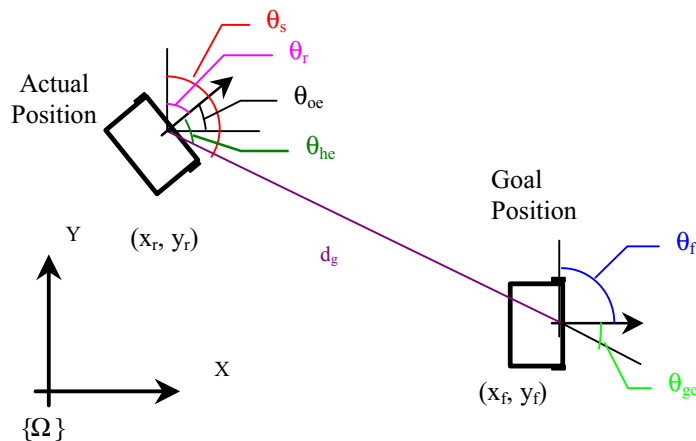


Figure 1. State variables and global coordinate system of the mobile robot.

The following equations derive the values of the state variables from the global coordinate system:

$$\text{Distance to goal: } d_g = \sqrt{(x_f - x_r)^2 + (y_f - y_r)^2} \tag{1}$$

$$\text{Goal error: } \theta_{ge} = \theta_s - \theta_f \tag{2}$$

$$\text{Heading error: } \theta_{he} = \theta_r - \theta_s \tag{3}$$

$$\text{Orientation error: } \theta_{oc} = \theta_r - \theta_f \tag{4}$$

$$\text{Where: } \theta_s = \tan^{-1} \left(\frac{y_f - y_r}{x_f - x_r} \right) \tag{5}$$

The input fuzzy sets are shown on Fig. 2 and the linguistic names for distance and error angles sets are shown on Tab. 1 and 2, respectively.

Table 1 - Linguistic names for distance.

Set Label	ZE	S	M	L
Set Name	Zero	Small	Medium	Large

Table 2 - Linguistic names for error angles.

Label	Set Name	Label	Set Name
NA	Neg. Away	PA	Pos. Away
NB	Neg. Big	PB	Pos. Big
NP	Neg. Perpendicular	PP	Pos. Perpendicular
NL	Neg. Large	PL	Pos. Large
NM	Neg. Medium	PM	Pos. Medium
NS	Neg. Small	PS	Pos. Small
NVS	Neg. Very Small	PVS	Pos. Very Small
NZ	Neg. Zero	PZ	Pos. Zero

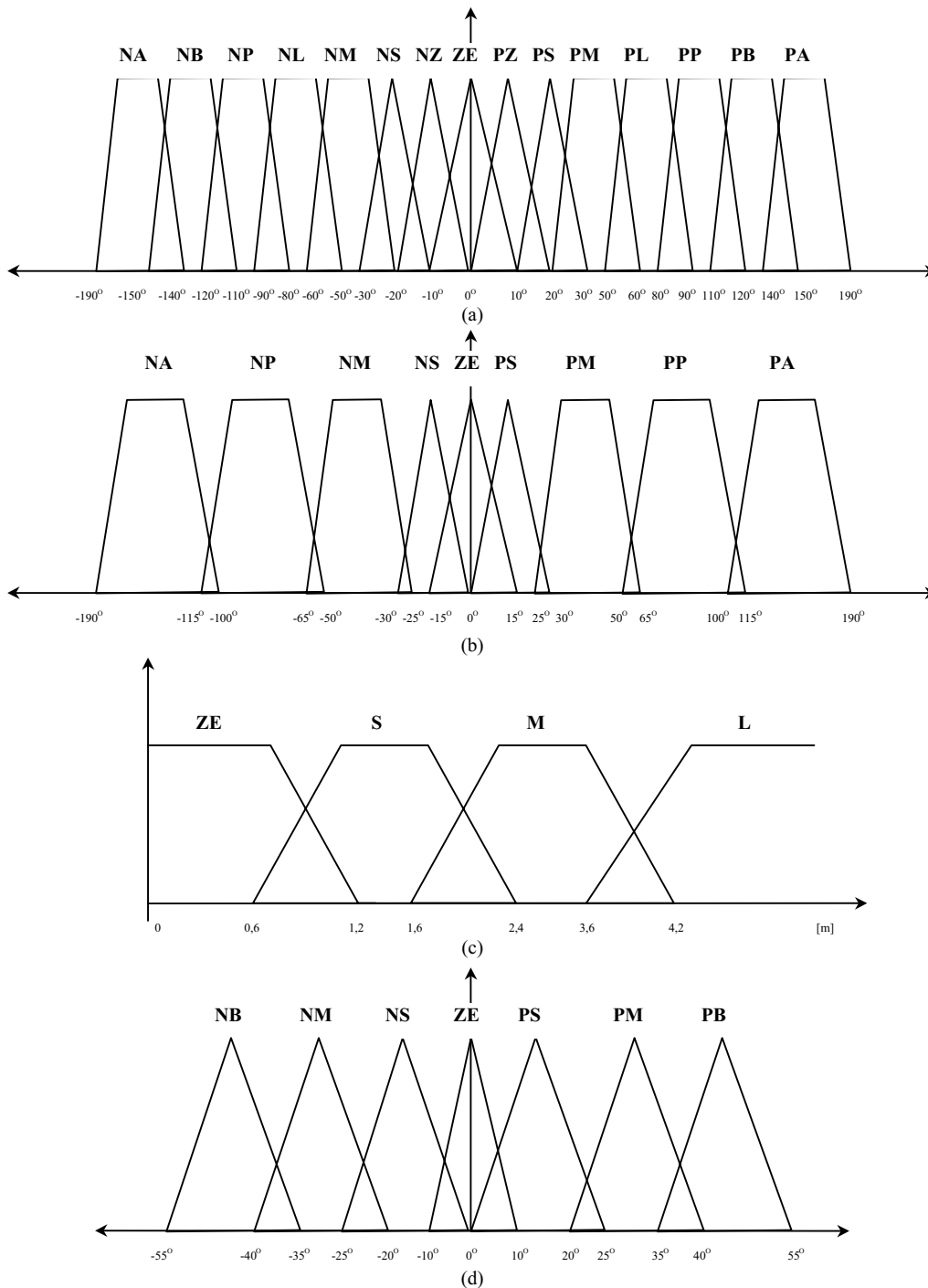


Figure 2. Input fuzzy sets for: (a) θ_{ge} ; (b) θ_{he} and θ_{oc} ; (c) d_g ; and (d) θ_{output} .

To see more clearly the interaction between the fuzzy rule sets, they are conveniently represented by a fuzzy associate memory bank matrix (Tab. 3 to 6). The number of rules is big due to the final orientation desired.
 Table 3 - Fuzzy associate memory matrix for d_g (ZE).

		θ_{ge}															
		PA	PB	PP	PL	PM	PS	PZ	ZE	NZ	NS	NM	NL	NP	NB	NA	
θ_{he}	PA	NB	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	NB	
	PP	ZE	ZE	ZE	ZE	ZE	ZE	NM	NB	NB	NB	NB	ZE	ZE	ZE		
	PM	PS	ZE	ZE	PS	PS	PS	ZE	NS	NM	NM	NB	NM	ZE	ZE	PS	
	PS	PM	ZE	ZE	PM	PS	PS	ZE	NS	ZE	NS	NM	NB	NM	ZE	PM	
	ZE	PB	ZE	PB	PM	PS	PS	ZE	ZE	ZE	NS	NS	NM	NB	ZE	PB	
	NS	PB	ZE	PB	PB	PM	PS	ZE	PS	ZE	NS	NS	NM	ZE	ZE	PB	
	NM	PB	ZE	ZE	PB	PB	PM	PM	PM	ZE	NS	NS	NS	ZE	ZE	PB	
	NP	PB	ZE	ZE	ZE	ZE	ZE	ZE	PB	ZE	ZE	ZE	ZE	ZE	ZE	PB	
	NA	PB	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	PB	

Table 4 - Fuzzy associate memory matrix for d_g (S).

		θ_{ge}															
		PA	PB	PP	PL	PM	PS	PZ	ZE	NZ	NS	NM	NL	NP	NB	NA	
θ_{he}	PA	ZE	ZE	NS	NS	NS	NM	NM	NM	NB	NB	NB	NB	NB	NB	NB	
	PP	PS	ZE	NS	NS	NS	NS	NM	NM	NB	NB	NB	NB	NB	NB	NB	
	PM	PM	PS	ZE	ZE	NS	NS	NS	NM	NM	NS	NS	NM	NB	NB	NB	
	PS	PB	PM	PM	PS	ZE	ZE	ZE	NS	NS	NS	NS	NM	NB	NB	NB	
	ZE	PB	PB	PB	PM	PM	PS	ZE	ZE	ZE	NS	NM	NM	NB	NB	NB	
	NS	PB	PB	PB	PM	PS	PS	PS	PS	ZE	ZE	ZE	NS	NM	NB	NB	
	NM	PB	PB	PB	PM	PS	PS	PM	PM	PS	PS	PS	ZE	ZE	NS	NM	
	NP	PB	PB	PB	PB	PB	PB	PB	PB	PM	PS	PS	PS	PS	PS	ZE	NS
	NA	PB	PB	PB	PB	PB	PB	PB	PB	PM	PM	PS	PS	PS	PS	ZE	ZE

Table 5 - Fuzzy associate memory matrix for d_g (M).

		θ_{ge}															
		PA	PB	PP	PL	PM	PS	PZ	ZE	NZ	NS	NM	NL	NP	NB	NA	
θ_{he}	PA	ZE	ZE	ZE	NS	NS	NS	NS	NB	NB	NB	NB	NB	NB	NB	NB	
	PP	PS	PS	ZE	ZE	ZE	NS	NS	NM	NM	NM	NM	NM	NB	NB	NB	
	PM	PM	PM	ZE	ZE	ZE	ZE	PS	NS	NS	NM	NM	NM	NB	NB	NB	
	PS	PB	PB	PM	PB	ZE	ZE	ZE	NS	NS	NS	NM	NM	NB	NB	NB	
	ZE	PB	PM	PM	PS	PS	ZE	ZE	ZE	ZE	ZE	ZE	NS	NM	NM	NB	
	NS	PB	PB	PB	PM	PS	PS	PS	ZE	ZE	ZE	ZE	NS	NM	NB	NB	
	NM	PB	PB	PB	PM	PM	PS	PS	PS	PS	PS	PS	ZE	ZE	ZE	NM	NM
	NP	PB	PB	PB	PM	PM	PS	PM	PS	PS	PS	PS	ZE	ZE	ZE	NM	NS
	NA	PB	PB	PB	PB	PB	PB	PB	PS	PS	PS	PS	PS	PS	PS	ZE	ZE

Table 6 - Fuzzy associate memory matrix for d_g (L).

		θ_{ge}															
		PA	PB	PP	PL	PM	PS	PZ	ZE	NZ	NS	NM	NL	NP	NB	NA	
θ_{he}	PA	NM	NM	NM	NM	NM	NM	NM	NB	NB	NB	NB	NB	NB	NB	NB	
	PP	NM	NM	NM	NM	NM	NM	NM	NM	NB	NB	NB	NB	NB	NB	NB	
	PM	NS	NS	NS	NS	NS	NS	NS	NS	NS	NS	NM	NM	NM	NM	NM	
	PS	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	NS	NM	NM	NM	NM	
	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	
	NS	PM	PM	PS	PS	PS	PS	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE	
	NM	PM	PM	PM	PM	PM	PM	PM	PS	PS	PS	PS	PS	PS	PS	PS	
	NP	PB	PB	PB	PB	PB	PB	PB	PM	PM	PM	PM	PM	PM	PM	PM	
	NA	PB	PB	PB	PB	PB	PB	PB	PB	PM	PM	PM	PM	PM	PM	PM	

A cruiser velocity is used during the path and a fuzzy velocity controller is used only in the final approach. In other words, the controller slows the vehicle down only as it approaches the goal position: “IF distance to goal position is SMALL then velocity is SLOW”.

2.2. Obstacle Avoidance Controller

In order to choose the better procedure to obtain the obstacle avoidance behavior, three different techniques were used: inhibition rules, proportional behavior selector, and velocity obstacle. They will be addressed in the next items.

2.2.1. Inhibition Rules

In this case, the obstacle avoidance procedure is added to the navigator by a set of inhibition rules of the form: “*IF obstacle is AHEAD LEFT then DONT steer ZERO, NEGATIVE SMALL, etc...*”. This type of rule set reduces the activation of an output set rather than increasing it, as would be the case if a positive avoidance rule were used (Baxter and Bumpy, 1995). The aim of the inhibition rules is to produce a mask vector which, when multiplied by the fuzzy vector, gives an overall output vector. The mask vector contains values indicating how acceptable each possible steering angle set is in view of the proximity of obstacles.

The input sets for the obstacle avoidance controller are linked to particular areas of the certainty grid relative to the vehicle. The area covered by a set is the area that the vehicle would occupy if a particular output set was activated. Each rule links one or more of these input sets to a degree of inhibition of the corresponding output steering angle set which ranges from 0, total inhibition, to 1, no inhibition. Sets covering areas close to the vehicle have the greatest inhibition effect and prevent any activation of their linked output set, while more distant sets merely reduce the activation (Fig. 3).

Table 7 - Avoidance rules set (Becker,2000).

Rule	Obstacle Position	Inhibition Rules
P1	$-100^\circ \leq \theta_{\text{obstacle}} \leq -35^\circ$	PB, PM, and PS
P2	$-40^\circ \leq \theta_{\text{obstacle}} \leq -20^\circ$	PB, PM, and PS
P3	$-25^\circ \leq \theta_{\text{obstacle}} \leq -5^\circ$	PB, PM, PS, ZE, and NS
P4	$-10^\circ \leq \theta_{\text{obstacle}} \leq 10^\circ$	PM, PS, ZE, NS and NM
P5	$5^\circ \leq \theta_{\text{obstacle}} \leq 25^\circ$	PS, ZE, NS, NM and NB
P6	$20^\circ \leq \theta_{\text{obstacle}} \leq 40^\circ$	NS, NM and NB
P7	$35^\circ \leq \theta_{\text{obstacle}} \leq 100^\circ$	NS, NM and NB

Table 8 - Degree of inhibition vs. distance to obstacle.

Distance [m]	$d \geq 4$	$4 < d \leq 3$	$3 < d \leq 2$	$2 > d$
inhibition	1,0	0,8	0,5	0

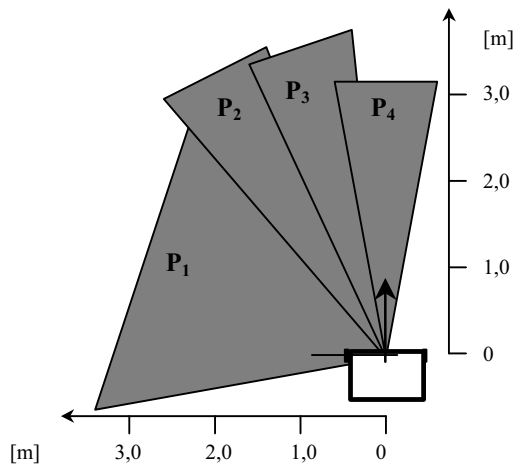


Figure 3 - Avoidance sets areas.

But, there are some problems with using inhibition rules and center of gravity defuzzification: the resultant output would still be to steer straight on as compromise between going left or right round the obstacle. To guarantee that the defuzzified output will not fall within one of inhibited sets, sliding window defuzzification is used. To provide alternative suggestions, in the situation when all the output sets activated by the navigation controller are inhibited by the obstacle avoidance rules, a technique called rule spreading is used (Baxter and Bumpy, 1995 and Becker, 1997). The rule spreading operates by using a bell function to add activation values to all the entries in the fuzzy fit vector, depending on their distance from the activated sets. The activation of the r^{th} set is multiplied by $e^{-k(i-r)^2}$ and added to the activation of the i^{th} set (see Fig. 4a). The sliding window defuzzification technique involves locating the set with the maximum activation and then applying fuzzy centroid defuzzification to this set and its immediate neighbors, defined by the window width (see Fig. 4b).

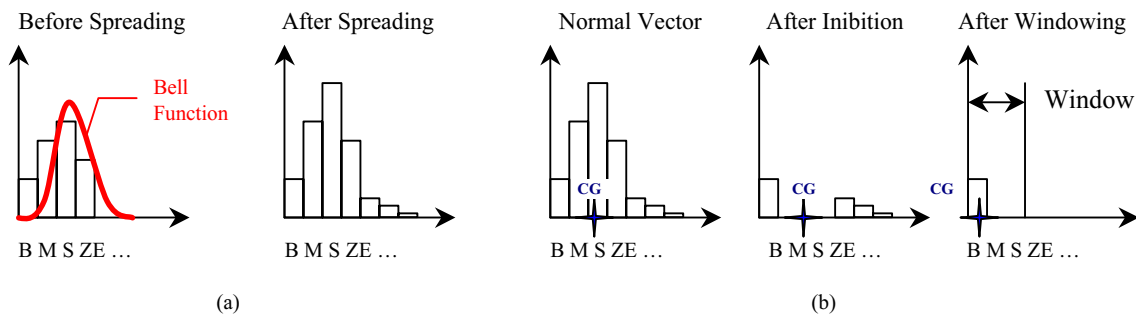


Figure 4. Examples of (a) rule spreading applied to a fuzzy fit vector and (b) slide windowing application.

2.2.2. Proportional Behavior Selector

In this case, a new fuzzy rule set is used to make the obstacle avoidance procedure. The proportional behavior selector takes its decision based on measurements of proximity of the goal position and the obstacle as follows: when an obstacle brings near, the navigation controller output is inhibited and the collision controller output is stimulated: the selector must avoid obstacles, independently of goal position. This situation occurs when the vehicle is near to the obstacles, but far from the target. When the vehicle is closer to the target than to any obstacle, the selector must go towards the target, independently of obstacles. And, when the target position is very close to an obstacle, the selector must find a compromise solution between the actions to be taken (Fabro and Gomide, 1996). The formulae to compute the selector activation are as follows:

$$\text{Collision: } a_c = d_{obst} \cdot e^{0.1(d_{obst}-15)} \tag{5}$$

$$\text{Navigation: } a_n = d_f \cdot e^{0.1(d_f-15)} \tag{6}$$

New linguistic terms are used in the fuzzy collision controller, for distance to obstacle and obstacle direction.

Table 9. Linguistic names for distance to obstacle.

Set Label	C	S	M	L
Set Name	Collision	Small	Medium	Large

Table 10. Linguistic names for obstacle direction.

Label	Set Name	Label	Set Name
NB	Neg. Big	PB	Pos. Big
NM	Neg. Medium	PM	Pos. Medium
NS	Neg. Small	PS	Pos. Small
NZ	Neg. Zero	PZ	Pos. Zero

The input fuzzy sets are as follows:

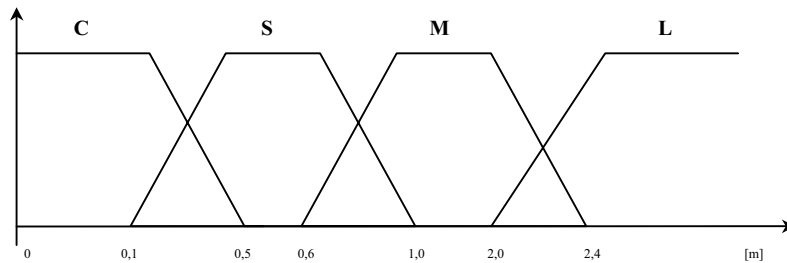


Figure 6. Fuzzy sets for distance to obstacle.

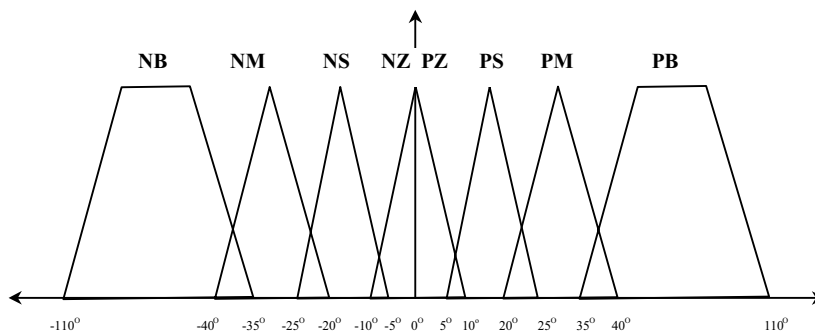


Figure 7. Fuzzy sets for obstacle direction ($\theta_{obstacle}$).

The fuzzy associate memory bank matrix for the collision controller is shown on Tab. 11. Let u_c and u_n be the defuzzified control actions from the fuzzy collision avoidance controller and the fuzzy navigator controller, respectively. Thus the output control action u to be adopted is determined by Eq. (7):

$$u = u_c \frac{a_c}{(a_c + a_n)} + u_n \frac{a_n}{(a_c + a_n)} \tag{7}$$

Table 11. Fuzzy matrix for collision controller.

		θ_{obstacle}							
		NB	NM	NS	NZ	PZ	PS	PM	PB
Distance to Obstacle	C	PS	PM	PB	PB	NB	NB	NM	NS
	S	PS	PS	PM	PB	NB	NM	NS	NS
	M	PS	PS	PS	PS	NS	NS	NS	NS
	L	ZE	ZE	ZE	ZE	ZE	ZE	ZE	ZE

2.2.3. Velocity Obstacle

This technique consists on three procedures, the first one to detect and classify the obstacles, the second one to track the moving obstacles, and the last one to generate the evasive maneuverings.

Obstacle detection: Based on the scan data, the robot can build a model of the environment and use this model to detect obstacles, motion, and to track the motion. However, how to represent and use these data in a real-time application? An efficient scheme for mapping scan data is the Occupancy Grid Representation (Elfes, 1989). In this approach, all the scan data are represented on a 2-D rectangular grid, where each grid element (cell) describes a small region of the real world. But, due to the necessity to completely initialize the grid set and each cell to some default value at step time t , this representation is too expensive in computer time (see Fig. 8a). All cells must be classified as occupied, free or undefined (cells behind the occupied cells). To avoid this problem and save computer time, a variation of this representation, called Time Stamp Map (Fiorini and Shiller, 1998), is used. In this case, only the occupied cells observed at each time step t are mapped, all the other cells in this grid remain untouched (see Fig. 8b). This procedure allows the implementation in real-time, avoiding spend time mapping free space.

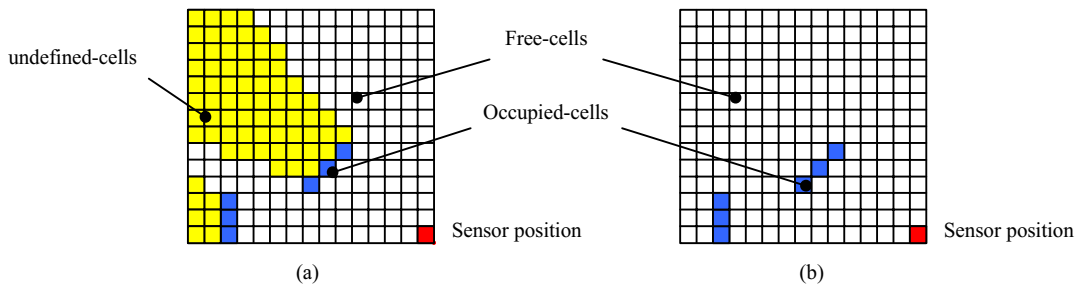


Figure 8. Comparison between (a) Occupancy Grid Representation and (b) Time Stamp Map.

Motion Estimation: The easiest way to identify changes in the environment is to consider a sequence of the Time Stamp Maps and to investigate where the steps of this sequence differ. A discrepancy between two subsequent steps is a strong indication of a potential modification in the environment. To estimate the velocity of the object (as a vector, i.e. with modulus and direction), the occupied cells are united into sets and the “centers of mass” (CG) of these sets are considered (see Fig. 9). A change in the CG indicates a possible moving obstacle.

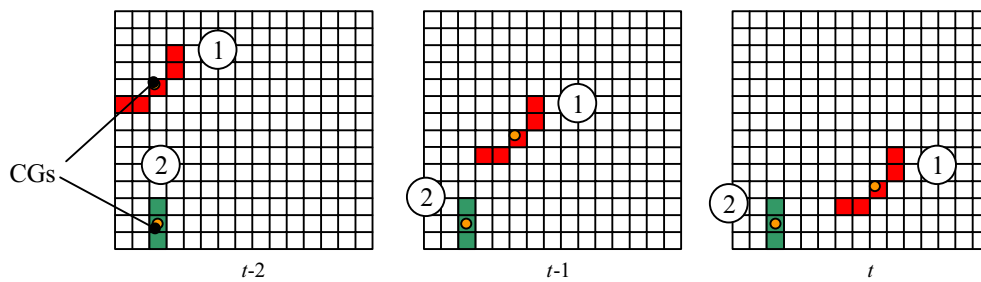


Figure 9. Sequence of Time Stamp Maps used to indicate possible moving objects. Observe the sets N. 1 and 2.

Evasive Maneuvering: The basic idea of this procedure is to obtain a “collision cone” specific to a particular pair of robot / obstacle (Fig. 10a). The robot and the obstacles are modeled as circles (with radius r_r and r_o , respectively). Initially the robot is considered as a particle and its radius is added to the obstacle radius (i.e. $R = r_o + r_r$). Then, two tangent lines are drawn from the robot position to the circle with radius R . The relative velocity vector between the robot and the obstacle is calculated ($V_{r,o}$). If this vector is positioned inside the collision cone, a collision between the robot and the obstacle is imminent. Nevertheless, when the robot is in an environment with many moving obstacles, it is easier to use an equivalent condition on the absolute velocity of the robot (“velocity obstacle”). It means, the collision cone is translated to the end of the obstacle velocity vector (V_o), see Fig. 10b. The robot velocity vectors (V_r) inside of the velocity obstacle would cause a collision between the robot and the obstacle, vectors outside of velocity obstacle would avoid collisions, and vectors on the boundaries of the velocity obstacle would result in the robot grazing the obstacle. Combining the results for each obstacle and the output of a navigation controller, it is possible to choose the

better control output, which results in the robot going to the goal position and avoiding the obstacles. Note on Fig. 10b that for the actual velocity of the robot (vector velocity inside of the velocity obstacle area) the collision is imminent. Nevertheless, any velocity orientation that provides a position outside of the filled area would avoid the collision.

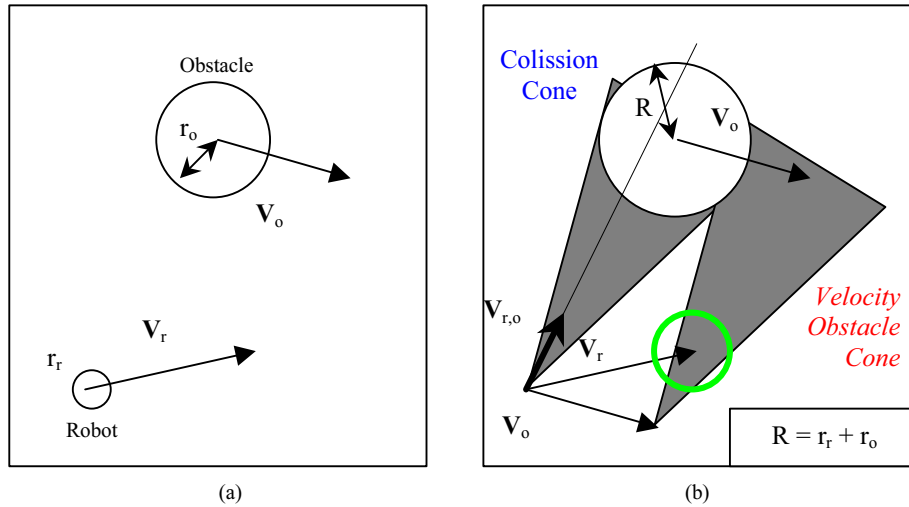


Figure 10. Collision and velocity obstacle cones.

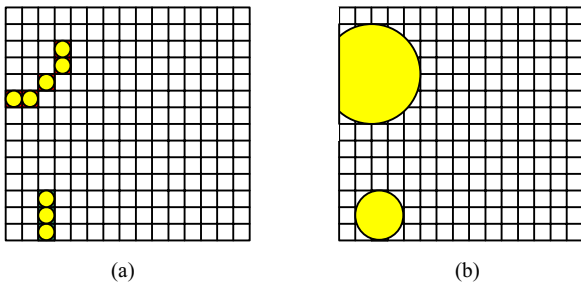


Figure 11. Comparison between the application of the evasive maneuverings using (a) isolated cells and (b) sets.

Observe that to apply evasive maneuverings using the velocity obstacle approach, the obstacles seen by the sensor are modeled as individual cells and not as sets of occupied cells. This procedure is adopted due to the free space close to the obstacle that can be used to avoid the collision grazing the obstacle (compare Fig. 11a and b). There are two options to avoid a collision: the first one is to change the orientation of the velocity vector, steering the robot and, the second one, to change the modulus of the velocity vector, accelerating the robot (positive or negative acceleration). As one of the principles adopted for the controller behavior is to use a cruiser velocity during the path, the modulus of the velocity vector is changed only if the maximum steering angle is exceeded.

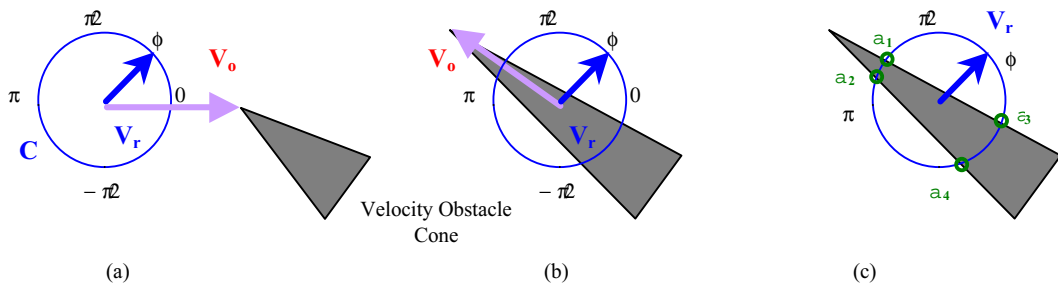


Figure 12. Application of the velocity obstacle using a cruiser velocity: (a) no intersection between C and velocity obstacle cone, (b) intersection between C and velocity obstacle cone, and (c) sets of dangerous angles.

Based on the above premises, a circle (C) with radius equal to the desired cruiser velocity is used to represent the set of possible orientation angles of the robot. Then, for each obstacle observed in the neighborhood, the velocity obstacle and the intersections between it and the circle are calculated. The results obtained for all obstacles are combined and result in the sets of dangerous orientation angles for the robot (on Fig. 12c, the angles between α_1 and α_2 , α_3 and α_4). In other words, if the orientation angle of the robot (ϕ) belongs to one of these sets, a collision between the robot and the obstacle will occur. Nevertheless, if the maximum steering angle is exceeded, the algorithm searches a new value for the modulus of the velocity vector that avoids the crash, increases the C circle radius, and does not exceed the maximum steering angle.

3. Results and Conclusions

To check the navigator behavior in environments with obstacles the software MatLab™ was used. Initially the three techniques were tested in a static environment with 10 fixed obstacles. Then, a dynamic environment with 5 moving

and 5 fixed obstacles was used. In all cases, the fuzzy navigation controller and a cruise velocity were used. Only the most representative results are shown. Figure 13 shows the behavior of the navigator with the proportional behavior selector. Even for static environments, this technique produces alternated output angles (Fig. 13b), causing a undesired zigzag movement of the robot. Figure 14 shows the simulated path for the robot with inhibition rules and the detail of the moment when the collision between the robot and an obstacle occurred. Even using the windowing and rule spreading techniques, this approach was not able to deal with several moving obstacles around the robot.

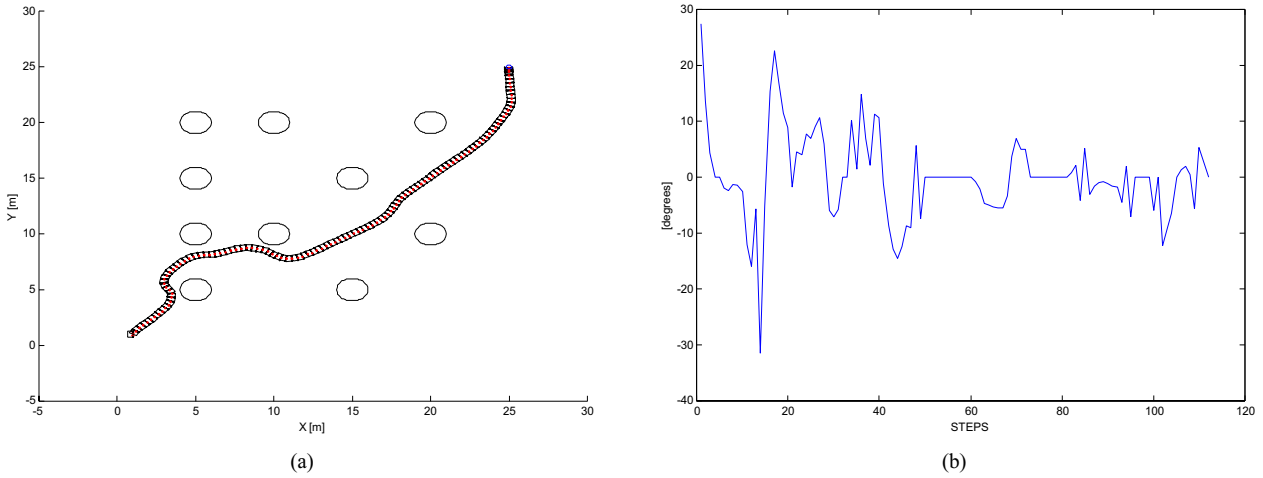


Figure 13. Simulated path of the robot (a) and output angle (b) using the navigator with proportional behavior selector.

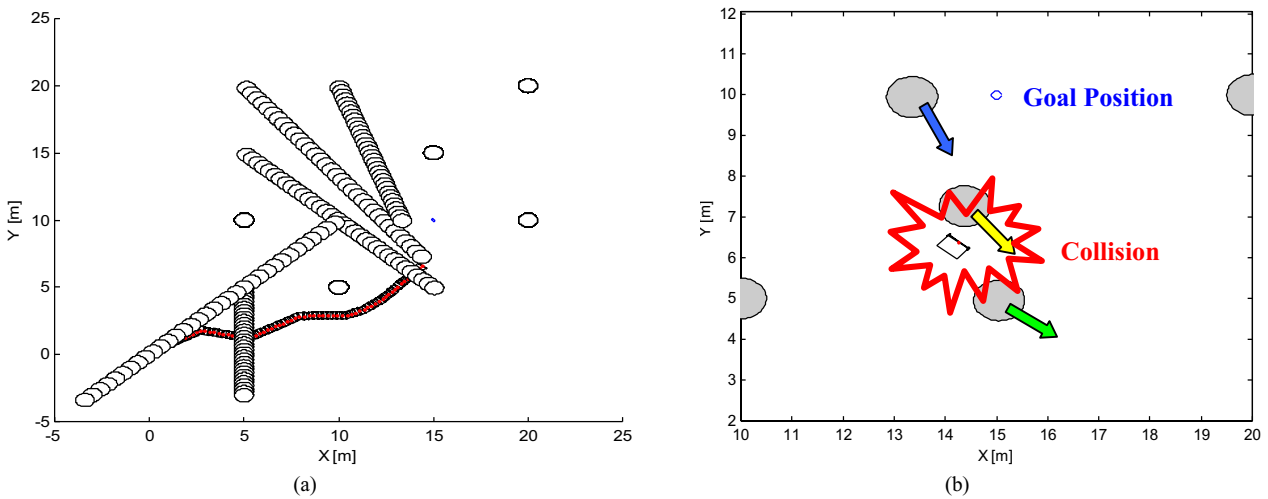


Figure 14. Simulated path of the robot with inhibition rules (a) and detail of the collision between the robot and an obstacle (b). The simulation was stopped on step 67, the moment of the collision. Dynamic environment.

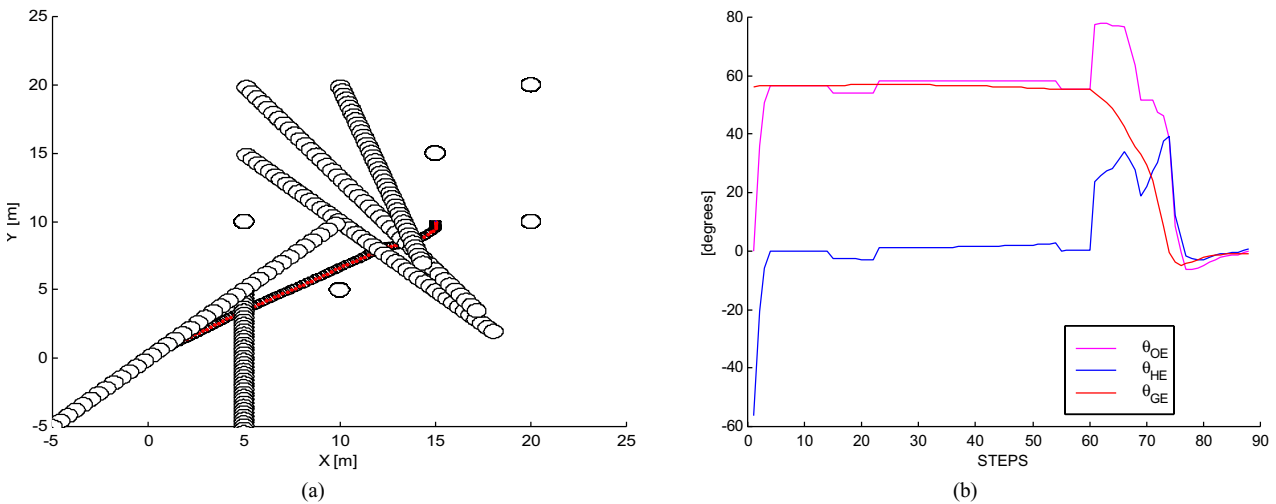


Figure 15. Simulated path of the robot with velocity obstacle (a) and error angles of the fuzzy navigation controller (b). Dynamic environment.

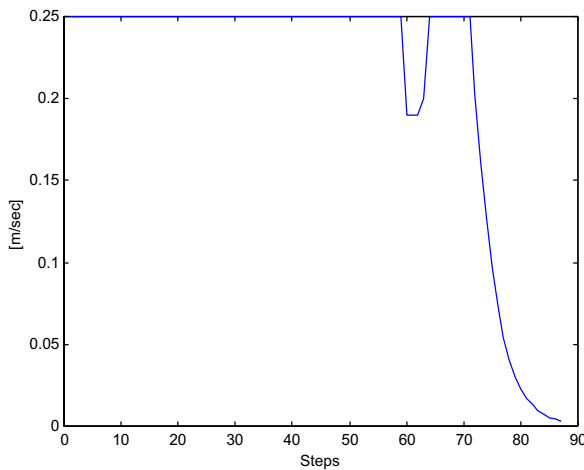


Figure 16. Simulated velocity of the robot with velocity obstacle during the path. Velocity obstacle approach.

Figures 15 and 16 show the simulated path for the robot with the velocity obstacle approach, the detail of the error angles used by the fuzzy navigation controller to generate the output angle, and the velocity of the robot during the path. Observe that the velocity of the robot was decreased when the maximum steering angle was exceeded (Fig. 16).

The two firsts approaches (proportional behavior selector and inhibition rules) were not able to avoid obstacles in dynamically changing environments. Nevertheless, the results obtained using the velocity obstacle technique were encouraged.

The capacity of the controller to predict and determine collision courses with several moving obstacles and compute an avoidance maneuver, which is close to its original heading as possible, allows the controller to avoid undesired and unnecessary accelerations and steering angles. The robot moves close to the obstacles, grazing them. Only when the maximum steering angle is exceeded, the controller changes the cruise velocity to compute a new output angle.

At this moment, the real implementation of the three procedures is in progress.

4. Acknowledgments

The author wish to thank the FAPEMIG for its financial support and all the staff of the Institute of Robotics of the Swiss Federal Institute of Technology (IfR-ETHZ), for their collaboration.

5. References

- Baxter, J.W. and Bumby, J.R., 1995, "Fuzzy control of a robotic vehicle", Proc. of Instn. Mech. Engrs. - Part I: Journal of Systems and Control Engineering, Vol. 209, pp. 79 - 91.
- Becker, M., 1997, "Robôs de locomoção: formas construtivas, dirigibilidade e controle", M.Sc. Dissertation (in Portuguese) School of Mechanical Engineering / FEM, State University of Campinas / Unicamp, Brazil.
- Becker, M., 1999, "Simulation of a fuzzy controller for autonomous vehicles in complex and dynamic environments", Proc. of COBEM99 – XV Brazilian Congress on Mechanical Engineering, Águas de Lindóia – SP, Brazil, Nov 22 to 26, 1999, CD-ROM.
- Becker, M., 2000, "Aplicação de técnicas de controle em cadeiras de rodas inteligentes", Ph.D. Thesis (in Portuguese) School of Mechanical Engineering / FEM, State University of Campinas / Unicamp, Brazil.
- Becker, M., Wullschleger, F. and Schweitzer, G., 2001, "Tracking moving objects in dynamically changing environments", Proc. of IX Diname – Dymanical Problems on Dymanics, pp .249 – 254.
- Beom, H.R. and Cho, H.S., 1995, "A sensor-based navigation for a mobile robot using fuzzy logic and reinforcement learning", IEEE Trans. on Syst., Man, and Cyber., Vol. 25, n. 3, pp. 464 - 477.
- Boby, S. and Lumelsky, V., 1999, "Control of dynamics and sensor based motion planning for a differential drive robot", Proc. of the 1999 IEEE International Conference on Robotics and Automation, May 1999, Detroit – MI - USA, pp. 157 – 162.
- Elfes, A., 1989, "Occupancy grids: a probabilistic framework for robot perception and navigation", Ph.D. Thesis, Electrical and Computer Engineering Department / Robotics Institute, Carnegie-Mellon University, USA.
- Fabro, J. A. and Gomide, F. A. C. , 1996, "Self-organising neuro-fuzzy control of complex systems", Applied Mathematics and Computer Sciences, vol. 6, n. 3, pp. 581 – 594.
- Feng, D. and Krogh, B.H., 1991, "Dynamic steering control of conventionally steered mobile robots", Journal of Robotic Systems, Vol. 8, n. 5, pp. 699 - 721.
- Fiorini, P. and Shiller, Z., 1998, "Motion planning in dynamic environments using velocity obstacles", International Journal of Robotics Research, Vol. 17, pp. 760 - 772.
- Fu, L.-C. and Liu, D.-Y., 1990, "An efficient algorithm for finding a collision-free path among polyhedral obstacles", Journal of Robotic Systems, Vol. 7, n. 1, pp. 129 - 137.
- Gomide, F. A. C. e Gudwin, R. R., 1994, *Modelagem, Controle, Sistemas e Lógica Fuzzy*, Controle & Automação, Vol. 4, n. 3, pp. 97 – 115.
- Prassler, E. et al., 1999b, "Tracking people in a railway station during rush hour", Proc. of 1st Int. Conf. on Computer Vision Systems (ICVS99), Gran Canaria, Spain, pp. 162 – 179.
- Prassler, E., et al., 1999a, "Navigating a robotic wheelchair in a railway station during rush hour", International Journal of Robotics Research, Vol. 18, pp. 711 - 727.
- Wang, L.-L. and Tsai, W.-H., 1991, "Collision avoidance by a modified least-mean-square error classification scheme for indoor autonomous land vehicle navigation", Journal of Robotic Systems, Vol. 8, n. 5, pp. 677 - 698.
- Zapata, R., et al., 1994, "Reactive behaviors of fast mobile robots", J. of Robotic Syst., vol. 11, n. 1, pp. 13 - 20.

DESIGN AND IMPLEMENTATION OF A HAPTIC INTERFACE FOR TELEPRESENCE

Denilson Laudares Rodrigues

GEAR – IPUC – Pontifical Catholic University, Belo Horizonte, MG, Brazil, 30535-610
denilsonlr@pucminas.br

Mario F. M. Campos

VERLab – DCC – Federal University of Minas Gerais, Belo Horizonte, MG, Brazil, 31270-010
mario@dcc.ufmg.br

Vijay Kumar

GRASP Lab – University of Pennsylvania, Philadelphia, PA, USA, 19104-6228
kumar@grasp.cis.upenn.edu

Abstract.

Recently an important new area has emerged from robotics and computer graphics working together to create powerful immersive interfaces called haptic interfaces. Haptic interfaces are devices that provide kinesthetic and force feedback information to the user through physical interaction. While using such interface, a person has the sensation of physically interacting with an environment, which can mirror a remote site, or even be a world completely synthesized by a computer. We present here a haptic interface device based on a commercial robot is designed and implemented. The system is modeled and experimentally identified. Several issues involved such as torque ripple, gravity and friction compensation are presented and effectively accomplished. Finally, the haptic interface is tested and evaluated in an actual telepresence system.

Keywords. robotics, dynamics, haptic interface, impedance control, human-computer interaction

1. Introduction

Largely driven by the needs of general positioning machines, robots have been widely employed in a large number of industrial applications. Yet they have crossed those borders and in recent years service robots have been gradually making their way into hospitals, offices, homes and construction sites (Sheridan, 1992). Applications in space exploration, defense, agriculture and healthcare are also typical targets of current research (Burdea, 1996). Recently an important new area has emerged from robotics and computer graphics working together to create powerful immersive interfaces called *haptic¹ interfaces*.

Haptic interfaces are devices that provide kinesthetic and force feedback information to the user through physical interaction. While using such interface, a person has the sensation of physically interacting with an environment, which can mirror a remote site, or even be a world completely synthesized by a computer. In the latter case, the term *virtual reality* (VR) is often applied. Virtual reality is a high-end human-computer interface allowing user interaction with simulated environments in real time and through multiple sensorial channels (Burdea, 1999). The oldest and most famous virtual reality systems are the flight simulators.

In virtual reality applications, haptic interfaces have two basic functions: measuring positions/forces (and their time derivatives) at the user's interaction port and displaying forces/positions back to the user under the control of the computer running the simulator. Haptic feedback is essential in many important virtual reality and telepresence applications. As one of the VR input/output devices, haptic interfaces have enabled highly realistic applications in areas as mechanical design, entertainment, teleoperation and medical training.

While there is not a general agreement on a single definition, *telepresence* can be described as the maximization of feeling physically present at the remote site with the help of visual, mechanical, and auditory stimuli. A sense of presence can dramatically increase the performance of telerobotics. When the remote site happens to be virtual then we speak about virtual reality or virtual presence which might combine seeing, touch, and hearing as well. Approaches to teleoperation and remote control have a long history in automatic-control and robotics. Telerobotics or remote supervisory control, are the names usually given to the area of research dealing with this class of problems (Sheridan, 1992). It has been widely used in such applications as entertainment, military, space, airborne, underwater, medical, and operations in hazardous environments. After industrial applications, telerobotics is the next largest application of robots.

Central to the subject matter of this paper is the issue of designing haptic interfaces for telepresence. Many of the previous works described in the literature focuses on simulating static forces such as those encountered when interacting with a stiff object, as present by Ansar (2001). Other implementations employ custom-made devices that are not readily available, which makes it difficult to be replicated and tested elsewhere.

A 2 DOF (degree-of-freedom) haptic interface device has been described in (Kazerooni, 1994). This device was built and programmed to impose arbitrary trajectory-dependent forces on the human arm. In the experiments described, a subject maneuvers the device in a vertical two-dimensional plane by a handle attached at the device's endpoint. A primary closed-loop position controller, which was called the primary stabilizing controller (PSC), was used. The

¹ The term "haptic", comes from the Greek *haptesthai* relating to or based on the sense of touch or tactile feedback (Webster, 1985).

design of the PSC enabled the designer to deal with side-effects caused by uncertainties of the device without worrying with the complex dynamics of the human operator. A variety of controllers may be used to stabilize the device in the presence of uncertainties and nonlinearities. The design of PSC must let the designer deal with effect of device uncertainties without concern for the dynamics of the human operator.

Adams (2000) also addressed the stability problem of a haptic display interface. Their haptic device, called Excalibur, a 3 DOF Cartesian robot, was designed to behave as a haptic interface. They approached the problem of stable haptic simulation as a two-port framework. The two-port approach allows for rigorous stability and performance analysis for a very general class of haptic displays. Sirouspour et al (2000) described the haptic rendering of virtual environments by drawing upon concepts developed in the area of teleoperation. They present the dynamic model of the particular haptic device and its control scheme. They also match the impedance control with the teleoperation approach using four types of data transmission modes between master and slave, sending forces and positions in both directions to guarantee stability.

Gillespie (1994) presented a haptic device to simulate medical procedures that involve needle insertion into human tissue. To perform this delicate manual procedure, an anesthesiologist must carefully guide a needle through various layers of tissue using only haptic cues for guidance. A simplifying aspect for the simulator design, all motions and forces involved in the task occur along a fixed line once insertion begins. A perceptual model was built based on current training techniques for the procedure provided by a skilled instructor. The control of haptic display follows the response measured when the instructor was performing the task.

Medical robotics and computer aided surgery in general, and robotic telesurgery in particular, are promising application of haptic interfaces. In (Cavusoglu, 1999) a haptic device with 4 DOF to be used in medical applications, namely as a laparoscopic telesurgical workstation is presented. It is important to note that the input is accomplished through a master device, whereas a slave manipulator is used to perform the task. They used PID controllers for each DOF and a force sensor in both robots (master and slave). In this approach the slave robot task was to follow the movements commanded by the master robot. At the endpoint of slave robot, force and torque are measured and sent back to the master arm, which was used to track the response of the system.

Even though commercial devices specifically designed as haptic interfaces are already available, results have been reported from the use of commercial robots, such as the PUMA and the WAM as a haptic input device. In (Yokokohji, 1999) is shown a haptic interface using a PUMA robot. The use of commercial robots presents a few drawbacks such as singularities and high mass/inertia. In general, the mass to be simulated must be larger than a value that depends on the mass properties of the haptic device. Otherwise the haptic device becomes oscillatory and unstable. For interfaces using industrial robots, their mass properties have larger values, because an industrial robot is built to be stiff. In this case the minimum value of mass to be simulated is usually large. Using the PUMA as a haptic interface presents another problem since it is very likely that it will reach a "wrist-singular posture". This singularity shows up within the working volume when the fifth joint stretches out.

Clearly many of these problems would be circumvented with custom-built devices. However, there are many issues that still need to be addressed concerning haptic interfaces, which would not be possible to stimulate with ready made devices. Since the goal of our work is to test both architectural and control issues, we opted to implement a haptic interface with a commercial direct driven robot called WAM - Whole Arm Manipulator. The WAM robot presents several interesting features such as a light mechanical structure and a suitable work volume.

The remainder of this paper is organized as follows: the next section presents and analyses the key features of an ideal haptic interface, followed by section 3, where the important factors necessary to use the WAM as a haptic interface are addressed. Section 4 describes the results, where the interface control and its implementation as an impedance display are detailed and evaluated experimentally, followed by conclusions and future directions for this research.

2. Haptic Interface Design

The design of a haptic interface is a complex task, since it involves mechanical design, control architecture, dynamic interaction and visual display. Nevertheless, there are a few important issues that must be considered from the outset:

1. *Backdriveability*. Looking the work done so far, the consensus seems to be that the "ideal" haptic interface device is one that is highly backdriveable. This is an important characteristic because it is necessary a maximum reality during the haptic task. The user will inform movement holding the end-effector of the robot and moving the end-effect around some region. In this case the robot must be backdriveable.
2. *Low Inertia*. Another important characteristics to haptic device is low inertia. This because the user must feel his hand free in the space when he is moving the haptic device in free space. If the haptic device has a heavy mechanical structure, the user wont be able to move his hand in free space without feeling some reaction (inertia).
3. *Zero Backlash*. As a robot, we do not want backlash in the mechanical structure. The shaft of the motors and mobile parts must be rigidly attached.
4. *Low friction*. Friction is another mechanical phenomena that we have to cancel or compensate. As a non-linear effect, the friction will be a noise during the task execution. In order to simulate free space, we have to take care about these last characteristics.

5. *Large force range.* During a haptic task, some time will be necessary to simulate static objects, like a wall for example. In this case a large force range will be an important characteristic.
6. *Working volume.* To use this haptic device for different applications a suitable working volume will be necessary.

Unfortunately simultaneously attaining all the features listed above can be conflicting. For instance, large force range and working volume require devices that are physically large, which normally implies larger inertia, coupled friction and low bandwidth. In order to reach a good trade off among these criteria it is necessary to focus on the application (Clover, 1999). If the goal is to simulate gross dynamic forces and motion when manipulating a substantial mass, force range and working volume are key considerations, and force bandwidth becomes a secondary consideration. Otherwise, if it is desired to simulate fine motion of small mass and feel high frequency vibrations, then force bandwidth becomes a dominant criteria.

The *control architecture* of haptic interfaces is another important issue in haptic interface design. Haptic feedback control systems are based on impedance/admittance theory (Hogan, 1985) and its application to robotics. Approaches using stiffness control is a simple solution in some applications (Asada, 1985). Typical systems measure the position of the interaction port (the user-device contact point where energy is exchanged) and then derive control signals based primarily on a desired force output when the user interacts with a static virtual object such as a stiff wall. This is referred to as an *impedance control* paradigm. Another approach is to measure the interaction forces and torques at the interaction port and then derive control signals based on a desired motion output. This scheme is referred to as an *admittance control* paradigm. As a general guideline, when the haptic device is light and backdriveable we should consider impedance control, otherwise, admittance control would be a better option for a robot that has heavy structure elements. Modeling the device dynamics can be accomplished with Newton-Euler, Lagrange or Kane

A key point in haptic design is the repertoire of virtual or remote environments simulators is the modeling and simulation tools upon which it is based (Gillespie, 1997). Most virtual environments are limited to static objects or systems of dynamics bodies whose motions are restricted to a plane and whose interconnections are modeled by relatively soft springs.

The visual display is very important to make the telepresence experience feel as closely related with the real task as possible. Head mounted displays (HMD) are often used to achieve this goal. Planar displays are also used in some cases (Yokokohji, 1999). The insertion of the image of the user's hand at the display, make a big difference during a simulation task, which can be accomplished with standard computer vision techniques. The control system cannot afford to spend too much time in image processing and synthesis, because there is a large computational load from other modules of the system such as collision detection, dynamic interaction and visual display. Previous works used different computers for each task, improving the result. In this last case attention must be paid to communication delays, which would seriously compromise the overall performance of the system.

3. WAM as a Haptic Interface

Due to several features of its design, we chose the Whole Arm Manipulator WAM - Fig (1) - as our test bed of a haptic interface device. The WAM robot we used is an anthropomorphic arm with four degrees of freedom. The structural elements are light and the robot employs a stiff cable transmission and brushless motors to permit to be intrinsically force controllable. Cable drive minimizes the effect of friction and backlash. The WAM has a large workspace (3.5m^3), 5m/s peak velocity and 30m/s^2 peak acceleration (both in Cartesian space) and 4.5Kg payload, being able to apply enough force and torque to simulate with different virtual and/or remote environments. To control the WAM robot, a dSPACE DS1103 controller board which has an onboard 333MHz PowerPC processor (PPC604e) for fast floating point computations. A TMS320240 Digital Signal Processor is also available with ADC, DAC and incremental encoder interfaces. Interaction forces and torques with the user are measured by an ATI Gamma F/T force sensor mounted on the end-effector of the last link of the WAM. System dynamics needs to be modeled before the WAM can be used as a haptic interface device. These issues are the *torque ripple identification*, *gravity compensation* and *friction compensation*.

3.1. Torque Ripple Identification

The motors used to drive torque to robot joints are permanent magnet synchronous motors. Their high torque capability makes them suitable in high performance servodrives and robotics applications. In classical control for these motors, a sinusoidal armature fed current is imposed regardless of the type of the machine. If the rotor flux and the armature magneto motive force distributions are sinusoidal, this current leads to a constant torque. If the rotor flux is non-sinusoidal, a sinusoidal current produces an important noise effect called torque ripple (Colamartino, 1999) which is dependent on the rotor positions and velocity. In many applications involving low-bandwidth and/or high friction mechanical systems, ripple is unnoticeable at the output. However, in (Ferreti, 1999) it is shown that the influence of the torque ripple affects the force control for an industrial robot. In the high bandwidth WAM robot it is important to compensate for this ripple otherwise, this ripple will be felt at the end-effector. It was observed for our setup that the main influence of torque ripple was in low velocities. The motor has a position sensor attached at the motor shaft. The position of each shaft is divided in 4096 digital positions and one of them is called zero position (index signal).



Figure 1. WAM robot installed at GRASP.

By measuring and recording the steady state torque required to hold the unloaded motor spindle at each of the 4096 discretely measurable positions, we create have sufficient information for torque ripple compensation, since it is not necessary to compensate torque ripple for velocities. Data were acquired while a PID digital controller that was running to spin the motor shaft at very low velocities (0.25 RPM). The torque value sent to the motor was acquired for each of the 4096 positions. The beginning of each measurement started at the index position. The compensation was performed using the value acquired and feed-forwarding this value to the motor.

It was observed that for motor 0 (joint1) the value of the torque ripple is very high when compared with other motors. Motors 1 and 2 jointly apply torque at joints 2 and 3. The torque ripple value for both is small when compared to joint 1, but it is very noisy. The motor controllers gain values (output current / input reference) for these two motors are twice as much the gain of motor 0 and motor 3. We observed that the controller gain (adjustable by a potentiometer inside the inverter) has a wide influence in the torque ripple values.

3.2. Gravity Compensation

In a typical situation, the user will interact with the arm changing position and orientation of the end-effector. The user's hand will be grasping at the tip of the robot in order to move it. In this case it is necessary to compensate for the gravity in the robot links. Gravity compensation makes the robot feel weightless, involving the calculation of the necessary joint torques required to balance the arm statically against gravitational loading and overlaying those torques through precise force control. Applying the Lagrange formulation for the arm dynamics and making the velocities equal to zero (equilibrium state), results in:

$$\frac{\partial U}{\partial \mathbf{q}_i} = \mathbf{t}_i \quad (1)$$

Where U is the potential energy of the entire system, \mathbf{q}_i is a 4x1 joint position vector and \mathbf{t}_i is a 4x1 joint torque vector. The potential energy of the system can be written as:

$$U = \sum_{i=1}^4 m_i \mathbf{g}^T \mathbf{r}_{ci}^0 = \sum m_i \mathbf{g}^T T_i^0 \mathbf{r}_{ci}^i \quad (2)$$

Attaching reference frames to each joint according to Denavit-Hartenberg (DH) notation results in:

$$\mathbf{t}_i = \frac{\partial U}{\partial \mathbf{q}_i} = m_3 \mathbf{g}^T \frac{\partial T_3^0}{\partial \mathbf{q}_i} \mathbf{p}_{gc3} + m_4 \mathbf{g}^T \frac{\partial T_4^0}{\partial \mathbf{q}_i} \mathbf{p}_{gc4} \quad (3)$$

Where T_x^0 is the homogenous transformation from link x to link 0, $\mathbf{p}_{gc3} = [p_{x3}, p_{y3}, 0]^T$ and $\mathbf{p}_{gc4} = [p_{x4}, 0, p_{z4}]^T$ are the position of center of mass of links 3 and 4 written in coordinate system 3 and 4 respectively. Solving this partial derivate, we have the following result:

$$\begin{bmatrix} \mathbf{t}_1 \\ \mathbf{t}_2 \\ \mathbf{t}_3 \\ \mathbf{t}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ C_2 C_3 & -S_2 & -(S_2 C_4 + C_2 C_3 S_4) & C_2 C_3 C_4 - S_2 S_4 \\ -S_2 S_3 & 0 & S_2 S_3 S_4 & -S_2 S_3 C_4 \\ 0 & 0 & -(S_2 C_3 C_4 + C_2 S_4) & C_2 C_4 - S_2 C_3 S_4 \end{bmatrix} \begin{bmatrix} g_z [m_3 (p_{x3} + l_3) + m_4 l_3] \\ g_z [m_3 (p_{y3} + l_1) + m_4 l_1] \\ g_z m_4 p_{z4} \\ g_z m_4 p_{x4} \end{bmatrix} \quad (4)$$

The set of equations for gravity compensation can be written as:

$$\tau_i = \Psi K \quad (5)$$

Where Ψ is a 4x4 matrix dependent of joint positions and K is the 4x1 vector of dynamic parameters for gravity compensation. By measuring joint positions and torques for N different static positions, it is possible to write:

$$\begin{bmatrix} \tau_{P1} \\ \tau_{P1} \\ \vdots \\ \tau_N \end{bmatrix} = \begin{bmatrix} Y_{P1} \\ Y_{P2} \\ \vdots \\ Y_N \end{bmatrix} \begin{bmatrix} K_{P1} \\ K_{P2} \\ \vdots \\ K_N \end{bmatrix} \quad (6)$$

Equation (6) represents an over-determined system of equations, which has no exact solution. Nevertheless, it can be solved using a least squares technique:

$$\hat{K} = (\Psi_N^T \Psi_N)^{-1} \Psi_N^T \tau_N \quad (7)$$

3.3. Dynamic Modeling and Identification

The design and implementation of model-based controller requires an accurate knowledge of the manipulator's kinematic and dynamic model and its parameters. Experimental robot identification is the only efficient way to accurately estimate the model parameters as well as indications on their precision and validity. The dynamic behavior of open-chain rigid manipulators can be modeled using classical mechanics concepts like Lagrange or Newton-Euler (Sciavicco, 1996). According previous work those models can be derived using DH (Antonelli, 1999) or modified DH notation (Kozlowski, 1998). In this work it was chosen the Lagrange formulation and DH notation, since the equations of motion provide explicit state equations for robot dynamics and can be used to analyze and design advanced joint variable space control strategies. The computation of the dynamic coefficients requires a fair amount of arithmetic operations, but since this operation is performed only once, this is not a restriction.

3.3.1. Dynamic model

In order to model the dynamic behavior of the WAM manipulator, Lagrange formulation was used and the results are expressed below in matrix form:

$$\tau(t) + \tau_{int}(t) = D(q(t))\ddot{q}(t) + h(q(t), \dot{q}(t)) + c(q(t)) \quad (8)$$

In above equation, τ is an 4x1 generalized torque vector applied at joints; τ_{int} is the interaction torque at joints imposed by the environment; $D(q(t))$ is an 4x4 inertial acceleration-related symmetric matrix; $h(q(t), \dot{q}(t))$ is an 4x1 nonlinear Coriolis and centrifugal force and $c(q(t))$ is an 4x1 gravity loading force vector. These equations do not include the effect of viscous friction and static friction. The influence of this force will be treated in the following subsection. A full parameter vector was used, i.e. all inertial parameters of individual links - their mass, center of mass, first and second moments of inertia (10 elements for each DOF) - as shown in Equation (9).

$$\mathcal{G}'' = [I_{xx1}, I_{yy1}, I_{zz1}, I_{yy1}, I_{yz1}, I_{zz1}, m_1, m_1 x_{c1}, m_1 y_{c1}, m_1 z_{c1}, \dots \\ I_{xx4}, I_{yy4}, I_{xz4}, I_{yy4}, I_{yz4}, I_{zz4}, m_4, m_4 x_{c4}, m_4 y_{c4}, m_4 z_{c4}] \quad (9)$$

All of the terms in Equation (9) are expressed in local coordinate frames assigned to individual joints. After applying the Lagrange formulation and with the final set of equations, it can be seen that some of inertial parameters have negligible effect on the joint torques. Then, these parameters are removed from the parameter vector by inspection, resulting in 25 parameters (vector γ'). From (An, 1988; Kozlowski, 1998) it can be seen that the equations of motion are linear in the vector of the unknown dynamic parameters and can be rewritten for one sample point as:

$$\tau = W'(q, \dot{q}, \ddot{q}) \gamma' \quad (10)$$

Where the (4x25) upper triangular matrix $W'(q, \dot{q}, \ddot{q})$ depends only on kinematics quantities (An, 1988).

3.3.2. Friction model

Friction is present in mechanical systems incorporating parts with relative motion. Many techniques have been well reported in literature (Armstrong, 1994). There are basically two methods of friction compensation: model-based and non-model-based. In a model-based method, a function describes the behavior of the friction force using system

information like position and velocity. Some approaches for compensating friction, estimate the friction value on-line otherwise some identification process will be needed. In non-model-based compensation, proper control schemes are used to accommodate errors from friction, avoiding the need of its identification. In order to simplify the friction model, in this work it was assumed that it composed only of static and viscous parts, so it depends linearly on these friction coefficients. The friction force is assumed to be independent of joint angle. To prevent the stick-slip situation, we defined a threshold velocity band of width $2d$ centered on the origin where the frictional torque is parabolic with respect to the joint velocity. Based on this assumption, the expression for friction torque, F_i for the i^{th} joint is given by (where d mean the dead zone in the friction model):

$$F_{f,i} = \begin{cases} F_{v,i} \dot{q}_i + F_{s,i} \text{sign}(\dot{q}_i) & \text{if } |\dot{q}_i| > d \\ F_{v,i} \dot{q}_i + \frac{F_{s,i}}{d} \left(\frac{\text{sign}(\dot{q}_i) \dot{q}_i^2}{d} + 2\dot{q}_i \right) & \text{if } |\dot{q}_i| \leq d \end{cases} \quad (11)$$

3.3.3. Experimental identification

Combining the dynamic behavior of the arm, Equation (10), and including the friction equations described in Equation (11), we obtain a new parameter vector γ . This new vector includes the viscous and static friction (parameters $F_{v,i}$, $F_{s,i}$) and has 33 elements. In the same way, the upper triangular matrix W' , it was changed for W (4x33) including the kinematics quantities for friction $(\dot{q}_i, \text{sign}(\dot{q}_i))$.

In order to get data for identification proposes, a finite sum of sine functions was used as excitation trajectory for each joint. Different sets of trajectories it were used with different values of frequency and amplitude for each joint. The values of frequencies were higher than 0.02Hz and lower than 5Hz. While the trajectories were generated, amplitude and frequency values are being changed as linear function with time. Using a PID controller for each joint, the trajectories were executed with different initial positions of the joints and different relations between frequencies and amplitudes. The robot controller was running at 5KHz, and motor torque and joint positions were acquired at 100Hz. For each different trajectory, a set of data was acquired. With this new data set (N values), Eq. (10), the new matrix W and the new vector γ , we are ready to implement the least-square estimation for the vector γ (both augmented with N data points).

$$W_N = [W_{(1)} \cdots W_{(N)}]^T, \quad \tau_N = [\tau_{(1)} \cdots \tau_{(N)}]^T \quad (12)$$

The least-squares estimation:

$$\hat{g} = (W_N^T W_N)^{-1} W_N^T \tau_N \quad (13)$$

As shown in An (1988) and Kozlowski (1998), the full set of dynamic parameters, and not a canonical (or reduced) model, present some singularities because the matrix W_N is not a full rank. Hence, it is clear that some columns are linear combination of others columns, i.e. some inertial parameters are completely unidentifiable while some others can only be identified in linear combinations. In (Antonelli, 1999) a Singular Value Decomposition (SVD) algorithm was applied to reduce the model. The authors in (Kozlowski, 1998) use the set of dynamic equations combining some columns to find a new set with only completely unidentifiable parameters.

In the present work, we determined the minimum set of inertial parameters of WAM robot. The determination of the minimum set of inertial parameters of robots contributes to the reduction of the computational cost of the dynamic models and simplifies the identification of the inertial parameters. These parameters were obtained from the classical inertial parameters by eliminating those that have no effect on the dynamic model and regrouping some others, as shown in (Gautier, 1990).

3.3.4. Model validation

The accuracy of the estimated parameters was verified using different trajectories from those used in estimation process. Figure (2) shows how the error signal is obtained. In Fig. (3) the response of joint 3 for one trajectory is shown. The first graph shows the torque acquired in the robot controller (dotted line), and the torque computed using \hat{g} (continuous line). The second graph shows the absolute value of error in the joint torque. Table (1) presents sample covariance values for five different trajectories during the validation process for all joints.

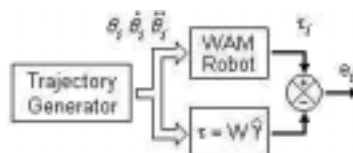


Figure 2. Diagram of the validation process.

Looking at the second graph of Fig. (3) we can see that the maximum error is below 20% of maximum torque.

Table 1. Covariance for some trajectories.

Trajectory	τ_1	τ_2	τ_3	τ_4
1	0.0460	0.0447	0.0139	0.0070
2	0.0669	0.0420	0.1042	0.0113
3	0.0332	0.0418	0.0093	0.0063
4	0.0258	0.0430	0.0110	0.0069
5	0.0294	0.0371	0.0091	0.0068

3.4. Dynamic Interaction

In order to better understand the dynamics of interaction, important results from the field of computational dynamics were surveyed, focusing on issues relevant to the construction of general purpose simulator with haptic display. There are different formalisms used to generate dynamic models of which Newton-Euler, Lagrange and Kane are some examples.

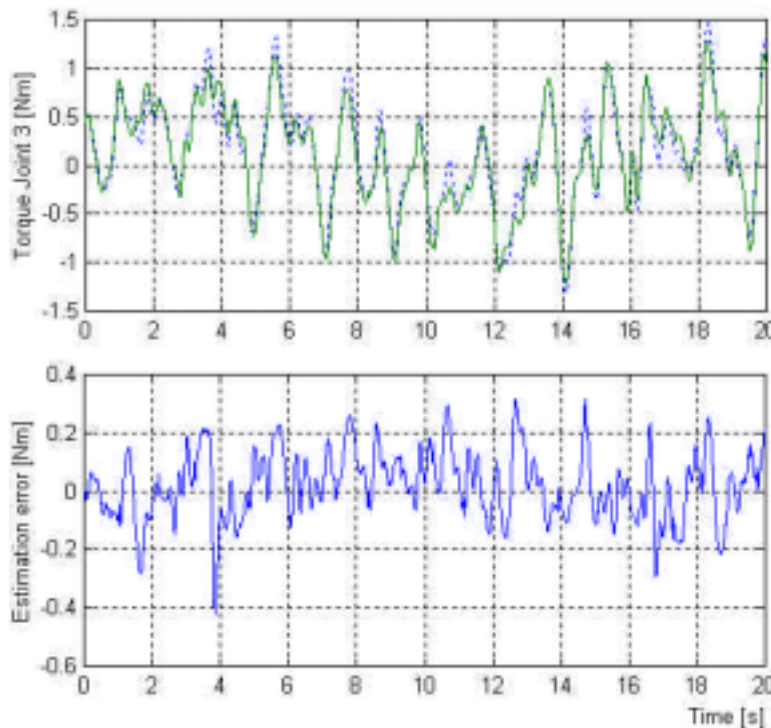


Figure 3. Joint torque and error at joint 3.

In (Gillespie, 1997) a survey of multibody dynamics for virtual environments is presented. To render the effects of intermittent contact and other transient phenomena as driven by a human haptic explorer, a computationally efficient model must be able to handle kinematics constrains. The ability to handle kinematics constrains is important to perform collision detection (also known as *interference detection* or *contact determination*). The goal of collision detection is to report a geometric contact when it is about to occur or has occurred. Geometric models may be polygonal objects, splines or algebraic surfaces. Lin (1998) presents a survey in collision detection between geometric models. In this work, polygonal approach is used to contact determination and environment modeling, because it has a simple representation and a widely hardware-accelerated rendering of polygon is available. Figure (4) shows a block diagram proposed dynamic interaction. We have implemented and tested our experiment without simulation of friction models.

We choose to implement the fixed bounding box (polygonal) as a general collision detection methodology because in our initial experiments, only polygonal objects are simulated. The main goal of this work is to get ready a haptic interface using a commercial robot (WAM). In the collision detection algorithm, the highest level identifies that objects that are close enough for collision and reports these potential collision pairs to exact or local collision detection algorithm. This approach is necessary because exact collision detection algorithm is time consuming.

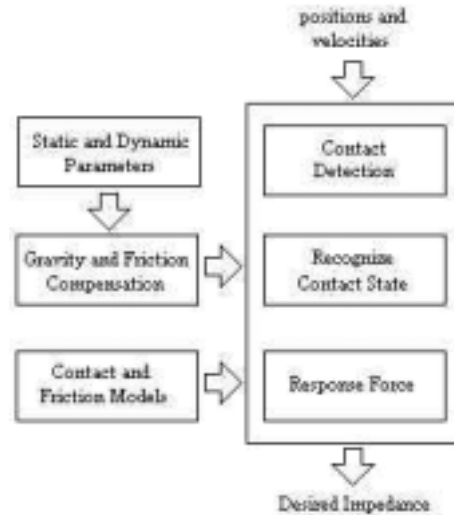


Figure 4. Proposed dynamic interaction block diagram.

3.5. Control of Haptic Device

For haptic displays, impedance and admittance displays have been proposed. In the case of impedance display, positions and velocities (motion) are acquired while the user moves the robot (haptic interface) and forces and torques are applied by the robot. For admittance display, forces and torques are measured in the interaction port and positions and velocities are displayed to the user and sensed while force is computed and displayed. In admittance simulators, force is sensed while motion is computed and displayed. In our approach we chose to use impedance control.

Positions and velocities are measured at the interaction port while forces and torques are computed by the control system. In order to implement the impedance control, we assume that the desired end-point behavior to be imposed on the manipulator is given by (Hogan, 1985):

$$M \frac{dV_{ep}}{dt} - B[V_d - V_{ep}] - K[X_d - X_{ep}] = F_{int} \quad (14)$$

Where M , B , and K are the mass, damping and spring values, V_d and V_{ep} the velocity desired and measured (calculated) of the end-point, and F_{int} the interaction force. Hogan derived the following control law by substitution:

$$\tau = D(\mathbf{q})J^{-1} \left[M^{-1} \left[K[X_d - L(\mathbf{q})] + B[V_d - J\dot{\mathbf{q}}] + F_{int} \right] - J\dot{\mathbf{q}} \right] + h(\mathbf{q}, \dot{\mathbf{q}}) + c(\mathbf{q}) - J^T F_{int} + F_f \quad (15)$$

Where J is the Jacobian and $L(\mathbf{q})$ is the linkage kinematic equations of WAM. This equation express the required behavior to be provided by the controller as impedance in actuator coordinates. It may be viewed as a nonlinear feedback law relating actuator torques to observations of actuator position, velocity and forces at the interaction port. The input (command) variables are the desired Cartesian position and velocity and the terms of the desired (possibly nonlinear) Cartesian dynamic behavior characterized by M , $B[J]$, and $K[J]$.

4. Results

In order to evaluate the performance of the haptic interface, the user moves the arm by holding a bar attached to the force sensor. Forces are measured at the tip of the WAM robot as the user moves the arm around the desired point. In our simulation we deal with a docking procedure. The WAM is used to simulate a remote docking station of a laptop. While the user moves and tries to insert the laptop in a remote dock station, forces and torques are acquired to evaluate the performance of the haptic interface. It was used a computer graphic representation (OpenGL) to display the laptop overlaid on a cardboard mounted at the end effector of the robot. We divided time of interaction with the virtual docking station into the following intervals:

1. $t_0 < t < t_1$ - (free space) - user moves laptop in front of docking station without touching it.
2. $t_1 < t < t_2$ - (front/back) - laptop in contact with front of docking station.
3. $t_2 < t < t_3$ - (inside) - laptop is sliding into docking station.
4. $t > t_3$ - (inside-bottom) - laptop touches back of docking station

From Fig. (5), it is clear that when the user is moving the laptop in free space the forces change smoothly. In the intermediate stage when the laptop makes contact with any surface of the docking station a force spike is observed and the user wearing the Head Mounted Display (HMD) “feels” the contact while seeing it. Once contact with the front of

the docking station is made, the user moves the laptop along its face until it is aligned with the slot. Once the alignment is done, the laptop slides until it reaches the back of the docking station.

The interaction force is computed using the impedance model Eq.(14) where K and B, respectively the spring and damping gain values. These values are constant for specific ranges of the position vector X in the Cartesian space of the task. The table below depicts the values of K and B that produced the best results during the docking procedure. By “best” we mean the subjectively overall better sensation perceived by several users that performed the procedure in the lab.

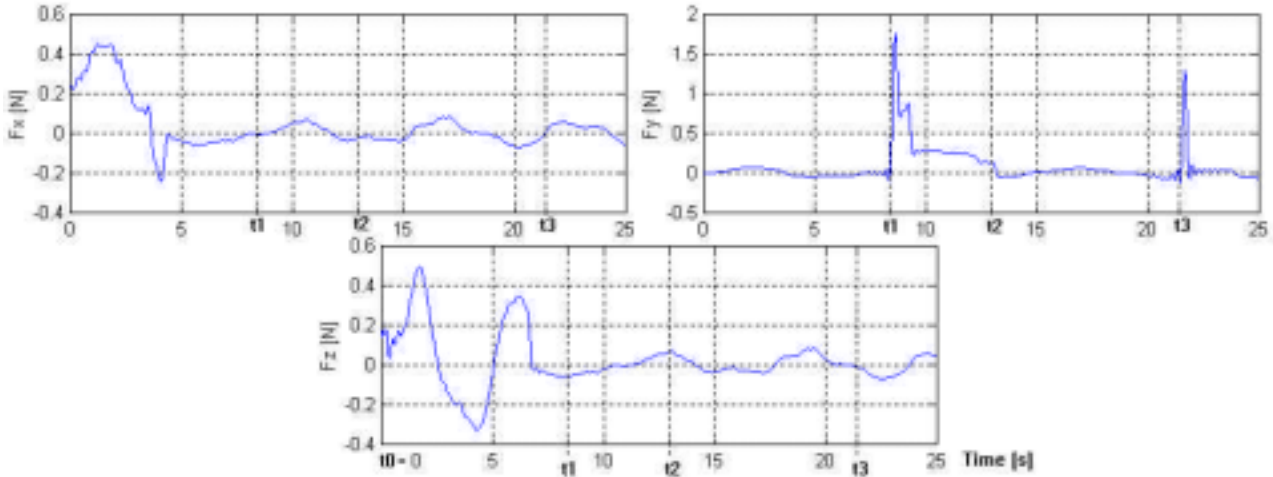


Figure 5. Interaction forces

In Tab. 2, H, M and L stand for high, medium and low gains, respectively. In free space, K and B are zero, and we only need to compensate for forces due to friction and gravity, making the arm “weightless”. At those positions where a collision is implied, different values of K and B are used, according to the previous table. This will provide different resulting forces being fed back to the user. Since a 4-DOF arm was used, evidently the current system does not respond to general forces and torques in Cartesian space (6-DOF). This will be addressed with the installation of an adequately designed 2-DOF wrist joint at the end of the last link. However, with the present configuration, our system is being used to evaluate the synchronization of the pose estimation performed by the Vision System module and the Haptic Display Controller for the docking procedure described above.

Table 2 – Impedance parameters.

Position	K_x	K_y	K_z	B_x	B_y	B_z
Free space	0	0	0	0	0	0
Top/bottom	0	0	H	0	0	M
Left/right	0	H	0	0	M	0
Front/back	H	0	0	M	0	0
Inside	0	H	H	L	H	H
Inside-bottom	H	H	H	L	H	H

5. Conclusions and future directions

This work addressed various aspects of a haptic interface design. First of all, it was introduced the haptic interfaces, design and applications, followed by an implementation using the WAM robot. All design steps are presented together with intermediate results. The main concerns for the design of a haptic interface can be summarized as: 1) fidelity in force-torque feedback; 2) stability-fidelity trade off – as pointed by several authors, fidelity and stability are contradicting factors and 3) performance under time delay. We successfully implemented an impedance control algorithm based on the Cartesian position of the end-effector. We have accomplished the first stage of the telepresence setup: the “haptic interface”. The depth recovery is sufficiently accurate to match the data obtained from the force interaction of the local object with the remote environment. In order to accomplish this we have a reasonably good model of the robot dynamics, including gravity compensation, friction modeling and torque ripple compensation. We have demonstrated the validity of our approach with the example of docking a laptop. There are several challenging issues that still need to be addressed. Some of these are visual representation and rendering in real time of an object in a remote environment and transferring the haptic information via the dynamic simulator.

6. References

Adams, R.J., Klowden, D., Hannaford, B., 2000, “Stable haptic interaction using Excalibur force display”, Proceedings of the 2000 IEEE International Conference on Robotics & Automation, San Francisco, CA, USA, pp. 770-775.

- An, C.H., Atkeson, C.G., Hollerbach, J.M., 1988, "Model-based control of a robot manipulator", Cambridge, MA, MIT Press.
- Ansar, A., Rodrigues, D.L., Desai, J.P., Daniilidis, K., Kumar, V., Campos, M.F.M., 2001, "Visual and haptic collaborative telepresence", Elsevier Science – Computer and Graphics, Special Issue on "Mixed Realities – Beyond Conventions", vol. 25, no. 5 (to be published in October 2001).
- Antonelli, G., Caccavale, F., Chiacchio, P., 1999, "A systematic procedure for the identification of dynamic parameters of robot manipulators", *Robotica*, Vol. 17, pp. 427-435.
- Armstrong-Helouvry, B., Dupont, P., Canudas De Wit, C., 1994, "A survey of models, analysis tools and compensation methods for the control of machines with friction", *Automatica*, Vol. 30, pp. 1083-1138.
- Asada, H., Slotine, J.J., 1985, "Robot analysis and control", John Wiley & Sons, New York, NY, USA.
- Burdea, G.C., 1996, "Force and touch feedback for virtual reality", John Wiley & Sons, New York, NY, USA.
- Burdea, G.C., 1999, "The synergy between virtual reality and robotics", *IEEE Transactions on Robotics and Automation*, Vol. 15, no. 3, pp. 400-410.
- Cavusoglu, M.C., Tendick, F., Cohn, M., Sastry, S.S., 1999, "A laparoscopy telesurgical workstation", *IEEE Transactions on Robotics and Automation*, Vol. 15, no. 4, pp 728-739.
- Clover, C.L., 1999, "A control-system architecture for robots used to simulate dynamic force and moment interaction between human and virtual objects", *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews*, Vol. 29, no. 4, pp. 481-493.
- Colamartino, F., Marchand, C., Razek, A., 1999, "Torque ripple minimization in permanent magnet synchronous servodrive", *IEEE Transactions on Energy Conversion*, Vol. 14, no. 3, pp. 616-621.
- Ferreti, G., Magnani, G., Rocco, P., 1999, "Force oscillations in contact motion of industrial robots: an experimental investigation", *IEEE/ASME Transactions on Mechatronics*, Vol. 4, no. 1, pp. 86-91.
- Gautier, M., Khalil, W., 1990, "Direct calculation of minimum set of inertial parameters of serial robots", *IEEE Transactions on Robotics and Automation*, Vol. 6, pp. 368-373.
- Gillespie, R.B., Colgate, J.E., 1997, "A survey of multibody dynamics for virtual environments", *Proceedings of the ASME International Mechanical Engineering Conference and Exposition, DSC-Vol. 6*, Dallas, TX, Nov 15-20, pp. 45-54.
- Gillespie, B., Rosenberg, L.B., 1994, "Design of high-fidelity haptic display for one-dimensional force reflection applications", *Telemanipulator and Telepresence Technology, Proceedings of the International Society of Optical Engineers – SPIE – East Cost Conference*, Boston, MA, Oct-31 Nov-1, pp. 44-54.
- Hogan, N., 1985, "Impedance control: an approach to manipulation", *Journal of Dynamic Systems, Measurement, and Control*, Vol. 107, pp. 1-7 (Part I) and pp. 8-16 (Part II).
- Kazerooni, H., Her, M.G., 1994, "The dynamics and control of a haptic interface device", *IEEE Transactions on Robotics and Automation*, Vol. 10, no. 4, pp 453-463.
- Kozlowski, K., 1998, "Modeling and identification in robotics", Springer-Verlag, London.
- Lin, M.C., Gottschalk, S., 1998, "Collision detection between geometric models: a survey", *Proceedings of IMA Conference on Mathematics of Surfaces*.
- L.Sciavicco and B.Siciliano, 1996, "Modeling and control of robot manipulators", McGraw-Hill, New York, NY, USA
- Sheridan, T.B., 1992, "Telerobotics, automation, and human supervisory control", MIT Press, Cambridge, MA, USA.
- Sirouspour, M.R., DiMaio, S.P., Salcudean, S.E., Abolmaesumi, P., Jones, C., 2000, "Haptic interface control – design issues and experiments with a planar device", *Proceedings of the 2000 IEEE International Conference on Robotics & Automation*, San Francisco, CA, USA, pp. 789-794.
- Yokokohji, Y., Hollis, R., Kanade, T., 1999, "Wysiwyf display: A visual/haptic interface to virtual environment", *Presence*, Vol. 8, no. 4, pp. 412-434.
- Webster's Ninth New Collegiate Dictionary, 1985, Merriam-Webster Inc., Springfield, MA, USA

7. Acknowledges

Part of this work was carried out at General Robotics Automation, Sensing and Perception Lab – GRASP in University of Pennsylvania – UPENN. The authors gratefully acknowledge the support of the CAPES (grant BEX1020/99-4) and PUC Minas that support D.L. Rodrigues, and CNPq that support M.F.M. Campos.

PIEZOELECTRIC ACTIVE SYSTEMS FOR MONITORING SURFACE BREAKING DEFECTS

Antonio Lopes Gama

Universidade Federal Fluminense, Departamento de Engenharia Mecânica, 24210-000, Niterói, RJ
agama@mec.uff.br

Sergio Ricardo Kokay Morikawa

Arthur Martins Barbosa Braga

PUC-Rio, Departamento de Engenharia Mecânica, 22453-900, Rio de Janeiro, RJ
abraca@mec.puc-rio.br

Rafael Ribeiro Correia

Universidade Federal Fluminense, Departamento de Engenharia Mecânica, 24210-000, Niterói, RJ

Abstract. *In this paper two active techniques using piezoelectric sensors and actuators to monitor the growth of surface breaking defects were investigated. The first method consists in exciting the structure with piezoelectric actuators, subjected to either a single frequency or broad-band signal, while recording the electromechanical response of sensors placed close to the defect. In this case, the piezoelectric sensors detect the damage growth by monitoring changes in the dynamic strain field induced by the actuator near the defect. The performance of this methodology was assessed through experiments in beams containing surface breaking fatigue cracks or machined slots. Finite element simulations were also performed to determine a correlation between sensor response, sensor location, and damage size. Results from tests performed in a three-dimensional framed structure are also presented. The second method is a impedance-based technique where a single piezoelectric element is used to excite and to monitor the growth of the defect. Tests in a beam with a machined slot shown variations of the electrical admittance (admittance is the reciprocal of impedance) of the piezoelectric sensor in a high frequency range (0.5 MHz to 1 MHz) due to change in the slot depth.*

Keywords. *Structural health monitoring, piezoelectric sensors and actuators, surface breaking defects.*

1. Introduction

At present there is a great interest in health monitoring of in-service structures on a continuous real-time basis due to the advantages that can be provided compared to conventional inspection techniques. The continuous monitoring can lead to a reduction of the downtime, extend life and help to avoid catastrophic failures. Searching for alterations in vibration modes due to structural damage has long been recognized as a potentially effective approach for monitoring the integrity of in-service structures (Doebing *et al.*, 1995, Dimarogonas, 1996, Salawu, 1997 and Zou *et al.*, 2000). Of the vibration-based methods for structural health monitoring, those that rely on changes in natural frequency are the ones that early on have attracted the most attention (Cawley and Adams, 1979). The effectiveness of such approach may, however, be limited due to the need of detecting, in a reliable way, very small variations in resonance frequency, often of the order of those associated with changes in ambient conditions (Salawu, 1997). Other equally appealing approaches have also been proposed (Liang *et al.*, 1994, Banks *et al.*, 1996 and Pines, 1997), all with the common purpose of detecting and locating the onset of structural damage. A less ambitious goal, but nonetheless of technological interest, is to simply monitor the evolution of localized, but yet acceptable damage, once it has been identified by other conventional technique. In certain cases, this may allow the operator to delay repair, keeping the structure in service while observing the evolution of damage, waiting the best opportunity to execute a maintenance stop. This is the motivation for the study reported in this paper.

Two techniques using piezoelectric sensors and actuators to monitor the growth of surface breaking defects were investigated. The first method consists in exciting the structure with piezoelectric actuators, either with a broad-band or single frequency (CW) signal, and recording the voltage of sensors placed close to the defect or in areas where localized damage is expected to occur. Piezoelectric sensors detect the damage growth or incipient failure by monitoring changes in the dynamic strain field induced by the actuator. For example, as a crack becomes deeper, the amplitude of surface strains close to the crack edge decreases and so does the electrical signal generated by the piezoelectric sensors. Placing the sensors close to the defect is what may allow the detection of very small changes in depth. Previous work on a similar methodology has been reported by Verreman, 1994, and Otegui *et al.*, 1991, who have employed electric resistance strain gages to monitor fatigue cracks in welded joints, being able to successfully detect variations of 20 μm in crack size. The advantage of using piezoelectric sensors is their high sensibility, which allows the low-level strains generated by the piezoelectric actuator to be properly sensed.

The performance of this methodology was assessed initially through experiments in beams containing surface breaking fatigue cracks and machined slots. Results have shown that the choice of adequate parameters, such as sensor size and its distance to the crack edge, allows the detection of small changes in defect depth, even in a frequency band within the modal domain of the structural response.

Finite element simulations were performed to determine a correlation between sensor response, sensor location, and damage size. Numerical and experimental results when the actuator is excited by a CW signal, have demonstrated that the sensor response to changes in defect depth does not dependent on the employed frequency as long as not within the resonance region. Tests conducted in a framed structure have also demonstrated the good potential of the method to applications in more complex structures

The second method is a impedance-based technique where a single piezoelectric element is used to excite and to monitor the growth of the defect (Liang *et al.*, 1996). In this case, the current flow through the piezoelectric is a direct indication of the interaction between the structure and the piezoelectric patch (Ayres, 1998). This method was evaluated through experiments in an aluminum beam where a machined slot of increasing depths was monitored by measuring the electrical admittance (admittance is the reciprocal of impedance) of a small piezoelectric ceramic bonded close to the slot. Using a electrical impedance analyser, the real and imaginary parts of electrical admittance of the piezoelectric sensor were recorded in a frequency range varying from 1Hz to 1 MHz. The results show that at high frequencies increases the sensitivity of the admittance variations.

2. Experimental Procedure

The health monitoring technique was preliminarily evaluated in experiments with either aluminum or steel beams containing, respectively, a machined slot or fatigue crack. Dimensions of the specimens, actuators, sensors and their position along the beam are listed in Tab. (1), while the experimental setup is presented in Fig. (1).

Table 1. Specimen dimensions in millimeters (abbreviations are shown in Figure 1).

Material	Type of defect	$W \times t$	L	L_a	L_s	X_a	X_s
Aluminum	Slot	9.6 x 9.6	380	20	7.0	45	268
Steel	Crack	20 x 24	350	50	8.0	20	240

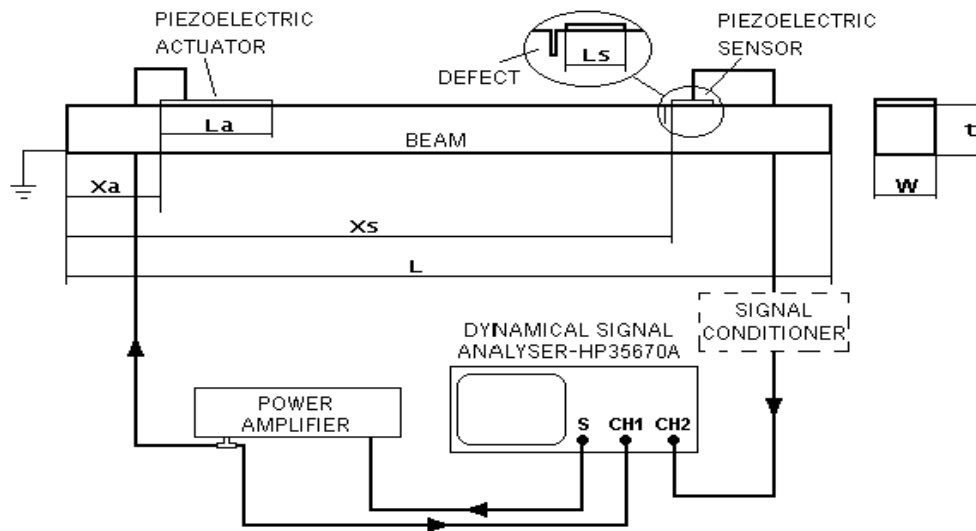


Figure 1. Experimental setup used to monitor the growth of surface breaking defects.

As depicted in Fig. (1), a pair of piezoceramic sensor and actuator is bonded to the beam's surface. Before introducing the defect — either a slot or a fatigue crack — the beam's FRF in a free-free condition is measured. This FRF, in the range from 0 to 52 kHz, was measured using a spectral analyzer. White-noise signal from the analyzer source is amplified and sent to the piezoelectric actuator. Both the excitation signal and that provided by the piezoelectric sensor are sent to the analyzer, and the FRF is determined using a Hanning window.

2.1. Aluminum beams with machined slots

Slots of 0.4-mm width were machined in the aluminum specimens at increasing depth increments of 0.5 mm, up to 5 mm, which corresponds to a defect with depth equal to 52% of the beam's thickness. Measurements were carried out after each depth increase. Fig. (2) shows results for one of the aluminum specimens. The FRF for various defect depths, measured up to 12 kHz, are shown in Fig. (2-a). One notices a significant decrease in amplitude, away from the

resonance frequencies, as the defect depth increases. The relative amplitudes of the FRF at some selected frequencies are plotted in Fig. (2-b) as a function of the slot depth. The decay in dB has been defined as:

$$dB = 20 \log_{10} (A/A_0) \tag{1}$$

where A is the beam's FRF amplitude at a given defect depth and A_0 is the reference amplitude measured before the defect was introduced. For small slot-depth to beam-thickness ratios, and all four frequencies chosen to construct Fig. (2-b), one notices an almost linear, close to 3 dB/mm, decay in the signal measured by the sensor as the slot becomes deeper. These results shown very good sensitivity to defect growth and practically the same relative amplitude decrease at different frequencies. It should be pointed out, as it will be shown later, that this sensitivity is dependent on the size and position of the sensor.

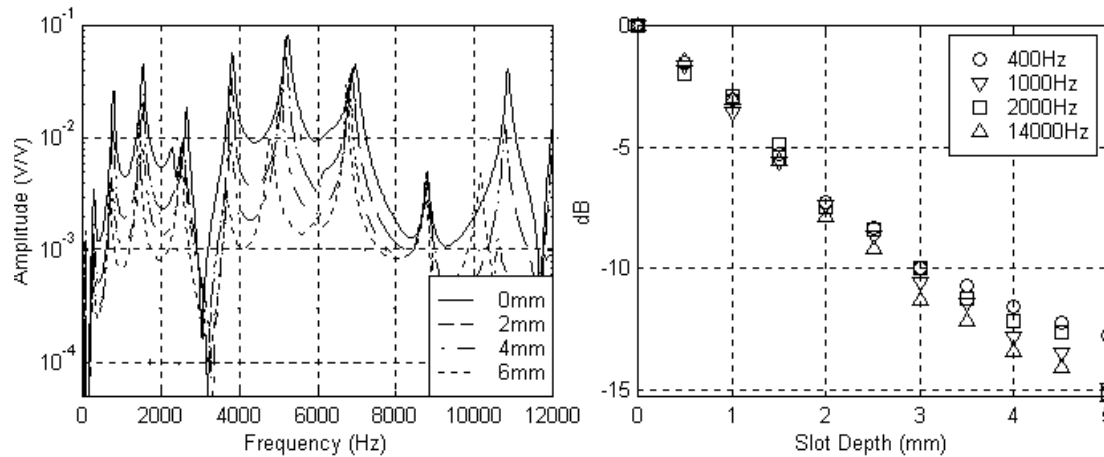


Figure 2. Results for aluminum beam with machined slots (dimensions in Tab. (1)): (a) Frequency response of the beam with slots of various depths; (b) Amplitude decay in dB (Eq. (1)) measured at fixed frequencies as the defect depth increases in increments of 0.5 mm.

2.2. Steel beams with fatigue cracks

The growth of more realistic defects was investigated by monitoring fatigue cracks introduced in steel beams. The specimen was prepared by, firstly, introducing a V-notch in a thicker steel (SAE-1045) beam to serve as the stress concentrator which would allow the nucleation of the fatigue crack. This beam was then subjected to a cyclic, three-point bending load, in a servo-hydraulic test machine, until a 3-mm deep, pre-crack was grown. Next the beam was machined to the dimensions shown in Tab. (1), yielding a specimen with a 2-mm deep crack. The actuator and sensor were then bonded to the beam surface as shown in Fig. (1). The experimental procedure was identical to that described for the aluminum specimens with slots. In this case, however, the reference FRF was measured for the already damaged beam, containing a 2-mm deep, surface breaking defect. After each FRF measurement, the crack was deepened in steps, of roughly 1 mm, by re-applying the three-point bending cyclic load with the aid of the servo-hydraulic test machine. Results for one of the specimens are presented in Fig. (3).

Although there still is a significant reduction in the amplitude of the signal measured by the piezoelectric sensor as the crack becomes deeper, the decay rate for the cracked beam is now lower than that verified for the slotted specimen. While in the first case the decay was of approximately 3 dB as the slot reached 10% of the beam thickness, the same relative decrease in amplitude was verified only when the crack depth was close to 25% of the steel specimen thickness. This behavior was in fact expected under the light of the results reported by Cawley and Ray, 1988, who have carried out a thorough study on the different sensitivity of changes in natural frequency produced by cracks and slots in beams. This lower sensitivity of the cracked specimen may also be attributed to the closure of crack surfaces, which allows partial transmission of loading across the joined faces of the defect. Since the FRF is measured for the beam in a free-free condition, with the input load provided only by the piezoelectric actuator, the crack is kept closed. Four steel specimens with fatigue cracks were monitored in this fashion. An excellent repeatability and stability of the electromechanical response of the piezoceramic sensor during the cyclic loading was verified.

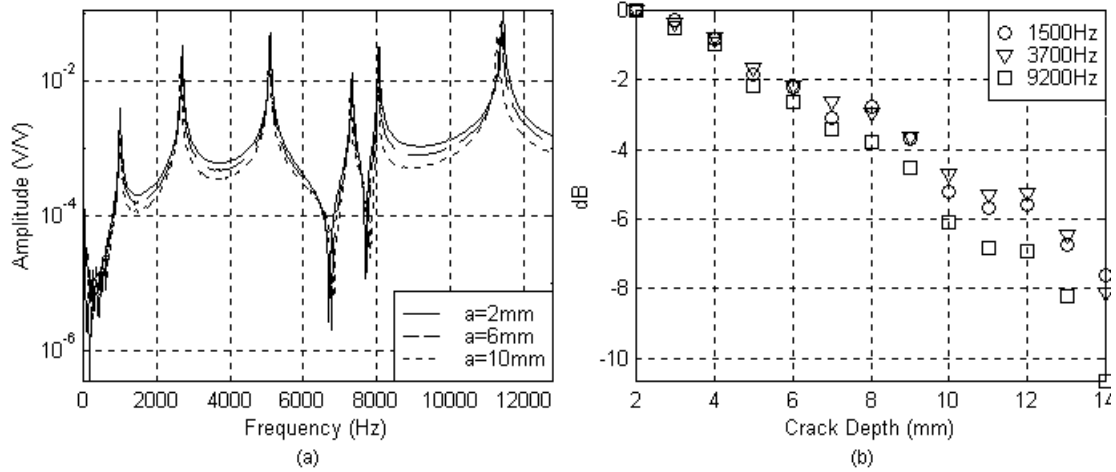


Figure 3. Results for steel beam with fatigue crack (dimensions in Table 1): (a) Frequency response for crack with various depths; (b) Amplitude decay in dB (Eq. 2) measured at fixed frequencies as the defect depth increases in increments of 1 mm.

The influence of crack closure level on the sensor response was studied with the aid of the experimental setup depicted in Fig. (4). A steel beam (dimensions in Tab. (1)) with a fatigue crack 10-mm deep was instrumented with three piezoceramic sensors. Two of them in the vicinity of the crack (sensors 1 and 2) and the third one away from the defect (sensor 3). The cantilever beam was mounted as shown in Fig. 4. The electromagnetic, static load was applied in increasing levels while the response of sensor 3 to a white-noise signal was recorded showing no variation. Signals from the other two sensors, on the other hand, suffered a significant decrease in amplitude as the load was increased. Observe that the upper portion of the cantilever beam was kept under tensile stresses as the magnetic load was applied, causing the crack to open. Finally, the response of sensors 1 and 2 to a CW signal applied to the actuator was recorded while the bending load was increased statically in steps. Results obtained for a CW excitation at 7 kHz are reproduced in Fig. (4), revealing that the reduction in the signal measured by the piezoceramic sensor is much more expressive when the crack is under higher tensile stresses. This indicate that the same levels of decay rate observed in tests with slotted beams may be achieved in applications where fatigue cracks are subjected to stress fields that will cause it to open.

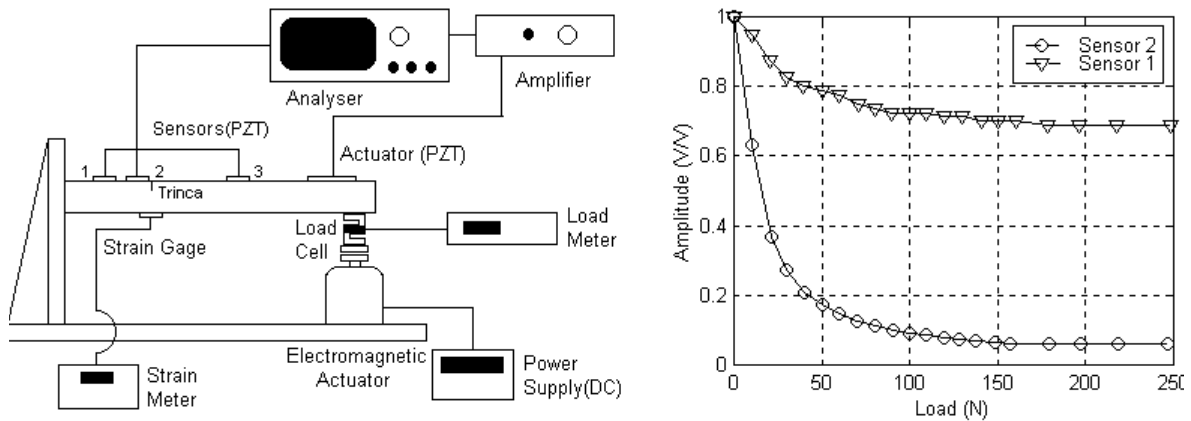


Figure 4. Experimental analysis of crack closure effect on sensor response. Plot shows decay in peak values of the signal measured by sensors 1 and 2 when the actuator is excited by a CW signal (7 kHz) while the load is applied statically, opening the crack but keeping its depth unchanged. Amplitudes of the signals are referred to their values in the unloaded condition.

2.3. The impedance-based method

This section reports results of monitoring the growth of a machined slot in a beam using a piezoelectric impedance-based technique. Various types of defects such as cracks, corrosions, delaminations etc, will change the mechanical impedance of the structure (Ayres *et al.*, 1998, Chiu *et al.*, 2000). The electromechanical coupling between a piezoelectric transducer material and the local structural system allows for the extraction of the mechanical impedance information from the electrical impedance measurement of the piezoelectric sensor (Liang *et al.*, 1996).

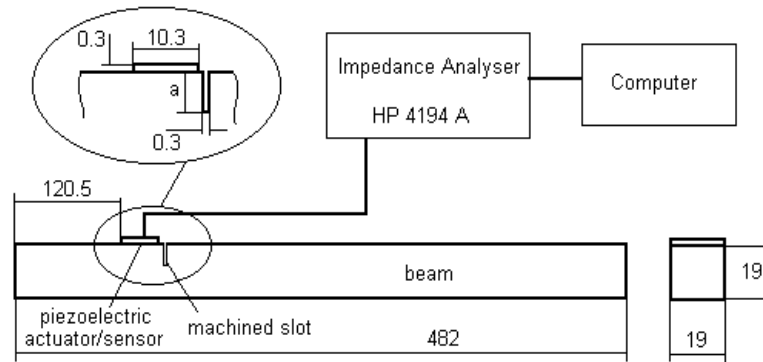


Figure 5. Experimental set up to monitor the growth of a machined slot in a beam through variations of the electrical admittance of a piezoelectric sensor.

The electrical impedance can be obtained using a commercial impedance analyser connected to the piezoelectric patch. The impedance analyser provides a constant voltage of excitation to the piezoelectric transducer over a chosen frequency range and the current draw of the piezoelectric is recorded in a real and imaginary admittance.

This process was applied to monitor the growth of a machined slot in an aluminum beam as depicted in Fig. (5). Initially the admittance of the piezoelectric sensor was measured in a broad frequency range before introducing the defect. The admittance in the range from 1 Hz to 1 MHz was measured using an HP 4194 A impedance analyser. Slots of approximately 0.3 mm width were machined at increasing depths close to the piezoelectric sensor and the admittance recorded in a real and imaginary part for each slot size. The variations in the electrical admittance may occur also due to the temperature change, humidity and contamination of the piezoelectric sensor. These factors can change the capacitance of the sensor and consequently change its admittance. Therefore, care must be taken during the experiment in order to avoid the influence of these factors.

Figure (6) shows a plot of conductance (real part of admittance) as a function of frequency for different slot sizes, in a range where the best sensitivity was found. The results shown that at high frequencies increases the sensitivity of admittance variations due to the defect size. This is because at high frequencies, the short wavelength of the induced vibrations makes the sensor response more sensitive to changes in the local mechanical impedance produced by the defect.

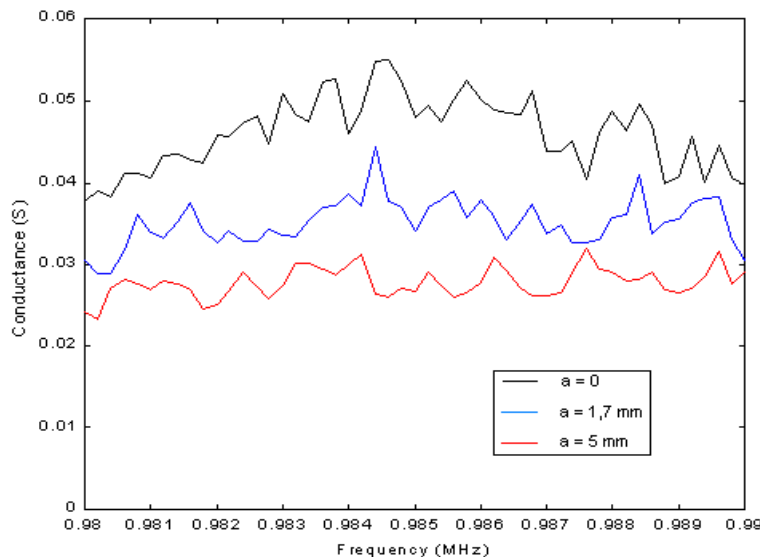


Figure 6. Variation of the conductance of the piezoelectric sensor for increasing slot depths in the aluminum beam shown in Fig.(5).

3. Numerical analysis

Finite element (FE) simulations and a numerical procedure (Braga *et al.*, 2000) based on the Reddy's layerwise laminate theory (Robbins and Reddy, 1991) were employed to investigate the influence of sensor location and defect

depth on the system response. The electromechanical behavior of a piezoelectric sensor placed near a slot is illustrated in Fig. (7), which shows typical results of a simulation using the method described in Braga *et al.*, 2000.

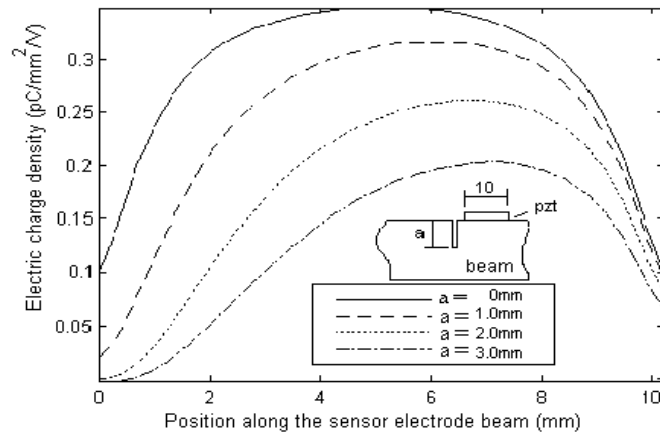


Figure 7. Charge distribution along the piezoelectric sensor for different slot depths.

The configuration is similar to the one depicted in Fig. (1). A CW signal is input to the actuator bonded to the surface of a free-free beam. The electric charge amplitude distribution along the sensor's top electrode is numerically estimated at different slot depths. As it should be expected, the charge distribution along the sensor decreases progressively while the slot depth increases. One also notices that the decrease in electric charge density on the electrode is greater at the region of the sensor closer to the slot. The electric charge density decay is attributed to a reduction in the dynamic strain field generated by the actuator in the vicinity of the slot as the defect is deepened.

The FE model of a beam excited by a piezoelectric actuator and monitored by a piezoelectric sensor was implemented with the aid of the commercial code ANSYSTM. The FE mesh of the simulated slotted aluminum beam, with the same dimensions as the experimental specimens in Tab.(1), is presented in Fig. (8-a). Results of a comparison between the numerical simulation and experimental results are shown in Fig. (8-b), where the decay rate of the measured signal due to a CW excitation is plotted against the depth of the slot. The agreement is quite good in spite of uncertainties in the mechanical and piezoelectric properties of the sensors and actuators. This should not be so surprising since the output signal is being referred to its value at the non-defective (zero slot depth) configuration.

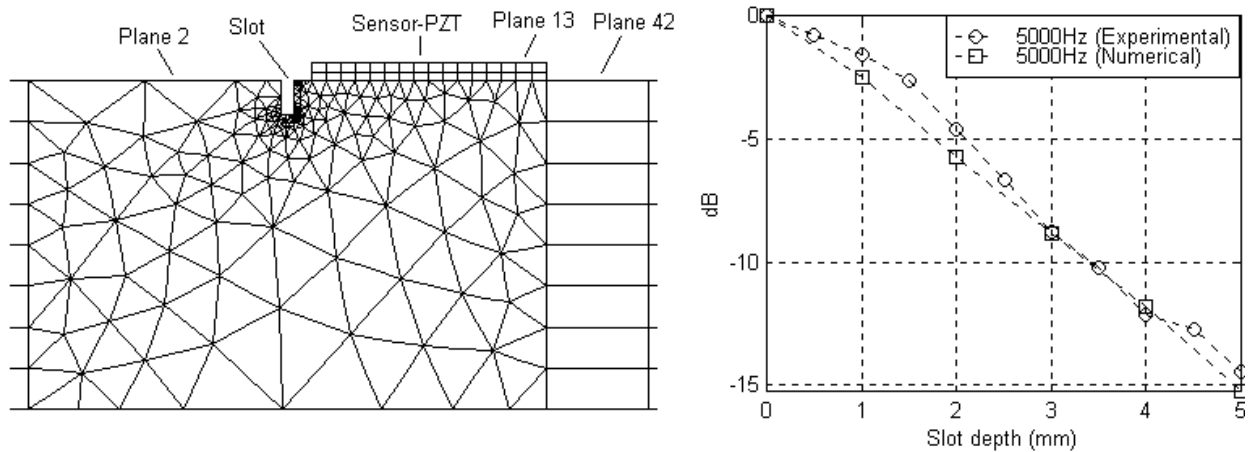


Figure 8. (a) Finite element mesh and (b) comparison between experimental and finite element results.

The very good agreement of the results predicted by the FE model with the experiments, motivated its use in additional simulations to determine a correlation between sensor response, sensor location and damage depth.

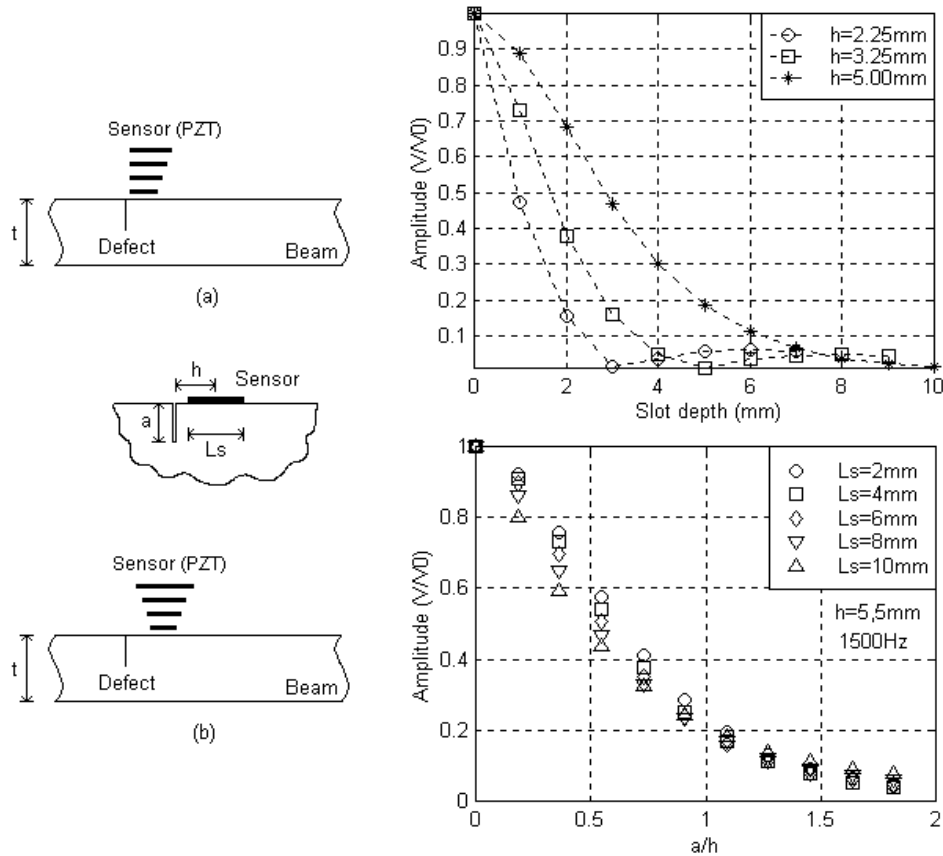


Figure 9. Results of FE simulations showing the influence of sensor position and size on sensibility.

The simulated responses of sensors of different sizes to a CW excitation are plotted in Fig. (9-a), which shows decay rates as the slot is deepened. In these FE simulations, the distance of the edge of the sensor to the slot was kept constant, equal to 0.5 mm, while the size of the sensor was varied. The parameter h is the distance of the center of the sensor to the defect. Thus, for instance, when h is 2.25 mm, the size of the sensor, L_s , is equal to 1.75 mm. By analyzing the results in Fig. (9-a), one may conclude that, if smaller sensors are much more sensitive to small damage growth, larger ones can nonetheless record a higher range of defect depths before reaching a saturation point. Results in Fig. (9-b), obtained when the distance h is fixed while the length of the sensor varies, show that the sensibility, or the decay rate, does not change substantially when the results are plotted with respect to the ratio a/h , where a is the slot depth. This good correlation of results with the ratio a/h was also observed in experiments conducted by Verreman, 1994, who used electric resistance strain gages to monitor fatigue cracks in welded joints.

The influence of the beam thickness was also investigated with the aid of FE models. Fig. (10) presents results obtained when both the length of the sensor, L_s , and its distance from the defect, h , were kept constant while the beam thickness, t , was changed. As shown in the plot, at small defect depths the decay rate does not depend significantly on the ratio h/t . Also, as the beam becomes thicker, the sensor response to depth changes in the defect, also varies very little with the parameter h/t .

A compilation of results from FE simulations of the response of slotted aluminum and steel beams with different sensor lengths, positions, and at various frequencies, are summarized in Fig. (11). All simulated defects had width of 0.4 mm. It must be pointed out, as discussed in an article by Cawley and Ray, 1988, that a lower sensibility should be expected for beams with surface breaking cracks. Results of the preceding section, however, tend to indicate that if the crack is subjected to high tensile stress such that it would be kept open, the same level of sensitivity verified for the slotted beams could be achieved.

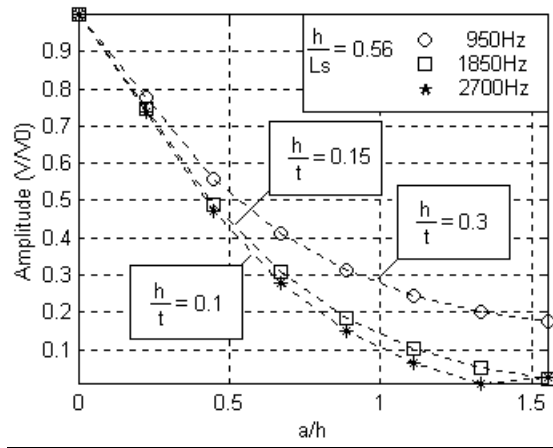


Figure 10. Effect of the slot-depth to thickness ratio.

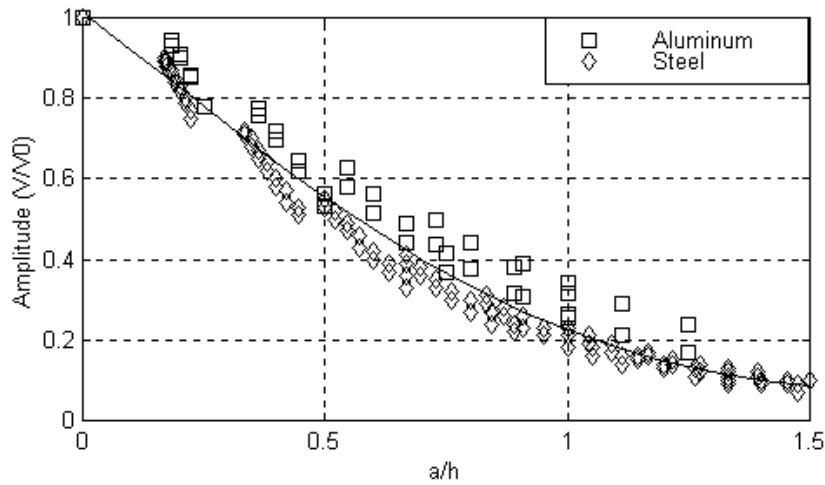


Figure 11. Compilation of results from various FE simulations.

4. Experiments in a framed structure

In this section we describe preliminary experiments carried out to evaluate the application of the proposed health monitoring technique to a more complex structure. A framed structure, built up from aluminum bars with square cross section was employed in the experiments. A machined slot (0.4-mm width) was gradually introduced in one of its members. The response of a piezoelectric sensor, placed in the vicinity of the defect, to either a broad band or CW excitation, applied by a piezoelectric actuator, was monitored after each time that the defect depth was increased. During some of the tests, the structure was also subjected to a broad band random-noise excitation applied by a magnetic shaker. The experimental setup is schematically drawn in Fig. (12), which shows the configurations for the tests where the random-noise excitation by the shaker was either turned off or on. During the tests, the structure was suspended by nylon wires while being also supported by the electromagnetic shaker.

The first series of tests were conducted in the same fashion as those carried out in beams with machined slots. With the shaker coupled to the truss but turned off, the piezoelectric actuator was excited by a broad band signal and, for each slot depth, the structure's FRF was measured using the experimental arrangement shown in Fig. (12-a). Results for one of the tests are presented in Fig. (13). As observed in the experiments with beams, one once again notices the decay in the FRF amplitude as the defect grows. Results from CW excitation, reproduced in Fig. (12-b), are also similar to those presented in Section 2.

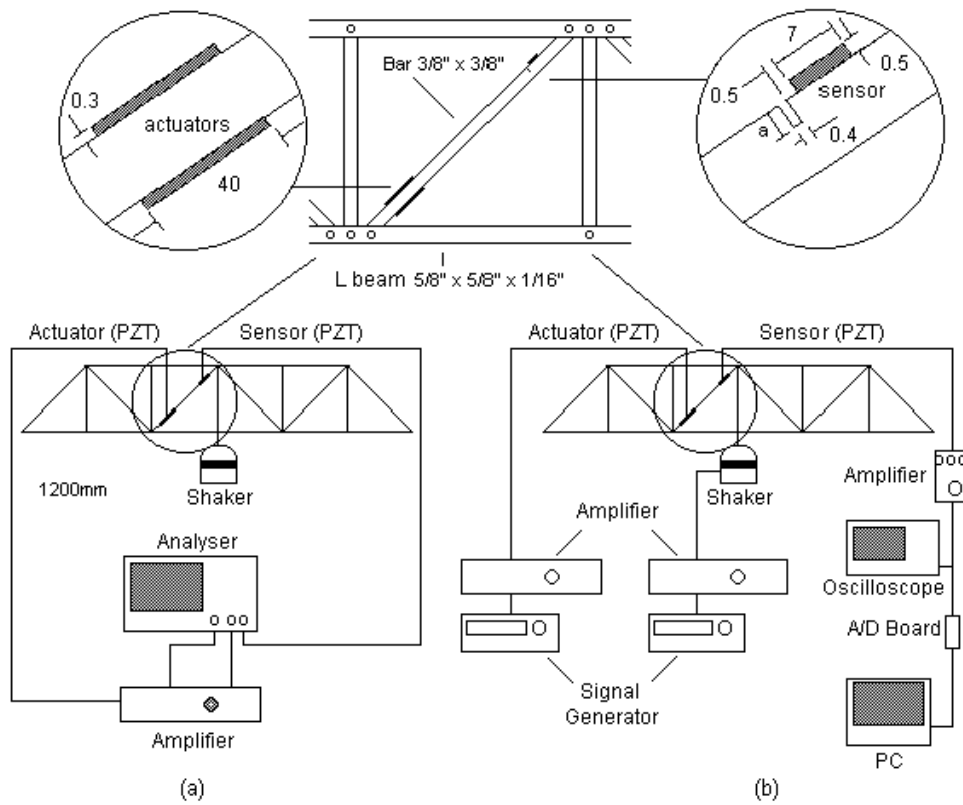


Figure 12. Experimental setup for tests with a framed structure.

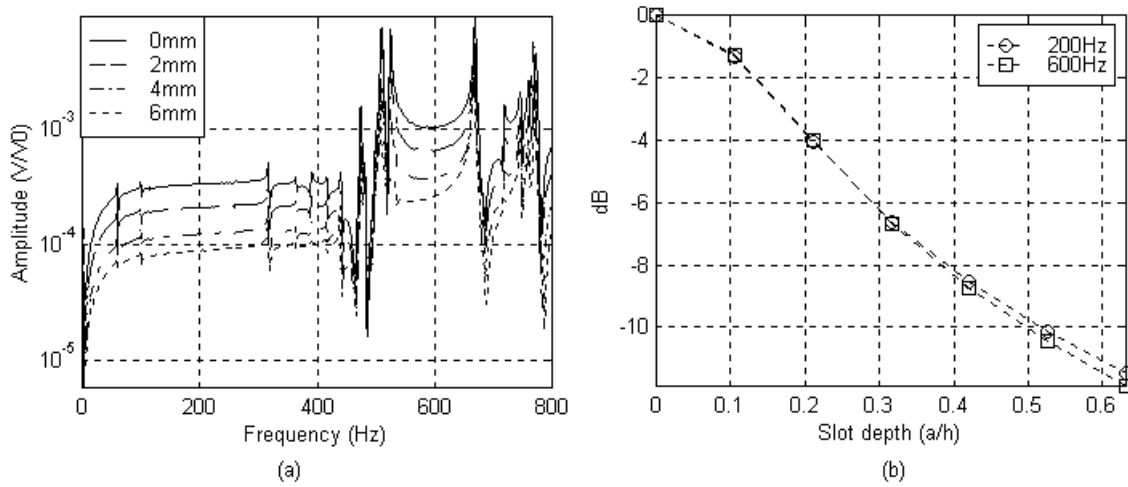


Figure 13. Results for tests with the framed structure (shaker turned off).

In another series of tests, the goal was to assess the performance of the method when the structure was also subjected to an external excitation other than that due to the piezoelectric actuator. The external load was then provided by the electromagnetic shaker excited by a white-noise signal with bandwidth of 800 Hz. A CW signal was applied to the piezoelectric actuator. The amplitude of the dynamic loading applied by the shaker was adjusted so that it would generate in the sensor signals five times higher than that due to the piezoelectric actuator alone. The tests were conducted with the experimental arrangement drawn in Fig. (13-b). The analogic signal from the sensor was digitized at a 15 kHz sampling rate and sent to a PC, where it was processed in a program developed to run under National Instrument's LabView™ ambient. The signal was then digitally filtered through a band-pass filter, centered at the frequency of the CW excitation provided by the piezoelectric actuator, and averaged several times before its root-mean-square amplitude was recorded. Comparisons of measurements made with the shaker turned off and on, presented in Fig. (14), have shown a quite good agreement.

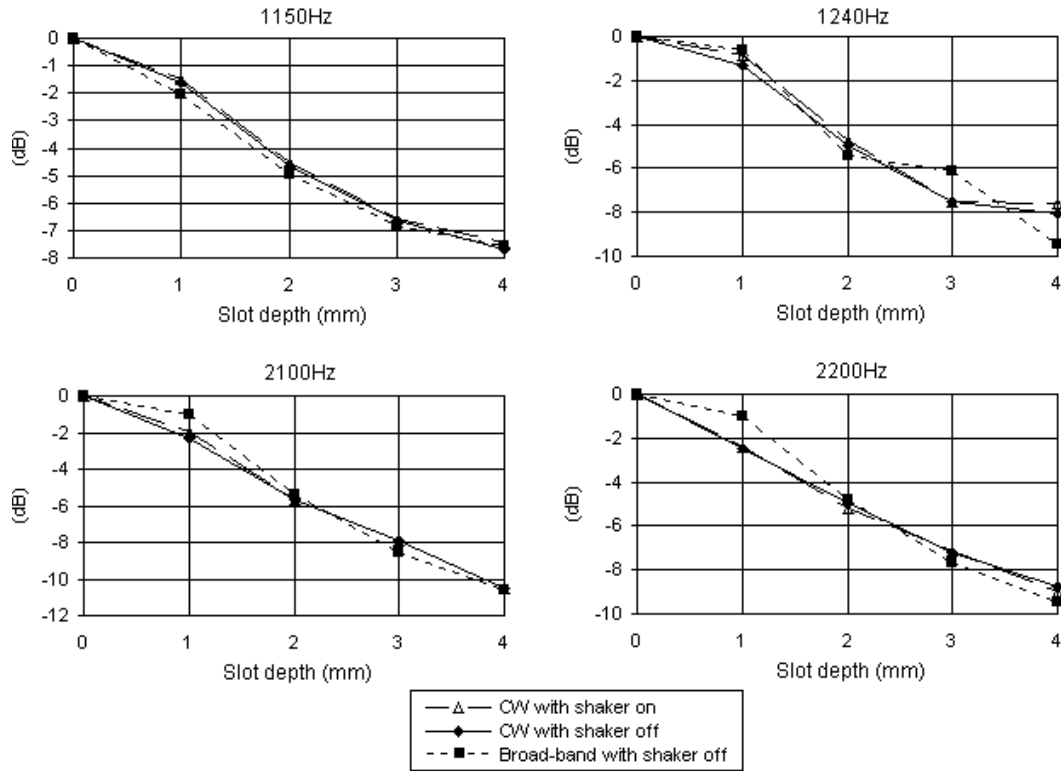


Figure 14. Comparison between results obtained by the broad band excitation with the shaker turned off and by CW signal with the shaker excited by a random-noise. Figure 1. Wavelets correlation coefficients.

5. Concluding Remarks

Two active piezoelectric health monitoring techniques were investigated : a transfer function analysis between piezoelectric sensors and actuators and a impedance-based method. Both methods detect the growth of defects due to changes in its neighborhood and demonstrated potential applications. The first method, has been evaluated through experiments in beams with machined slots or fatigue cracks. Results have shown that when the piezoelectric sensor is placed close to the defect, its response to a single frequency excitation by the actuator experiences a substantial reduction in amplitude as the defect depth increases. Experiments have shown that the good sensibility of the method verified in tests with slotted beams, might also be achieved when monitoring the growth of fatigue cracks under tensile stresses that will keep them open. Finite element simulations of slotted beams were also performed, and their results have indicated that the response to damage growth may be increased by employing small sensors placed very close to the defect. Tests in a framed structure have also demonstrated the good potential of the method to applications in more complex structures.

The impedance-based technique was preliminarily evaluated through tests in an aluminum beam with machined slots. The results shown that the method has potential applications and that at high frequencies increases the sensitivity of the admittance variations due to the defect size. Additional tests and simulations are being implemented by the authors in order to correlate the admittance variations with the defect size.

In spite of these encouraging results, there are still a number of issues to be addressed before applying the proposed health monitoring methodology to real structures. Further investigation must be carried out in order to develop techniques to account for variations in the sensor response not correlated with damage growth. These variations may be due to temperature changes or modifications in the structure's dynamic response. As a future improvement to be implemented, the authors are investigating the use of piezoelectric sensors installed at locations not affected by the damage growth, to compensate for temperature variations and other factors. The use of various sensors, one of them placed close to the defect, could also be explored in another method under investigation by the authors. In this case, there would be no need for the actuator, and the correlation between the response of the various sensors to the external dynamic excitation, would provide the means to monitoring the growth of an existing defect.

6. Acknowledgements

The authors are grateful for the support provided by the Brazilian Ministry of Science and Technology, through grants awarded by its agencies CNPq and FINEP. They also acknowledge support from the Brazilian National Regulatory Petroleum Agency (ANP).

7. Reference list

- Ayres, J.W., "Qualitative impedance-based health monitoring of civil infrastructures", *Smart Mater. Struct.*, Vol. 7 (1998), pp. 599-605.
- Banks, H. T., Inman, D. J., D. J. Leo, and Wang, Y., "An experimentally validated damage detection theory in smart structures," *J. Sound and Vibration*, Vol. 191, pp. 859-880, 1996.
- Braga, A. M. B., de Barros L. P. F., and Gama, A. L., "Models for the high frequency response of active piezoelectric composite beams," *Applied Mechanics and Engineering*, Vol. 5, pp. 47-61, 2000.
- Cawley, P., and Adams, D., "The location of defects in structures from measurement of natural frequencies," *J. Strain Analysis*, Vol. 14(2), pp. 49-57, 1979.
- Cawley, P., and Ray, R., "A comparison of the natural frequency changes produced by cracks and slots," *ASME J. Vibration, Acoustics, Stress, and Reliability in Design*, Vol. 110, pp. 366-370, 1988.
- Chiu, W.K., Galea, S.C., Koss, L.L. and Rajic, N., "Damage detection in bonded repairs using piezoceramics", *Smart Mater. Struct.*, Vol. 9 (2000), pp. 466-475.
- Dimarogonas, A. D., "Vibration of cracked structures: a state of the art review," *Engng Fracture Mech*, Vol. 55, pp. 831-857, 1996.
- Doebling, S. W., Farrar, C. R., Prime, M. B. and Shultz, D. W., "Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review", Report LA-13070-MS, Los Alamos National Laboratory, Los Alamos, NM, 1995.
- Liang, C., Sun, F. P., and Rogers, C. A., "An impedance method for dynamic analysis of active material systems," *ASME J. Vibration and Acoustics*, Vol. 116, pp. 120-128, 1994.
- Liang, C., Sun, F. P., and Rogers, C. A., "Electro-mechanical impedance modeling of active material systems", *Smart Mater. Struct.*, Vol. 5 (1996), pp. 171-186.
- Otegui, J. L., Mohaupt, U. H., and Burns, D. J., "A strain gage technique for monitoring small fatigue cracks in welds," *Engng. Fracture Mechanics*, Vol. 40(3), pp. 549-569, 1991.
- Pines, D. J., "The use of wave propagation models for structural damage identification," *Proceedings of the International Workshop on Structural Health Monitoring*, Stanford, CA, pp. 665-677, 1997.
- Robbins, D. H. and Reddy, J. N., "Finite element analysis of piezoelectric actuated beams," *Comput. & Struct.*, Vol. 41, pp. 147-169, 1991.
- Salawu, O. S., "Detection of structural damage through changes in frequency: a review," *Engng Struct*, Vol. 19, pp. 718-723, 1997.
- Verreman, Y., "Monitoring short fatigue cracks with miniature strain-gages," *Experimental Mechanics*, Vol. 21, pp. 208-216, 1994.
- Zou, Y., Tong L., and Steven, G. P., "Vibration-based model-dependent damage (delamination) identification and health monitoring for composite structures — a review," *J. Sound and Vibration*, Vol. 230(2), pp. 357-378, 2000.

THERMAL ALARM USING A SHAPE MEMORY ALLOY HELICAL SPRING

Carlos José de Araújo**Edgard Augusto da Silva Rocha****Cícero da Rocha Souto**

Departamento de Engenharia Mecânica

Universidade Federal da Paraíba – Campus II

Caixa Postal: 10069 Cep: 58109-970

Campina Grande – PB

Carlos@dem.ufpb.br

César Henrique Gonzalez

GEMPPM, UMR CNRS 5510, INSA de Lyon

20, Avenue Albert Einstein – Bat 502

F-69621, Villeurbanne - France

Abstract: This work describes the fabrication of helical springs using Ti-Ni shape memory wires and its application as thermomechanical actuator in a small sound thermal alarm. For this one, a coil spring having about 8mm mean diameter and 8 active coils was manufactured using a 0.66mm equiatomic Ti-Ni wire. After a particular training procedure, a deflection of about 1mm in the expansion sense was measured with the obtained coil spring heated above 65°C without any applied load. The thermomechanical characterization has demonstrated that an external load improves this deflection by two-way memory effect (TWME) and the maximum measured displacement was 5mm under an optimal load of 0.4N. This SMA coil spring was used to design first prototype of a small sound thermal alarm demonstrating the potential of these actuators generating force and displacement by thermal energy conversion.

Keywords: Shape memory alloys, helical springs, actuators, thermal alarm.

1. Introduction

Shape memory alloys (SMA's) are very interesting "smart" metallic materials that can respond to a temperature change when are placed in a previously strained state. Therefore, these SMA's intrinsically demonstrates sensor and/or actuator functions potentially applicable in many engineering fields (Van Humbeeck, 1999; Duerig *et al.*, 1999). These thermomechanical sensor/actuation functions are associated to a solid-state thermoelastic martensitic transformation involving formation (by cooling or loading) and reversion (by heating or unloading) of martensite variants into the material as a function of the temperature and/or mechanical loading (Funakubo, 1987; Otsuka and Wayman, 1998). SMA's are characterized by a low temperature phase, named martensite, and by a high temperature phase, named austenite.

In fact, there are basically four operation modes regarding the shape memory effect (SME) phenomena (Patoor and Berveiller, 1994):

- *One Way Memory Effect (OWME)*: occurring when the SMA is placed at the low temperature martensite phase ($T < M_f$, martensite finish), plastically strained by a loading/unloading mechanical procedure and heated to form the austenite phase ($T > A_f$, austenite finish) reverting to its original shape in absence of external loads. Afterwards, if the SMA is submitted to an another thermal cycle under stress free condition, there is no more motion. To repeat a OWME cycle is always necessary a deformation of the SMA;
- *Stress Assisted Two Way Memory Effect (SATWME)*: in this case, the SMA is strained by a mechanical load that remains during the heating-cooling procedure ($M_f > T > A_f$). Therefore, a displacement under load (external work) can be obtained or, if the SMA is constrained to prevent motion, a mechanical effort is generated. Considering that the applied load is always present, the SATWME can be repeated many times;
- *Two Way Memory Effect (TWME)*: is associated to the spontaneous shape change of the SMA, in absence of mechanical loading, when it is alternately heated and cooled. This phenomenon demand a previous special thermomechanical treatment of the SMA, called "training" (Perkins and Hodgson, 1990);
- *Superelasticity (SE)*: this behavior can be considered a mechanical SME because it involves formation and reversion of the martensite phase by mechanical loading. If the SMA is placed in the austenitic phase ($T > A_f$) and loaded, there will be formation of martensite variants when a critical stress is exceeded and its reversion occurs when the SMA is unloaded. The term superelasticity is used because in this case reversible non-linear strains of the order of 10% can be easily obtained.

Each one of the above described SME characteristics has found a variety of applications (Van Humbeeck, 1999; Duerig *et al.*, 1999). For example, OWME has been successfully used to develop thermomechanical SMA couplings and fasteners to connect tubings (Proft and Duerig, 1990). The SATWME and TWME properties has been employed to design “two-way SMA actuators” that finds application in robotics, thermal protection devices, thermally actuated electric switches, thermostatic mixing valves, air flow control for air conditioners and many others (Otsuka and Wayman, 1998). SE has mainly being used for medical and dental applications (Duerig *et al.*, 1999). Raynaerts and Van Brussel (1998) gives general design guidelines for building high performance shape memory actuators including vantages and drawbacks in comparison with classical technologies.

There exist two main groups of SMA’s commercially available today: copper based alloys (Cu-Zn-Al and Cu-Al-Ni) and Ti-Ni based alloys (sometimes called Nitinol). Ti-Ni based SMA’s are more expensive than Cu-based alloys but present best shape memory properties, fatigue and corrosion resistance. In addition, Ti-Ni based alloys is the main material used to design robotic actuators due to its larger resistivity that allow electrical resistive heating by applying smaller currents.

In Brazil, even considering laboratory scale, very few applications of SMA actuators has been realized. Recently, Nunes *et al.* (1999) has designed a holddown/release device prototype for satellite solar panels to be used in the missions using the framework of the scientific satellite SACII. Da Silva and Mesquita (2000) has studied an adaptive flexible beam actuated by SMA aiming to control actively the vibration level of the beam.

In this sense, the main goal of this work is to verify the possibility of manufacture SMA helical springs by a simple thermomechanical procedure and to design a small sound thermal alarm with this element demonstrating the potential of these actuators generating force and displacement by thermal energy conversion.

2. Designing the SMA spring: theoretical analysis

Among the possibilities concerning SMA actuators, a coil spring is sometimes preferable since this form makes possible a large stroke in comparison with the shape recovery of a straight wire (Otsuka and Wayman, 1998). The design of SMA springs is defined by the force and deflection characteristics required, as well as by the space available for the particular device. The basic dimensions of a spring are the coil diameter (D), the wire diameter (d) and the number of active coils (N). In our case, a Ti-Ni equiatomic wire having a diameter of about 0,66mm was selected. In addition, it is assumed that a small thermomechanical device can be designed with a compression spring having D=8mm and a displacement by SATWME of the order of 5mm.

The design of SMA springs employs the same equations of conventional spring design (Shigley and Mischke, 1989). The fundamental equations are:

- Deflection (δ):
$$\delta = \frac{8PD^3 N}{Gd^4} \quad (2.1)$$

- Shear strain (γ):
$$\gamma = \frac{\delta \cdot d}{\pi \cdot N \cdot D^2} \quad (2.2)$$

- Shear stress (τ):
$$\tau = \frac{8PDK}{\pi \cdot d^3} \quad (2.3)$$

- Stress correction factor (K) using Wahl’s equation:
$$K = \frac{4C - 1}{4C - 4} + \frac{0,615}{C} \quad (2.4)$$

- Stress-strain relationship:
$$\tau = G \cdot \gamma \quad (2.5)$$

where the others undefined parameters are:

- P – load on spring;
- G – shear modulus;
- C – spring index ($C=D/d$)

The characteristic feature of SMA’s is the change in G modulus as the material is heated above and cooled below the phase change transformation temperatures. Therefore, three main parameters are required to estimate the deflection and force capacities of the spring: the maximum shear stress (τ_{max}), the shear modulus at the high temperature (G_H) and the shear modulus of the SMA at the low temperature (G_L). Typical values for these parameters for commercial Ti-Ni SMA’s are: $\tau_{max}=120\text{MPa}$, $G_L=8\text{GPa}$ and $G_H=23\text{GPa}$ (Otsuka and Wayman, 1998). It should be noted that if the maximum shear strain γ_{max} is imposed, the operating strain of the SMA spring must be kept within γ_{max} . For a good fatigue life, it is recommended $\gamma_{max} \leq 1\%$.

With the previously established geometrical parameters, the spring index and the stress correction factor is calculated according to equation (2.4): $C=12,1$ and $K=1,12$. The next step is the calculation of the number of active coils (N) by equation 2.2. Figure 1 illustrate that the stroke length δ is equal to the difference between the deflection at high and low temperatures

($\delta_L - \delta_H$). Limiting $\gamma_L = \gamma_{max} = 0,75\%$ (Tsuji *et al.*, 1991), the shear strain γ_δ associated to the stroke length δ during heating can be calculated as follow:

$$\gamma_\delta = \gamma_{max} - \frac{\tau_{max}}{G_H} = 0,0075 - \frac{120}{23000} = 0,0023$$

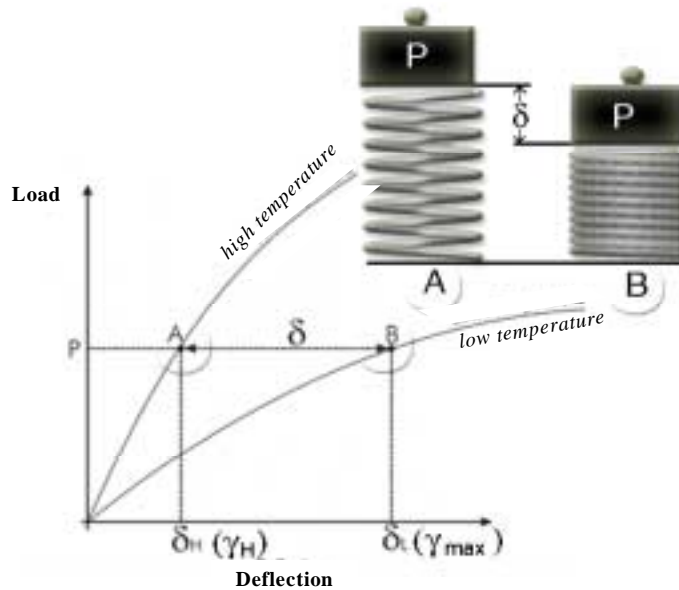


Figure 1 – Illustration of the load-deflection behavior of the low and high temperature phases in SMA springs.

The number of active coils can be obtained from equation (2.2): $N = \frac{\delta \cdot d}{\pi \cdot \gamma_\delta \cdot D^2}$. For $\delta=5\text{mm}$, $d=0,66\text{mm}$, $\gamma_\delta=0,0023$

and $D=8\text{mm}$: $N=7,8 \approx 8$ active coils.

The load capacity (P) of the coil spring can be estimated by substituting equations (2.2) and (2.3) in equation (2.5):

$P = \frac{\delta \cdot d^4 \cdot G}{8 \cdot N \cdot D^3 \cdot K}$. Replacing G by G_H in this equation leads to $P=0,59N \approx 61\text{g}$. In conclusion, theoretically a coil SMA compression spring with $d=0,66\text{mm}$, $D=8\text{mm}$ and $N=8$ active coils can elevate of 5mm by heating a mass of about 60g.

3. Making and training the SMA spring

The method of making SMA springs is chosen largely by the quantity required. If only one or a few springs are needed, it is easiest to wind the chosen wire on a cylindrical mandrel having a diameter close to the internal diameter of the spring (Otsuka and Wayman, 1998). In our case, an 18UNF – 5/16” screw was employed as cylindrical mandrel and both ends of the SMA wire were fastened to this one. The heat treatment chosen to set the shape and properties of the spring was an annealing at 450°C for 3 minutes followed by rapid cooling via water quench. Figure 2 illustrate schematically the procedure above described and figure 3 shows the obtained SMA spring already cut off to about 8 active coils.

The dimensions of the spring are 11mm height, 8mm mean diameter and a mass of about 400mg. The SMA spring was trained employing a thermal cycling under constant load procedure (De Araújo, 1999). In this way, the SMA spring showed in figure 3 was introduced into a cylindrical rod and submitted to a mass of about 75g, which is slightly higher than the load capacity theoretically estimated in the previous section. This set was thermal cycled 100 times between a hot bath (boiling water) and an another cool bath (water at ~25°C). After this procedure and when heated, the trained SMA spring has demonstrated an expansion without applied load (δ for $P=0$ in figure 1) of the order of 1mm, corresponding to a TWME of about 10%.

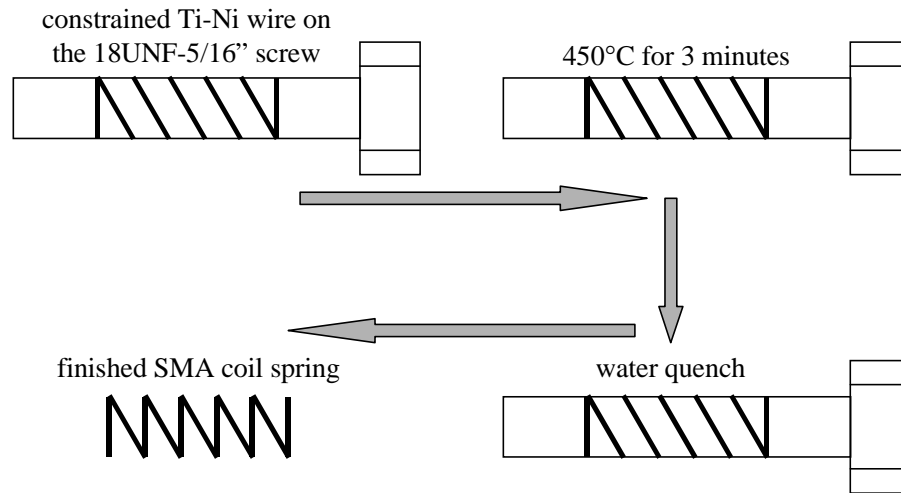


Figure 2 – Making the SMA coil spring



Figure 3 – Picture of the SMA coil spring obtained by the thermomechanical procedure indicated in figure 2.

4. Thermomechanical characterization

The transformation temperatures of the SMA spring, after the thermomechanical procedure described in figure 2, were measured using a Differential Scanning Calorimeter (DSC) Mettler TA3000. The sample for this test was a piece of about 11mg cut off of the SMA spring. Figure 4 reveals the two peaks associated to the reversible martensitic transformation during heating and cooling. It can be verified in figure 4(a) that for temperatures higher than $A_f=57^\circ\text{C}$ the SMA spring is fully transformed from martensite to austenite. During cooling, figure 4(b) reveals that reversion of this transformation starts at $M_s=56^\circ\text{C}$ which is very close to the austenite finish temperature A_f . Additionally, a complete SATWME (δ in figure 1) was observed when the trained SMA spring was loaded by a constant mass of 75g and heated at about 65°C (A_f under load) using a hot plate. This difference between the A_f temperature obtained by DSC and the one measured with the SMA spring under load is mainly due to the fact that the transformation temperatures are increased by the application of mechanical stresses following a modified Clausius-Clayperon law for SMA's (Funakubo, 1987).

After the training procedure described in the previous section, the thermomechanical response of the SMA coil spring under constant load was verified. For this one, the SMA coil spring loaded by growing weights was heated to above 65°C using a high power thermal air blower (max. 350°C). Figure 5 shows the load versus deflection behavior of the fully transformed SMA coil spring ($T > A_f$). The deflection by SATWME during heating was measured using a LVDT displacement sensor.

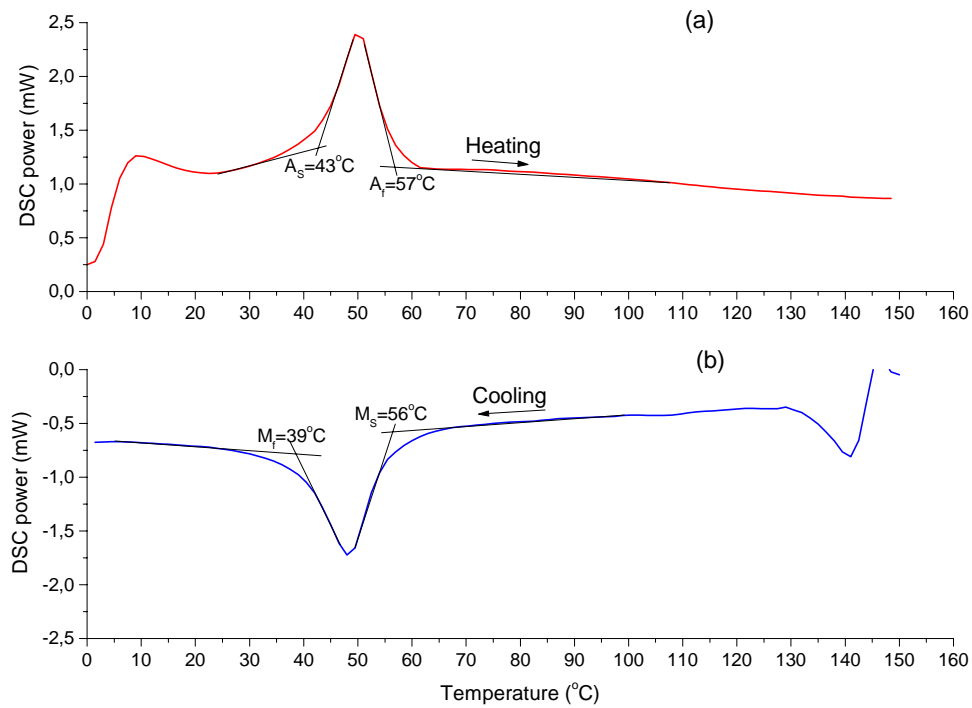


Figure 4 – DSC peaks for the heat-treated SMA (figure 2). (a) heating. (b) cooling. Cooling-heating rate: 15°C/min.

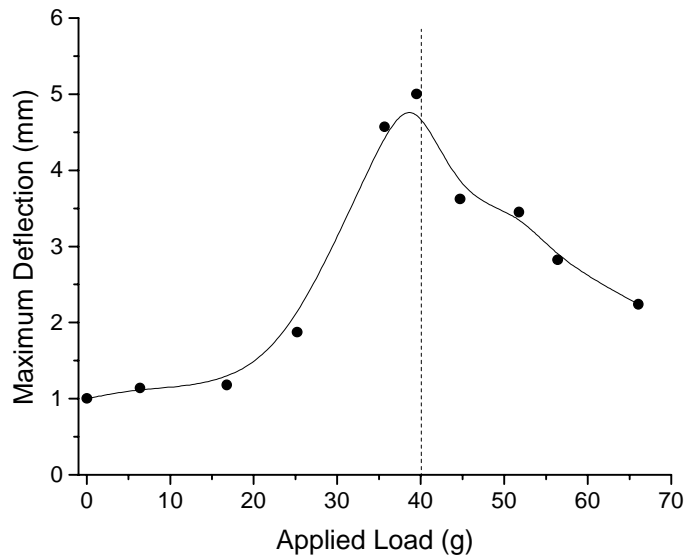


Figure 5 – Maximum deflection of the SMA coil spring by heating under applied load.

From the figure 5 it can be verified that the deflection of the SMA coil spring during heating is practically unaffected by applied loads until 15g, but a fast increase is observed between 20g and 40g. The best response of the SMA spring (~5mm) is obtained for a constant load corresponding to a mass of about 40g. Therefore, a SATWME of the order of 45% was measured for an optimum load of 0.4N. Figure 5 also reveals that applied loads higher than this optimum level leads to a loss of the SATWME capacity. It is very important to remark that the theoretical load capacity during heating was 60g, estimated for a displacement of 5mm (section 2). This difference between theory and experimentation can be attributed to the uncertainties in the thermomechanical properties used for theoretical calculations. An another feature that can contribute to this difference is a

small contraction of the SMA spring in the diameter sense during heating, causing friction between the inner diameter of the spring and the surface of the vertical rod used to guide the SMA spring, reducing its load capacity.

The main advantage of a SMA spring is that it generates a large macroscopic displacement out with a relatively small microscopic strain, however the stress distribution over the cross-section of the spring is not constant. This implies that a greater material volume is needed for generating the same force. Defining the energy density (ED) of an SMA actuator as follow:

$$ED = \frac{P \cdot \delta}{m_{SMA}} \tag{3.1}$$

where P is the load capacity, δ is the maximum displacement during heating (for $T > A_f$) and m_{SMA} is the mass of the SMA element.

By applying equation (3.1) for our SMA coil spring, an ED of about 5J/Kg is obtained. Obviously, using wires in uniaxial tension as an active element has the advantage of optimal use of the material because the stress distribution over the cross-section is constant. For comparison, equation 3.1 applied to the results obtained by De Araujo (1999) leads to ED values between 350 and 1350 kJ/kg, depending on the applied load. These values are in agreement with the ones presented by Reynaerts and Van Brussel (1998).

5. Thermal alarm

Based on the work capacity of the SMA coil spring demonstrated by figure 5, a first prototype of a small sound thermal alarm was designed to demonstrate the potential of this SMA sensor/actuator element. Figure 6(a) and 6(b) show respectively an expanded and a global view of this on/off device.

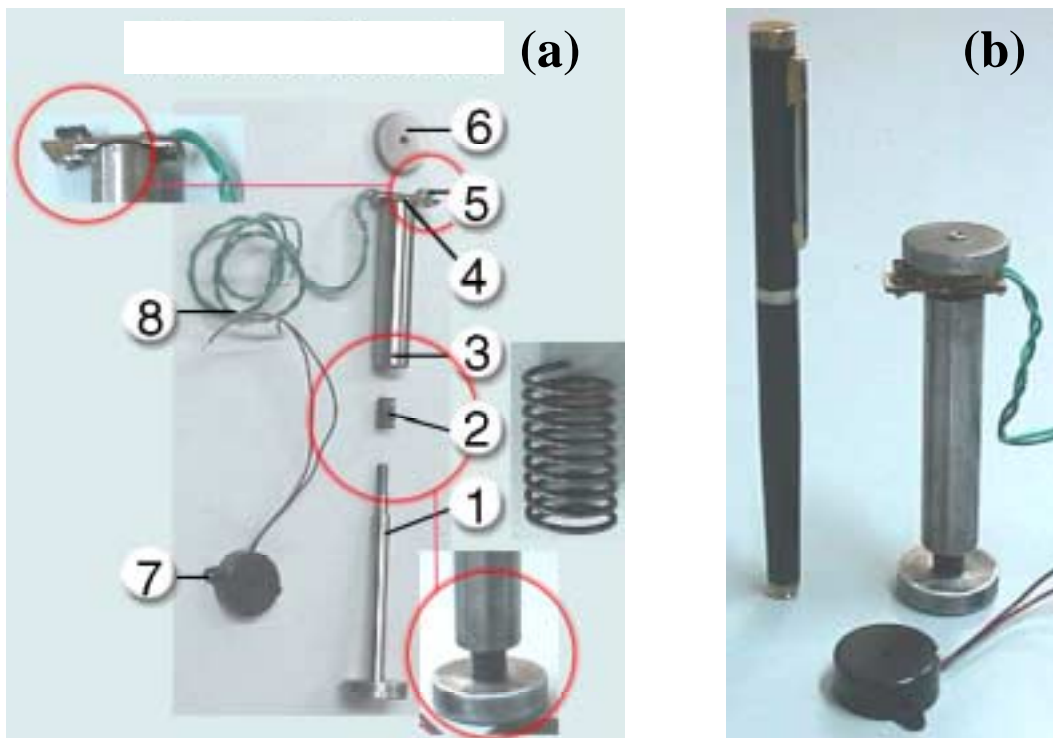


Figure 6 – First prototype of the thermal alarm using a SMA spring actuator. (a) expanded view and (b) general view.

In this device, the SMA coil spring (2) is placed in a steel guide rod (1) and compressed by a steel cylinder (3) having a weight equivalent to the optimum load determined in figure 5 (~0.4N). An electrical switch (5) separated by an insulating sheet (4) is mounted on the steel cylinder (3) and a flat cylindrical piece (6) is adjusted in the upper extremity of the steel guide rod (1). The electrical switch (5) is linked to a buzzer (7) by electrical wires (8). In this state, the small spring of the electrical switch (5) is slightly pre-stressed. In the operation mode, a hot air flow is addressed to the inferior region indicated in figure 6(a) causing the heating of the SMA spring. However, as the expansion (~5mm) by SATWME of the SMA spring is prevented by the flat cylindrical piece (6), a force is generated in the upper extremity of the device. This force is sufficient to activate the electrical switch and consequently the buzzer (7), which is energized using a 12V/6A power supply.

This device was satisfactorily tested in hot (room temperature $\sim 40^{\circ}\text{C}$) and acclimatized environments (room temperature $\sim 15^{\circ}\text{C}$). For this one, the SMA spring actuator was activated by heating it to above 65°C using a high power thermal air blower (max. 350°C). Cooling was naturally induced by natural convection. Obviously, operation in acclimatized environments leads to a larger number of on/off cycles, but measurements of the cooling and heating time is a work to be done.

6. Conclusions

In this work a SMA spring actuator was manufactured by a simple thermomechanical procedure that employs an 18UNF – 5/16” screw as cylindrical mandrel. The obtained SMA coil spring has 11mm height, 8 active coils, 8mm mean diameter and a mass of about 400mg. In absence of applied loads and after training, this element has demonstrated an expansion by TWME of the order of 1mm ($\sim 10\%$). By application of external loads, it was verified an optimization of the thermomechanical behavior of the SMA spring actuator which can elevate of 5mm by heating a mass of about 40g, corresponding to a SATWME of 45% and an energy density of 5J/kg. This SMA spring actuator was successfully used to design a first prototype of an on/off device operating as sound thermal alarm for environments where temperature is a critical parameter.

7. References

- Da Silva, E. P. and Mesquita, A. L. A., 2000. “Vibration Control of Flexible Beams Using Shape Memory Alloy Actuators” (in portuguese), Proceedings of the National Conference on Mechanical Engineering 2000, CD-ROM, Natal-RN, Brazil.
- De Araujo, C. J., 1999. “Comportement cyclique de fils en alliage à mémoire de forme Ti-Ni-Cu: analyse electro-thermomécanique, dégradation et fatigue par cyclage thermique sous contrainte”. Thèse de Doctorat: INSA de Lyon, France, 176p.
- Duerig, T., Pelton, A. and Stockel, D., 1999. “An Overview of Nitinol Medical Applications”, Materials Science Engineering A, Vol. A273-275, pp. 149-160.
- Funakubo, H., 1987. “Shape Memory Alloys”, Gordon and Breach Publishers, London, 273p.
- Otsuka, K. and Wayman, C.M., 1998. “Shape Memory Materials”, Cambridge University Press, Cambridge, UK, 284p.
- Nunes, A. L. L. V., Almeida, S. F. M. and Palerosi, A. C., 1999. “Design of a Holddown/Release Device for the Solar Panels of the Scientific Satellite SACII Using a Shape Memory Alloy Actuator” (in portuguese), Proceedings of the 15th Brazilian Congress of Mechanical Engineering, CD-ROM, Águas de Lindóia-SP, Brazil.
- Patoor, E. and Berveiller, M., 1994. “Technologie des Alliages à Mémoire de Forme”, Hermès, Paris, France, 286p.
- Proft, J. L. and Duerig, T. W., 1990. in “Engineering Aspects of Shape Memory Alloys”, Edited by T. W. Duerig *et al.*, Butterworth-Heinemann, UK, pp. 115-129.
- Perkins, J. and Hodgson, D., 1990. “The Two-Way Shape Memory Effect”, in *Engineering aspects of shape memory alloys*. Edited by T. W. Duerig. London: Butterworth-Heinemann, 195-206.
- Reynaerts, D. and Van Brussel, H., 1998. “Design Aspects of Shape Memory Actuators”, Mechatronics, Vol. 8, pp. 635-656.
- Shigley, E. S. and Mischke, C. R., 1989. “Mechanical Engineering Design”, McGraw-Hill International Editions, Singapore, 779p.
- Tsuji, K., Takegawa, Y. and Ito, T., 1991. “Phase Transformation Property of Ni-Ti-Cu Alloy Coil Under an Applied Stress”, Materials Science and Engineering A, Vol. A142, pp. 203-208.
- Van Humbeeck, J., 1999. “Non-Medical Applications of Shape Memory Alloys”, Materials Science Engineering A, Vol. A273-275, pp. 134-148.

PROJETO DE CONTROLE DE UM SISTEMA DE POSICIONAMENTO PNEUMÁTICO ROTATIVO**Marcos Costa Hunold**

Escola Politécnica da Universidade de São Paulo
Departamento de Sistemas Mecatrônicos
Avenida Mello Moraes, 2231
mahunold@usp.br

Eduardo Lobo Lustosa Cabral

Escola Politécnica da Universidade de São Paulo
Departamento de Sistemas Mecatrônicos
Avenida Mello Moraes, 2231
elcabral@usp.br

Resumo. Este trabalho apresenta o projeto de controle de um sistema de posicionamento pneumático rotativo que é implementado em uma bancada experimental. Este sistema, devido às suas características de precisão e rigidez, pode ser utilizado em sistemas de acionamento de robôs e de máquinas CNC. O sistema de posicionamento é constituído por um motor pneumático de palhetas, um redutor planetário a tração e uma válvula pneumática proporcional de 4 vias. Um controlador digital PD (proporcional-derivativo) convencional controla o orifício de abertura da válvula. O comportamento dinâmico e em regime permanente do sistema de posicionamento com a válvula de controle são estudados e um modelo matemático é proposto e utilizado no projeto do controle do sistema. O projeto de controle do sistema de posicionamento é elaborado com controladores digitais convencionais do tipo PD (proporcional-derivativo), PID (proporcional-integral-derivativo) e de realimentação de estados. Estes controladores são implementados e testados. Resultados mostram que o controlador de realimentação de estados apresenta o menor erro de posicionamento, embora a sua resposta dinâmica não seja adequada. Assim, conclui-se que o sistema de posicionamento pneumático rotativo possui características de precisão adequadas, havendo necessidade de melhorar a resposta transitória do sistema.

Palavras Chaves: *motores pneumáticos, sistemas pneumáticos rotativos, controle de posição.*

1. Introdução

Atualmente, em sistemas de posicionamento, são utilizadas três tecnologias de acionamento: a pneumática, a elétrica e a hidráulica. Usualmente, as aplicações da área pneumática são frequentes onde não são exigidos grandes esforços, com posicionamentos de baixa precisão e com movimentos lineares ou rotacionais parciais. Assim, os acionadores que posicionam os braços robóticos e as mesas de máquinas de controle numérico com alta precisão são, invariavelmente, elétricos ou hidráulicos. No entanto, há diversos estudos (Hunold, 1999; Pandiam et al, 1999; Mahgoub e Craighead, 1996 e Pu et al, 1991) que apontam o motor pneumático como atuador de sistemas de posicionamento pois, ao contrário dos motores elétricos, o motor pneumático tem um arranjo mecânico simples e compacto, pode trabalhar travado, em áreas de risco de explosão e “sujas” e apresenta alta relação potência-peso e potência-tamanho. Porém, não na sua forma atual pois o motor pneumático possui limitações sérias quanto à rigidez devido ao efeito de compressibilidade do ar e possui não-linearidades como o atrito. O problema de rigidez é solucionado com a associação do motor pneumático a um redutor que possua alta rigidez e alta relação torque-inércia ou torque-peso, conforme descrito em Hunold (1999). Deste modo, o sistema de posicionamento pneumático rotativo aqui proposto pode ser aplicado no acionamento de robôs e máquinas de comando numérico. Este sistema é constituído por uma válvula pneumática proporcional de 4 vias, um motor pneumático de palhetas e um redutor planetário a tração, conforme apresentado na Fig. (1). O conjunto formado pelo motor-redutor é denominado como MPAR (Motor Pneumático de Alta Rigidez).

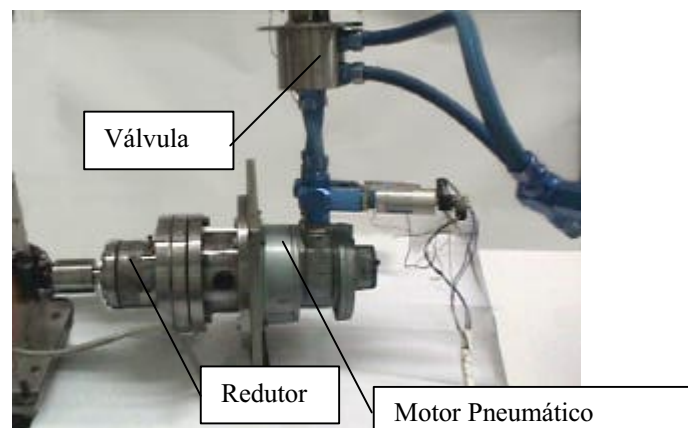


Figura 1. Foto do sistema experimental MPAR-Válvula.

O redutor planetário a tração utiliza como princípio ativo na transmissão de torque e de velocidade, o atrito entre duas superfícies metálicas, no lugar dos dentes de engrenagens. Com isto, este redutor não apresenta folga, é compacto e a sua rigidez é maior do que a de um redutor convencional. A válvula pneumática proporcional é projetada especialmente para este sistema, possuindo uma concepção inovadora, pois tem um único estágio de atuação formado por um elemento deslizante tipo disco, que facilita a sua operação e, principalmente, porque permite orifícios de pressão e de escape de áreas diferentes. Segundo Gouveia (1996), esta última característica permite que a válvula aumente a eficiência de qualquer carga associada à mesma. Finalmente, a tecnologia atual de controle digital, integrada à instrumentação utilizada, fornece ao sistema flexibilidade na escolha da melhor estratégia de controle, além de permitir uma alta precisão de posicionamento mesmo diante do atrito existente no motor pneumático. O trabalho apresenta a seguir, o projeto de controladores para o sistema de posicionamento descrito.

2. Projeto de controle do sistema de posicionamento

A estratégia de controle adotada para o sistema MPAR-Válvula é representada no diagrama de blocos da Fig. (2). Como se observa, o sistema de controle opera em malha fechada para posicionar a válvula, sendo que o valor de referência deste controle é gerado pelo controlador da posição angular do eixo de saída do redutor. Para realizar o projeto do controlador do MPAR e do controlador da válvula são obtidos modelos lineares a partir dos modelos matemáticos desenvolvidos para o sistema MPAR e para a válvula, descritos em Hunold (1999). Os dois controladores são implementados em um microcomputador, que recebe os sinais dos sensores do sistema (sinais dos “encoders” e das pressões da entrada e saída do motor pneumático). O projeto destes controladores, bem como o funcionamento da válvula proporcional e do sistema MPAR-Válvula estão descritos a seguir.

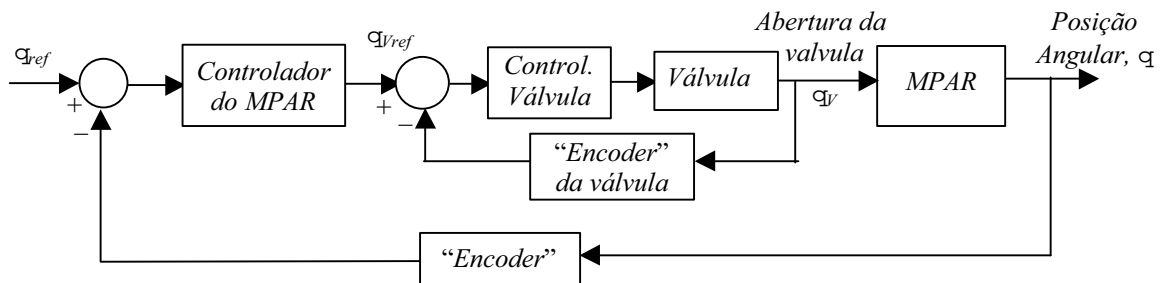


Figura 2. Diagrama de blocos da malha fechada de controle de posição do sistema MPAR-Válvula.

Ressalta-se que a entrada do sistema MPAR-Válvula é a posição do disco da válvula. Com isso, o sistema de controle de posicionamento da válvula e a própria dinâmica da válvula é incorporada ao MPAR como sendo a dinâmica do atuador do sistema. De acordo com os resultados apresentados a seguir, o tempo de resposta da válvula será bem menor do que o tempo de resposta do MPAR, que é da ordem de 75ms. Assim, a dinâmica da válvula não interfere com o sistema de controle do MPAR. Para melhor entendimento do projeto do controlador da válvula proporcional descreve-se a seguir o seu funcionamento e os componentes do seu sistema de acionamento. A válvula proporcional está representada na Fig. (3).

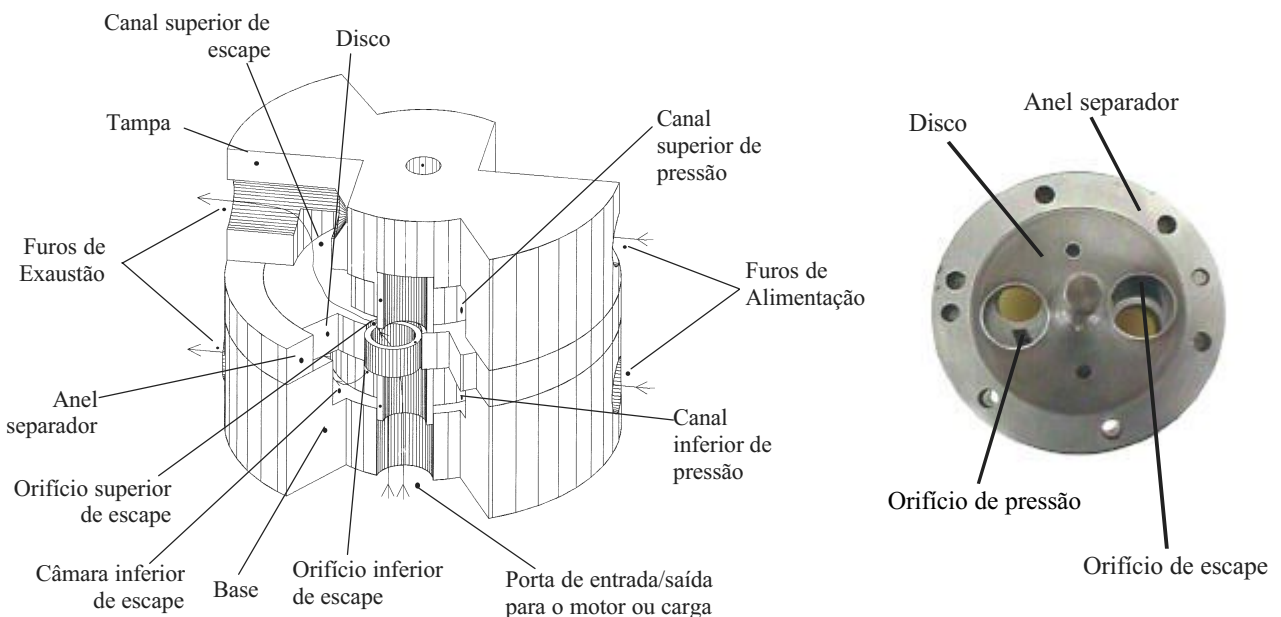
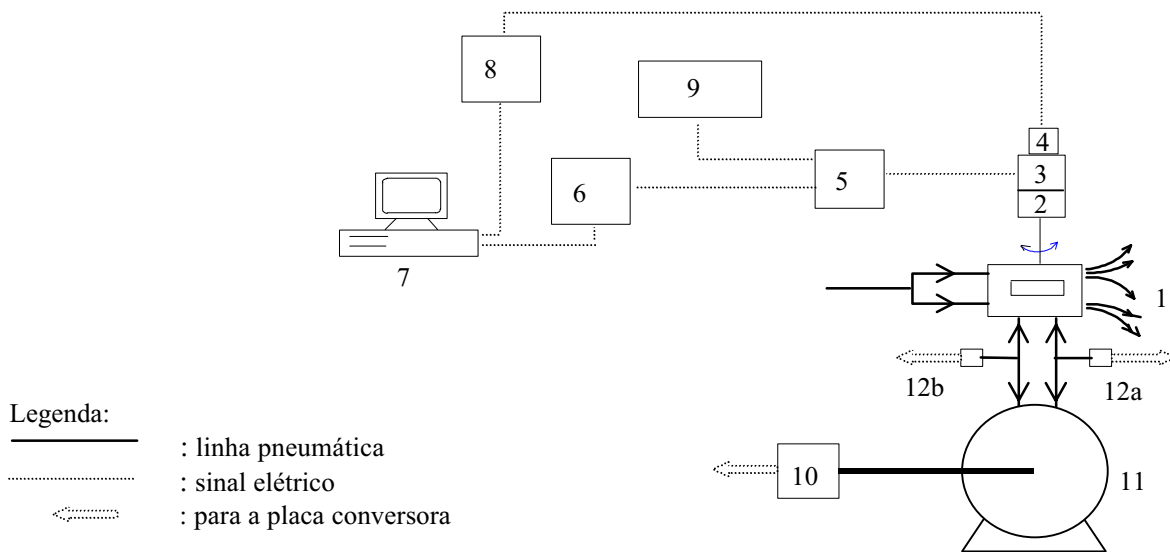


Figura 3. Desenho tridimensional em corte da válvula e foto com indicação dos orifícios de pressão e escape da válvula.

Através do desenho e da foto, verifica-se que o ar proveniente de um sistema de compressão é introduzido nos dois orifícios de entrada (furos de alimentação) da válvula, atingindo os canais de pressão (superior e inferior). De acordo com a posição angular do disco, o ar dos dois canais penetra no orifício de pressão alimentando o motor pneumático. O ar que retorna do motor, penetra, então, pelo orifício de escape alcançando os dois canais de escape. Em seguida, o ar é liberado para o ambiente, através dos orifícios (furos) de exaustão da válvula.

A posição angular do disco é determinada por um sistema de acionamento (atuador), que é acoplado ao eixo da válvula, o qual está preso ao disco, conforme apresentado na Fig. (4). Este sistema utiliza um motor CC com escovas, acoplado a um redutor de engrenagens para movimentar o eixo preso ao disco. O motor CC é alimentado através de um sinal de tensão pulsante proveniente de um acionador PWM, que por sua vez recebe um sinal de tensão vindo de uma placa de aquisição de dados, a qual está inserida no microcomputador. A alimentação do acionador PWM é realizada por uma fonte CC. Um "encoder", acoplado ao eixo do motor CC, envia pulsos para um circuito contador cuja contagem, na forma de um sinal digital de 16 bits, é enviada a placa de aquisição. Este sinal digital fornece a posição angular do eixo da válvula e portanto, a área de abertura dos orifícios da válvula. Tanto este sinal, como o sinal de tensão enviado para o motor são manipulados a partir de um programa instalado no microcomputador, que realiza o controle da posição angular do eixo de saída do redutor, e conseqüentemente, o controle da abertura dos orifícios da válvula. Ressalta-se que o tempo de amostragem utilizado na aquisição de dados, bem como no controle do sistema de posicionamento é de 2ms.



Descrição dos elementos da bancada:

- | | |
|-------------------------------------|---|
| 1- válvula proporcional; | 7- microcomputador; |
| 2- redutor de engrenagens; | 8- circuito contador dos sinais do "encoder"; |
| 3- motor de corrente contínua (CC); | 9- fonte de alimentação CC; |
| 4- "encoder"; | 10- "encoder" do motor pneumático; |
| 5- acionador PWM; | 11- conjunto motor pneumático-redutor; |
| 6- placa conversora A/D e D/A; | 12- sensores de pressão. |

Figura 4. Esquema simplificado da bancada experimental.

O projeto do controlador da válvula é baseado na técnica da resposta temporal do sistema de acionamento, utilizando o método de posicionamento de pólos através do lugar das raízes, descrito em Ogata (1987). Este projeto tem a função de fornecer um valor inicial para os parâmetros do controlador, pois existem não linearidades que alteram a resposta real do sistema. Para aplicar a técnica do lugar das raízes, é necessário determinar um modelo linear do sistema de acionamento e a partir deste, a função de transferência em tempo discreto que relaciona a palavra digital da saída do microcomputador e a posição angular do "encoder". A Fig. (5) representa as conexões dos elementos que constituem o modelo matemático.

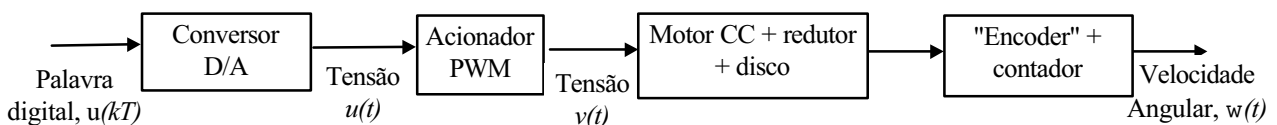


Figura 5. Diagrama de blocos dos elementos do modelo matemático do sistema de acionamento da válvula.

O modelo final é a versão linear de um modelo não-linear do sistema de acionamento. Este último é determinado através das relações físicas e constitutivas do motor CC, redutor, disco e demais elementos. A versão linear é representada por uma equação de diferenças cujos parâmetros são determinados a partir de um processo de identificação de sistemas. Este processo utiliza o modelo ARMAX associado ao método de estimativa dos mínimos quadrados, apresentado em Ljung (1985). Para realizar o cálculo dos parâmetros da equação de diferenças é necessário utilizar dados da velocidade angular do eixo do motor CC, em função de transientes de variação em degrau da entrada $u(kT)$, realizados na bancada experimental.

Com o modelo elaborado, procede-se pelo projeto do controlador, optando-se por um controlador PD (Proporcional Derivativo). Acrescenta-se ao controlador PD um pólo para filtrar dinâmicas de altas frequências, que, em função da parte derivativa do controlador, poderiam instabilizar o sistema em malha fechada. O projeto do controlador, isto é, a determinação do ganho, do zero e do pólo do controlador PD, foi feito segundo a análise da resposta temporal para uma entrada degrau de tensão aplicada sobre o acionador de potência do motor CC. Como característica do transitório verificou-se a necessidade de uma resposta com um ligeiro sobressinal e o menor tempo de subida possível. Além disto o posicionamento foi realizado com o menor erro de regime. A Fig. (6) apresenta os resultados de um transitório experimental e da simulação do modelo em malha fechada da válvula com controlador, para um degrau na referência de posição da válvula. No caso, é dado um degrau de 0 para $7,2^\circ$ na referência. Como se verifica, o sistema em malha fechada fornece uma resposta dentro das especificações propostas: o erro de regime, não considerando a folga do redutor, para este transitório é igual a 1,44 minutos de arco, o tempo de subida está em torno de 15ms e o sobressinal é cerca de 10%. Ressalta-se que, foram feitos diversos transitórios e que houve repetibilidade de resultados, dentro da faixa de erro de regime especificada. Com estas características a válvula com o seu controlador em malha fechada está apta para ser o atuador do MPAR.

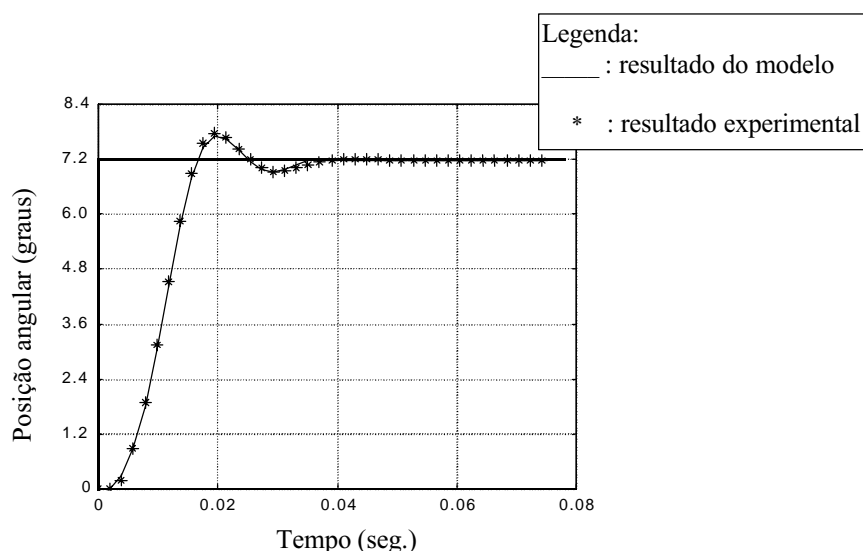


Figura 6. Gráfico da posição angular da abertura da válvula para um transiente de posicionamento com referência igual a $7,2^\circ$.

Com o controle da válvula funcionando, o controlador do MPAR da Fig. (2) é projetado baseado na resposta temporal do sistema para uma entrada na forma de degrau. Para tanto, como proposta para desempenho do sistema em malha fechada, especificam-se as seguintes condições de projeto:

- Erro máximo de posicionamento de 10 minutos de arco;
- Repetibilidade da mesma ordem do erro de posicionamento;
- Transitório com tempo de assentamento da ordem de 200ms;
- Sobressinal máximo de 15%.

Para realizar o projeto dos controladores do sistema MPAR-Válvula é necessário obter um modelo linear, a partir da linearização e posterior discretização do modelo físico. Este modelo é baseado nas equações de balanço de massa e de momento sobre o sistema MPAR e das equações do escoamento do ar através da entrada e saída do motor pneumático, considerando-se a válvula pneumática. Neste modelo, as variáveis de estado são a pressão de entrada e de saída do motor, a velocidade angular e a posição angular do eixo de saída do redutor.

O modelo linear elaborado para o sistema tem como entrada o valor de referência da abertura da válvula e como saída a posição angular do eixo de saída do MPAR. O mesmo método de identificação de sistemas aplicado no modelo da válvula é utilizado para determinar os coeficientes do modelo linear. O método é aplicado em cada uma das equações do modelo linear, descritas a seguir.

Neste modelo, a dinâmica da válvula não é considerada explicitamente. A dinâmica da válvula pode ser desprezada porque é muito mais rápida do que a dinâmica do MPAR, como citado anteriormente. Além disso, o modelo do sistema, sem considerar a válvula é muito mais simples e fácil de ser tratado.

Considera-se ainda um atraso de 3 intervalos de amostragem para o sistema, existente na relação entre o valor de referência da posição angular da válvula e as pressões de entrada e saída do motor pneumático, em função de observações experimentais dos transientes do sistema MPAR-Válvula. Com estas premissas, as equações lineares em tempo discreto do modelo são dadas por:

$$P_1(kT + T) = a_1 P_1(kT) + a_2 \Omega(kT) + b_1 \varphi_{ref}(kT - 3T); \tag{1}$$

$$P_2(kT + T) = a_3 P_2(kT) + a_4 \Omega(kT) + b_2 \varphi_{ref}(kT - 3T); \tag{2}$$

$$\Omega(kT + T) = a_5 \Omega(kT) + a_6 P_1(kT) + a_7 P_2(kT); \tag{3}$$

$$\varphi(kT + T) = \varphi(kT) + T \Omega(kT). \tag{4}$$

onde φ_{ref} é o valor de referência da posição angular da válvula, P_1 e P_2 são as pressões de entrada e saída do motor, Ω é a velocidade angular do eixo de saída do redutor, φ é a posição angular do eixo de saída, T é o tempo de amostragem e $a_1, a_2, a_3, a_4, a_5, a_6, a_7, b_1$ e b_2 são os coeficientes das equações a serem identificados. A Tab. (1) apresenta os valores dos coeficientes das equações (1), (2) e (3) calculados a partir do método de identificação mencionado anteriormente.

Tabela 1. Coeficientes do modelo linear em tempo discreto do sistema MPAR-Válvula.

Coeficiente	Valor
a_1	0,9053
a_2	$0,6661 \times 10^{-6}$
a_3	0,0532
a_4	$-0,2297 \times 10^{-5}$
a_5	0,9852
a_6	$-3,2293 \times 10^3$
a_7	$1,4409 \times 10^3$
b_1	$-0,2649 \times 10^{-3}$
b_2	$0,2513 \times 10^{-3}$

Estas equações podem ser representadas na forma de espaço de estados, onde os estados são as pressões de entrada e saída do motor, P_1 e P_2 , a velocidade angular do eixo de saída do redutor, Ω , e a posição angular do eixo de saída, φ . Considera-se ainda como variáveis de estado, os três atrasos presentes no sistema, dados por x_1, x_2 e x_3 . O sistema descrito pelas equações anteriores pode ser colocado na forma matricial, resultando no seguinte:

$$\begin{bmatrix} \varphi(kT + T) \\ \Omega(kT + T) \\ P_1(kT + T) \\ P_2(kT + T) \\ x_1(kT + T) \\ x_2(kT + T) \\ x_3(kT + T) \end{bmatrix} = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 \\ 0 & a_5 & a_6 & a_7 & 0 & 0 & 0 \\ 0 & a_2 & a_1 & 0 & b_1 & 0 & 0 \\ 0 & a_4 & 0 & a_3 & b_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi(kT) \\ \Omega(kT) \\ P_1(kT) \\ P_2(kT) \\ x_1(kT) \\ x_2(kT) \\ x_3(kT) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \varphi_{ref}(kT). \tag{5}$$

Observa-se que a ordem das variáveis de estado não segue a ordem de apresentação das equações fornecidas anteriormente. Como se deseja controlar a posição angular, a saída do sistema é representada pela seguinte equação matricial:

$$y(kT) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi(kT) \\ \Omega(kT) \\ P_1(kT) \\ P_2(kT) \\ x_1(kT) \\ x_2(kT) \\ x_3(kT) \end{bmatrix}. \tag{6}$$

Simplificadamente, as equações de estado e da saída do sistema são dadas por:

$$\bar{x}(kT+T) = \underline{A}\bar{x}(kT) + \underline{B}\alpha_{ref}(kT), \tag{7}$$

$$y(kT) = \underline{C}\bar{x}(kT) + \underline{D}\alpha_{ref}(kT), \tag{8}$$

onde, para o sistema em malha aberta, a matriz \underline{A} e o vetor \underline{B} são fornecidos na Eq. (5), \underline{D} é um vetor nulo e \underline{C} é o vetor da Eq.(6). Desta forma, obtém-se a representação dos espaços de estados para o sistema em malha aberta.

A partir daí, são analisados três tipos de controladores para o MPAR. Dois controladores são baseados em técnicas clássicas de controle, os controladores PD e PID. O terceiro controlador está baseado na técnica moderna de realimentação de estados, com uso de um integrador cujo projeto será apresentado de forma detalhada, já que é o controlador que apresenta os melhores resultados.

No caso do projeto dos controladores PD e PID adota-se o seguinte procedimento: (1) primeiramente, determina-se a função de transferência global do sistema em malha aberta, que inclui o modelo linear do MPAR e a equação do controlador utilizado; (2) com a função de transferência da malha aberta calculam-se os ganhos do controlador a partir da análise do lugar das raízes; (3) a resposta obtida é uma primeira aproximação, pois o modelo linear não considera os efeitos de saturação da válvula, zona morta da válvula e atrito; (4) os ganhos do controlador determinados através da análise pelo lugar das raízes são testados com ensaios realizados na bancada experimental; (5) as etapas (1) a (4) são repetidas algumas vezes, até que se obtenha uma resposta otimizada ou se verifique que o controlador não é aplicável.

O primeiro controlador projetado e implementado é o controlador PD. Verifica-se, entretanto, que existe um erro de posicionamento elevado devido à existência de um atrito estático considerável no sistema MPAR-Válvula. Isto impede que o sistema atinja a posição de regime desejada, mesmo com a presença do integrador natural do sistema. Além disto, o sistema MPAR-Válvula apresenta uma zona morta, dentro da qual, mesmo a válvula estando ligeiramente aberta, o MPAR não se movimenta. O atrito não pode ser reduzido e nem alterado. No entanto, a zona morta pode ser considerada no controlador PD, de forma a impedir que a válvula opere nesta região. Assim, modifica-se o controlador, introduzindo-se um comando que impede que a válvula opere dentro da região da zona morta.

A Fig. (7) ilustra uma resposta experimental típica do controlador PD modificado, chamado de controlador PD com compensação de zona morta. Neste caso, o erro de posicionamento é da ordem de 20 minutos de arco, o sobressinal está adequado, mas o sistema apresenta um comportamento oscilatório e o tempo de assentamento aumentou significativamente, estando em torno de 400ms. Assim, o controlador não segue as especificações de desempenho.

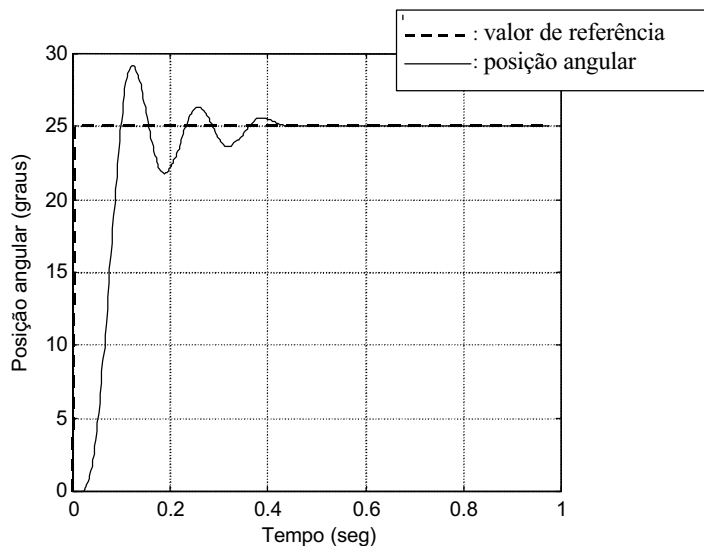


Figura 7. Transitório de posicionamento do sistema MPAR-Válvula para o controlador PD com compensação de zona morta.

Na tentativa de reduzir o erro de posicionamento utiliza-se um controlador tipo PID. A implementação do controlador PID é feita com a equação de diferenças associada à sua função de transferência, considerando-se a compensação da zona morta da válvula. A compensação da zona morta foi introduzida neste controlador para torná-lo mais rápido, pois a eliminação do erro causado pelo atrito estático é feita pela parte integral do controlador de maneira muito lenta. O controlador é testado na bancada experimental com um transitório na forma de um degrau de posicionamento com amplitude de 25°. A Fig. (8) apresenta uma resposta típica do MPAR com o controlador PID. O erro de posicionamento neste caso é baixo, cerca de 30 minutos de arco, mas o sistema apresenta um alto sobressinal, da ordem de 60% e um tempo de assentamento muito grande, cerca de 0,55s. Assim, este controlador não obedece as especificações desejadas.

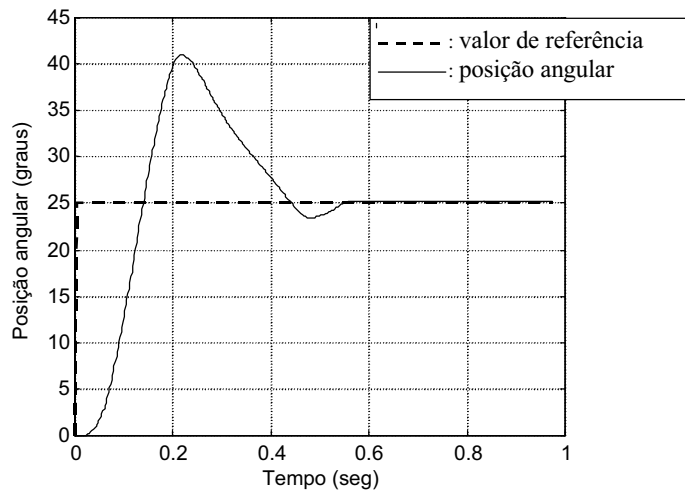


Figura 8. Transitório de posicionamento do sistema MPAR-Válvula para o controlador PID com compensação de zona morta.

Como visto, os controladores PD e PID não fornecem uma resposta dentro das especificações de desempenho desejadas. As respostas destes controladores não são adequadas devido à existência do atrito estático e ao fato do sistema ser altamente não-linear. Assim, optou-se por avaliar uma outra técnica de controle, no caso, a técnica de controle baseada na realimentação dos estados. Este tipo de controlador pode fornecer melhores resultados, pois consegue perceber a relação dinâmica entre as diversas variáveis do sistema.

A metodologia utilizada para o controlador com realimentação de estados é a seguinte: (1) primeiramente obtém-se o modelo do sistema segundo uma representação de espaço de estados; (2) introduz-se um integrador no sistema como sendo mais uma variável de estado; (3) projeta-se um regulador para os estados do sistema, através da alocação de pólos; (4) transforma-se o regulador em um servossistema com a introdução da referência do sistema, isto é, a posição angular de referência para o eixo de saída do redutor. Este valor é comparado com a saída do sistema e a diferença, ou erro, é aplicado ao controlador do sistema; (5) o controlador é implementado e testado na bancada experimental.

A primeira etapa deste procedimento já foi realizada, quando da representação do modelo linear no espaço de estados, dada pelas Eq. (5) e Eq. (6). Um integrador é introduzido no sistema, na forma de mais um estado associado à saída do sistema da seguinte forma:

$$\begin{bmatrix} \varrho(kT+T) \\ \Omega(kT+T) \\ P_1(kT+T) \\ P_2(kT+T) \\ x_1(kT+T) \\ x_2(kT+T) \\ x_3(kT+T) \\ i(kT+T) \end{bmatrix} = \begin{bmatrix} 1 & T & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_5 & a_6 & a_7 & 0 & 0 & 0 & 0 \\ 0 & a_2 & a_1 & 0 & b_1 & 0 & 0 & 0 \\ 0 & a_4 & 0 & a_3 & b_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \varrho(kT) \\ \Omega(kT) \\ P_1(kT) \\ P_2(kT) \\ x_1(kT) \\ x_2(kT) \\ x_3(kT) \\ i(kT) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 0 \end{bmatrix} \varrho_{ref}(kT), \tag{9}$$

onde $i(kT)$ é o estado do integrador. Simplificadamente, as equações de estado e da saída do sistema com o integrador podem ser escritas da seguinte forma:

$$\bar{x}_a(kT+T) = \underline{A}_a \bar{x}_a(kT) + \underline{B}_a \varrho_{ref}(kT), \tag{10}$$

$$y(kT) = \underline{C}_a \bar{x}_a(kT) + \underline{D}_a \varrho_{ref}(kT), \tag{11}$$

onde a matriz \underline{A}_a e o vetor \underline{B}_a são fornecidos na Eq. (9), \underline{D}_a é um vetor nulo e o vetor \underline{C}_a é dado por:

$$\underline{C}_a = [1 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0 \ 0]. \tag{12}$$

Um controlador no espaço de estados é descrito pela seguinte expressão:

$$\varrho_{ref}(kT) = -\underline{K} \bar{x}_a(kT), \tag{13}$$

onde \underline{K} é uma matriz de ganhos, cujos elementos estão associados a cada um dos estados. Substituindo-se a Eq. (13) na Eq. (10), obtém-se a equação do sistema em malha fechada, dada por:

$$\bar{x}_a(kT + T) = \underline{A}_{CL} \bar{x}_a(kT), \tag{14}$$

onde \underline{A}_{CL} é a matriz de estados da malha fechada, dada por:

$$\underline{A}_{CL} = (\underline{A}_a - \underline{B}_a \underline{K}). \tag{15}$$

O sistema em malha fechada descrito pela Eq. (14) é um regulador, já que não possui uma entrada. Em um regulador, o propósito de controle é manter o sistema em uma mesma condição de equilíbrio. Projetar o regulador é, na verdade, determinar o vetor de ganhos \underline{K} . Como se verifica, os pólos do regulador são os autovalores da matriz \underline{A}_{CL} . Assim, impondo-se valores para os pólos do regulador determina-se a matriz de ganhos \underline{K} . Tal processo é conhecido como alocação de pólos, pois reposiciona os pólos do sistema para que o sistema em malha fechada apresente a resposta desejável. Este procedimento está descrito com detalhes em Ogata (1987). Os pólos da malha aberta do sistema estão posicionados no plano z em:

$$p_{ma} = [1 \ 0.0568 \ 0.943 + 0.026i \ 0.943 - 0.026i \ 0 \ 0 \ 0 \ 1]. \tag{16}$$

Como se observa, os pólos dominantes de malha aberta são elementos integradores. Define-se, então, para os pólos da malha fechada no plano z os seguintes valores:

$$p = [0.965 \ 0.955 \ 0.86 + 0.01i \ 0.86 - 0.01i \ 0.15 \ 0.05 \ 0.10 \ 0.96]. \tag{17}$$

Como pode se observar, os valores dos pólos da malha fechada estão relativamente próximos dos pólos da malha aberta. Do ponto de vista de controle no domínio de tempo discreto, deseja-se afastar os pólos do círculo unitário. Com isto, obtém-se um sistema com resposta mais rápida. No entanto, se os pólos forem posicionados muito longe dos pólos de malha aberta, os valores dos ganhos serão elevados podendo ocorrer saturação da entrada ou excitação de altas frequências do sistema não modeladas. Assim, o sistema não irá responder de forma adequada. Dessa forma, existe um compromisso no posicionamento dos pólos para que a escolha dos mesmos forneça uma resposta do sistema de posicionamento de acordo com as especificações de desempenho propostas. Com o valor dos pólos desejados para o regulador e o procedimento descrito anteriormente, calcula-se a matriz de ganhos \underline{K} , cujos valores são apresentados na Tab. (2), dada a seguir.

Tabela 2. Valores da matriz de ganhos da alocação de pólos proposta.

Coeficiente	Valor
K_1	22214,78
K_2	804,33
K_3	232,71
K_4	-212,74
K_5	0,085
K_6	0,1106
K_7	-0,1793
K_8	259,07

Observa-se que os ganhos obtidos foram calculados a partir do escalamento dos parâmetros do modelo linear. O controlador também escala o valor das variáveis de estado e, desta forma, consegue-se uma maior precisão na implementação. O próximo passo do projeto é obter um servossistema a partir do regulador. Este servossistema é composto pelo sistema em malha aberta e pelo controlador, onde se realimenta a posição angular, comparando-a com um valor de referência. Assim, a equação dos estado do servossistema é dada por:

$$\bar{x}_a(kT + T) = \underline{A}_{CL} \bar{x}_a(kT) + \underline{B}_S \varpi_{ref}(kT), \tag{18}$$

onde \underline{B}_S é o vetor associado à entrada da referência da posição angular do eixo de saída do redutor, $\varpi_{ref}(kT)$. O vetor \underline{B}_S é igual a:

$$\underline{B}_S = [0 \ 0 \ 0 \ 0 \ 0 \ 0 \ K_1 \ -1]^T. \tag{19}$$

O controlador é composto pela equação do integrador, porém aplicado não mais sobre $\alpha(kT)$ mas sobre a diferença entre $\alpha_{ref}(kT)$ e este valor, isto é, o erro. A equação do controlador com realimentação dos estados é dada por:

$$\alpha_{ref}(kT) = K_1[\alpha_{ref}(kT) - \alpha(kT)] - K_2\Omega(kT) - K_3P_1(kT) - K_4P_2(kT) - K_5x_1(kT) - K_6x_2(kT) - K_7x_3(kT) - K_8i(kT) \quad (20)$$

e

$$i(kT + T) = i(kT) - \alpha_{ref}(kT) + \alpha(kT) . \quad (21)$$

As Eq. (20) e Eq. (21) são implementadas no microcomputador e o controlador é testado. A Fig. (9) apresenta um resultado típico de um transitório de posicionamento do eixo de saída do redutor, para um degrau no valor de referência de 0° para 25° . Verifica-se que o sistema em malha fechada tem um erro de posicionamento da ordem de 2 a 5 minutos de arco, sobressinal de 30% e tempo de assentamento de 300ms. Como nos demais casos o tempo de assentamento e o sobressinal estão acima das especificações de desempenho, embora o erro de posicionamento seja adequado às especificações de precisão.

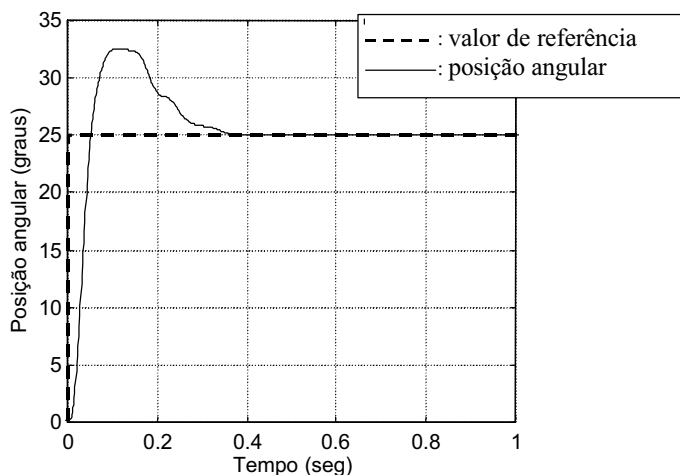


Figura 9. Transitório de posicionamento do sistema MPAR-Válvula para o controlador de espaço de estados

3. Conclusões

De acordo com as respostas ao degrau, constatou-se que o sistema de posicionamento possui um tempo de assentamento e sobressinal elevados. A primeira característica se justifica em função do tamanho do motor pneumático, que confere ao mesmo uma alta potência e elevada inércia. Outro fato que prejudicou a ação dos controladores foi o atrito que o motor pneumático apresenta que como foi citado é altamente não-linear. Foram examinadas três técnicas de controle e nenhuma satisfaz, por completo, as especificações de desempenho desejadas. Mais especificamente sobre o controlador, a técnica de controle que apresentou a melhor resposta foi a da realimentação de estados, pois foi a que forneceu o menor erro de posicionamento. Porém, trata-se de uma forma de controle bastante complexa, pois necessita do uso de sensores para medir as pressões, que são estados do sistema. Qualquer estratégia de controle baseada no espaço dos estados, necessitará da informação destes sinais. Uma alternativa, ainda que complexa, é estimar os valores de pressão com base em um modelo de referência ou com um filtro de Kalman.

4. Referências

- Gouveia, G. M., 1996, "Válvula Pneumática Proporcional de Quatro Vias", Dissertação de Mestrado, Depto. de Eng. Mecânica, Escola Politécnica da USP, São Paulo.
- Hunold, M. C., 1999, "Sistema de Posicionamento Pneumático Rotativo", Tese de Doutorado, Depto. de Eng. Mecânica, Escola Politécnica da USP, São Paulo.
- Ljung, L., 1985, "Theory and Practice of Recursive Identification", The MIT Press, Cambridge, Massachusetts, USA.
- Mahgoub, H. M. and Craighead, I. A., 1996, "Robot Actuation Using Air Motors", International Journal of Advanced Manufacturing Technology", Vol. 11, pp 221-229.
- Ogata, K., 1987, "Discrete-Time Control Systems", Prentice Hall Inc., Englewood Cliffs, N. J., USA.
- Pandiam, S. R., Takemura, F., Hayakawa, Y. and Kawamura, S., 1999, "Control Performance of an Air Motor – Can Air Motors Replace Electric Motors?", Proceedings of the 1999 IEEE International Conference on Robotics and Automation, Vol. 1, Detroit, Michigan, pp 518 – 524.
- Pu, J., Moore, P. R. and Weston, R. W., 1991, "Digital Servo Motion of Air Motors", International Journal of Production Research, Vol. 29, n° 3, pp 599-618.

THE CONTROL DESIGN OF A ROTARY PNEUMATIC POSITIONING SYSTEM

Marcos Costa Hunold

Politecnical School of University of São Paulo
Department of Mechatronics
Mello Moraes Avenue, 2231
mahunold@usp.br

Eduardo Lobo Lustosa Cabral

Politecnical School of University of São Paulo
Department of Mechatronics
Mello Moraes Avenue, 2231
elcabral@usp.br

Abstract. This work describes the control design of a rotary pneumatic positioning system that is implemented in an experimental apparatus. This system, due to the achieved precision and stiffness, can be used to drive robots and CNC machine tools. The positioning system consists of a rotary vane air motor, a planetary traction drive speed reducer and a pneumatic four way proportional valve. The proportional valve is used to adjust the air flow to the system. A conventional PD digital controller controls the valve flow area. The dynamic and steady state behaviours of the positioning system with the valve control are studied and a model is proposed and used in the control design of this system. The control design of the rotary pneumatic positioning system is done. Conventional PD, PID and state feedback digital controllers are implemented and tested. Results show that the state feedback controller is the best because it gives the smallest position errors although its dynamic response is not very adequate. Therefore, it is concluded that this rotary pneumatic positioning system has good precision but that further research is needed to provide it with an adequate closed loop dynamic response.

Key words: *air motors, rotary pneumatic systems, position control.*

IMPLEMENTAÇÃO DE CONTROLADORES FUZZY EM MICROCONTROLADORES APLICADOS NO CONTROLE DE VIBRAÇÕES EM ESTRUTURAS TIPO VIGA

Gustavo Luiz C. M. de Abreu

Universidade Federal de Uberlândia – FEMEC – Faculdade de Engenharia Mecânica
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
gmanhaes@mecanica.ufu.br

Patrick Magalhães Cardoso

Universidade Federal de Uberlândia – FEMEC – Faculdade de Engenharia Mecânica
patrick@mecanica.ufu.br

Resumo. Este trabalho apresenta a aplicação de controladores fuzzy implementados em microcontroladores para o controle de vibrações em estruturas flexíveis do tipo viga utilizando atuadores piezelétricos. Para a implementação do controlador fuzzy foi utilizado o microcontrolador PIC16F877 da Microchip®. O controlador fuzzy foi concebido de forma intuitiva onde as restrições de velocidade de processamento e tamanho (memória ocupada) foram criteriosamente avaliadas atendendo às limitações do microcontrolador. É apresentada uma bancada de testes que tem suas frequências naturais identificadas através de uma série de ensaios. Tal bancada é usada para avaliar a ação do controlador fuzzy implementado no microcontrolador.

Palavras chave: *Microcontroladores, Materiais Piezelétricos, Controladores Fuzzy, Controle de Vibrações e Estruturas Flexíveis do tipo Viga.*

1. Introdução

A Lógica Fuzzy vem se tornando nas últimas décadas uma opção interessante na solução de problemas em várias áreas da engenharia. A sua utilização no controle tem evoluído de forma crescente durante os últimos anos, pois possibilita lidar com sistemas que envolvam informações muitas vezes vagas, imprecisas e não adequadamente quantificadas em que decisões objetivas de operação precisam ser tomadas. Ao contrário da teoria clássica de controle cujo requisito principal é o conhecimento dos modelos dos elementos do sistema a ser controlado (equações exatas e valores numéricos), a lógica fuzzy permite tratar com sistemas complexos usando um alto nível de abstração a partir do conhecimento e da experiência a respeito do sistema.

Em vários tipos de estruturas flexíveis, a presença de vibrações indesejadas provenientes de perturbações externas ocasiona uma diminuição no tempo de vida do sistema, demanda manutenções periódicas mais frequentes e, em determinadas situações, podem provocar fadigas mecânicas que afetam severamente a estrutura.

Recentemente, uma nova alternativa no controle de vibrações em estruturas flexíveis tem sido alvo de investigação de vários estudiosos da área (Taylor et al., 1985). Segundo essa alternativa, uma estrutura pode ter a sua resposta minimizada, utilizando de forma integrada elementos ativos, como sensores e atuadores constituídos de materiais piezelétricos. As cerâmicas piezelétricas desenvolvem um campo elétrico, quando sujeitas a uma força (ou pressão) e apresentam uma deformação mecânica, quando sujeitas a um campo elétrico. Este é o fenômeno da piezoelectricidade que tem grande aplicabilidade em diversas áreas da ciência e principalmente na área de controle de estruturas flexíveis. Um dos primeiros relatos sobre o uso de materiais piezoelétricos como atuadores no controle de estruturas, foi apresentado por Crawley & de Luis (1987), que construíram três protótipos de estruturas com cerâmicas piezelétricas incorporadas, quais sejam: uma viga de alumínio com atuadores piezelétricos colados sobre suas superfícies, uma viga de grafita/epoxy e outra com fibra de vidro/epoxy, ambas com material piezelétrico imerso nas estruturas e empregaram um controlador por realimentação proporcional à velocidade. Uma das dificuldades no projeto deste tipo de controlador decorre da necessidade de atender os seguintes passos:

- compreensão do sistema físico e de seus critérios de desempenho;
- desenvolvimento de um modelo matemático que inclua a planta, sensores e atuadores;
- utilização da teoria de controle linear para determinar os parâmetros principais do controlador;
- desenvolvimento do algoritmo de controle e
- simulação do sistema em malha fechada incluindo os efeitos não lineares, ruído e variações paramétricas.

Caso o desempenho do controle não for satisfatório, é preciso remodelar o sistema (em alguns casos esta informação é muito complexa ou de difícil obtenção), reprojeter o controlador e reescrever o algoritmo de controle.

Em contrapartida, quando se utiliza a lógica fuzzy alguns passos da metodologia clássica são eliminados e algumas etapas devem ser superadas, quais sejam:

- compreensão e caracterização do comportamento do sistema utilizando o conhecimento e a experiência;
- desenvolvimento do algoritmo de controle usando regras fuzzy que descrevem os princípios do controlador em termos da relação entre as entradas e as saídas e por fim
- simulação do sistema com o controle fuzzy.

Neste caso, se o desempenho do controle não for satisfatório, algumas regras fuzzy podem ser modificadas cuja estrutura incorpora informações verbais ou linguísticas de sentido intuitivo. Como resultado, a lógica fuzzy reduz substancialmente o ciclo de desenvolvimento do controlador e a característica importante das soluções alcançadas é a robustez, que proporciona bom desempenho mesmo diante de variações nos parâmetros dos sistemas físicos.

Ao lado do avanço teórico a despeito dos controladores, evoluiu também nos últimos anos, e de forma muito acelerada, as conquistas no campo tecnológico. Podem ser citados os sistemas eletrônicos para a aquisição, transmissão e tratamento de sinais, sem falar no grande desenvolvimento e popularização dos computadores. Proliferaram as técnicas de miniaturização eletrônica e surgiram os circuitos integrados e microprocessadores que reúnem dentro de um pequeno componente diversos outros, que se interagem para a realização de uma determinada tarefa. Hoje em dia chega-se a se ter microprocessadores do tamanho de um grão de areia que são capazes de processar o mesmo número de dados que um computador de grande porte de 15 anos atrás.

Toda esta tecnologia miniaturizada é um grande atrativo para se implementar fisicamente os controladores. Sofisticados controladores fuzzy podem ser implementados em microcontroladores, que possuem dimensões máximas da ordem de 5 cm², com grande facilidade. Outro fator importante está no baixo custo destes microprocessadores. Por US\$ 20,00 adquire-se máquinas com configurações bastante sofisticadas, que propiciam uma fácil implementação de controladores razoavelmente complexos (Microchip, 2001).

Em um único microcontrolador podem ter: várias portas de entrada e saída de dados, conversores analógicos-digitais (AD's) de sinais, saídas do tipo PWM (modularização por largura de pulso), contadores de tempo automáticos, interrupções, e outras funções. Além destas características os microprocessadores atuais apresentam uma grande capacidade de processamento possuindo um bom espaço de memória destinada a programa (ROM), o que possibilita a implementação de programas mais complexos. Apresentam também uma memória para as variáveis de programa (RAM) bem dimensionada, que permite a execução de programas relativamente extensos.

Todos estes microcontroladores operam com uma base de tempo externa que pode ser dada por um simples temporizador RC (resistor/capacitor) ou por osciladores de cristal, sendo que em ambos os casos as configurações são de baixíssimo custo. Tem-se hoje em dia microcontroladores bastante populares, que operam em frequências de até 40MHz, resultando em tempos de instrução (ciclo de instrução) da ordem de 100 ns (100 nano segundos), possibilitando a realização de cálculos complexos em um tempo muito reduzido. Existem naturalmente microcontroladores que operam em frequências superiores, porém são de custo um pouco maior e não muito populares.

Aliando-se a eficiência, simplicidade e robustez dos controladores fuzzy com a implementação rápida, barata e simples em microprocessadores, tem-se como resultado um controlador confiável, que pode ser projetado rapidamente e a um custo baixo. Tais características de confiabilidade, baixo custo e rápido desenvolvimento são objetivo permanente do mercado mundial, uma vez que existe uma grande concorrência em praticamente todas as áreas da indústria.

Frente a todas estas vantagens apresentadas, este trabalho propõe a construção de um módulo de controle baseado em lógica fuzzy para sistemas de uma entrada e uma saída. Este módulo é implementado fisicamente utilizando-se microprocessadores e outros componentes eletrônicos auxiliares para o controle de vibrações em estruturas flexíveis do tipo viga contendo atuadores piezelétricos incorporados.

2. Conceitos Básicos da Teoria da Lógica Fuzzy

A utilização dos sistemas Fuzzy (sistemas nebulosos) no controle moderno foi introduzido a partir do trabalho de Lotfi Zadeh (1965) e o grande apelo da tecnologia fuzzy em controle é a possibilidade de se operar com "meias verdades ou meias mentiras", ou seja, pode-se admitir uma incerteza na definição das variáveis de entrada e de saída. Na lógica Booleana um determinado elemento pertence ou não a um conjunto específico, ou formalmente, seja U uma coleção de objetos, chamado universo de discurso, u um elemento particular deste universo e A um dado conjunto contido em U , então, ou $u \in A$ ou $u \notin A$. Para cada elemento do universo de discurso pode-se, portanto, determinar sem ambigüidades quando pertence ou não a um conjunto. Na teoria dos conjuntos Fuzzy faz-se uma generalização desta teoria clássica. A extensão está na possibilidade de um determinado elemento poder pertencer a um conjunto com um valor chamado grau de pertinência (Zadeh, 1965). Assim, um elemento não simplesmente pertence ou não pertence a um conjunto, como na teoria clássica, mas poderá pertencer a um conjunto com grau de pertinência que varia no intervalo $[0,1]$, onde o valor 0 indica uma completa exclusão (não – membro), o valor 1 representa completa pertinência (totalmente membro) e os valores deste intervalo representam graus intermediários de pertinência do objeto com relação ao conjunto. A função que define os graus de pertinência dos elementos é chamada função de pertinência e é uma generalização da função característica da teoria clássica, uma vez que associa para todo elemento do universo de discurso um valor do intervalo $[0,1]$ ao invés do conjunto de apenas dois elementos $\{0,1\}$.

As abordagens dos controladores convencionais mais utilizados procuram tratar as não linearidades dos sistemas físicos através de aproximações. Ou os sistemas são considerados simplesmente lineares, lineares por partes, ou são descritos por extensas tabelas de buscas ("lookup tables"), que procuram mapear as entradas e saídas do processo. Classicamente a escolha entre um procedimento ou outro depende dos custos envolvidos, da complexidade de implementação e do desempenho desejado. Os controladores fuzzy por sua vez não ignoram as não linearidade dos processos, pelo contrário incorporam-nas na sua metodologia de projeto. Tal metodologia envolve a construção de regras, operadores lógicos e funções de pertinência, que mapeam, através de um processo de inferência, as variáveis de

entrada em variáveis de saída. Qualquer função, linear ou não, que faça este mapeamento pode ser aproximada por uma função construída a partir da lógica fuzzy.

Para desenvolver um controlador fuzzy convencional há seis passos básicos a serem superados: 1) definição das variáveis de entrada e saída; 2) caracterização do intervalo de valores que as variáveis de entradas e de saída podem assumir; 3) definição de um conjunto de funções, denominadas funções de inferência, que mapeam as variáveis de entrada e saída no universo $[0,1]$. 4) definição de um conjunto de regras, usando operadores lógicos, que buscam estabelecer uma relação entre valores da entrada e da saída; 5) inferência fuzzy e por fim, 6) a defuzzificação e agregação (veja Fig. 1).

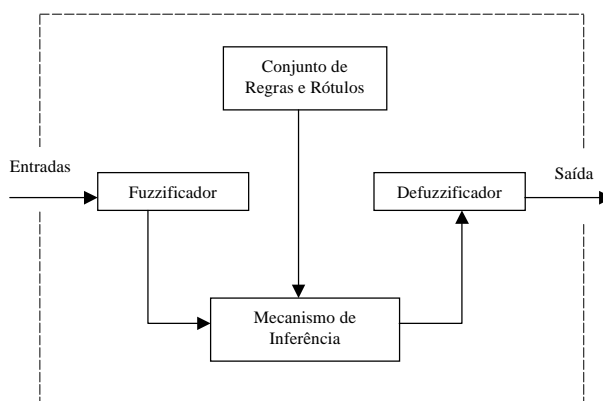


Figura 1. Sistema de inferência fuzzy.

O problema de controle usando a lógica fuzzy envolve a superação de algumas etapas bem definidas:

1. Definição das variáveis de entrada e saída. As variáveis de entrada/saída de um controlador fuzzy podem ser divididas em variáveis do sistema e variáveis lingüísticas. A maioria dos controladores fuzzy empregam as variáveis do sistema: erro e variação do erro como entrada e força, voltagem ou outra variável da lei de controle como saída.
2. Caracterização do intervalo de valores que as variáveis de entradas e de saída podem assumir. Esta etapa não apresenta nenhuma dificuldade a ser vencida.
3. Definição de um conjunto de funções, denominadas funções de inferência, que mapeam as variáveis de entrada e saída no universo $[0,1]$. Estas funções recebem nomes (rótulos) que procuram traduzir verbalmente algum significado para o fenômeno físico modelado. A definição do número de funções e a forma das mesmas é um campo aberto à investigações.
4. Definição de um conjunto de regras, usando operadores lógicos, que buscam estabelecer uma relação entre valores da entrada e da saída. A base regras do controlador fuzzy é importante para o bom funcionamento do sistema de controle. A base de regras ou base de conhecimento, contendo vários se-então é construída como segue:

$$R_i: \quad \text{Se } x_1 \text{ é } A_{i1} \text{ e } x_2 \text{ é } A_{i2} \dots \text{ e } x_n \text{ é } A_{in} \text{ então } y_i \text{ é } B_i \quad (1)$$

onde R_i é a i -ésima regra, A_{ik} é um conjunto fuzzy, x_k é uma variável de entrada, y_i é a saída da i -ésima regra e B_i é o conjunto fuzzy da saída que pode ser representado por um conjunto de funções de pertinência ou por uma combinação algébrica das variáveis de entrada (Takagi e Sugeno, 1985) descrita da seguinte forma:

$$B_i = c_o^i + c_1^i x_1 + \dots + c_n^i x_n \quad (2)$$

onde c_k^i são os parâmetros do conseqüente no modelo fuzzy Takagi-Sugeno.

O estabelecimento das regras, ou seja a natureza e o número das mesmas é uma variável de projeto que não dispensa naturalmente o uso da experiência e do bom senso. Um número elevado de regras não significa necessariamente um melhor desempenho. A relação custo benefício entre a demanda adicional de memória, o tempo de processamento e a melhoria no desempenho do controlador precisa ser criteriosamente avaliada quanto da construção do conjunto de regras.

5. Mecanismo de inferência fuzzy. O mecanismo de inferência fuzzy que baseia-se no produto composicional Max-Min (Zadeh, 1965) foi empregado neste trabalho.

6. Uma vez definidas as regras, derivadas a partir de uma linguagem simbólica e com significado bem intuitivo para o projetista, passa-se à fase da tradução matemática da linguagem simbólica construída. Isto é conduzido através da utilização de operadores lógicos definidos pela teoria dos conjuntos fuzzy. Esta tarefa se divide na verdade em três subetapas: A primeira transforma os valores reais das variáveis de entrada em graus de pertinência a um determinado conjunto (fase de "fuzzificação"), a segunda opera com as regras, os rótulos e o resultado fase de "fuzzificação" e gera um conjunto de variáveis fuzzy através do mecanismo de inferência a terceira e última subetapa transforma os resultados da inferência em uma saída numérica ("fase de defuzzificação"). Neste trabalho, o centróide de área foi empregado no cálculo da saída final do controle.

Após esta breve revisão dos principais conceitos da lógica fuzzy apresenta-se a seguir o problema do controle das vibrações de um sistema flexível constituído por uma estrutura do tipo viga empregando-se atuadores piezelétricos.

3. Descrição do Problema de Controle

O problema que se pretende resolver neste trabalho é controlar as vibrações presentes em um sistema vibratório constituído por uma estrutura flexível tipo viga utilizando-se das metodologias dos sistemas fuzzy implementadas em microcontroladores.

A bancada experimental é constituída de uma estrutura do tipo viga de material aço inox com duas placas de material piezelétrico coladas nas duas faces opostas da estrutura (proporciona o movimento de flexão da estrutura), como pode ser vista na Fig. (2). As dimensões principais do modelo experimental da viga são mostradas na Fig. (3) e a condição de contorno pretendida é engastada-livre.

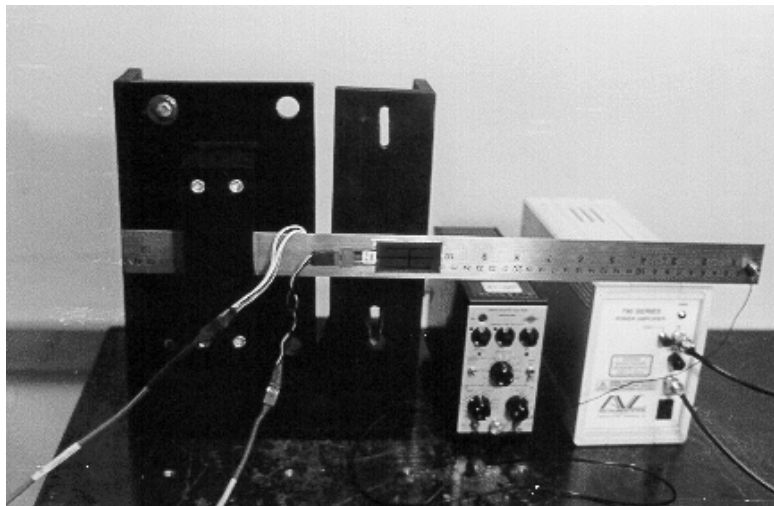


Figura 2. Bancada Experimental.

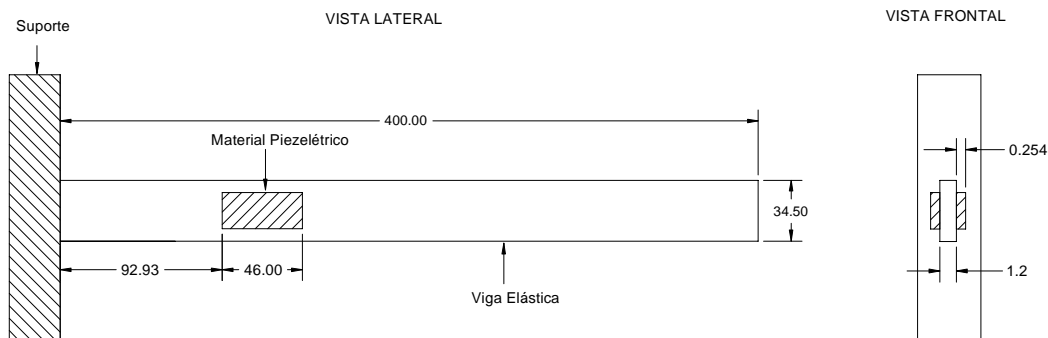


Figura 3. Características dimensionais do modelo experimental.

O elemento piezelétrico, fornecido pelo fabricante da empresa ACX (Active Control eXperts), é constituído por duas placas de material piezelétrico comandados por uma mesma tensão elétrica de sinais contrários, de modo que sua deformação é transmitida à viga de aço. Estes elementos, por sua vez, são alimentados por um amplificador de potência da ACX, que amplifica a tensão de entrada com um ganho na ordem de 30 V/V.

4. Análise Modal Experimental

O ensaio experimental para a determinação das frequências naturais do sistema físico foi feito através da análise da sua resposta em frequência quando excitado por uma força impulsiva aplicada na extremidade livre da estrutura.

Para a determinação das frequências naturais do sistema utilizou-se um acelerômetro piezelétrico instalado próximo à extremidade livre. O sistema foi excitado por um martelo de impacto tipo 8202, da Brüel & Kjaer, instrumentado com uma célula de carga piezelétrica que gera uma entrada impulsiva. O sinal do acelerômetro do tipo B&K 4375 foi condicionado por um amplificador de sinais B&K.

A Fig. (4) mostra o gráfico da função de transferência entre os sinais de saída e de entrada.

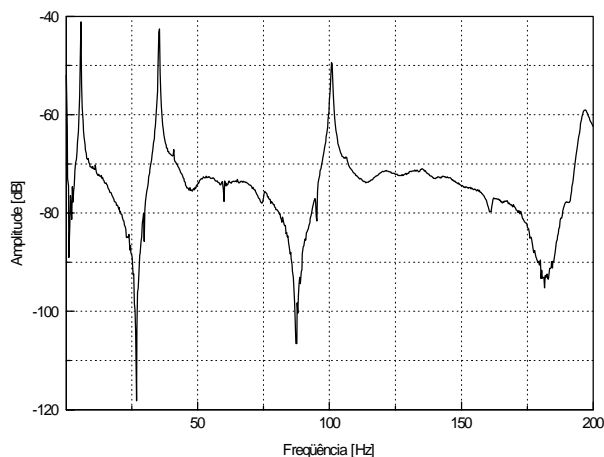


Figura 4. Resposta em frequência entre os sinais de saída e de entrada.

A partir da Fig. (4) construiu-se a Tab. (1) que mostra as 4 primeiras frequências naturais da estrutura com o elemento piezelétrico incorporado.

Tabela 1 – Frequências naturais experimentais.

Modo	Frequência (Hz)
1	5.75
2	35.5
3	100.75
4	196.80

Objetiva-se a partir da análise da função resposta em frequência projetar intuitivamente e implementar num microcontrolador o controlador fuzzy para o controle das vibrações desta estrutura flexível no primeiro modo de vibrar (5.75 Hz), utilizando o elemento piezelétrico como atuador.

5. Projeto do Controlador Fuzzy

Neste trabalho, a construção do controlador fuzzy, foi realizada, à princípio, a partir de dois componentes postos como fundamentais:

- A concepção do controlador deveria ser feita de forma intuitiva com base numa estratégia de controle realizada por um operador humano a partir da observação e análise do comportamento do sistema quando sujeito a uma perturbação externa;
- A solução deveria ser passível de ensaios em laboratório, ou seja, ser de fácil implementação num microcontrolador de baixo custo, não demandando recursos sofisticados de hardware e software e nem um tempo excessivo de execução (restrições de velocidade de processamento e memória ocupada) atendendo, desta forma, às limitações do microcontrolador.

Diante destes dois aspectos, construiu-se e implementou-se o controlador fuzzy a partir de uma estratégia de controle elaborada através da observação e análise do comportamento do sistema diante de perturbações sofridas pela estrutura.

5.1. Estratégia de Controle Utilizada

Considerando os efeitos causados pelo acoplamento dos elementos piezelétricos na estrutura, é possível extrair verbalmente as estratégias de controle que traduzem a operação de controle do sistema físico mediante a aplicação de uma perturbação externa P .

No instante em que aplicamos um potencial elétrico na cerâmica, há um aparecimento de uma deformação induzida no elemento piezelétrico (PZT) que se traduz como um momento. Este esquema de funcionamento tem como função produzir um movimento de flexão na viga contrário ao movimento causado pela perturbação externa P . Neste sentido, podemos construir verbalmente, de forma simplificada, duas *regras* de controle dadas pelas seguintes expressões:

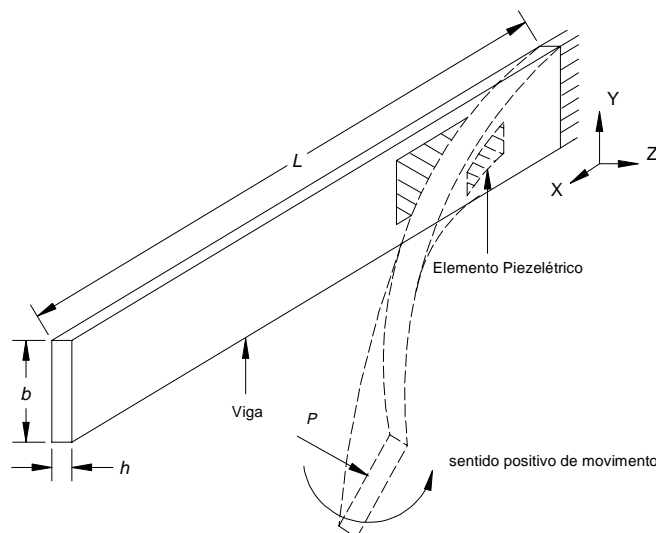


Figura 5. Movimento causado pela perturbação externa P aplicada na viga com elemento piezelétrico.

Se o deslocamento da viga é positivo *então* o momento gerado pelo PZT é negativo

e

Se o deslocamento da viga é negativo *então* o momento gerado pelo PZT é positivo.

Desta forma, o momento gerado pelo elemento cerâmico tem a função de anular os efeitos da força de perturbação P que atua na viga no sentido de desestabilizá-la. Naturalmente a eficiência deste procedimento depende do número e da posição dos atuadores (controlabilidade) e sensores (observabilidade) piezelétricos utilizados (Ribeiro, 1995), bem como da magnitude do potencial elétrico aplicado.

5.2. Definição das Variáveis de Entrada e Saída do Controlador

A primeira abordagem a ser analisada na construção do controlador fuzzy é a definição das variáveis de entrada do controlador. Neste trabalho, optou-se por utilizar um controlador fuzzy constituído por duas variáveis de entrada: *a*) erro entre uma referência e o deslocamento lido pelo sensor e *b*) variação temporal do erro ou variação temporal do erro no instante atual em relação ao instante anteriormente calculado, ou seja:

a) erro

$$e(k) = r(k) - z(k) \quad (3)$$

b) Variação temporal do erro

$$\Delta e(k) = \frac{e(k) - e(k-1)}{\Delta t} \quad (4)$$

onde k e Δt referem-se ao instante e ao tempo de amostragem utilizados, respectivamente.

Evidentemente, a variável de saída do controlador fuzzy é dada, por sua vez, por uma tensão elétrica aplicada no elemento cerâmico.

6. Características do Microcontrolador utilizado

O microprocessador de 8 bits (PIC16F877) utilizado na implementação do controlador fuzzy apresenta, como características mais importantes:

- possui até 8 canais para entrada analógica com conversor analógico/digital de 10 bits;
- opera com um oscilador externo de 20Mhz máximo, resultando em um tempo de instrução de 200ns e
- a tensão de operação pode variar de 2.0V a 5.5V.

Neste trabalho o sinal de controle foi gerado utilizando o sinal PWM do microcontrolador. Foi escolhido, para o PWM, uma frequência de trabalho de 4.9 KHz. Desta forma, a tensão média de saída do controle é alterada pela variação da largura do pulso, conforme mostra a Fig. (6).

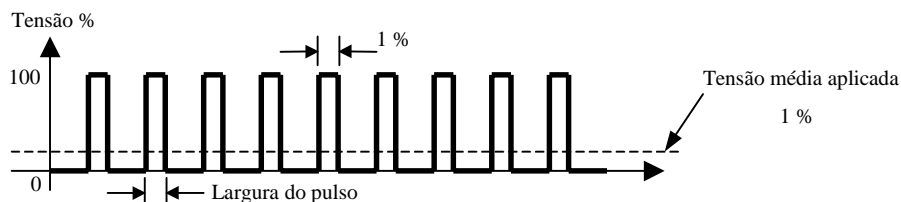


Figura 6. Variação da tensão média abrindo e fechando em tempos controlados.

Variando-se a largura do pulso podemos controlar a tensão média aplicada. Assim, quando a largura do pulso varia de zero até o máximo, a tensão média também varia na mesma proporção.

7. Implementação do Controlador Fuzzy no Microcontrolador

Quanto à implementação do controlador no microcontrolador, as restrições de velocidade de processamento e memória ocupada foram criteriosamente avaliadas atendendo às limitações do microcontrolador. Desta forma, o controlador fuzzy intuitivo foi implementado apresentando as seguintes características:

- três funções de pertinência (*Pequeno, Médio e Grande*) do tipo triangular igualmente espaçadas definidas dentro do universo de discurso das variáveis de entrada (veja Fig. (7));
- saída do controlador do tipo Sugeno de ordem zero e
- 5 regras fuzzy (veja Tab. 2).

Os sinais de entrada foram escalonados analogicamente de 0 a 3V que internamente (no microcontrolador) equivale ao intervalo de 0 a 1023 (valor correspondente a 10 bits), ou seja, os intervalos fuzzy das variáveis de entrada (sinal do erro e da variação do erro) foram definidos tomando-se os valores de entrada do microcontrolador sempre positivos. Desta forma, foi possível trabalhar com a maioria das variáveis e operações em formato tipo inteiro, reduzindo fortemente o tempo de cálculo do sinal de controle e a quantidade de memória de programa utilizada.

Da mesma forma, os parâmetros das funções de pertinência das variáveis de entrada (veja Fig. (7)), os valores Sugeno da saída e as regras (Tab. (2)) foram fornecidos através de pequenos vetores com valores do tipo inteiro para as variáveis de entrada e saída, e em uma matriz, no formato inteiro, para as regras. Este modo de interpretação também contribui fortemente para a redução do tamanho do programa e do tempo de cálculo.

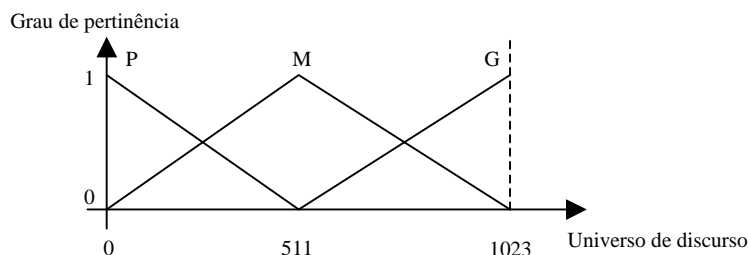


Figura 7. Funções de pertinência e intervalos fuzzy das variáveis de entrada do controlador.

Tabela 2 – Base de regras utilizada.

		Variação do erro		
		P	M	G
Erro	P	¹ 1023		
	M	² 1023	³ 511	⁴ 0
	G			⁵ 0

Cada célula da Tab. (2) representa uma regra construída da seguinte forma:

$$R_1 : \text{Se o erro é } P \text{ e variação do erro é } P \text{ então a saída do controlador é } 1023.$$

A saída final do controle é obtida através da média ponderada dos respectivos valores de saída das regras ativadas. Desta forma, o valor da tensão média aplicada no amplificador de tensão (alimenta o elemento piezelétrico) é obtida a partir do sinal de controle calculado que é dado por um número de 10 bits onde o valor 1023 corresponde ao valor máximo de controle, 0 corresponde ao valor mínimo e 511 ao valor nulo. Pode-se notar que a saída PWM é sempre uma tensão positiva que varia de 0 à tensão de alimentação (neste caso 5V). Neste sentido, para obtermos uma saída também negativa foi construído um circuito analógico que amplifica a saída PWM com um ganho de 2.0 e subtrai a tensão resultante de 5V. Desta forma o circuito de controle passa a operar de $-5V$ a $5V$.

O fluxograma do controle implementado no microcontrolador seguiu os seguintes passos:

1. leitura do valor do sinal de referência;
2. geração do sinal de excitação senoidal na primeira frequência natural da estrutura durante um período de 0,5s;
3. leitura do sinal do sensor;
4. cálculo do sinal de controle e finalmente
5. saída PWM.

Para a implementação do controlador fuzzy no microcontrolador as seguintes etapas foram necessárias:

- elaboração do programa em linguagem C utilizando-se um compilador específico;
- conversão do programa baseado em linguagem C para código assembler e hexadecimal específico do microcontrolador;
- verificação e análise do código em assembler observando o comportamento dos registradores, portas A/D, I/O, etc. e por fim
- gravação do programa no microcontrolador.

8. Avaliação Experimental do Controlador Fuzzy

Após a realização da etapa anterior, procedeu-se a seleção final dos componentes eletrônicos e do microcontrolador (PIC16F877). Com todos os componentes disponíveis, o módulo de controle foi então montado em um "protoboard". Utilizou-se um tempo de controle de 20 ms que proporciona o controle do sistema flexível em questão até a frequência de 25 Hz, ou seja, até a primeira frequência natural da estrutura (5,75 Hz).

O diagrama esquemático do experimento de controle utilizado pode ser visto na Fig. (8).

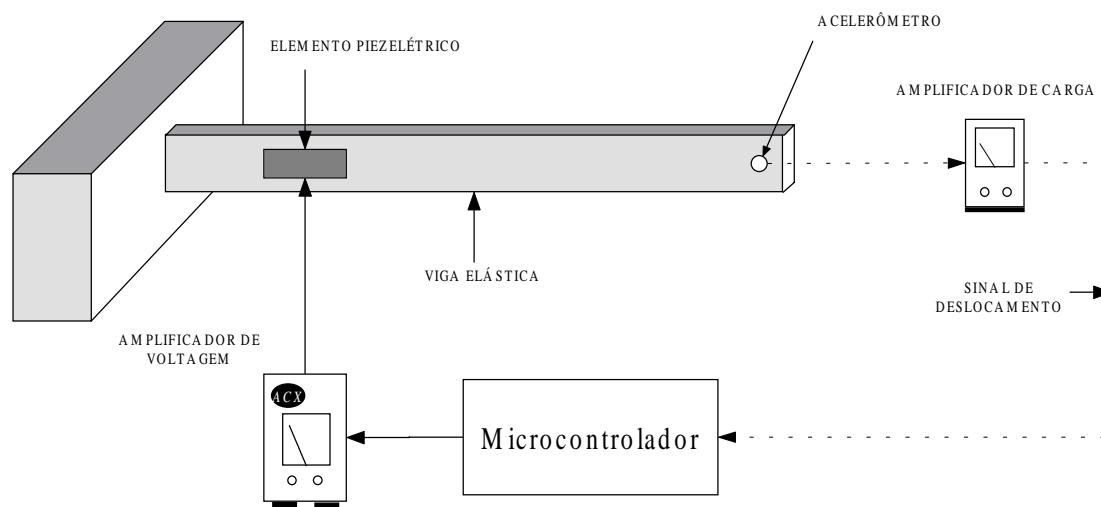


Figura 8. Diagrama esquemático do experimento de controle utilizado.

Segundo o diagrama mostrado na Fig. (8), o sinal do acelerômetro é carreado para a entrada A/D do microcontrolador que faz o interfaceamento entre o acelerômetro e o amplificador de tensão que alimenta duas placas de elementos piezelétricos de modo a proporcionar o movimento de flexão na estrutura..

Para a verificação da eficiência do controlador fuzzy proposto são apresentadas a seguir os testes experimentais realizados para o sistema sem controle e controlado.

8.1. Estudo do Posicionamento do Sensor na Estrutura

O sensor utilizado nos testes experimentais foi o acelerômetro do tipo B&K 4375 condicionado por um amplificador de sinais B&K que forneceu o sinal de deslocamento da viga com a utilização de circuitos de integração. O acelerômetro foi posicionado na extremidade livre da viga onde as deflexões do primeiro modo são maiores.

8.2 Resultados Experimentais e Discussões

Em ensaio experimental, o sistema sem controle, foi excitado pelos atuadores piezelétricos na sua primeira frequência natural num tempo de 0.5s através de um sinal senoidal gerado pelo microcontrolador. Os deslocamentos da viga, gerados por esta excitação, podem ser vistos na Fig. (9).

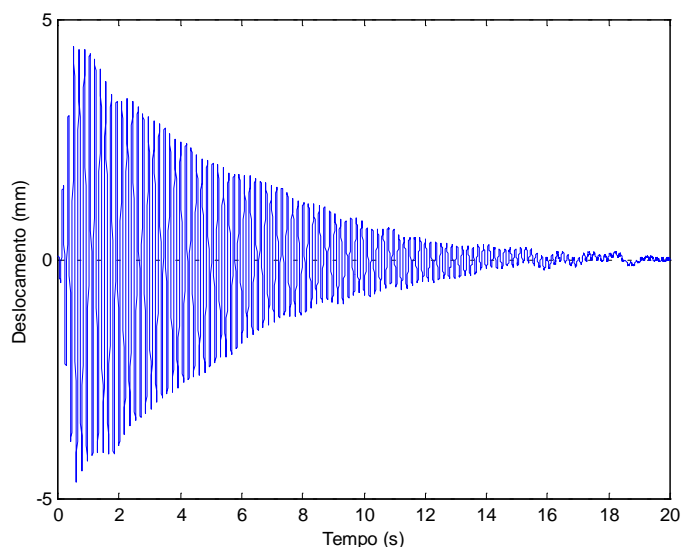


Figura 9. Deslocamento da extremidade livre da viga para o sistema sem controle.

Como pode ser visto na Fig. (9), o sistema leva aproximadamente 20 segundos para atingir patamares de deslocamento significativamente reduzidos, o que reforça a idéia da utilização de uma estratégia de controle fuzzy para este problema.

Com a mesma configuração utilizada na etapa anterior, o circuito de controle foi então acionado e a resposta temporal dos deslocamentos da estrutura foi adquirida e pode ser vista na Fig. (10).

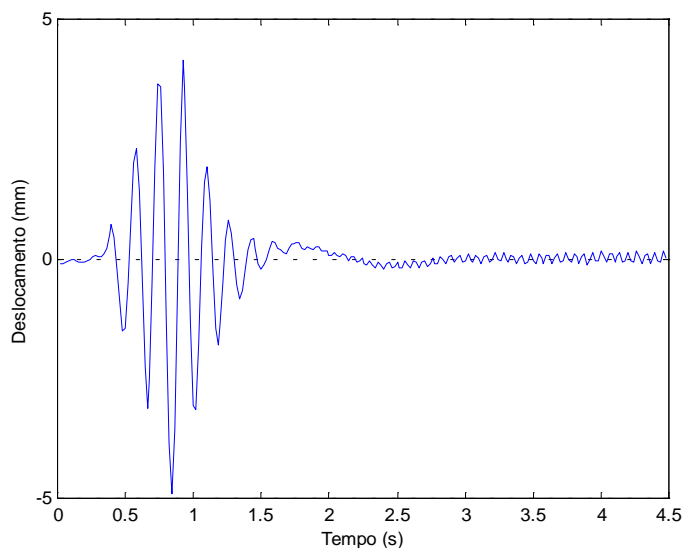


Figura 10. Deslocamento da extremidade da viga e saída PWM para o sistema controlado com um tempo de amostragem de 20 ms.

Observa-se através dos resultados apresentados o bom desempenho do controlador fuzzy. As oscilações iniciais do sistema foram amortecidas a patamares bem reduzidos em torno de 2.0s, comprovando com isso o seu ótimo desempenho.

9. Conclusões

Neste trabalho foi apresentada uma metodologia de controle fuzzy projetado intuitivamente, implementado num microcontrolador e aplicado em uma viga elástica com material piezelétrico incorporado. Num primeiro momento, a bancada de testes construída teve as suas frequências naturais experimentalmente identificadas. Experimentos foram conduzidos com o objetivo de validar a metodologia de controle desenvolvida e de avaliar a ação do controlador fuzzy proposto. O controlador mostrou-se eficiente para o problema em questão. Demonstrou, no domínio do tempo, características de desempenho bastante satisfatórias e não apresentou dificuldades de implementação. A eficiência do microcontrolador foi comprovada mediante os testes realizados não apresentando dificuldades de operação. O elemento piezelétrico, utilizado como atuador neste trabalho, teve sua eficiência constatada a partir dos resultados experimentais realizados na bancada de testes. Uma etapa natural e evidente deste trabalho é a utilização do material piezelétrico como sensor ao invés do emprego do acelerômetro e a utilização de microcontroladores com maior velocidade de processamento e maior capacidade de memória para implementação de controladores fuzzy adaptativos aplicados no controle de vibrações em placas finas.

10. Agradecimentos

Os autores agradecem o apoio financeiro proporcionado a este trabalho pela CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) e pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

11. Referências

- Crawley, E. F. e de Luis, J., 1987, "Use of Piezoelectric Actuators as Elements of Intelligent Structures", AIAA Journal, vol. 25, no. 10, pp. 1373-1385.
- Microchip Technology Inc., <http://www.microchip.com>.
- Ribeiro, J. Francisco, 1995, "Controle Robusto aplicado ao Problema de Vibrações em Rotores Flexíveis", Tese de Doutorado, São José dos Campos, p181.
- Takagi, T. e Sugeno, M., 1985, "Fuzzy Identification of Systems and Its Applications to Modeling and Control", IEEE Transactions on Systems, Man and Cybernetics, vol. 15, No. 1, pp. 116-132.
- Taylor, G.W.; Gagnepain, J.J.; Meeker, T.R.; Nakamura, T. e Shuvalov, L. A., 1985, "Piezoelectricity - Ferroelectricity and Related Phenomena", Gordon and Breach Science Publishers, Vol. 4, p406.
- Zadeh, L. A., 1965, "Fuzzy Sets, Information and Control", Vol. 8, pp. 338-353.

APPLICATION OF FUZZY CONTROLLERS IN MICROCONTROLLERS TO THE CONTROL OF VIBRATIONS IN FLEXIBLE BEAM TYPE STRUCTURE

Gustavo Luiz Chagas M. de Abreu

Federal University of Uberlândia – FEMEC – School of Mechanical Engineering
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
gmanhaes@mecanica.ufu.br

Patrick Magalhães Cardoso

Federal University of Uberlândia – FEMEC – School of Mechanical Engineering
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
patrick@mecanica.ufu.br

Abstract. The great progress experimented by the theory of the fuzzy controllers in the last years has been opening new possibilities of practical application for these controllers. This work presents the application of fuzzy controllers implemented in microcontrollers to control the vibrations of flexible beam type structure bonded with piezoelectric sensors and actuators. For the implementation of the fuzzy controller, constructed from a set of fuzzy IF-THEN rules, the microcontroller PIC16F877 (Microchip) is used and the fuzzy controller is designed to guarantee the basic requirements of the microcontroller like processing speed and memory size. An experimental apparatus is used to evaluate the action of fuzzy controller implemented on the microcontroller.

Keywords. *Microcontrollers, Piezoelectric Materials, Fuzzy Controllers, Control of Vibrations and Flexible Beam Type Structures.*

STRUCTURAL VIBRATION CONTROL USING THICKNESS-SHEAR MODE OF PIEZOELECTRIC ACTUATORS

Marcelo A. Trindade

Laboratório de Dinâmica e Vibrações, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ 22453-900
trindade@mec.puc-rio.br

Abstract. Piezoelectric materials are widely used for structural vibration control, either bonded on or embedded in the structure. Recently, developments in composites design have brought attention to embedded actuators, using the thickness-shear mode of piezoelectric actuators. This is achieved through longitudinally poled piezoelectric patches that, when subjected to a transverse electric field, present shear deformations. This “shear actuation mechanism” induces distributed actuation moments in the structure, unlike extension actuators which induce boundary forces, leading to less problems of debonding in actuators boundaries and to minor dependence of the control performance on actuators position and length. This work aims to present a comparison of active control performances of shear and extension actuation mechanisms. In particular, analysis of optimal parameters regions are studied and used to compare both mechanisms. Thereafter, some conclusions are drawn about the applications and operation conditions for which shear actuators are more effective.

Keywords: piezoelectric materials, thickness-shear mode, vibration control, smart structures

1 Introduction

Piezoelectric materials are widely used for structural vibration control due to their high performance, modal selectivity and low weight. They may be either bonded on or embedded in the host structure. In the first case, they are normally poled in the thickness direction so that, when submitted to an electric field through thickness, they present membrane forces that control structural vibrations. This “extension actuation mechanism” has been widely used on active control applications and hybrid active-passive damping treatments (Trindade *et al.*, 2000). However, extension piezoelectric actuators bonded on the host structure surfaces are subjected to high normal stresses and may be exposed to impacts, possibly leading to fracture in the normally brittle piezoceramic actuator.

The recent developments and increasing number of applications of composites design have brought attention to the use of embedded actuators. This may be obtained by replacing a laminae, or part of it, in a laminate composite structure by a piezoelectric actuator. However, although this concept may yield less structural integrity problems for the actuator, embedded extension actuators are not very efficient to control bending vibrations in this case. This is because their induced membrane forces lead to smaller bending moments when they are close to the neutral line. Some recent works (Sun and Zhang, 1995) proposed the use of the thickness-shear mode of piezoelectric actuators. In this case, the piezoelectric patches are poled in the axial direction and, when subjected to a transverse through thickness electric field, induce thickness shear stress in the structure. This leads to the less known “shear actuation mechanism”. Sun and Zhang (1995,1996) have shown that embedded shear actuators are subjected to lower stresses, under actuation.

The shear actuation mechanism was also studied by Benjeddou *et al.* (1997,1999). They have developed and validated sandwich beam finite element models able to treat extension and shear actuation mechanisms. These models were then used to compare the actuation mechanisms through a static analysis (Benjeddou *et al.*, 2000). The comparison showed that shear actuators induce distributed actuation moments in the structure unlike extension actuators which induce boundary forces. Therefore, shear actuation mechanism may lead to less problems of debonding in actuators boundaries and to minor dependence of the control performance on actuators position and length. Aldraihem and Khdeir (2000) performed a comparison between first-order and high-order beam theories to represent extension and shear actuation mechanisms. Later, Trindade *et al.* (1999) presented a comparison between active control performances of shear and extension actuation mechanisms. They showed that shear actuators are generally more suitable to control bending vibrations of stiff structures. Raja *et al.* (2000) also studied active damping performance in composite materials using shear actuators as compared to that using the widespread extension actuators. Their results showed that shear actuators have promising features for vibration control applications.

The present work extends the comparison of active control performances of shear and extension actuation mechanisms to higher modes of vibration with fixed maximum deflection. In particular, analysis of the damping performance for varying parameters is performed and used to compare both damping mechanisms. Thereafter, some conclusions are drawn about the applications and operation conditions for which the promising shear actuators are effective.

2 Theoretical formulation

A multilayer sandwich beam, with elastic and/or piezoelectric layers, is considered. One of the layers is allowed to perform membrane, bending and shear strains and will be named core layer. Thus, one may denote the upper and lower layers, relative to the core, as face sub-layers. These undergo only membrane and bending strains and hence are modeled

using classical laminate theory. This leads to a classical sandwich theory (face/core/face) with laminate faces. Euler-Bernoulli assumptions are considered for the laminate faces whereas, those of Timoshenko are retained for the core. The piezoelectric face sub-layers are supposed transversely poled and subject to transverse electric fields. On the contrary, the piezoelectric core layer is supposed longitudinally poled but also subject to transverse electric fields. Elastic layers are assumed insulated and are obtained by annulling the piezoelectric constants. All layers are assumed perfectly bonded and in plane stress state. The length, width and thickness of the beam are denoted by L , b and h , respectively. The subscripts a_j , b_j and c refer to quantities relative to the j -th sub-layer of upper a and lower b faces and to the core, respectively.

2.1 Kinematics

As stated before, classical sandwich beam theory is used here. This means that the same displacement field u_k ($k = a, b$) is considered for all sub-layers k_j of the laminate face k . Hence, despite the number n and m of sub-layers in the faces a and b respectively, only three displacements fields may be considered. Axial displacements $u_i(x, y, z)$ of the i -th layer ($i = a, b, c$) are assumed to vary in x -direction, vanish in y -direction and be linear in z -direction (through thickness). As for the transverse deflections $w_i(x, y, z)$, they are supposed to vary only in x -direction. From Euler-Bernoulli hypotheses, the cross-section rotation of the faces β_k ($k = a, b$) is related to the deflection derivative w' as $\beta_k = -w'$, where w' is used to denote $\partial w / \partial x$. The mid-plan of the core is set to coincide with the origin of the z -axis. Then, using the displacement continuity conditions between layers, one may write the displacements fields as

$$\begin{aligned} u_a(x, z) &= [\bar{u}(x) + \tilde{u}(x)/2] - (z - z_a)w(x)' \\ u_b(x, z) &= [\bar{u}(x) - \tilde{u}(x)/2] - (z - z_b)w(x)' \quad ; \quad w_a(x, z) = w_b(x, z) = w_c(x, z) = w(x) \\ u_c(x, z) &= [\bar{u}(x) + d w(x)'] + z [\tilde{u}(x)/h_c + \lambda w(x)'] \end{aligned} \quad (1)$$

where

$$d = (h_a - h_b)/4; \quad \lambda = (h_a + h_b)/2h_c; \quad z_k = \pm(h_k + h_c)/2; \quad h_k = \sum_{j=1}^{n,m} h_{k_j}$$

and the mean and relative axial displacements of the laminate faces \bar{u} and \tilde{u} are

$$\bar{u} = \frac{u_a + u_b}{2}; \quad \tilde{u} = u_a - u_b \quad (2)$$

Notice that these displacement fields are the same as those of (Benjeddou *et al.*, 1999), they are repeated here for clarity. Using the usual strain-displacement relations for each layer, in conjunction with relations (1), the axial ϵ_1 and shear ϵ_5 strains of the i -th layer ($i = a, b, c$) can be written as

$$\epsilon_{1i} = \epsilon_i^m + (z - z_i)\epsilon_i^b; \quad \epsilon_{5i} = \epsilon_i^s \quad (3)$$

where

$$\begin{aligned} \epsilon_a^m &= \bar{u}' + \frac{\tilde{u}'}{2}; & \epsilon_b^m &= \bar{u}' - \frac{\tilde{u}'}{2}; & \epsilon_c^m &= \bar{u}' + d w''; \\ \epsilon_a^b &= -w''; & \epsilon_b^b &= -w''; & \epsilon_c^b &= \frac{\tilde{u}'}{h_c} + \lambda w''; \\ \epsilon_a^s &= 0; & \epsilon_b^s &= 0; & \epsilon_c^s &= \frac{\tilde{u}}{h_c} + (\lambda + 1)w'. \end{aligned}$$

These generalized strains ϵ_i^m , ϵ_i^b and ϵ_i^s define the membrane, bending and shear strains of each layer. One may notice that for $d = 0$, that is for symmetric construction ($h_a = h_b$), the membrane strain of core becomes uncoupled to the bending strain, thus d is named membrane-bending coupling parameter. The parameter λ increases the influence of the cross-section rotation in the shear strain of the core. It augments with the faces to core thickness ratio, meaning that for fixed cross-section rotation and core thickness, thicker faces lead to higher shear strains.

A constant transverse electrical field is assumed for the piezoelectric layers and the remaining in-plane components are supposed to vanish. Although electrostatic equilibrium equation is only satisfied with a linear electrical field assumption in the faces (Benjeddou *et al.*, 1997; Rahmoune *et al.*, 1998), it was found that the linear part is negligible for small patches bonded on long structures. Consequently it is, for the k_j -th face piezoelectric sub-layer and core c ,

$$E_{3k_j} = -\frac{V_{k_j}}{h_{k_j}}; \quad E_{3c} = -\frac{V_c}{h_c} \quad (4)$$

where V_{k_j} , V_c are the difference of electric potential of the k_j -th and c laminae, defined by $V_{k_j} = V_{k_j}^+ - V_{k_j}^-$ and $V_c = V_c^+ - V_c^-$, where $V_{k_j}^+, V_c^+$ and $V_{k_j}^-, V_c^-$ are the voltages applied on the upper and lower skins of the k_j -th face and core c piezoelectric sub-layer.

2.2 Reduced constitutive equations

Linear orthotropic piezoelectric materials with material symmetry axes parallel to those of the beam are considered. c_{ij} , e_{lj} and ϵ_{ll} ($i, j = 1, \dots, 6; l = 1, 2, 3$) denote their elastic, piezoelectric and dielectric constants. For simplicity of notation, all layers will be considered piezoelectric. Elastic layers are obtained by making their piezoelectric constants vanish.

The three-dimensional linear constitutive equations of an orthotropic piezoelectric layer, poled in the thickness direction, can be reduced to (Benjeddou *et al.*, 1997)

$$\begin{Bmatrix} \sigma_{1k_j} \\ D_{3k_j} \end{Bmatrix} = \begin{bmatrix} c_{11}^{*k_j} & -e_{31}^{*k_j} \\ c_{31}^{*k_j} & \epsilon_{33}^{*k_j} \end{bmatrix} \begin{Bmatrix} \epsilon_{1k} \\ E_{3k_j} \end{Bmatrix}; \quad \epsilon_{33}^{*k_j} = \epsilon_{33}^{k_j} + \frac{e_{33}^{k_j 2}}{c_{33}^{k_j}}, \quad e_{31}^{*k_j} = e_{31}^{k_j} - \frac{c_{13}^{k_j}}{c_{33}^{k_j}} e_{33}^{k_j}, \quad c_{11}^{*k_j} = c_{11}^{k_j} - \frac{c_{13}^{k_j 2}}{c_{33}^{k_j}} \quad (5)$$

σ_{1k_j} and D_{3k_j} are axial stress components and transverse electrical displacement. Notice that electromechanical coupling in the piezoelectric faces sub-layers is between axial strain and transverse electrical field. This is the conventional piezoelectric *extension actuation mechanism*.

The piezoelectric material considered for the core is poled in the longitudinal direction. Hence, its constitutive equations may be obtained by a rotation of the three-dimensional constitutive equations of an orthotropic piezoelectric layer such that the directions x and z interchange. Thereafter, using the plane stress assumption, they can be reduced to

$$\begin{Bmatrix} \sigma_{1c} \\ \sigma_{5c} \\ D_{3c} \end{Bmatrix} = \begin{bmatrix} c_{33}^{*c} & 0 & 0 \\ 0 & c_{55}^c & -e_{15}^c \\ 0 & e_{15}^c & \epsilon_{11}^c \end{bmatrix} \begin{Bmatrix} \epsilon_{1c} \\ \epsilon_{5c} \\ E_{3c} \end{Bmatrix}; \quad c_{33}^{*c} = c_{33}^c - \frac{c_{13}^{c 2}}{c_{11}^c} \quad (6)$$

The axial electrical field E_{1c} is coupled with the axial strain of the core. However, its contribution to the electromechanical energy is neglected here, since only bending vibration is of interest. Consequently, it is reasonable to suppose the axial strain to be negligible, as the electrical field it induces. One may notice that the electromechanical coupling is between the transverse electrical field and the shear strains. This property characterizes the *shear actuation mechanism*.

2.3 Variational formulation

In this section, strains (3) and electric fields (4) expressions and constitutive equations (5) and (6) are used to derive a variational formulation in order to develop a finite element model. Hence, let us start with d'Alembert's principle extended to piezoelectric media,

$$\delta T - \delta H + \delta W = 0; \quad \forall \delta \bar{u}, \delta \bar{u}, \delta w, \delta V_{k_j}, \delta V_c \quad (7)$$

where δT , δH and δW are the virtual work of inertial, electromechanical internal and applied mechanical forces, respectively. δT and δW are described in (Trindade, 2000) and are not repeated here for simplicity. The virtual work of electromechanical internal forces virtual of the piezoelectric multilayer faces sandwich beam is

$$\delta H = \delta H_c + \sum_{k=a}^b \sum_{j=1}^{n,m} \delta H_{k_j} \quad (8)$$

where,

$$\delta H_{k_j} = \int_{\Omega_{k_j}} (\sigma_{1k_j} \delta \epsilon_{k1} - D_{3k_j} \delta E_{3k_j}) d\Omega_{k_j}; \quad \delta H_c = \int_{\Omega_c} (\delta \epsilon_{1c} \sigma_{1c} + \delta \epsilon_{5c} \sigma_{5c} - \delta E_{3c} D_{3c}) d\Omega_c \quad (9)$$

Ω_{k_j} and Ω_c are the volume of the k_j -th sub-layer and the core c , respectively.

Using strain (3) and electric field (4) relations and constitutive equations (5) and (6), then integrating through thickness, the above equations, for the piezoelectric k_j -th face sub-layer and core c , become

$$\delta H_{k_j} = \int_0^L \left[c_{11}^{*k_j} \left(A_{k_j} \delta \epsilon_k^m \epsilon_k^m + \bar{I}_{k_j} \delta \epsilon_k^m \epsilon_k^b + \bar{I}_{k_j} \delta \epsilon_k^b \epsilon_k^m + I_{k_j} \delta \epsilon_k^b \epsilon_k^b \right) + e_{31}^{*k_j} \left(A_{k_j} \delta \epsilon_k^m + \bar{I}_{k_j} \delta \epsilon_k^b \right) \frac{V_{k_j}}{h_{k_j}} + e_{31}^{*k_j} \frac{\delta V_{k_j}}{h_{k_j}} \left(A_{k_j} \epsilon_k^m + \bar{I}_{k_j} \epsilon_k^b \right) - \epsilon_{33}^{*k_j} A_{k_j} \frac{\delta V_{k_j}}{h_{k_j}} \frac{V_{k_j}}{h_{k_j}} \right] dx \quad (10)$$

$$\delta H_c = \int_0^L \left[c_{33}^{*c} \left(A_c \delta \epsilon_c^m \epsilon_c^m + I_c \delta \epsilon_c^b \epsilon_c^b \right) + k_c c_{55}^c A_c \delta \epsilon_c^s \epsilon_c^s + e_{15}^c A_c \delta \epsilon_c^s \frac{V_c}{h_c} + e_{15}^c A_c \frac{\delta V_c}{h_c} \epsilon_c^s - \epsilon_{11}^c A_c \frac{\delta V_c}{h_c} \frac{V_c}{h_c} \right] dx \quad (11)$$

where k_c is the shear correction factor. Notice that here, unlike in the three-layer sandwich beam model of (Benjeddou *et al.*, 1999), there are membrane-bending coupling terms due to the multilayer characteristic of the faces. A_{k_j} , \bar{I}_{k_j} and I_{k_j} are, respectively, the area and the first and second moments of inertia of the k_j -th face sub-layer cross section. These are

$$[A_{k_j}, \bar{I}_{k_j}, I_{k_j}] = \int_{-b/2}^{b/2} \int_{z_{k_j}-h_{k_j}/2}^{z_{k_j}+h_{k_j}/2} [1, (z-z_k), (z-z_k)^2] dz dy \quad (12)$$

where the local z -axis of the k_j -th sub-layer is situated at

$$z_{k_j} = \pm \frac{h_{k_j} + h_c}{2} \pm \sum_{r=1}^{j-1} h_{k_r}; \quad k = a(+), b(-) \quad (13)$$

The area and second moment of inertia of the core are $[A_c, I_c] = b [h, h^3/12]$ and its first moment of inertia vanishes since the core is symmetric. One may notice from (10) that for an applied difference of potential V_{k_j} on the k_j -th face piezoelectric sub-layer, the variations δV_{k_j} vanish, in the last two terms of δH_{k_j} , and its second term results in the virtual work $\delta H_{k_j p}$ of a generalized piezoelectric load equivalent to

$$\delta H_{k_j p} = - \int_0^L e_{31}^{*k_j} \left(A_{k_j} \delta \epsilon_k^m + \bar{I}_{k_j} \delta \epsilon_k^b \right) \frac{V_{k_j}}{h_{k_j}} dx \quad (14)$$

when moved to the right hand side of (7). For homogeneous properties in the axial direction, according to (3), this may also be expressed as the virtual work of boundary generalized piezoelectric loads,

$$\delta H_{k_j p} = -e_{31}^{*k_j} \left[A_{k_j} \left(\delta \bar{u} \pm \frac{\delta \tilde{u}}{2} \right) - \bar{I}_{k_j} \delta w' \right]_0^L \frac{V_{k_j}}{h_{k_j}} \quad (15)$$

One may notice from the last term of (15) that the piezoelectric layers may induce bending on the sandwich beam due to layered faces asymmetry, unlike in the case of single-layer faces for which only mean and relative axial displacements of the faces are induced (Benjeddou *et al.*, 1999).

Performing the same analysis for an applied electric field in the piezoelectric core, one may notice from (11) that, unlike for extension actuators and even for homogeneous properties, shear actuators always induce distributed loads which virtual work may be expressed as

$$\delta H_{c p} = e_{15}^c A_c \left\{ \int_0^L \left[\frac{\delta \tilde{u}}{h_c} + (\lambda + 1) \delta w' \right] dx \right\} \frac{V_c}{h_c} \quad (16)$$

3 Finite element formulation

From the variational formulation and virtual work expressions presented in the last section, a finite element model was developed for the laminate faces adaptive sandwich beam. It assumes Lagrange linear shape functions for the mean and relative axial displacements, \bar{u} and \tilde{u} and Hermite cubic ones for the transverse deflection w . The difference of electric potentials V_{k_r} ($r = 1, \dots, (\hat{n}, \hat{m})$) and V_c of the \hat{n}, \hat{m} face piezoelectric sub-layers and core c are assumed constant in the element (Figure 1). This leads to the following elementary degrees of freedom (dof) column vector $\hat{\mathbf{q}}_e$,

$$\hat{\mathbf{q}}_e = \text{col}(\bar{u}_1, w_1, w_1', \bar{u}_1, \bar{u}_2, w_2, w_2', \bar{u}_2, V_{a_1 e}, \dots, V_{a_{\hat{n}} e}, V_{b_1 e}, \dots, V_{b_{\hat{m}} e}, V_{c e}) \quad (17)$$

Based on this dof vector and relations (10) and (11), the discretized virtual work of the elementary electromechanical internal forces of the face sub-layers $\delta H_{k_j}^e$ and the core δH_c^e are

$$\begin{aligned} \delta H_{k_j}^e &= \delta \hat{\mathbf{q}}_e^T \left(\hat{\mathbf{K}}_{k_j m}^e - \hat{\mathbf{K}}_{k_j p}^e - \hat{\mathbf{K}}_{k_j p}^{e T} + \hat{\mathbf{K}}_{k_j d}^e \right) \hat{\mathbf{q}}_e = \delta \hat{\mathbf{q}}_e^T \hat{\mathbf{K}}_{k_j}^e \hat{\mathbf{q}}_e \\ \delta H_c^e &= \delta \hat{\mathbf{q}}_e^T \left(\hat{\mathbf{K}}_{c m}^e - \hat{\mathbf{K}}_{c p}^e - \hat{\mathbf{K}}_{c p}^{e T} + \hat{\mathbf{K}}_{c d}^e \right) \hat{\mathbf{q}}_e = \delta \hat{\mathbf{q}}_e^T \hat{\mathbf{K}}_c^e \hat{\mathbf{q}}_e \end{aligned} \quad (18)$$

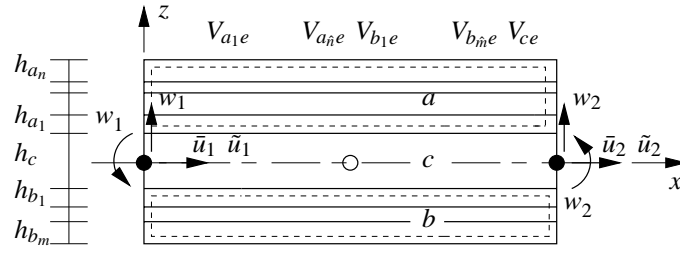


Figure 1. Piezoelectric laminated sandwich beam finite element.

where the elementary mechanical $\hat{\mathbf{K}}_{kjm}^e$, $\hat{\mathbf{K}}_{cm}^e$, piezoelectric $\hat{\mathbf{K}}_{kjp}^e$, $\hat{\mathbf{K}}_{cp}^e$ and dielectric $\hat{\mathbf{K}}_{kjd}^e$, $\hat{\mathbf{K}}_{cd}^e$ stiffness matrices have the expressions

$$\begin{aligned}\hat{\mathbf{K}}_{kjm}^e &= \int_0^{L_e} c_{11}^{*kj} [A_{kj} \mathbf{B}_{km}^T \mathbf{B}_{km} + \bar{I}_{kj} (\mathbf{B}_{km}^T \mathbf{B}_{kb} + \mathbf{B}_{kb}^T \mathbf{B}_{km}) + I_{kj} \mathbf{B}_{kb}^T \mathbf{B}_{kb}] dx \\ \hat{\mathbf{K}}_{cm}^e &= \int_0^{L_e} [c_{33}^{*c} (A_c \mathbf{B}_{cm}^T \mathbf{B}_{cm} + I_c \mathbf{B}_{cb}^T \mathbf{B}_{cb}) + k_c c_{55}^c A_c \mathbf{B}_{cs}^T \mathbf{B}_{cs}] dx \\ \hat{\mathbf{K}}_{kjp}^e &= - \int_0^{L_e} c_{31}^{*kj} \frac{1}{h_{kj}} [A_{kj} \mathbf{B}_{km}^T + \bar{I}_{kj} \mathbf{B}_{kb}^T] \mathbf{N}_{pkj} dx; \quad \hat{\mathbf{K}}_{cp}^e = - \int_0^{L_e} e_{15}^c \frac{A_c}{h_c} \mathbf{B}_{cs}^T \mathbf{N}_{pc} dx \\ \hat{\mathbf{K}}_{kjd}^e &= - \int_0^{L_e} \epsilon_{33}^{*kj} \frac{A_{kj}}{h_{kj}^2} \mathbf{N}_{pkj}^T \mathbf{N}_{pkj} dx; \quad \hat{\mathbf{K}}_{cd}^e = - \int_0^{L_e} \epsilon_{11}^c \frac{A_c}{h_c^2} \mathbf{N}_{pc}^T \mathbf{N}_{pc} dx\end{aligned}\quad (19)$$

L_e is the element length. \mathbf{B}_{km} , \mathbf{B}_{cm} , \mathbf{B}_{kb} and \mathbf{B}_{cb} are the, faces (k) and core (c), membrane (m) and bending (b) strain operators. \mathbf{B}_{cs} is the core shear strain operator and, \mathbf{N}_{pkj} and \mathbf{N}_{pc} are the difference of electric potential interpolation matrices for the faces and core, respectively.

Assembling on the faces sub-layers, then on the beam layers, the discretized variational equation (7) reduced to the element level can be written as,

$$(\hat{\mathbf{M}}_f^e + \hat{\mathbf{M}}_c^e) \ddot{\mathbf{q}}_e + (\hat{\mathbf{K}}_f^e + \hat{\mathbf{K}}_c^e) \mathbf{q}_e = \hat{\mathbf{F}}_m^e \quad (20)$$

where $\hat{\mathbf{M}}_f^e = \sum_k \sum_j \hat{\mathbf{M}}_{kj}^e$ and $\hat{\mathbf{K}}_f^e = \sum_k \sum_j \hat{\mathbf{K}}_{kj}^e$. One may notice that, since electrical dofs inertia vanishes, mechanical and electrical dofs are coupled statically only (time-independent relationship). Therefore, two different cases may be considered for each piezoelectric sub-layer: applied difference of potentials (actuator) or unknown difference of potentials (sensor). Let us define the corresponding subgroups \mathbf{V}_A^e and \mathbf{V}_S^e of the elementary electrical dofs \mathbf{V}^e . Then, the vector \mathbf{q}_e in (17) may be decomposed such that $\mathbf{q}_e = \text{col}(\mathbf{q}_e, \mathbf{V}_S^e, \mathbf{V}_A^e)$. Consequently, the system (20) becomes

$$\begin{bmatrix} \mathbf{M}^e & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{Bmatrix} \ddot{\mathbf{q}}_e \\ \ddot{\mathbf{V}}_S^e \\ \ddot{\mathbf{V}}_A^e \end{Bmatrix} + \begin{bmatrix} \mathbf{K}_m^e & -\mathbf{K}_{meS}^e & -\mathbf{K}_{meA}^e \\ -\mathbf{K}_{meS}^{eT} & \mathbf{K}_{eS}^e & \mathbf{0} \\ -\mathbf{K}_{meA}^{eT} & \mathbf{0} & \mathbf{K}_{eA}^e \end{bmatrix} \begin{Bmatrix} \mathbf{q}_e \\ \mathbf{V}_S^e \\ \mathbf{V}_A^e \end{Bmatrix} = \begin{Bmatrix} \mathbf{F}_m^e \\ \mathbf{0} \\ \mathbf{0} \end{Bmatrix} \quad (21)$$

Since the electrical dofs \mathbf{V}_A^e are imposed, their virtual variations $\delta \mathbf{V}_A^e$ vanish. Therefore, the third equation of (21) is automatically satisfied and may be ignored. Also the corresponding term to \mathbf{V}_A^e in the first equation can be moved to the right hand side as an equivalent electrical work, defined by

$$\mathbf{F}_e^e = \mathbf{K}_{meA}^e \mathbf{V}_A^e \quad (22)$$

The second equation of (21) can then be used to express the unknown potentials \mathbf{V}_S^e in terms of the mechanical dofs \mathbf{q}_e , such as

$$\mathbf{V}_S^e = \mathbf{K}_{eS}^{e-1} \mathbf{K}_{meS}^{eT} \mathbf{q}_e \quad (23)$$

Replacing expressions (22) and (23) in the first equation of (21) leads to the following electrical dof condensed elementary system

$$\mathbf{M}^e \ddot{\mathbf{q}}_e + (\mathbf{K}_m^e - \mathbf{K}_{meS}^e \mathbf{K}_{eS}^{e-1} \mathbf{K}_{meS}^{eT}) \mathbf{q}_e = \mathbf{F}_m^e + \mathbf{F}_e^e \quad (24)$$

Hence, the mechanical dofs, due to mechanical and/or electrical loads \mathbf{F}_m^e and \mathbf{F}_e^e , may be evaluated first, then unknown electrical dofs are found through a post-processing calculation using (23). This method not only leads to faster calculations since matrices dimensions are lower but also prevents ill-conditioning problems of solving directly equation (21). Therefore, both piezoelectric actuators and sensors can be considered in a closed-loop analysis.

Then, one may assemble this elementary system to get the corresponding global mass and stiffness matrices, and mechanical and electrical load vectors, \mathbf{M} , \mathbf{K}_f , \mathbf{K}_c , \mathbf{F}_m and \mathbf{F}_e . Also, a standard viscous damping matrix \mathbf{D} may be considered *a posteriori*. Consequently, the assembled system becomes

$$\mathbf{M}\ddot{\mathbf{q}} + \mathbf{D}\dot{\mathbf{q}} + (\mathbf{K}_f + \mathbf{K}_c)\mathbf{q} = \mathbf{F}_m + \mathbf{F}_e \quad (25)$$

4 Model reduction

In order to apply the equations of motion to control design, a model reduction is normally required. Here, equations (25) are reduced through projection on a truncated modal basis Φ such that the finite element dofs vector \mathbf{q} are approximated by $\mathbf{q} = \Phi\chi$, with χ being the modal displacements. Thus, (25) may be rewritten as

$$\ddot{\chi} + 2\zeta\Omega\dot{\chi} + \Omega^2\chi = \mathcal{F}_m + \mathcal{F}_e \quad (26)$$

where the eigenvectors Φ are mass-normalized such that $\Phi^T\mathbf{M}\Phi = \mathbf{I}$ and Ω states for the undamped eigenfrequencies matrix. The damping matrix \mathbf{D} is chosen such that the resulting projected modal damping matrix is diagonal and proportional to the eigenfrequencies matrix and ζ defines a modal damping factor equal for all modes considered. Then, one gets

$$\Omega^2 = \Phi^T(\mathbf{K}_f + \mathbf{K}_c)\Phi \quad (27)$$

Notice that these eigenfrequencies already account for induced electric fields in the piezoelectric sensor layers. The modal perturbation and control forces contributions are obtained through projection of \mathbf{F}_m and \mathbf{F}_e in the modal basis Φ ,

$$\mathcal{F}_m = \Phi^T\mathbf{F}_m; \quad \mathcal{F}_e = \Phi^T\mathbf{F}_e \quad (28)$$

The projection of piezoelectric control forces \mathbf{F}_e in the modal basis presents an important meaning for control design. This is because the magnitude of the elements of \mathcal{F}_e represents a measure of the controllability of the corresponding mode. Hence, the greater the projection of \mathbf{F}_e for a given mode, the greater is the ability of this force to control this mode.

For control design, the second order equations (26) are transformed into state-space form as follows

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{p} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} \end{aligned}, \quad \text{with } \mathbf{x} = \begin{Bmatrix} \chi \\ \dot{\chi} \end{Bmatrix} \quad (29)$$

The output variables vector \mathbf{y} is written as combinations of the state variables, through matrix \mathbf{C} . However, the state variables, that are modal displacements χ and velocities $\dot{\chi}$, are difficult to measure. Nevertheless, as these may be expressed in terms of the finite element dofs and their derivatives, one may consider an output vector $\mathbf{y} = \mathbf{C}_p\mathbf{q} + \mathbf{C}_d\dot{\mathbf{q}}$, so that matrices \mathbf{C}_p and \mathbf{C}_d allow to measure nodal displacements and velocities. Then, the output matrix \mathbf{C} may be written in terms of \mathbf{C}_p and \mathbf{C}_d as shown below. Similarly, \mathbf{A} , \mathbf{B} and \mathbf{p} , which are the system dynamics, input distribution and perturbation matrices, respectively, are given by

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\Omega^2 & -2\zeta\Omega \end{bmatrix}; \quad \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathcal{F}_e^* \end{bmatrix}; \quad \mathbf{p} = \begin{bmatrix} \mathbf{0} \\ \mathcal{F}_m \end{bmatrix}; \quad \mathbf{C} = [\mathbf{C}_p\Phi \quad \mathbf{C}_d\Phi] \quad (30)$$

To account for the control input \mathbf{u} , which is here the voltage applied to the actuators, the input distribution matrix \mathbf{B} is written in terms of a potential factored-out piezoelectric force vector \mathcal{F}_e^* . This is defined as the piezoelectric force \mathcal{F}_e for a unit applied voltage on the corresponding actuator.

5 Control design

The state-space system (29) is now applied to the design of an optimal controller. The control algorithm considered here is an iterative version of the Linear Quadratic Regulator (LQR) in order to find an optimal linear feedback controller while respecting a prescribed maximum voltage applied to the piezoelectric actuators to prevent their depoling. Hence, a full state feedback control $\mathbf{u} = -\mathbf{K}_g\mathbf{x}$ is considered to minimize the functional

$$J = \frac{1}{2} \int_0^{\infty} (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt \quad (31)$$

subjected to the linear constraints (29) and to the voltage limitation $\mathbf{u} < V_{max}$. Evidently, the performance of this controller depends on the state \mathbf{Q} and input \mathbf{R} weight matrices, where \mathbf{Q} defines the relative weight of each state variable and \mathbf{R} , the relative weight of each actuator voltage. The latter is supposed to be in the form $\mathbf{R} = \gamma \mathbf{I}$ and the factor γ is then adjusted automatically to limit the maximum voltage, through an iterative algorithm. Consequently, the control gain matrix $\mathbf{K}_g = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P}$ is evaluated, for each value of γ , by solving for \mathbf{P} the following algebraic Riccati equation

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - \mathbf{P} \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} + \mathbf{Q} = \mathbf{0} \quad (32)$$

It is clear that the computational cost of this iterative algorithm increases greatly with the dimension of the state-space system. However, the convergence is generally very fast. One may also notice that the maximum control voltage must be evaluated in terms of the dynamics of the system for a given perturbation. Hence, the voltage limitation leads to a perturbation-dependent performance of the controller.

6 Numerical results

The reduced model together with the control algorithm are now used to investigate the active control performances of extension and shear actuation mechanisms. For that, consider the cantilever sandwich beam with faces and core made of steel and rigid foam respectively as shown in Figure 2. This host structure may be controlled either by bonding an extension piezoactuator in the upper surface of the sandwich (Figure 2a) or by inserting a shear piezoactuator in the sandwich core, replacing the existing foam (Figure 2b). The geometrical properties of the beam are: $L = 200$ mm, $h_b = 10$ mm, $h_c = 0.2$ mm, $p = 10$ mm and $a = 50$ mm. The piezoactuators are made of PZT5H piezoceramic material whose properties are: $\rho = 7500$ kg m⁻³, $c_{11}^* = 65.5$ GPa, $e_{31}^* = -23.2$ C m⁻², $\epsilon_{33}^* = 1.54 \cdot 10^{-8}$ F m⁻¹, for the extension actuator and $c_{33}^* = 60.9$ GPa, $c_{55} = 23$ GPa, $e_{15} = 17$ C m⁻², $\epsilon_{11} = 1.3 \cdot 10^{-8}$ F m⁻¹, for the shear actuator. Steel properties are: $c_{11}^* = 210$ GPa and $\rho = 7800$ kg m⁻³ and those of the rigid foam are: $c_{11}^* = 35.3$ MPa, $c_{55} = 12.6$ MPa and $\rho = 32$ kg m⁻³.

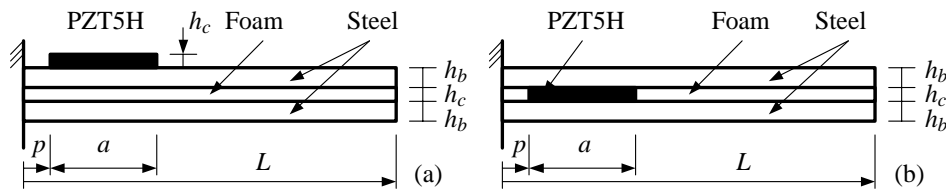


Figure 2. Cantilever beam controlled by extension (a) and shear (b) actuation mechanisms.

The first fifteen bending modes are considered in the reduced model (29), using 50 piezoelectric multilayer sandwich beam finite elements. The control design is carried out using the iterative LQR algorithm with the following state weight matrix $\mathbf{Q} = [\omega_1^2, \dots, \omega_{15}^2, 0, \dots, 0]$. The control gain matrix \mathbf{K}_g is then evaluated for a transverse perturbation force applied on the tip of the beam, such that the maximum deflection amplitude is $1 \mu\text{m}$ and the maximum voltage applied to the piezoactuators is 100 V, that is an electric field of 500 V/mm.

Figure 3 presents the open- (solid line) and closed- (dashed line) loop frequency response functions for the tip displacement of the beam actuated by (a) extension and (b) shear actuators. One may notice that the first and fourth mode are better damped by the extension actuator, but the damping of other modes are generally higher due to the shear actuator control. It is clear though, from Figure 3, that the systems resulting from extension and shear actuation mechanisms present different behaviors.

For clarity purposes, the closed-loop damping factors for the first fifteen bending modes using extension (red bar) and shear (blue bar) actuators are presented in Figure 4. As guessed previously, from Figure 3, the first mode is much more actively damped by extension actuators. This is clearly due to the spatial shape of the deformed configuration resulting from the two actuation mechanisms, which is closer to the first mode for extension actuators (Benjeddou *et al.*, 1997; Trindade *et al.*, 1999). One may also observe that the fourth and sixth modes are better damped by extension actuators. However, the damping performance for all the other modes is higher when using shear actuators. This is better understood if one observes that the closed-loop damping furnished by the extension actuators (red bars) present an overall decrease with mode number. On the other hand, the damping performance of shear actuators is quite uniform globally. This fact has two consequences: i) shear actuators may be better to control higher modes since their actuation induces highly deformed shapes in the structure, and ii) shear actuation mechanism should be a better option for the control within higher frequency ranges.

An alternative manner of evaluating the suitability of each actuation mechanism to control specific vibration modes consists in an analysis of the projection of piezoelectric control forces in the modal basis. This provides a measure of

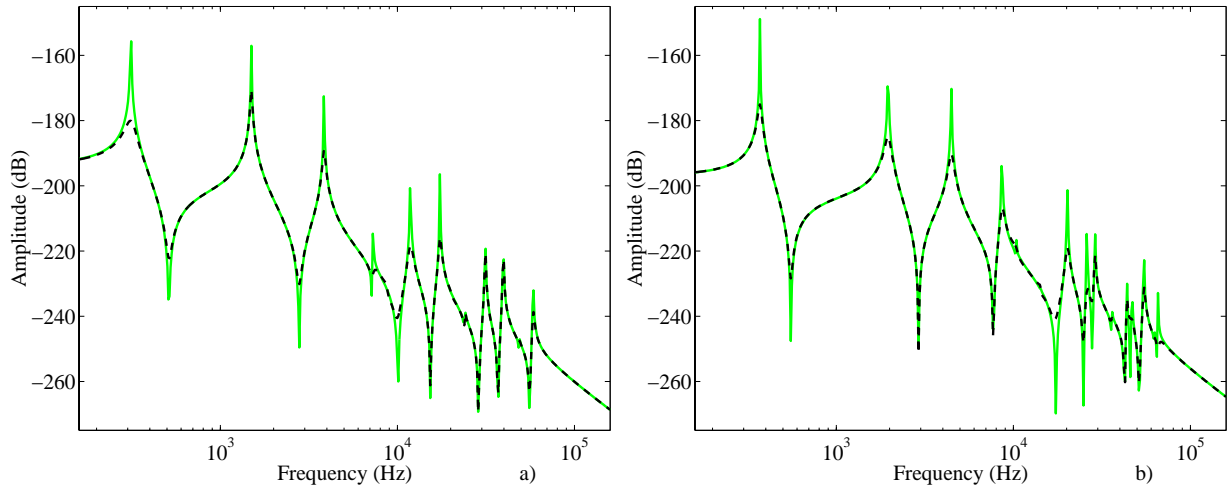


Figure 3. Open- (solid line) and closed- (dashed line) loop frequency response functions for the tip displacement of the beam actuated by (a) extension and (b) shear actuators.

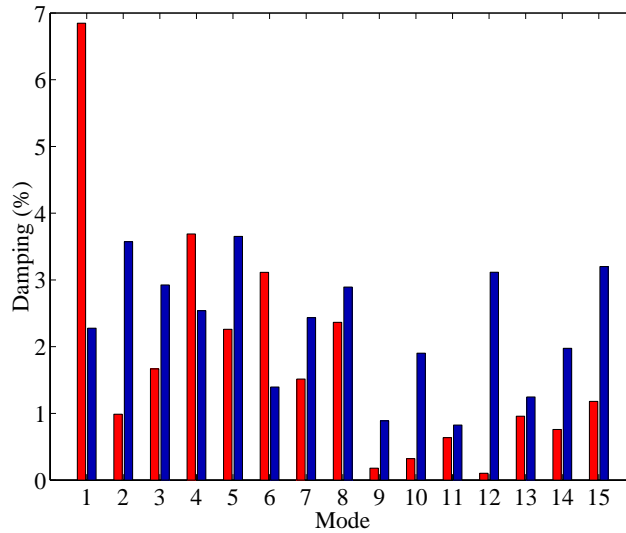


Figure 4. Closed-loop modal damping for the first fifteen bending modes using extension (red bar) and shear (blue bar) actuators.

the controllability of the each mode. Hence, Table 1 shows the normalized controllabilities of the first fifteen bending modes for the extension and shear actuated beams. These are obtained by equalization of the control vectors \mathcal{F}_e^* evaluated for both actuation mechanisms. One may notice, from Table 1, that except for the first, fourth and sixth modes, shear actuator controllabilities are much higher than the corresponding extension actuator ones. This is in conformity with the previous observations of damping performance. However, comparison of each mode controllability for a single actuation mechanism does not provide a reliable measure of damping performance ability, since damping performance depends highly also on the state weight matrix. That is why, for instance, although controllability of first mode is higher than that of fifteenth mode for the extension actuator, the first mode is better controlled by this actuator. On the other hand, one may observe that it is almost impossible to control the twelfth mode with extension actuators and this is confirmed by the weak damping performance for this mode (Figure 4). Moreover, the opposite is true for shear actuators which controls very well this mode.

Table 1. Normalized controllabilities of the first fifteen bending modes for the extension and shear actuated beams.

	1	2	3	4	5	6	7	8	9	10	11
Extension	2.09	1.42	6.22	26.28	19.47	35.64	25.68	46.24	3.45	9.31	19.90
Shear	1.49	13.67	22.45	39.25	67.43	35.77	84.84	143.75	47.16	115.83	79.08
	12	13	14	15							
Extension	1.00	42.29	35.63	65.97							
Shear	256.28	124.15	280.65	355.94							

It is also interesting to analyze the influence of actuator position and foam stiffness in the active damping performance of both mechanisms. For sake of clarity, these analyses are carried out with the mean of the first fifteen modal damping factors as measure of performance. Figure 5 shows that the performance of extension actuators (red bars) is almost uniform in the position range, whereas that of shear actuators (blue bars) increases up to 3.5% in the medium position range. It is clear that this mean damping factor provides only a general behavior and, while first mode damping is optimal for extension actuators near the clamped end, as expected, higher modes are better damped for extension actuators near the beam tip. Nevertheless, although the designer may perform an optimization following the damping requirements for each mode, here the objective is to show that shear actuators may be quite effective for unexpected configurations.

Another important design parameter is the foam stiffness. Figure 6 shows its influence in the mean modal damping using extension (red bar) and shear (blue bar) actuators. Notice that the \log_{10} of a magnifying foam stiffness factor, with reference to the foam modulus given previously, is used in Figure 6. One may observe that extension actuators loose performance with the increase of foam stiffness, which means that, as expected, these actuators become ineffective for stiffer beams. On the other hand, shear actuators provide an almost uniform mean damping performance within the foam stiffness range. Nevertheless, as shown by Trindade *et al.* (1999), the optimal foam stiffness for shear actuation mechanism varies from a mode to another.

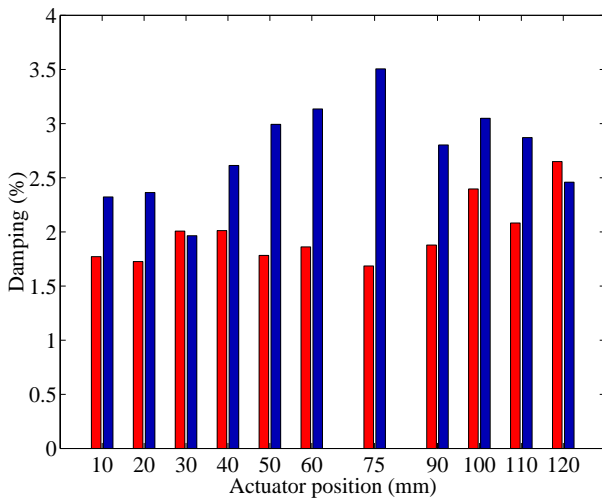


Figure 5. Variation of the mean modal damping using extension (red bar) and shear (blue bar) actuators with actuator position.

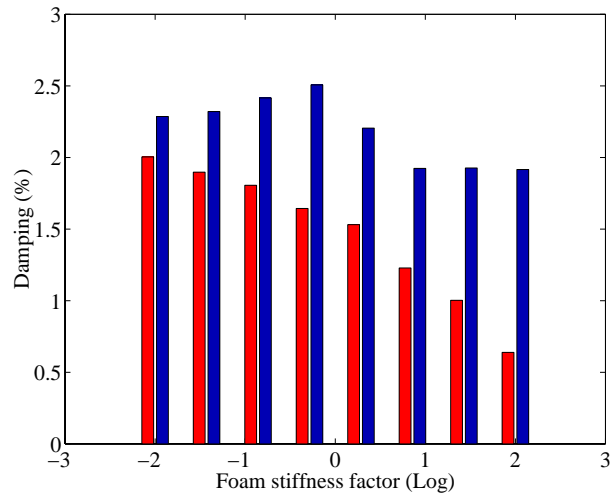


Figure 6. Variation of the mean modal damping using extension (red bar) and shear (blue bar) actuators with foam stiffness factor.

7 Conclusions

An analysis of the application of thickness-shear mode of piezoelectric actuators to active vibration control was presented. This was done using a piezoelectric multilayer sandwich beam finite element model proposed in a previous work. This model was then reduced through projection in an appropriate modal basis and applied to the design of an input constrained optimal control algorithm.

The active control performances of the proposed shear actuation mechanism and the widespread used extension one were presented. The comparison has shown that shear actuators may be more effective than extension ones to control small amplitude vibrations in a wider frequency range. They were shown to be particularly effective for stiff structures.

This work is now being extended to account for several piezoactuators distributed in both axial and transverse directions for application in adaptive composites design. New configurations combining shear and extension actuation mechanisms are also being accounted for.

8 Acknowledgment

The author gratefully acknowledges the financial support of Brazilian “Conselho Nacional de Desenvolvimento Científico e Tecnológico” (CNPq) through a research scholarship, grant No.300512-2000/7.

9 References

- Aldraihem, O.J. and Khdeir, A.A., 2000, “Smart beams with extension and thickness-shear piezoelectric actuators”, *Smart Materials and Structures*, Vol.9, No.1, pp.1–9.
- Benjeddou, A., Trindade, M.A. and Ohayon, R., 1997, “A unified beam finite element model for extension and shear piezoelectric actuation mechanisms”, *Journal of Intelligent Material Systems and Structures*, Vol.8, No.12, pp.1012–1025.

- Benjeddou, A., Trindade, M.A. and Ohayon, R., 1999, "New shear actuated smart structure beam finite element", *AIAA Journal*, Vol.37, No.3, pp.378–383.
- Benjeddou, A., Trindade, M.A. and Ohayon, R. 2000, "Piezoelectric actuation mechanisms for intelligent sandwich structures", *Smart Materials and Structures*, Vol.9, No.3, pp.328–335.
- Raja, S., Prathap, G. and Sinha, P.K., 2000, "Active vibration control of composite sandwich beams with distributed piezoelectric extension-bending and shear actuators", *SPIE's Smart Materials and MEMS*, Vol.4234, paper #13.
- Rahmoune, M., Benjeddou, A., Ohayon, R. and Osmont, D., 1998, "New thin piezoelectric plate models", *Journal of Intelligent Material Systems and Structures*, Vol.9, No.12, pp.1017–1029.
- Sun, C.T. and Zhang, X.D., 1995, "Use of thickness shear mode in adaptive sandwich structures", *Smart Materials and Structures*, Vol.4, No.3, pp.202–206.
- Trindade, M.A., 2000, "Contrôle hybride actif-passif des vibrations de structures par des matériaux piézoélectriques et viscoélastiques : poutres sandwich/multicouches intelligentes", *Thèse de Doctorat, Conservatoire National des Arts et Métiers, Paris.*
- Trindade, M.A., Benjeddou, A. and Ohayon, R., 2000, "Modeling of frequency-dependent viscoelastic materials for active-passive vibration damping", *Journal of Vibration and Acoustics*, Vol.122, No.2, pp.169–174.
- Trindade, M.A., Benjeddou, A. and Ohayon, R., 1999, "Parametric analysis of the vibration control of sandwich beams through shear-based piezoelectric actuation", *Journal of Intelligent Material Systems and Structures*, Vol.10, No.5, pp.377–385.
- Zhang, X.D. and Sun, C.T., 1996, "Formulation of an adaptive sandwich beam", *Smart Materials and Structures*, Vol.5, No.6, pp.814–823.

CONTROLADOR NEURAL ADAPTATIVO DIRETO APLICADO EM UM ATUADOR HIDRÁULICO

José Antônio Riul

Departamento de Tecnologia Mecânica – CT–UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, e-mail:riul@ct.ufpb.br

José Felício da Silva

Departamento de Tecnologia Mecânica – CT–UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, e-mail:felicio@funape.ufpb.br

José Homero. F. Cavalcanti

Departamento de Tecnologia Mecânica – CT–UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, e-mail:homero@ct.ufpb.br

Resumo. Este trabalho tem como objetivo a apresentação de um sistema de controle não-linear para o posicionamento de uma carga acionada por atuadores hidráulicos. A estratégia de controle proposta é realizada em duas etapas, utilizando-se redes neurais multi-camadas. Na primeira etapa os parâmetros de uma rede neural são ajustados, através de treinamento supervisionado; e na segunda, o controle da planta é efetuado de forma similar ao treinamento, a partir dos parâmetros obtidos na primeira etapa. Resultados de simulação são mostrados para a avaliação do desempenho do sistema proposto.

Palavras-chave: Controle de sistemas hidráulicos, Atuadores hidráulicos, Redes neurais.

1. Introdução

O objetivo do presente trabalho é o projeto de um sistema de controle não-linear para posicionamento de uma carga acionada por atuadores hidráulicos. Estes atuadores são utilizados onde cargas elevadas devem ser manipuladas, uma vez que fornecem força em alta escala. Como a compressibilidade do fluido pode causar efeitos indesejáveis na resposta dinâmica de sistemas acionados por atuadores hidráulicos, estes não são usados em aplicações onde exige-se pequenos erros de posicionamento, como por exemplo em robôs manipuladores. Riul et al. (1999) projetaram um controlador neural inverso direto para posicionamento de um atuador hidráulico, e constataram que o erro de posicionamento era pequeno quando não existia perturbação no sistema. Se o sistema fosse perturbado, o erro aumentava e a resposta não acompanhava o sinal de referência imposto. Visando eliminar o problema verificado com a utilização do controlador neural inverso direto, quando o sistema era submetido a perturbações, Riul et al. (2000) projetaram um controlador híbrido (neural inverso direto e proporcional) e verificaram que o erro de posicionamento do atuador hidráulico diminuiu consideravelmente.

Neste trabalho usa-se um controlador neural adaptativo direto para o posicionamento proposto. O controlador é projetado em duas etapas. Na primeira, uma rede neural multi-camadas (Haykin, 1994), (Beale et al., 1991) é treinada ‘off line’ tendo como entradas a posição, a velocidade e a aceleração de referência, especificadas para a planta, e como saída a variável de controle da planta. O controlador neural tem seus pesos ajustados pelo algoritmo de propagação retroativa do erro entre a saída desejada e a obtida para a carga, através da minimização de um índice de desempenho. Na segunda etapa, o controle do sistema é realizado ‘on line’ de forma similar ao do treinamento, a partir dos parâmetros determinados na primeira etapa. Para avaliação do desempenho do sistema de controle projetado, resultados de simulação são mostrados e comparados com os obtidos através de um controlador proporcional-derivativo.

2. Modelo Dinâmico do Atuador Hidráulico

O atuador hidráulico a ser analisado é composto basicamente de uma válvula e de um cilindro hidráulico conforme mostrado na Fig. (1). A válvula é do tipo carretel de quatro vias e centro crítico (Guenther e De Pieri, 1997). Na Fig. (1), P_s é a pressão de suprimento, P_d é a pressão de descarga, P_1 e P_2 são as pressões nas câmaras 1 e 2 do cilindro hidráulico, v_1 e v_2 são os volumes das câmaras 1 e 2 do cilindro hidráulico, Q_1 é a vazão da válvula para a câmara 1, Q_2 é a vazão da câmara 2 para o reservatório, M é a massa do sistema, c é o coeficiente de atrito viscoso, F_p é uma força perturbadora, u é a posição do carretel da válvula e y é a posição da carga do sistema.

Para obter-se o modelo dinâmico do atuador hidráulico, usa-se a equação de vazão em orifícios, a equação da continuidade e o princípio de D’Alembert. O modelo resultante (Cunha et al., 1998) que relaciona a saída y e a entrada u é não-linear e representado conforme Eq. (1) e Eq. (2).

$$M\ddot{y} + c\dot{y} - F_p = AP_c \tag{1}$$

$$\dot{P}_c = - \frac{\beta v}{\left(\frac{v}{2}\right)^2 - (Ay)^2} A\dot{y} + \frac{\beta v}{\left(\frac{v}{2}\right)^2 - (Ay)^2} K_1 \sqrt{P_s - \text{sign}(u)P_c} \cdot u \tag{2}$$

onde: A é a área da seção transversal do pistão hidráulico, $P_c = P_1 - P_2$ é a diferença de pressão nas câmaras 1 e 2 do cilindro, β é o módulo de elasticidade volumétrico do fluido, $v = v_1 + v_2$ é o volume total do cilindro e K_1 é uma constante obtida em função da massa específica do fluido, do gradiente da área da válvula e do coeficiente de arraste.

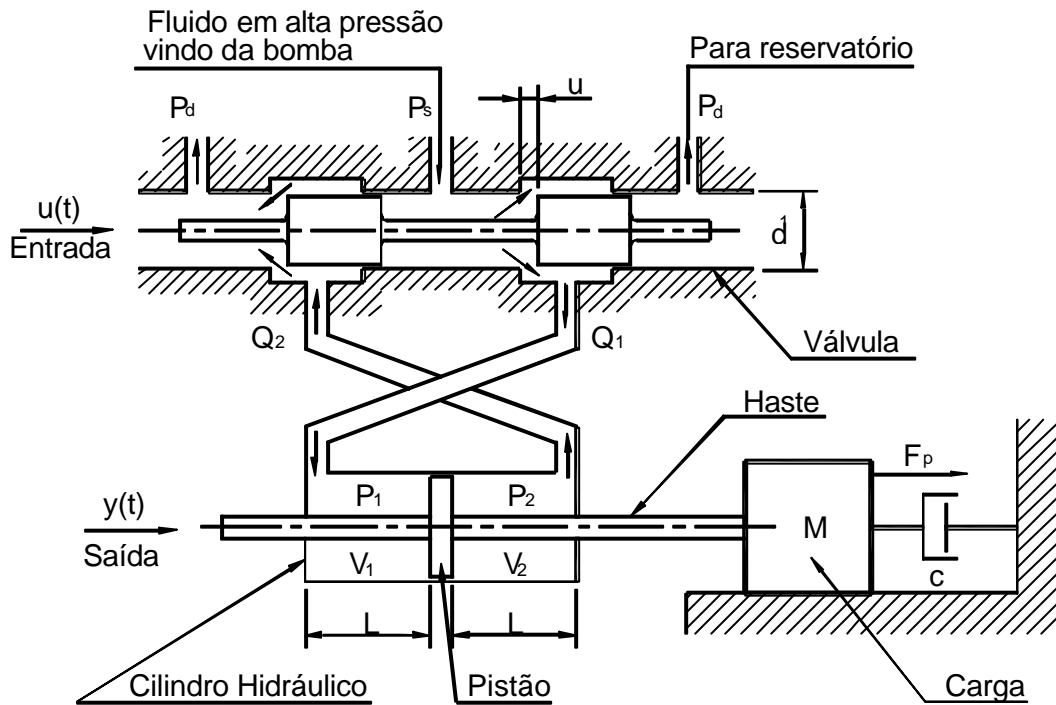


Figura 1. Esquema do Atuador Hidráulico.

3. Controlador Neural Adaptativo Direto

A Figura (2) mostra um esboço do controlador neural adaptativo direto (Cavalcanti, 1994) e da planta. O controlador é treinado “off line” para que a saída da planta siga a referência y_r . Isto é obtido pela minimização do índice de desempenho I_w dado pela Eq. (3). A minimização é realizada como segue: primeiro, o alvo da rede é determinado usando-se o algoritmo de propagação retroativa do erro (APR) de forma a minimizar o índice de desempenho I_u dado pela Eq. (4). Definindo-se como u_r o valor de saída da rede, então o valor de entrada da planta u_p é obtido pela Eq. (5), que é o resultado da minimização do índice de desempenho I_u . Dessa forma, o alvo da rede u_p depende do erro entre y_r e y , do jacobiano da planta e do fator de adaptação η . O jacobiano é calculado impondo-se uma sequência de entrada impulsiva $u(K)$ e determinando-se a sequência de saída $y(K)$ da planta; e através dessas duas sequências a curva $y(u)$ é obtida e conseqüentemente sua derivada. E, na sequência, os pesos de entrada W_e e de saída W_s da rede neural são ajustados utilizando-se o algoritmo anterior, para obtenção do sinal de controle u com a minimização do índice de desempenho I_w dado pela Eq. (6).

$$I_w (K) = \frac{1}{2} [\text{erroy}(K)]^2 \tag{3}$$

onde: $\text{erroy}(K) = y_r(K) - y(K)$

$$I_u (K) = \frac{1}{2} [\text{erroy}(K)]^2 \tag{4}$$

$$u(K+1) = u(K) - \eta \nabla I_u(K) \tag{5}$$

onde: $\nabla I_u = -\text{erro} \cdot \frac{\partial y(K)}{\partial u(K)}$

$$u(K+1) = u_p \text{ e } u(K) = u_r.$$

$$I_w(K) = \frac{1}{2} [u(K+1) - u(K)]^2 \tag{6}$$

O ajuste dos pesos da rede é obtido conforme Eq. (7) e Eq. (8).

$$W_s(K+1) = W_s(K) - \mu \nabla I_{W_s}(K) \tag{7}$$

$$W_e(K+1) = W_e(K) - \mu \nabla I_{W_e}(K) \tag{8}$$

onde μ é o fator de treinamento.

A rede neural usada para o controlador adaptativo direto (CNAD) é composta por uma camada de entrada com função de ativação linear (L), uma intermediária com função de ativação tangente hiperbólica (tgh) e a de saída com função linear (L). A camada de entrada da rede tem três neurônios e as entradas são: trajetória, velocidade e aceleração de referência. A camada intermediária contém sete neurônios e a de saída um, que é a variável de controle da planta. A fase de controle “on line” é realizada da mesma forma que a “off line”, a partir dos pesos obtidos durante o treinamento.

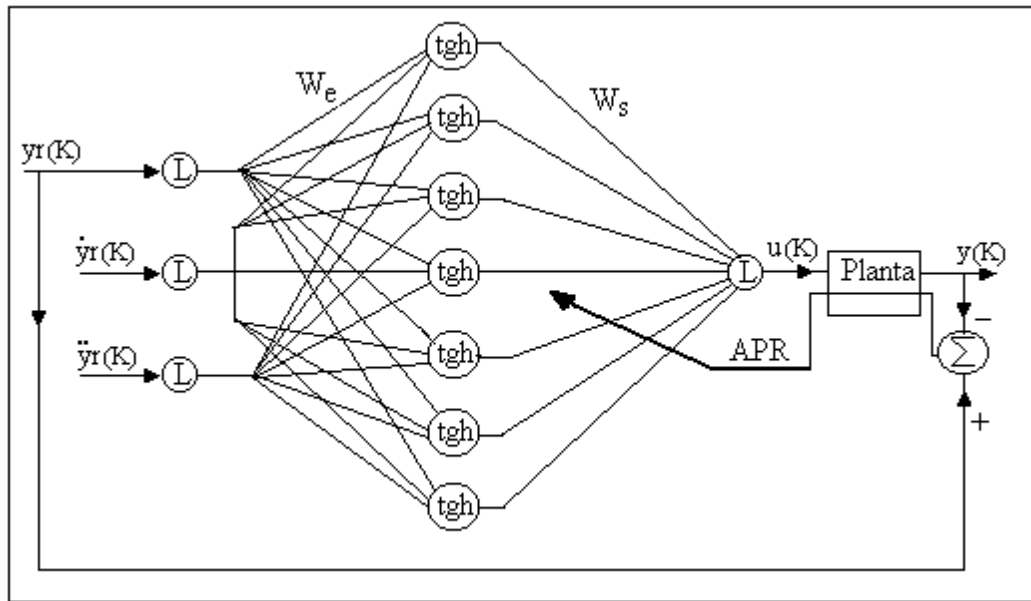


Figura 2. Rede do Controlador Neural Adaptativo Direto e Planta.

4. Simulação e Resultados

Para a fase de treinamento da rede utilizou-se como sinal de entrada o polinômio de grau 3, $y_r(t) = 0,3t^2 - 0,2t^3$ para $0,0 \leq t < 1,0$ s e $y_r(t) = 0,1$ m para $t \geq 1,0$ s e sua primeira e segunda derivadas. Os parâmetros utilizados para planta foram: $M = 128,7$ kg, $c = 162,73$ Ns/m, $A = 1,075 \times 10^{-3}$ m², $v = 0,874 \times 10^{-3}$ m³, $\beta = 14,0 \times 10^8$ N/m² e $K_1 = 2,8 \times 10^{-4}$ (m⁵/kg)^{1/2}, resultando no jacobiano com valor igual a 6,84. O treinamento efetuado foi concluído após 450 épocas, com um índice de desempenho I_w mínimo de $4,4 \times 10^{-6}$ m², para um fator de treinamento $\mu = 0,3$, como mostrado na Fig. (3). Na fase “on line”, as simulações foram realizadas tendo como sinais de referência o polinômio de grau 3 usado no treinamento da rede, variando-se sua amplitude, e a função senoidal $y_r(t) = 0,1 \sin(0,5\pi t)$. O controlador neural adaptativo direto (CNAD) foi testado e os resultados foram comparados com os obtidos através de um controlador proporcional-derivativo (CPD) com ganhos $K_p = 1,2 \times 10^{-2}$ e $K_d = 4,81 \times 10^{-5}$ (Ogata, 1982), (D’Azzo, 1988) determinados a partir do limite de estabilidade calculado com base em um modelo linear da planta e simulado no modelo não-linear em estudo.

A Figura (4) mostra a resposta da planta à referência polinomial de grau 3, com variação na amplitude, usando-se o CNAD. Verificou-se um erro de posição da ordem de décimos de milímetro com a amplitude de referência de 0,1m que foi a usada no treinamento da rede e de milímetros para as outras. A Figura (5) mostra a resposta da planta à referência polinomial de grau 3 obtida com os controladores neural adaptativo direto e proporcional-derivativo. O teste visou comparar o desempenho do sistema usando-se os dois controladores.

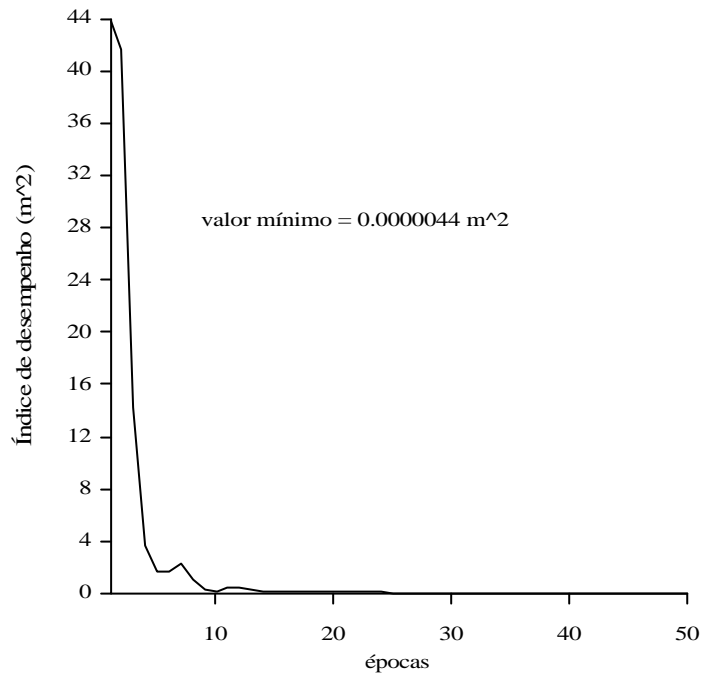


Figura 3. Índice de Desempenho do Controlador Neural Adaptativo Direto.

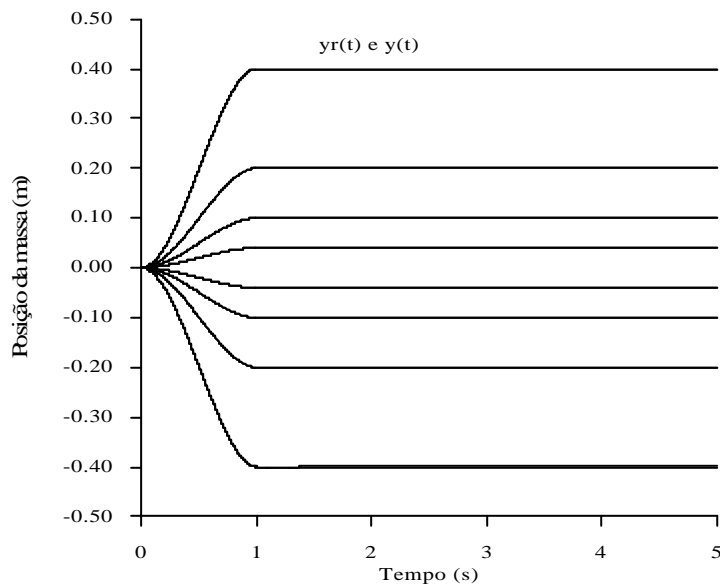


Figura 4. Resposta à referência polinomial de grau 3, com variação de amplitude, da planta, com o CNAD.

O erro de posição foi da ordem de décimos de milímetro com o CNAD e da ordem de centímetros com o CPD. Analisando o resultado mostrado na Fig. (6), que trata de seguimento do sinal de referência senoidal, verificou-se que a resposta da planta apresentou um atraso de tempo com o controlador proporcional-derivativo, que gera um erro da ordem de centímetros, enquanto que com o CNAD, praticamente não existe atraso, sendo o erro da ordem de décimos de milímetro.

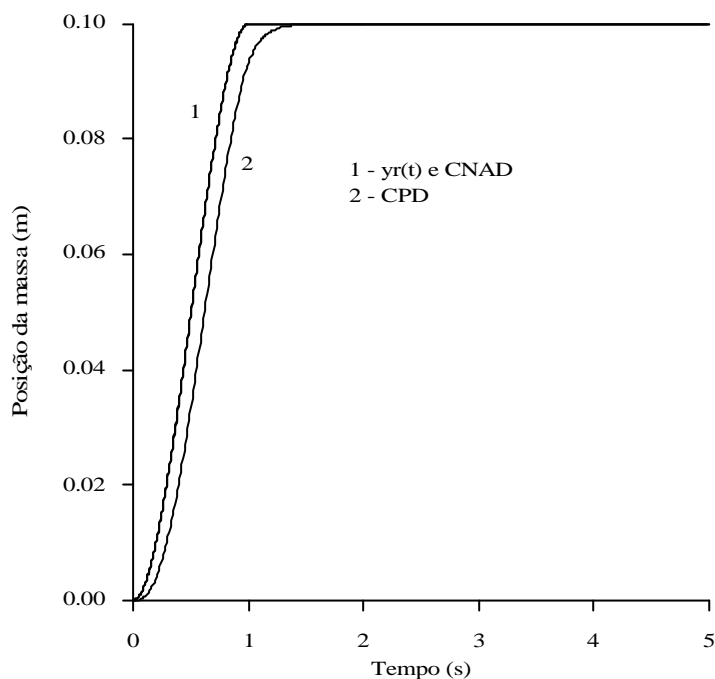


Figura 5. Resposta à referência polinomial de grau 3 da planta, com o CNAD e o CPD.

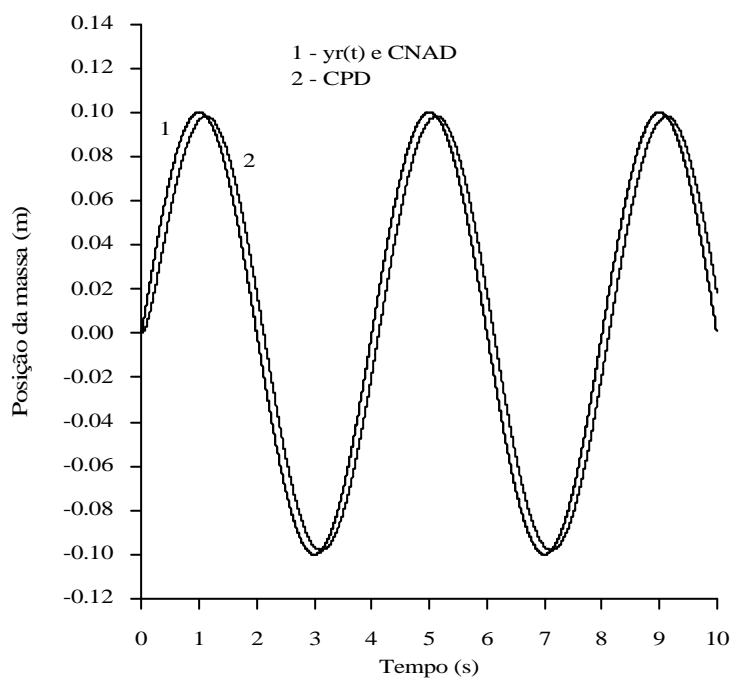


Figura 6. Resposta à referência senoidal da planta, com o CNAD e o CPD.

5. Conclusão

Neste trabalho projetou-se um controlador neural adaptativo direto para posicionamento de uma carga acionada por um atuador hidráulico. O projeto foi realizado em duas etapas; a primeira, para treinamento de uma rede neural multi-camadas e a segunda, para efetuar o controle da posição da carga.

Realizou-se simulações para posicionamento da carga usando-se o controlador projetado e um proporcional-derivativo. O Controlador Neural Adaptativo Direto (CNAD) foi testado, com o polinômio de grau 3 usado no treinamento, tendo sua amplitude variada, e com a função senoidal. Através dos resultados obtidos para o posicionamento da carga, observou-se que o CNAD apresentou boa capacidade nos testes de generalização, e que o desempenho do sistema controlado pelo CNAD foi muito superior ao obtido quando controlado pelo proporcional-derivativo. Pesquisas futuras serão desenvolvidas com o intuito de implementar novas técnicas de controle usando redes neurais, para o posicionamento de sistemas acionados por atuadores hidráulicos e pneumáticos.

6. Referências

- Beale, R., Jackson, T., 1991, "Neural Computing: An Introduction", Editora Adam Hilger, New York.
- Cavalcanti, J. H. F., 1994, "Controladores Neurais Adaptativos", Tese de doutorado, Universidade Federal da Paraíba, Campina Grande, Paraíba, Brasil.
- Cunha, M. A. B., Guenther, R., De Pieri, E. R., 1998, "Controlador em cascata com adaptação de parâmetros aplicado a um atuador hidráulico", Proceedings of XII Brazilian Automatic Conference - XII CBA, September 14-18, Uberlândia, MG, Brazil, vol. 1, pp. 211-216.
- D'Azzo, J. J., Houpis, C. H., 1988, "Análise e Projeto de Sistemas de Controle Lineares", Editora Guanabara S. A., Rio de Janeiro - RJ.
- Guenther, R., De Pieri, E. R., 1997, "Cascade Control of Hydraulic Actuators", RBCM – Journal of the Brazilian Society of Mechanical Sciences, vol. XIX, n. 2, pp. 108-120.
- Haykin, S., 1994, "Neural Networks - A Comprehensive Foundation", Macmillan, New York.
- Ogata, K., 1982, "Engenharia de Controle Moderno", Editora Prentice/Hall do Brasil Ltda, Rio de Janeiro - RJ.
- Riul, J. A., Cavalcanti, J. H. F., Silva, J. F., 1999, "Neurocontrolador Inverso Direto Aplicado a um Atuador Hidráulico", XV Congresso Brasileiro de Engenharia Mecânica, Aguas de Lindóia, SP, Brasil.
- Riul, J. A., Silva, J. F., Cavalcanti, J. H. F., 2000, "Controlador Híbrido Aplicado a um Atuador Hidráulico", I Congresso Nacional de Engenharia Mecânica, Natal, RN, Brasil.

DIRECT ADAPTIVE NEURAL CONTROL APPLIED TO A HYDRAULIC ACTUATOR

José Antônio Riul

Departamento de Tecnologia Mecânica – CT-UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, e-mail:riul@ct.ufpb.br

José Felício da Silva

Departamento de Tecnologia Mecânica – CT-UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, e-mail:felicio@funape.ufpb.br

José Homero. F. Cavalcanti

Departamento de Tecnologia Mecânica – CT-UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, e-mail:homero@ct.ufpb.br

Abstract. *This work has as main objective to show a system of non-linear control for the positioning of a actuated load by hydraulic actuator. The control strategy proposed is made in two steps by using of neural network. In the first step all neural network parameters are learned by supervised training. For the second step the plant control is made in similar way to supervised training from parameters obtained in the first step. The simulation results are presented to evaluate the performance of the proposed system.*

Keywords. *Control System, Neural Network, Non-linear System*

IMPROVING THE POSITIONING ACCURACY OF POWERFUL MANIPULATORS WITH APPLICATION IN NUCLEAR MAINTENANCE

Marco Antonio Meggiolaro

Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) - Dept. Eng. Mecânica
Rua Marquês de São Vicente, 225 Gávea, Rio de Janeiro, RJ 22453-900
meggi@mec.puc-rio.br

Steven Dubowsky

Department of Mechanical Engineering, Massachusetts Institute of Technology
77 Massachusetts Ave., Cambridge, MA 02139
dubowsky@mit.edu

Abstract. *Powerful robotic manipulators are needed in nuclear maintenance, field, undersea and medical applications to perform high accuracy tasks requiring the manipulation of heavy payloads. The nozzle dam positioning task for maintenance of a nuclear power plant steam generator is an example of a task that requires a strong manipulator with very fine absolute positioning accuracy. Absolute accuracy, rather than simple repeatability, is required for autonomous operation or for teleoperation with advanced virtual aides, such as virtual viewing. However, high accuracy is generally unattainable in manipulators capable of producing high task forces due to such factors as high joint, actuator, and transmission friction and link geometric distortions. A method called Base Sensor Control (BSC) has been developed to compensate for nonlinear joint characteristics, such as high joint friction, to improve system repeatability. A method to identify and compensate for system geometric distortion positioning errors in large manipulators has also been proposed to improve absolute accuracy in systems with good repeatability. This technique is called Geometric and Elastic Error Compensation (GEC). Here, it is shown experimentally that the two techniques can be effectively combined to enable strong manipulators to achieve high absolute positioning accuracy while performing tasks requiring high forces. These developed control techniques are being implemented at the Korean and French power companies.*

Keywords: *robot calibration, friction compensation, nuclear maintenance.*

1. Introduction

Large robot manipulators are needed in field, service and medical applications to perform high accuracy tasks. Examples are manipulators that perform decontamination tasks in nuclear sites, space manipulators such as the Special Purpose Dexterous Manipulator (SPDM) and manipulators for medical treatment. In these applications, a large robotic system is often needed to have very fine precision. Its accuracy specifications may be very small fractions of its size. Achieving such high accuracy is difficult because of the manipulator's size and its need to carry heavy payloads. Hydraulic robot's high load carrying capacity is attractive for such applications, but high joint friction and actuator nonlinearities make them difficult to control.

A number of approaches exist for improving fine motion manipulator performance through friction compensation. Some of these require modeling of the difficult to characterize joint frictional behavior (Canudas de Wit et al., 1996; Popovic et al., 1994). Some require the use of specially designed manipulators that contain complex internal joint-torque sensors (Pfeffer et al., 1989).

A simple, yet effective control method has been developed that is modelless and does not require internal joint sensors (Iagnemma et al., 1997; Morel and Dubowsky, 1996). The method, called Base Sensor Control (BSC), estimates manipulator joint torques from a self-contained external six-axis force/torque sensor placed under the manipulator's base. The joint torque estimates allow for accurate joint torque control that has been shown to greatly improve repeatability of both hydraulic and electric manipulators.

Even with improved repeatability, high absolute positioning accuracy is still difficult to achieve with a strong manipulator. In such systems, two principal error sources create significant end-effector errors. The first is kinematic errors due to the non-ideal geometry of the links and joints of manipulators, such as errors due to machining tolerances. These errors are often called geometric errors. Task constraints often make it impossible to use direct end-point sensing in a closed-loop control scheme to compensate for these errors. Therefore, there is a need for model-based error identification and compensation techniques, often called robot calibration.

The second error source that can limit the absolute accuracy of a large manipulator is the elastic errors due to the distortion of a manipulator's mechanical components under large task loads or even its own weight. Classical error compensation methods cannot correct the errors in large systems with significant elastic deformations, because they do not explicitly consider the effects of task forces and structural compliance.

Considerable research has been performed in robot calibration (Roth et al. 1986; Hollerbach 1988; Hollerbach et al. 1996; Zhuang et al. 1996). In these methods robot position accuracy is improved using compensation methods that essentially identify a more accurate functional relationship between the joint transducer readings and the workspace position of the end-effector based on experimental calibration measurements. A major component of this process is the development of manipulator error models, some of which consider the effects of manipulator joint errors, while others

focus on the effects of link dimensional errors (Waldron et al. 1979; Wu 1984; Vaichav et al. 1987; Mirman et al. 1993). Some researchers have studied methods to find the optimal configurations during the calibration measurements to reduce the manipulator errors by calibration (Borm et al. 1991; Zhuang et al. 1996). Solution methods for the identification of the manipulator's unknown parameters have been studied for these model-based calibration processes (Dubowsky et al. 1975; Zhuang et al. 1993). Most calibration methods have been applied to industrial or laboratory robots, achieving good accuracy when geometric errors are dominant.

However, with only one or two exceptions, calibration methods do not explicitly compensate for elastic errors due to the wrench at the end-effector, or they require explicit structural modeling of the system (Drouet et al. 1998; Drouet 1999). While conceptually very similar to the classical geometric problem, the combined problem is far more complex. Compensating for geometric errors requires building a model that is a function of the n (usually 6) joint variables. To compensate for a general 6 variable end-point task wrench (three end-point forces and three end-point moments) requires a model that is a function of both the joint variables and the end-point wrench variables, or a function of at least 12 variables. The number of measurements required to characterize this 12 dimensional space is far larger than required for the 6 dimensional space. The time and cost of the physical calibration measurements often dominates the calibration problem. Simple calculations suggest that a brute force identification would require several million calibration measurements.

Recent work has resulted in a method to correct for errors in the end-effector position and orientation caused by geometric and elastic errors in large manipulators (Meggiolaro et al., 1999). The method, called Geometric and Elastic Error Compensation (GEC), yields measurement based error compensation algorithms that predict the manipulator's end-point position and orientation as a function of the configuration of the system and the task forces. Given the task loads from a conventional wrist force/torque sensor and the joint angles of the manipulator, the algorithm compensates for the combined elastic and geometric errors. It does not require detailed modeling of the manipulator's structural properties. Instead it uses a relatively small set of offline end-point experimental measurements to build a "generalized error" representation of the system (Mavroidis et al., 1997). In the GEC method each generalized error parameter can be represented as a function of only a few of the system variables. As a result, the number of measurements required to characterize the system is dramatically smaller than might be expected. This method can substantially reduce the absolute errors in manipulators with good inherent repeatability.

In this research, an approach is developed that substantially improves the absolute accuracy in strong powerful manipulators lacking good repeatability and having significant geometric and elastic errors. The method uses base force/torque sensor information to apply BSC in concert with GEC, which uses wrist sensor information to achieve greatly improved absolute accuracy in a strong manipulator exerting high task loads. The algorithm does not require joint velocity or acceleration measurements, a model of the actuators or friction, or the knowledge of manipulator mass parameters or link stiffnesses, yet it is able to substantially improve its absolute positioning accuracy.

In addition, the error formulation introduces redundant parameters, often non-intuitive, that may compromise the robustness of the calibration. The existing numerical methods to eliminate such errors are formulated on a case-by-case basis. In this paper, the general analytical expressions of the redundant parameters are developed for any serial link manipulator, expressed through its Denavit-Hartenberg parameters. These expressions are used to eliminate the redundant parameters from the error model of any manipulator prior to the identification process, allowing for systematic robot calibration with improved accuracy.

The combined methods are applied to an important application in nuclear maintenance. The nozzle dam positioning task for maintenance of a nuclear power plant steam generator is an example of a task that requires a strong manipulator with very fine absolute positioning accuracy (Zezza, 1985). Absolute accuracy, rather than simple repeatability, is required for autonomous operation or for teleoperation with advanced virtual aides, such as virtual viewing. Here, it is shown experimentally that the two techniques can be effectively combined to enable strong manipulators to achieve high absolute positioning accuracy while performing tasks requiring high forces.

2. Analytical background

2.1. Base sensor control (BSC)

Here the basis for BSC is briefly reviewed. The complete development is presented in (Morel and Dubowsky, 1996). A simplified version of the algorithm sufficient and effective for fine-motion control is formulated in (Iagnemma et al., 1997).

As shown in Fig. (1), the wrench, \mathbf{W}_b , exerted by the manipulator on its base sensor can be expressed as the sum of three components $\mathbf{W}_g + \mathbf{W}_d + \mathbf{W}_e$, where \mathbf{W}_g is the robot gravity component, \mathbf{W}_d is caused by manipulator motion, and \mathbf{W}_e is the wrench exerted by the payload on the end-effector. Note that joint friction does not appear in the measured base sensor wrench. In the fine-motion case, it is assumed that the gravity wrench is essentially constant, and this wrench can be approximated by the initial value measured by the base sensor. Hence, the complexity of computing the gravitational wrench, such as identification of link weights and a static manipulator model, is eliminated. Under this assumption, the Newton Euler equations of the first i links are:

$$\begin{cases} \mathbf{W}_{0 \rightarrow 1} = -\mathbf{W}_b \\ \mathbf{W}_{1 \rightarrow 2} = \mathbf{W}_{0 \rightarrow 1} - \mathbf{W}_{d_1} \\ \vdots \\ \mathbf{W}_{i \rightarrow i+1} = \mathbf{W}_{i-1 \rightarrow i} - \mathbf{W}_{d_i} \\ \vdots \\ -\mathbf{W}_e = \mathbf{W}_{n-1 \rightarrow n} - \mathbf{W}_{d_n} \end{cases} \quad (1)$$

where $\mathbf{W}_{i \rightarrow i+1}$ is the wrench exerted by link i on link $i+1$, and \mathbf{W}_{d_i} is the dynamic wrench for link i .

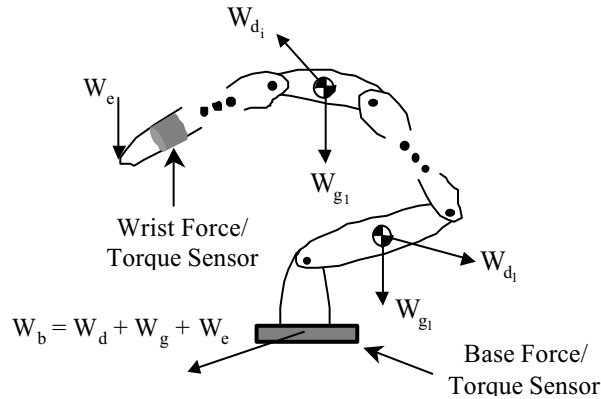


Figure 1. External and dynamic wrenches.

For fine tasks it is assumed that the manipulator moves very slowly so that \mathbf{W}_d can be neglected. Therefore, for slow, fine motions, only the measured wrench at the base is used to estimate the torque in joint $i+1$. The estimated torque in joint $i+1$ is obtained by projecting the moment vector at the origin O_i of the i^{th} reference frame along the joint axis z_i :

$$\tau_{i+1} = -z_i^T \cdot \mathbf{W}_b^{O_i} \quad (2)$$

The value of τ_{i+1} depends only on the robot's kinematic parameters, joint angles and base sensor measurements.

With estimates of the joint torque, high performance torque control can be achieved to greatly reduce the effects of joint friction and nonlinearities. This results in greatly improved repeatability. This method will not compensate for sources of random repeatability errors, such as limited encoder resolution. In addition, a manipulator with good repeatability may not have fine absolute position accuracy.

2.2. Robot calibration

The main sources of absolute accuracy errors in a manipulator with good repeatability are mechanical system errors (resulting from machining and assembly tolerances), elastic deformations of the manipulator links, and joint errors (bearing run-out). These can be grouped into geometric and elastic errors. Although these physical errors are relatively small, their influence on the end-effector position of a large manipulator can be significant. A brief review of the error compensation method used here is presented below.

The end-effector position and orientation error, \mathbf{DX} , is defined as the 6×1 vector that represents the difference between the real position and orientation of the end-effector and the ideal or desired one:

$$\mathbf{DX} = \mathbf{X}^{\text{real}} - \mathbf{X}^{\text{ideal}} \quad (3)$$

where \mathbf{X}^{real} and $\mathbf{X}^{\text{ideal}}$ are 6×1 vectors composed of the three positions and three orientations of the end-effector reference frame in the inertial reference system for the real and ideal cases respectively.

The error compensation method assumes that physical errors slightly displace manipulator joint frames from their expected, ideal locations (Mavroidis et al., 1997). The actual position and orientation of a frame with respect to its ideal location is represented by 3 translations $\epsilon_{x,i}$, $\epsilon_{y,i}$ and $\epsilon_{z,i}$ (along the X, Y and Z axes respectively, defined using the Denavit-Hartenberg representation) and 3 consecutive rotations $\epsilon_{s,i}$, $\epsilon_{r,i}$, $\epsilon_{p,i}$ around the Y, Z and X axes respectively, see Fig. (2). The subscripts s, r, and p represent spin (yaw), roll, and pitch, respectively. The 6 parameters $\epsilon_{x,i}$, $\epsilon_{y,i}$, $\epsilon_{z,i}$, $\epsilon_{s,i}$, $\epsilon_{r,i}$ and $\epsilon_{p,i}$ are called generalized error parameters, which can be a function of the system geometry and joint variables. For an n degree of freedom manipulator, there are $6(n+1)$ generalized errors which can be written in the form of a $6(n+1) \times 1$ vector $\mathbf{e} = [\epsilon_{x,0}, \dots, \epsilon_{x,i}, \epsilon_{y,i}, \epsilon_{z,i}, \epsilon_{s,i}, \epsilon_{r,i}, \epsilon_{p,i}, \dots, \epsilon_{p,n}]$, with i ranging from 0 to n , assuming that both the

manipulator and the location of its base are being calibrated. The generalized errors that depend on the system geometry, the system task loads and the system joint variables can be calculated from the physical errors link by link. Note that actual system weight effects can be included in the model as a simple function of joint variables.

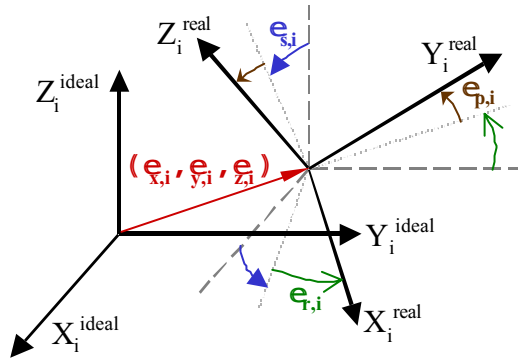


Figure 2. Definition of the translational and rotational generalized errors for i^{th} link.

Since the generalized errors are small, \mathbf{DX} can be calculated by the following linear equation in \mathbf{e} :

$$\mathbf{DX} = \mathbf{J}_e \mathbf{e} \tag{4}$$

where \mathbf{J}_e is the $6 \times 6(n+1)$ Jacobian matrix of the end-effector error \mathbf{DX} with respect to the elements of the generalized error vector \mathbf{e} , also known as Identification Jacobian matrix (Zhuang et al. 1999). As with the generalized errors, \mathbf{J}_e depends on the system configuration, geometry and task loads.

If the generalized errors, \mathbf{e} , can be found from calibration measurements, then the correct end-effector position and orientation error can be calculated using Eq. (4) and be compensated. Figure (3) shows schematically an error compensation algorithm based on Eq. (4). The method to obtain \mathbf{e} from experimental measurements is explained below.

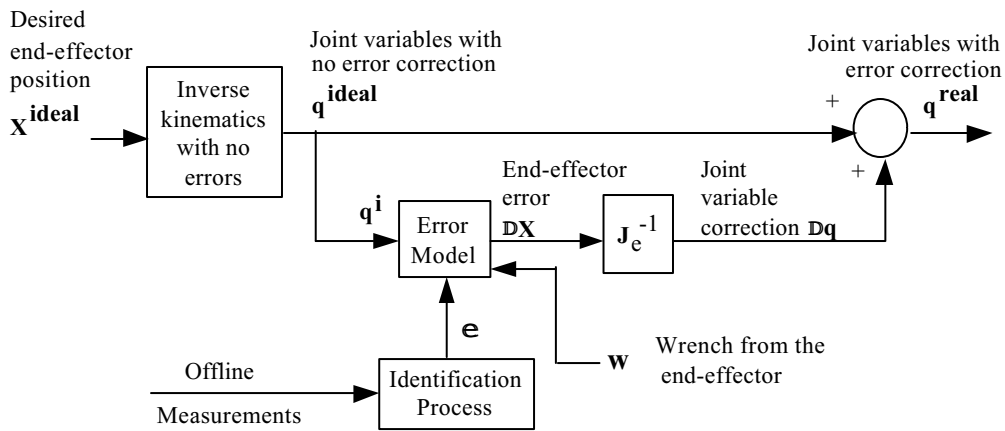


Figure 3. Error compensation scheme.

To calculate the generalized errors \mathbf{e} it is assumed that some components of vector \mathbf{DX} can be measured at a finite number of different manipulator configurations. However, since position coordinates are much easier to measure in practice than orientations, in many cases only the three position coordinates of \mathbf{DX} are measured.

Assuming that all 6 components of \mathbf{DX} can be measured, for an n degree of freedom manipulator, $6(n+1)$ generalized errors \mathbf{e} can be calculated by measuring \mathbf{DX} at m different configurations, defined as $\mathbf{q}_1, \mathbf{q}_2, \dots, \mathbf{q}_m$, then writing Eq. (4) m times:

$$\Delta \mathbf{X}_t = \begin{bmatrix} \Delta \mathbf{X}_1 \\ \Delta \mathbf{X}_2 \\ \dots \\ \Delta \mathbf{X}_m \end{bmatrix} = \begin{bmatrix} \mathbf{J}_e(\mathbf{q}_1) \\ \mathbf{J}_e(\mathbf{q}_2) \\ \dots \\ \mathbf{J}_e(\mathbf{q}_m) \end{bmatrix} \cdot \mathbf{e} = \mathbf{J}_t \cdot \mathbf{e} \tag{5}$$

where $\Delta \mathbf{X}_t$ is the $m \times 1$ vector formed by all measured vectors \mathbf{DX} at m different configurations and \mathbf{J}_t is the $6m \times 6(n+1)$ matrix formed by the m Identification Jacobian matrices \mathbf{J}_e at m configurations, called here Total Identification

Jacobian. To compensate for the effects of measurement noise, the number of measurements m is, in general, much larger than n .

If the generalized errors \mathbf{e} are constant, then a unique least-squares estimate $\hat{\mathbf{e}}$ can be calculated by:

$$\hat{\mathbf{e}} = (\mathbf{J}_t^T \mathbf{J}_t)^{-1} \mathbf{J}_t^T \cdot \mathbf{DX}_t \quad (6)$$

However, if the Identification Jacobian matrix $\mathbf{J}_e(\mathbf{q})$ contains linearly dependent columns, Eq. (6) will produce estimates with poor accuracy (Hollerbach et al. 1996). This occurs when there is redundancy in the error model, in which case it is not possible to distinguish the error contributed by each generalized error component. Conventional calibration methods also cannot be successfully applied when some of the generalized errors depend on the manipulator configuration \mathbf{q} or the end-effector wrench \mathbf{w} , namely $\mathbf{e}(\mathbf{q}, \mathbf{w})$, such as when elastic deflections that depend on the configuration and applied forces at the end-effector are significant. Below, methods are presented for finding the generalized errors (\mathbf{e}) where there is a singular Identification Jacobian matrix and for the case where there is significant elastic deformation combined with conventional geometric errors.

3. Geometric and elastic error compensation

In the GEC method (Geometric and Elastic Error Compensation), elastic deformation and classical geometric errors are considered in a unified manner. The method can identify and compensate for both types of error, without an elastic model of the system. Two steps are necessary to successfully apply the GEC method: the redundant error parameters must be eliminated from the identification model (as with any calibration method), and the error model must be extended to explicitly consider the task loading wrench and configuration dependency of the errors. These two steps are presented below.

3.1. Eliminating the redundant error parameters

In robot calibration, redundant errors must be eliminated from the error model prior to the identification process. This is usually done in an ad hoc or numerical manner by reducing the columns of the Identification Jacobian matrix \mathbf{J}_e to a linearly independent set, thus obtaining a non-singular form of \mathbf{J}_e , called \mathbf{G}_e . This is accomplished by grouping the generalized errors into a smaller independent set, called \mathbf{e}^* , in accordance with the columns of the submatrix \mathbf{G}_e . If \mathbf{J}_e is replaced by its submatrix \mathbf{G}_e in Eq. (5), then application of Eq. (6) will result in estimates with improved accuracy.

By definition, the dependent error parameters eliminated from \mathbf{e} do not affect the end-effector error, resulting in the identity

$$\mathbf{DX} = \mathbf{J}_e \mathbf{e} \equiv \mathbf{G}_e \mathbf{e}^* \quad (7)$$

Using the above identity and the linear combinations of the columns of \mathbf{J}_e calculated in (Meggiolaro et al., 2000), it is possible to obtain all relationships between the generalized error set \mathbf{e} and its independent subset \mathbf{e}^* .

Defining $\mathbf{J}_{x,i}$, $\mathbf{J}_{y,i}$, $\mathbf{J}_{z,i}$, $\mathbf{J}_{s,i}$, $\mathbf{J}_{r,i}$ and $\mathbf{J}_{p,i}$ as the columns of \mathbf{J}_e associated with the generalized error components $\epsilon_{x,i}$, $\epsilon_{y,i}$, $\epsilon_{z,i}$, $\epsilon_{s,i}$, $\epsilon_{r,i}$ and $\epsilon_{p,i}$ respectively ($0 \leq i \leq n$), Eq. (4) can be rewritten as

$$\mathbf{DX} = [\mathbf{J}_{x,0}, \dots, \mathbf{J}_{x,i}, \mathbf{J}_{y,i}, \mathbf{J}_{z,i}, \mathbf{J}_{s,i}, \mathbf{J}_{r,i}, \mathbf{J}_{p,i}, \dots, \mathbf{J}_{p,n}] [\epsilon_{x,0}, \dots, \epsilon_{x,i}, \epsilon_{y,i}, \epsilon_{z,i}, \epsilon_{s,i}, \epsilon_{r,i}, \epsilon_{p,i}, \dots, \epsilon_{p,n}]^T \quad (8)$$

To obtain the non-singular Identification Jacobian matrix \mathbf{G}_e , columns $\mathbf{J}_{z,(i-1)}$ and $\mathbf{J}_{r,(i-1)}$ must be eliminated from the matrix \mathbf{J}_e for all values of i between 1 and n . If joint i is prismatic, then columns $\mathbf{J}_{x,(i-1)}$ and $\mathbf{J}_{y,(i-1)}$ must also be eliminated. For an n DOF manipulator with r rotary joints and p (p equal to $n-r$) prismatic joints, a total of $2r+4p$ columns are eliminated from the Identification Jacobian \mathbf{J}_e to form its submatrix \mathbf{G}_e . This means that $2r+4p$ generalized errors cannot be obtained by measuring the end-effector pose.

Once the linearly dependent columns of matrix \mathbf{J}_e are eliminated, the independent generalized error set \mathbf{e}^* can be obtained. Note that the independent errors \mathbf{e}^* are a subset of the generalized errors \mathbf{e} . If joint i is revolute ($1 \leq i \leq n$), then the generalized errors $\epsilon_{z,(i-1)}$ and $\epsilon_{r,(i-1)}$ are eliminated, and their values are incorporated into the independent error parameters $\epsilon_{y,i}^*$, $\epsilon_{z,i}^*$, $\epsilon_{s,i}^*$ and $\epsilon_{r,i}^*$:

$$\begin{cases} \epsilon_{y,i}^* \equiv \epsilon_{y,i} + \epsilon_{z,(i-1)} \sin \alpha_i + \epsilon_{r,(i-1)} \cdot a_i \cos \alpha_i \\ \epsilon_{z,i}^* \equiv \epsilon_{z,i} + \epsilon_{z,(i-1)} \cos \alpha_i - \epsilon_{r,(i-1)} \cdot a_i \sin \alpha_i \\ \epsilon_{s,i}^* \equiv \epsilon_{s,i} + \epsilon_{r,(i-1)} \sin \alpha_i \\ \epsilon_{r,i}^* \equiv \epsilon_{r,i} + \epsilon_{r,(i-1)} \cos \alpha_i \end{cases} \quad (9)$$

where the manipulator parameters are defined using the D.H. representation: link lengths a_i , joint offsets d_i , joint angles θ_i , and skew angles α_i .

If joint i is prismatic, then the translational errors $\varepsilon_{x,(i-1)}$ and $\varepsilon_{y,(i-1)}$ are also eliminated, and their values are incorporated into $\varepsilon_{x,i}^*$, $\varepsilon_{y,i}^*$ and $\varepsilon_{z,i}^*$. In this case, Eq. (9) becomes:

$$\begin{cases} \varepsilon_{x,i}^* \equiv \varepsilon_{x,i} + \varepsilon_{x,(i-1)} \\ \varepsilon_{y,i}^* \equiv \varepsilon_{y,i} + \varepsilon_{y,(i-1)} \cos \alpha_i + \varepsilon_{z,(i-1)} \sin \alpha_i + \varepsilon_{r,(i-1)} \cdot a_i \cos \alpha_i \\ \varepsilon_{z,i}^* \equiv \varepsilon_{z,i} - \varepsilon_{y,(i-1)} \sin \alpha_i + \varepsilon_{z,(i-1)} \cos \alpha_i - \varepsilon_{r,(i-1)} \cdot a_i \sin \alpha_i \\ \varepsilon_{s,i}^* \equiv \varepsilon_{s,i} + \varepsilon_{r,(i-1)} \sin \alpha_i \\ \varepsilon_{r,i}^* \equiv \varepsilon_{r,i} + \varepsilon_{r,(i-1)} \cos \alpha_i \end{cases} \quad (10)$$

Note that Eqs. (9) and (10) are obtained when both position and orientation of the end-effector are considered. In cases where only the end-effector position is measured, its orientation can take any value, resulting in additional linear combinations of the columns of \mathbf{J}_e . Since the three rotational errors $\varepsilon_{s,n}$, $\varepsilon_{r,n}$ and $\varepsilon_{p,n}$ of the end-effector frame do not influence the end-effector position (they only affect the orientation, which is not being measured), these errors cannot be identified and must be eliminated to form \mathbf{e}^* . In this case, the three last columns of \mathbf{J}_e are also eliminated to form the submatrix \mathbf{G}_e . Further linear combinations exist if the last joint is revolute and its link length a_n is zero, refer to (Meggiolaro et al., 2000) for more information on the elimination of the redundant errors.

If the vector \mathbf{e}^* containing the independent errors is constant, then the matrix \mathbf{G}_e can be used to replace \mathbf{J}_e in Eq. (5), and Eq. (6) is applied to calculate the estimate of the independent generalized errors \mathbf{e}^* , completing the identification process. However, if non-geometric factors are considered such as link compliance or gear eccentricity, then it is necessary to represent the parameters of \mathbf{e}^* as a function of the system configuration and task loadings prior to the identification process. This extended modeling is presented below.

3.2. Polynomial approximation of the generalized errors

For a system with significant geometric and elastic errors, the independent generalized errors \mathbf{e}^* are a function of the manipulator configuration \mathbf{q} and the end-effector wrench \mathbf{w} , or $\mathbf{e}^*(\mathbf{q}, \mathbf{w})$. To predict the endpoint position of the manipulator for a given configuration and task wrench, it is necessary to calculate the generalized errors from a set of offline measurements. The complexity of these calculations can be substantially reduced if the generalized errors are parameterized using polynomial functions. The i^{th} element of vector \mathbf{e}^* is approximated by a polynomial series expansion of the form:

$$\varepsilon_i^* = \sum_{j=1}^{n_i} c_{ij} \cdot f_{ij}(\mathbf{q}, \mathbf{w}), \quad f_{ij}(\mathbf{q}, \mathbf{w}) \equiv (w_{mj})^{a_{0,j}} \cdot (q_1^{a_{1,j}} \cdot q_2^{a_{2,j}} \cdot \dots \cdot q_n^{a_{n,j}}) \quad (11)$$

where n_i is the number of terms used in each expansion, c_{ij} are the polynomial coefficients, w_{mj} is an element of the task wrench \mathbf{w} , and q_1, q_2, \dots, q_n are the manipulator joint parameters. It has been found that good accuracy can be obtained using only a few terms n_i in the above expansion (Meggiolaro et al., 1998). From the definition of the generalized errors, the errors associated with the i^{th} link depend only on the parameters of the i^{th} joint. If elastic deflections of link i are considered, then the generalized errors created by these deflections would depend on the weight wrench w_i applied at the i^{th} link. For a serial manipulator, this wrench is due to the wrench at the end-effector and to the configuration of the links after the i^{th} . Hence, the wrench w_i depends only on the joint parameters q_{i+1}, \dots, q_n . Thus, the number of terms in the products of Eq. (11) is substantially reduced. Each generalized error parameter is then represented as a function of only a few of the system variables, greatly reducing the number of measurements required to characterize the system using the GEC method.

The constant coefficients c_{ij} are grouped into one vector \mathbf{c} , becoming the unknowns of the problem. The total number of unknown coefficients, called n_c , is the sum of the number of terms used in Eq. (11) to approximate each generalized error, i.e. $n_c = \sum n_i$. The n_c functions $f_{ij}(\mathbf{q}, \mathbf{w})$ are then incorporated into the non-singular Identification Jacobian matrix \mathbf{G}_e by substituting Eq. (11) into (7). Equation (7) becomes:

$$\mathbf{DX} = \mathbf{G}_e(\mathbf{q}) \cdot \mathbf{e}^*(\mathbf{q}, \mathbf{w}) \equiv \mathbf{H}_e(\mathbf{q}, \mathbf{w}) \cdot \mathbf{c} \quad (12)$$

where \mathbf{H}_e is the $(6 \times n_c)$ Jacobian matrix of the end-effector error \mathbf{DX} with respect to the polynomial coefficients c_{ij} . The matrix \mathbf{H}_e , called here Extended Identification Jacobian matrix, can be obtained from Eqs. (11) and (12):

$$\mathbf{H}_e(\mathbf{q}, \mathbf{w}) \equiv [\mathbf{G}_1 \cdot f_{1,1}, \dots, \mathbf{G}_1 \cdot f_{1,n_1}, \dots, \mathbf{G}_i \cdot f_{i,1}, \mathbf{G}_i \cdot f_{i,2}, \dots, \mathbf{G}_i \cdot f_{i,n_i}, \dots] \quad (13)$$

where \mathbf{G}_i is the column of matrix \mathbf{G}_e associated to the generalized error component ε_i^* .

An estimate of the coefficient vector \mathbf{c} is then calculated by replacing \mathbf{J}_e with the matrix \mathbf{H}_e in Eq. (5) and applying Eq. (6), completing the identification process. Once the polynomial coefficients, \mathbf{c} , are identified, the end-effector

position and orientation error \mathbf{DX} can be calculated and compensated using Eq. (12). The method of identifying the generalized errors as a function of the manipulator configuration and the end-effector wrench is summarized in Fig. (4).

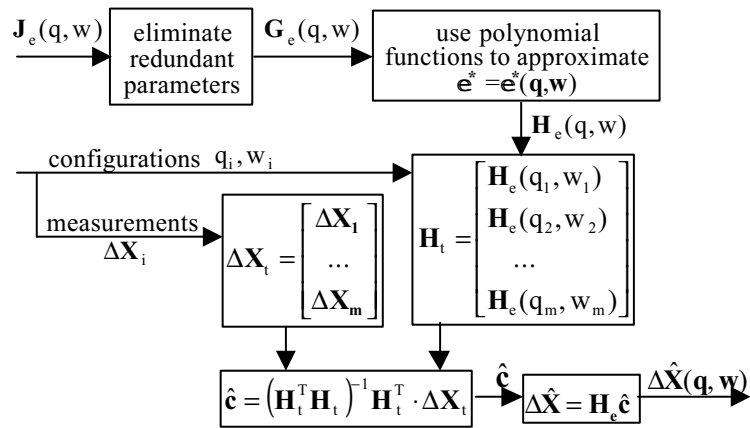


Figure 4. Flow-chart of the method to identify generalized errors.

4. Application to the nozzle dam task

The precision control algorithms presented in this paper are being developed for a task in the nuclear power industry. In order for workers to inspect and repair a nuclear power plant’s steam generator, two very large pipes (1 meter in diameter) must be sealed with a device called a nozzle dam. The center section of the nozzle dam weighs approximately 60 kg and it must be inserted into a ring with clearances of a few millimeters. In this operation, workers receive high doses of radiation. Hence, performing this task with a robotic manipulator would be very desirable. A simulated robotic nozzle dam placement can be see in Fig. (5), where the manipulator is moving the nozzle dam side plate into its position in the nozzle ring. The center plate will then be inserted within the side plate.

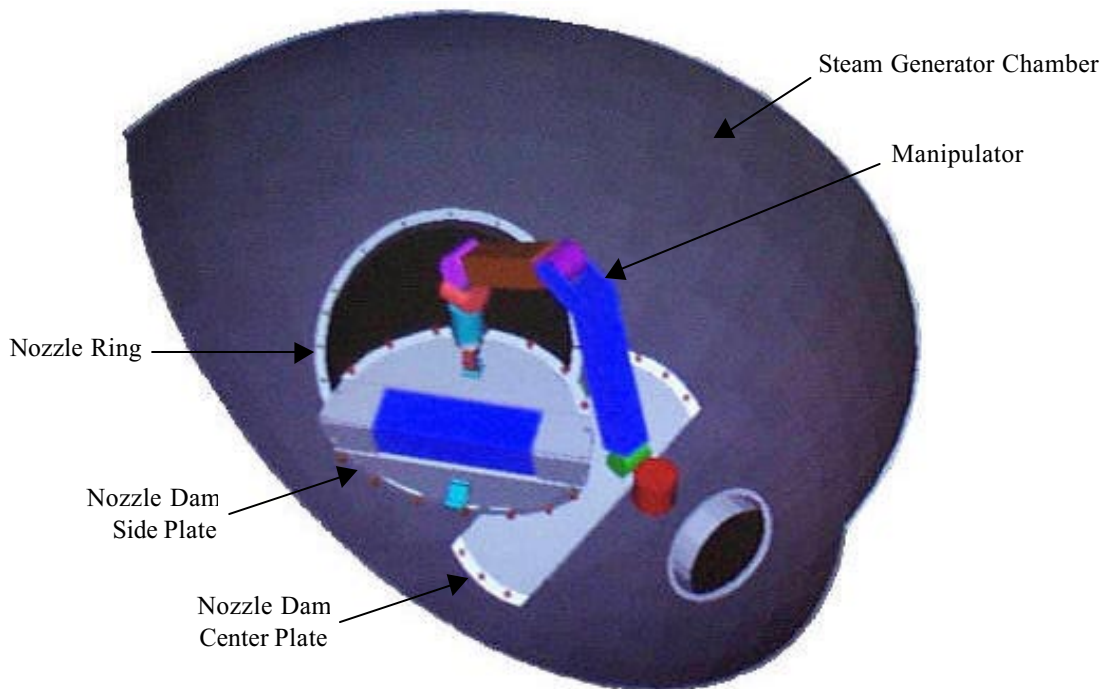


Figure 5. Simulated robotic nozzle dam task.

Attempts to place the dam with a manipulator have taken too long because of the combination of poor operator visibility and lack of manipulator accuracy. It costs tens of thousands of dollars per hour to keep a nuclear power plant offline. Improving manipulator accuracy is a key to shortening this time. The typical repeatability of manipulators capable of handling the required load is in the range of 10 to 20 mm. The absolute accuracy can be several times these amounts. The automation of this task would require absolute accuracy of a few mm. In this work, the combined BSC/GEC method was experimentally evaluated for this application.

Figure (6) shows the experimental test-bed constructed for this study. The manipulator chosen for this system is a Schilling Titan II, a six DOF hydraulic robot capable of handling payloads in excess of 100 kg. Its position accuracy is approximately 34 mm (RMS), many times the specification of a few mm. A good part of its lack of accuracy is due to its underlying lack of repeatability. This can be traced to high seal friction in its joints. It has been found that this friction is very difficult to characterize (Habibi et al., 1994; Merritt, 1967). Hence, model based friction methods are difficult to apply successfully. This system is a good candidate for BSC to improve its repeatability. For this experimental system, the achievable repeatability is limited by the particular control electronics used. The joint resolver signals, standard on the Schilling, are converted to quadrature encoder waveforms using a special purpose Delta Tau Data/PMAC controller design. The joint angle resolution of this configuration is limited to ± 0.087 degree, which leads to as much as ± 5 mm errors in the end-effector positioning.

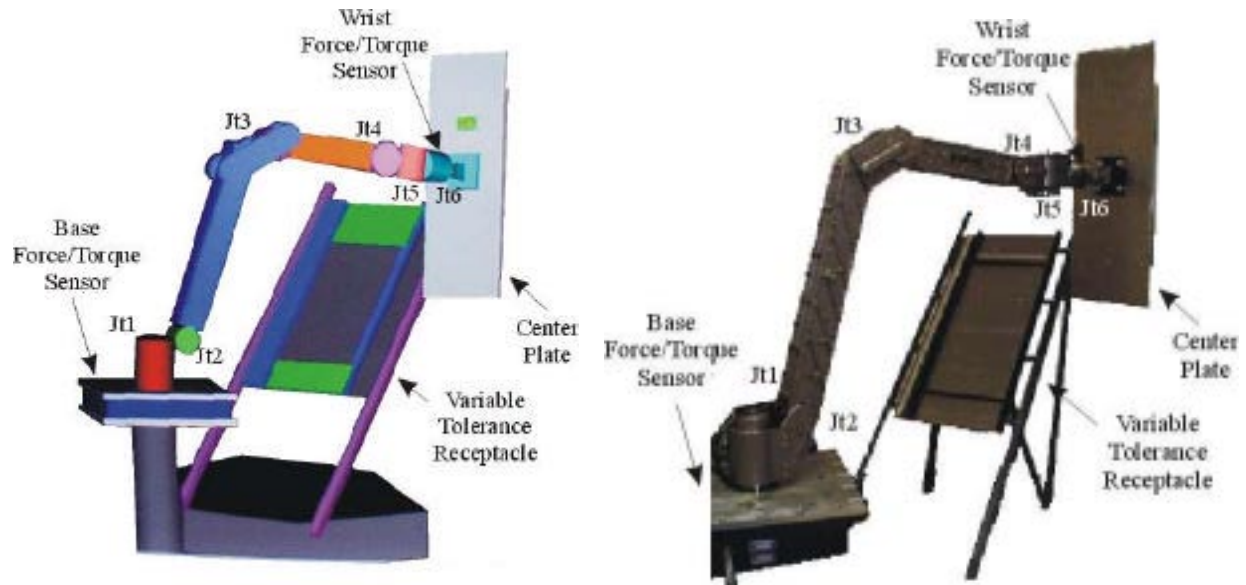


Figure 6. Simulated and real experimental system.

A 6-axis force/torque base sensor is mounted under the manipulator to provide wrench measurements for the BSC algorithm. An 18kg replica of the nozzle dam center-plate was built along with an adjustable plate receptacle that permits the clearances to be varied from interference to several cm. An algorithm to successfully place the rectangular center plate within the receptacle would be easily extendable to perform the other high precision tasks necessary to complete the entire nozzle dam installation, either through teleoperation or as an autonomous subtask.

A pair of Pentax optical theodolites was used to accurately locate the end-effector in 3D space to generate the correction matrix, evaluate weight dependent deflections, and verify the algorithm performance. The resolution of the theodolites was 30 arc seconds, leading to measurement errors of 0.29 mm.

A fixed reference frame, F_0 , is used to express the coordinates of all points. The origin of this reference lies at the intersection of the top of the base sensor and the joint 1 axis. Its z-axis is vertical and its x-axis is defined by a specific horizontal reference direction.

A PC based graphical user interface provides the operator with workspace visualization as well as manipulator control functionality. For all experiments, the sampling rate was ten milliseconds, which was sufficiently fast for the experiments.

The objective of the experiment was to see if the method outlined in Fig. (4) could be applied to the experimental system to improve its repeatability and its absolute accuracy. The object was to have the residual error approach the limit set by the position sensing resolution of the system. In this work, 400 measurements were used to evaluate the basic accuracy of the Schilling. Different payloads were used, with weights up to 45kg. Most of the measurements focused on two specific payloads: one with no weight and another with an 18kg weight (the replica nozzle dam plate).

End-effector measurements of the manipulator under PI control determined the baseline uncompensated system repeatability and accuracy. The relative positioning root mean square error was used as a measure of the system repeatability. Recall that the 12-bit discretization of the resolver signal leads to random errors up to 5.0 mm, and imposes a lower limit of 2.0mm (RMS) on the system repeatability, which sets the accuracy limit of any error compensation algorithm.

The results show that the BSC algorithm was able to reduce the repeatability errors by a factor of 4.8 over PI control. Data was taken by moving the manipulator an arbitrary distance from the test point and then commanding it back to its original position. The maximum errors without BSC were 21.0 mm, and the repeatability was 14.3 mm (RMS). BSC reduced the maximum errors to 5.5 mm with a repeatability of only 3.0 mm (RMS).

Although the BSC algorithm greatly reduced the repeatability errors, there are still 35 mm (RMS) errors in absolute accuracy. Since BSC reduced the system repeatability to 3.0 mm, a model based error correction method can be applied to reduce the accuracy errors.

In order to implement GEC, the geometric and elastic deformation correction matrix was calculated using approximately 350 measurements of the end-effector in different configurations and with different payloads. The remaining points were used to verify the efficiency of the GEC method.

From the system kinematic model with no errors, the ideal coordinates of the end-effector were calculated and subtracted from the experimentally measured values to yield the vector $\mathbf{DX}(\mathbf{q}, \mathbf{W}_e)$ in Eq. (5). The redundant error parameters are eliminated from the error model using Eqs. (9) and (10). The generalized errors are then calculated with Eq. (6). By treating generalized errors as constant in their respective frames, the system absolute accuracy was improved to 13.4 mm (RMS). Since the GEC method allows for the use of polynomials to describe each generalized error, second order polynomials achieved an absolute accuracy of 7.3mm (RMS), an additional 100% improvement.

Figure (7) shows the convergence of original positioning errors as large as 55.1 mm (34.3 mm RMS) to corrected errors of less than 10.7 mm (7.3 mm RMS) with respect to the base frame F_0 . This demonstrates an overall factor of nearly 4.7 improvement in absolute accuracy by using the GEC algorithm. With this improvement in performance, it should make feasible such tasks as the nozzle dam insertion.

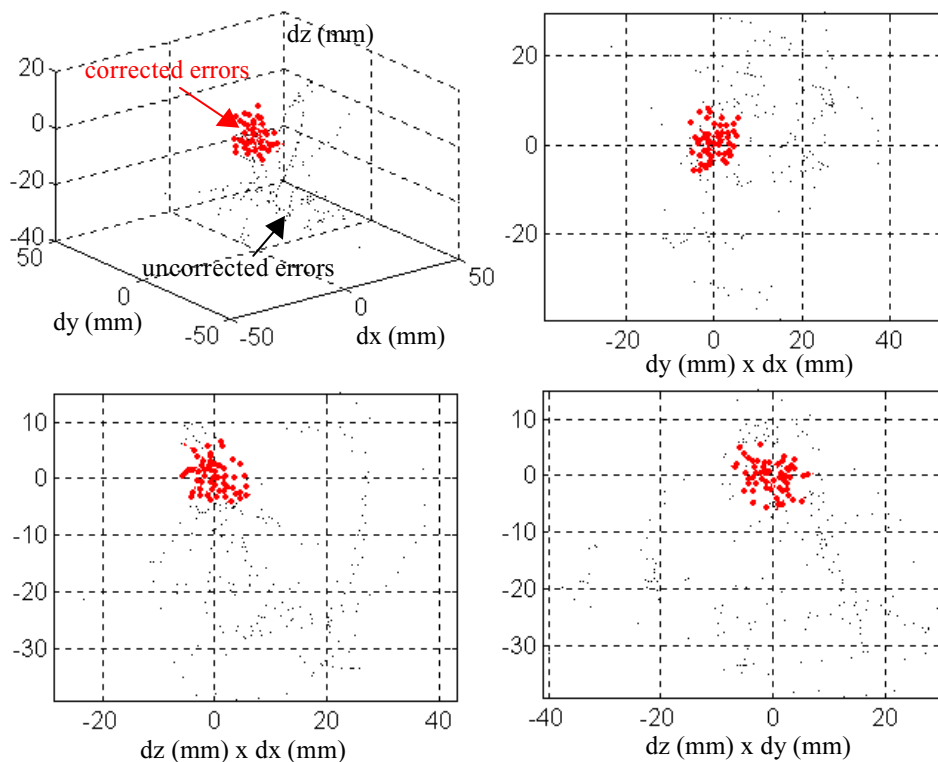


Figure 7. Measured and residual errors after compensation.

5. Conclusions

In this paper, the simplified, model-free form of Base Sensor Control (BSC) is applied to a hydraulic manipulator. The BSC uses a base force/torque sensor to accurately control joint torques, thereby compensating for joint friction. This in turn, substantially improves the manipulator's poor position repeatability. The BSC controller is then combined with a method, called GEC, that compensates for geometric and elastic errors that degrade the absolute positioning accuracy in large manipulators with inherently good repeatability. To improve the accuracy of GEC, a general analytical method to eliminate redundant error parameters in robot calibration is presented. These errors, often non-intuitive, must be eliminated from the error model prior to the identification process, otherwise the robustness of the calibration can be compromised. The results showed that applying the combined error compensation algorithm improved the absolute accuracy of the manipulator by a factor of 4.7 over pure BSC. These developed control techniques are being implemented at the Korean and French power companies.

6. Acknowledgments

The assistance and encouragement of Dr. Byung-Hak Cho of the Korean Electric Power Research Institute (KEPRI) and Mr. Jacques Pot of the Electricité de France (EDF) in this research is most appreciated, as the financial support of KEPRI and EDF.

7. References

- Borm, J.H., Menq, C.H., 1991, "Determination of Optimal Measurement Configurations for Robot Calibration Based on Observability Measure, *Int. Journal of Robotics Research*," Vol.10, No. 1, pp. 51-63.
- Canudas de Wit, C., Olsson, H., Astrom, K.J., Lischinsky, P., 1996, "A New Model for Control of Systems with Friction," *IEEE Transactions on Automatic Control*, Vol.40, No. 3, pp. 419-425.
- Drouet, P., Dubowsky, S., Mavroidis, C., 1998, "Compensation of Geometric and Elastic Deflection Errors in Large Manipulators Based on Experimental Measurements: Application to a High Accuracy Medical Manipulator," *Proceedings of the 6th International Symposium on Advances in Robot Kinematics*, Austria, pp. 513-522.
- Drouet, P., 1999, "Modeling, Identification and Compensation of Positioning Errors in High Accuracy Manipulators under Variable Loading: Application to a Medical Patient Positioning System," Ph.D. Thesis, Un. Poitiers, France.
- Dubowsky, S., Maatuk, J., Perreira, N.D., 1975, "A Parametric Identification Study of Kinematic Errors in Planar Mechanisms," *Trans. of ASME, Journal of Engineering for Industry*, pp. 635-642.
- Habibi, S.R., Richards, R.J., Goldenberg, A.A., 1994, "Hydraulic Actuator Analysis for Industrial Robot Multivariable Control," *Proceedings of the American Control Conference*, Vol.1, pp. 1003-1007.
- Hollerbach, J., 1988, "A Survey of Kinematic Calibration," *Robotics Review*, Khatib ed., MIT Press, Cambridge, MA.
- Hollerbach, J.M., Wampler, C.W., 1996, "The Calibration Index and Taxonomy for Robot Kinematic Calibration Methods," *International Journal of Robotics Research*, Vol. 15, No. 6, pp. 573-591.
- Iagnemma, K., Morel, G., Dubowsky, S., 1997, "A Model-Free Fine Position Control System Using the Base-Sensor: With Application to a Hydraulic Manipulator," *Symposium on Robot Control, SYROCO '97*, Vol.2, pp. 359-365.
- Mavroidis, C., Dubowsky, S., Drouet, P., Hintersteiner, J., Flanz, J., 1997, "A Systematic Error Analysis of Robotic Manipulators: Application to a High Performance Medical Robot," *Proceedings of the 1997 IEEE Int. Conference of Robotics and Automation*, Albuquerque, New Mexico, pp. 980-985.
- Meggiolaro, M., Mavroidis, C., Dubowsky, S., 1998, "Identification and Compensation of Geometric and Elastic Errors in Large Manipulators: Application to a High Accuracy Medical Robot," *Proceedings of the 25th Biennial Mechanisms Conference, ASME*, Atlanta.
- Meggiolaro, M., Jaffe, P.C.L., Dubowsky, S., 1999, "Achieving Fine Absolute Positioning Accuracy in Large Powerful Manipulators", *Proceedings of the International Conference on Robotics and Automation (ICRA '99)*, IEEE, Detroit, Michigan, pp.2819-2824.
- Meggiolaro, M., Dubowsky, S., 2000, "An Analytical Method to Eliminate the Redundant Parameters in Robot Calibration," *Proceedings of the International Conference on Robotics and Automation (ICRA '2000)*, IEEE, San Francisco, pp. 3609-3615.
- Merritt, H., 1967, "Hydraulic Control Systems," John Wiley and Sons, New York, USA.
- Mirman, C. and Gupta, K., 1993, "Identification of Position Independent Robot Parameter Errors Using Special Jacobian Matrices," *International Journal of Robotics Research*, Vol.12, No. 3, pp. 288-298.
- Morel, G., Dubowsky, S., 1996, "The Precise Control of Manipulators with Joint Friction: A Base Force/Torque Sensor Method," *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol.1, pp. 360-365.
- Pfeffer, L.E., Khatib, O., Hake, J., 1989, "Joint Torque Sensory Feedback of a PUMA Manipulator," *IEEE Transactions on Robotics and Automation*, Vol.5, No. 4, pp. 418-425.
- Popovic, M.R., Shimoga, K.B., Goldenberg, A.A., 1994, "Model-Based Compensation of Friction in Direct Drive Robotic Arms," *Journal of Studies in Information and Control*, Vol.3, No. 1, pp. 75-88.
- Roth, Z.S., Mooring, B.W., Ravani, B., 1986, "An Overview of Robot Calibration," *IEEE Southcon Conference*, Orlando, Florida, pp. 377-384.
- Vaichav, R., Magrab, E., 1987, "A General Procedure to Evaluate Robot Positioning Errors," *International Journal of Robotics Research*, Vol.6, No.1, pp. 59-74.
- Waldron, K., Kumar, V., 1979, "Development of a Theory of Errors for Manipulators," *Proceedings of the Fifth World Congress on the Theory of Machines and Mechanisms*, pp. 821-826.
- Wu, C., 1984, "A Kinematic CAD Tool for the Design and Control of a Robot Manipulator," *International Journal of Robotics Research*, Vol.3, No. 1, pp. 58-67.
- Zeza, L.J., 1985, "Steam Generator Nozzle Dam System," *Trans. American Nuclear Society*, Vol.50, pp. 412-413.
- Zhuang, H., Roth, Z.S., 1993, "A Linear Solution to the Kinematic Parameter Identification of Robot Manipulators," *IEEE Transactions in Robotics and Automation*, Vol.9, No. 2, pp. 174-185.
- Zhuang, H., Wu, J., Huang, W., 1996, "Optimal Planning of Robot Calibration Experiments by Genetic Algorithms," *Proc. IEEE Int. Conf. Robotics and Automation*, Minneapolis, pp. 981-986.
- Zhuang, H., Motagheidi, S.H., Roth, Z.S., 1999, "Robot Calibration with Planar Constraints," *Proc. IEEE International Conference of Robotics and Automation*, Detroit, Michigan, pp. 805-810.

MODELAGEM, SIMULAÇÃO E CONTROLE DE FORÇA DE UM MANIPULADOR ROBÓTICO FLEXÍVEL SUJEITO A IMPACTO

Carlos Eduardo Ingar Valer

Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio
Departamento de Engenharia Mecânica
Rua Marquês de São Vicente 225, Gávea, RJ, Brasil. CEP 22453-900.
cingar@mec.puc-rio.br

Rubens Sampaio

Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio
Departamento de Engenharia Mecânica
Rua Marquês de São Vicente 225, Gávea, RJ, Brasil. CEP 22453-900.
rsampaio@mec.puc-rio.br

Resumo. Neste trabalho é estudada a resposta dinâmica de um manipulador robótico flexível em presença de impacto. O modelo dinâmico do manipulador foi obtido através das equações de Lagrange e do método dos modos supostos. Para o impacto foi usado um modelo de força contínuo baseado na lei de contato de Hertz mas modificado para considerar deformações elásticas e plásticas. Este modelo de contato permitiu prever a evolução das forças entre os corpos em colisão durante o impacto. Com a finalidade de estudar a influência da flexibilidade na resposta dinâmica ao impacto, os resultados da simulação foram comparados com a resposta dinâmica de um manipulador rígido. Algumas diferenças substanciais entre os comportamentos dinâmicos foram encontradas. Adicionalmente, com a finalidade de avaliar o desempenho de um controlador em sistemas envolvendo impacto, foi implementado também um sistema de controle de força com ação integral. Foi observado que a flexibilidade contribuiu favoravelmente reduzindo a severidade dos impactos. Este trabalho pretende servir de base para futuras pesquisas na área de impacto envolvendo corpos flexíveis incluindo o desenvolvimento de técnicas de controle mais complexas.

Palavras Chave: Impacto, Estruturas flexíveis, Robótica, Controle de força.

1. Introdução

Impacto é um fenômeno que tem interessado cientistas e engenheiros por um longo período de tempo devido a sua presença relativamente freqüente em sistemas mecânicos. Para citar apenas alguns exemplos podemos mencionar os impactos que acontecem durante o funcionamento de mecanismos intermitentes, mecanismos com pequenas folgas, máquinas de forja ou similares, impressoras, interruptores e conectores elétricos, satélites sujeitos a impactos de pequenos escombros, ou simplesmente o impacto de um bastão de beisebol ou raquete de tênis em uma bola (Cyril, 2000; Margaritu, 1997; Cross, 1999).

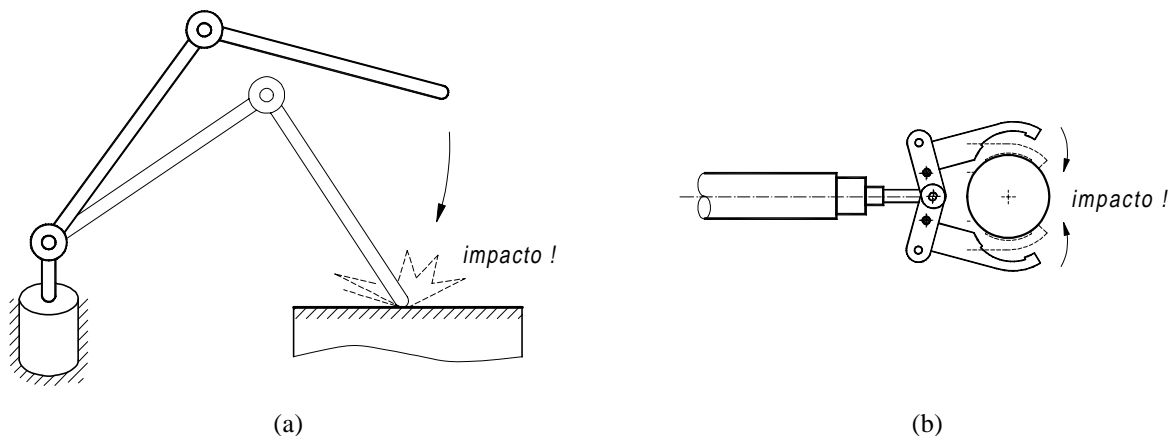


Figura 1. Algumas operações em robótica que podem envolver impacto. (a) Transição desde o movimento livre ao movimento restringido. (b) Captura de um objeto.

Particularmente em robótica acontece impacto em algumas operações feitas através de manipuladores, tais como: durante o martelamento de um prego, durante a captura de um objeto através de uma garra, ou durante a transição de um manipulador robótico desde um movimento não-restrigido a um movimento restringido com uma velocidade relativa não-nula. A última operação é necessária quando o manipulador deve manter contato com o meio de trabalho como parte da sua tarefa. A Fig. (1) apresenta algumas dessas operações.

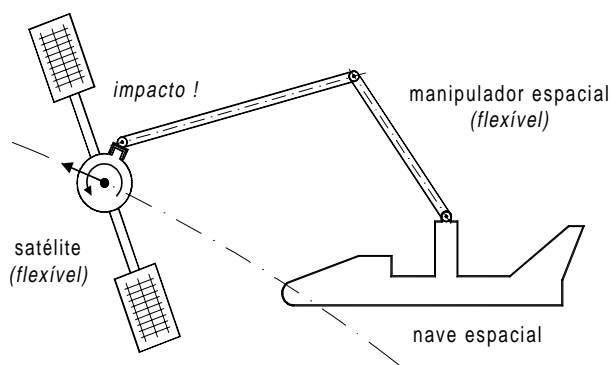


Figura 2. Manipulador espacial flexível no instante da captura de um satélite com movimento relativo.

No passado, os manipuladores robóticos eram controlados para se movimentar lentamente de maneira a evitar impactos. No entanto, com a crescente necessidade de alta velocidade na execução de tarefas, o impacto é inevitável e portanto o controle de manipuladores com relação ao fenômeno de impacto está se tornando indispensável. Adicionalmente, o problema de modelagem e controle de manipuladores sujeitos a impacto fica mais complexo devido às flexibilidades que em algumas situações não podem mais ser ignoradas. A primeira fonte de flexibilidade é devida à tendência atual de redução das massas dos membros dos manipuladores com a finalidade de diminuir a potência de acionamento. Isto faz com que a flexibilidade dos membros tenha que ser considerada em muitos casos abandonando assim a hipótese de rigidez. Isto é mais comum em manipuladores espaciais como no caso apresentado na Fig. (2). Uma outra fonte de flexibilidade é a transmissão de torque. Para multiplicar o torque e aumentar a precisão, muitas vezes uma caixa de transmissão é incluída. Dependendo da sua construção, essas caixas introduzem pequenas ou grandes deformações elásticas. Esta elasticidade é especialmente evidente em redutores de acionamento-harmônico já que a elasticidade é uma propriedade essencial para sua operação (Vukobratovic and Potkonjak, 1999). A Fig. (3) apresenta esquematicamente as principais fontes de flexibilidade nos manipuladores robóticos.

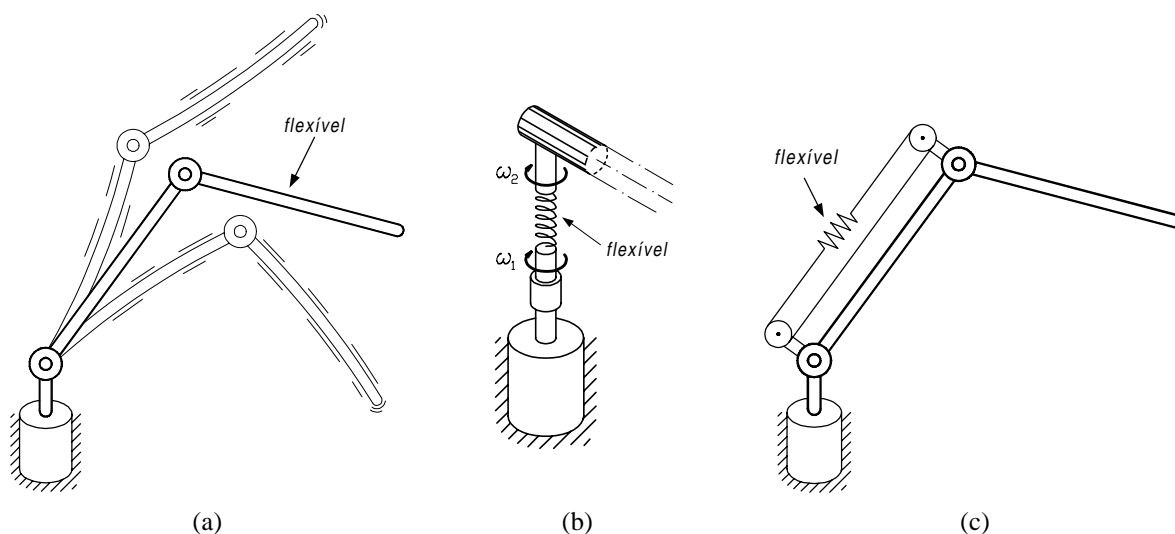


Figura 3. Flexibilidade em manipuladores robóticos. (a) Elos flexíveis. (b) Juntas flexíveis. (c) Transmissão flexível.

A modelagem e/o controle de manipuladores robóticos sujeitos a impacto é uma área desafiante, cujos resultados poderão afetar consideravelmente aplicações avançadas de robôs na indústria e incrementará sua eficiência e produtividade. Contudo, nos últimos dez anos poucas tentativas têm sido feitas para tornar aplicáveis as técnicas de controle em problemas envolvendo contato/impacto e além disso, a literatura de modelagem e/ou controle de impacto em sistemas mecânicos com flexibilidades apreciáveis é quase inexistente (Cataldo and Sampaio, 2001; Sampaio and Tavares, 1998).

Neste artigo é estudado o fenômeno de impacto em um manipulador simples porém flexível. Através deste modelo pretende-se entender melhor o fenômeno de impacto em manipuladores flexíveis analisando e interpretando cuidadosamente os resultados das simulações. Adicionalmente, com a finalidade de avaliar o desempenho de um controlador em sistemas envolvendo impacto, foi incluído também um sistema de controle de força tipo integral. Em todas as simulações o caso de um manipulador rígido é também considerado com a finalidade de poder avaliar a influência da flexibilidade na dinâmica do impacto. Este trabalho pretende servir de base para futuras pesquisas na área incluindo o desenvolvimento de técnicas de controle mais complexas.

2. Modelo do manipulador e modelo de impacto

Um esquema do sistema estudado é a apresentado na Fig. (4). Este sistema representa de uma maneira simplificada um manipulador flexível com apenas um membro o qual é elástico e será considerado como uma viga flexível. O acionamento é feito através de um motor DC colocado numa extremidade da viga. Na outra extremidade considera-se um pequeno carregamento o qual é representado por uma massa concentrada. O manipulador tem liberdade para girar mas existe uma parede que limitará o seu movimento provocando impacto.

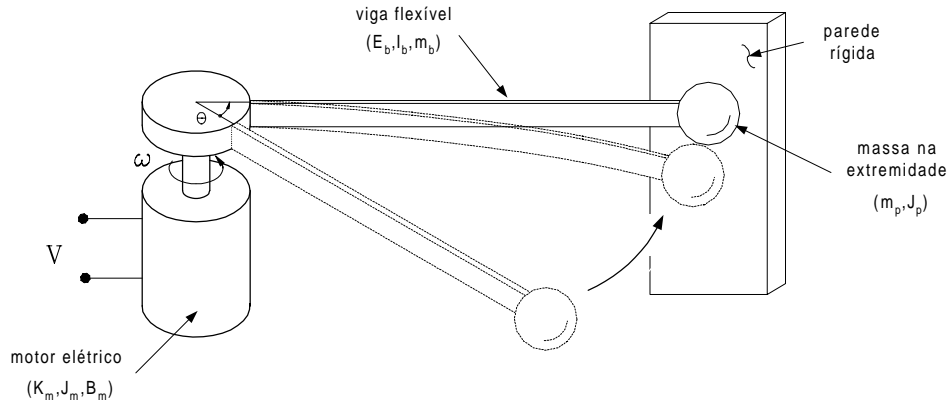


Figura 4. Esquema do sistema considerado para estudar a dinâmica do impacto em manipuladores flexíveis.

Como apresentado a Fig. (5), durante o funcionamento do manipulador, a viga flexível experimenta movimento rotacional de corpo rígido e vibração devido a sua flexibilidade. A parte flexível foi modelada através do método dos modos supostos, onde foi considerado que o deslocamento da viga flexível $w(x,t)$ pode ser escrito da seguinte maneira:

$$w(x,t) = \sum_{i=1}^n \Phi_i(x) q_{v_i}(t) \quad (1)$$

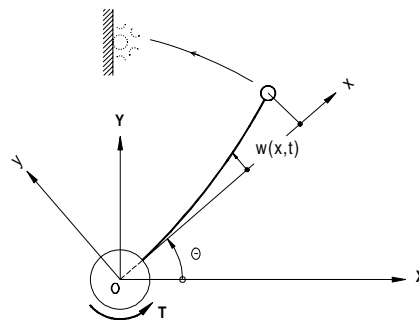


Figura 5. Rotação e deformação da viga flexível do manipulador.

onde Φ_i são um conjunto de funções admissíveis e q_{v_i} são as coordenadas modais (Meirovitch, 1986). A seguir a energia potencial elástica e cinética de todo o manipulador foi calculada e por aplicação direta das equações Lagrange, a equação de movimento, do sistema é obtida. Por simplicidade apresentamos aqui apenas a equação resultante:

$$\mathbf{M} \ddot{\mathbf{r}} + \mathbf{D} \dot{\mathbf{r}} + \mathbf{K} \mathbf{r} = \mathbf{B}_T \mathbf{T}_m + \mathbf{B}_f \mathbf{f}$$

$$\mathbf{M} = \begin{bmatrix} J_r & \mathbf{p}^T \\ \mathbf{p} & \mathbf{M}_v \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} D_r & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_v \end{bmatrix}, \quad \mathbf{K} = \begin{bmatrix} 0 & \mathbf{0} \\ \mathbf{0} & \mathbf{K}_v \end{bmatrix} \quad (2)$$

onde \mathbf{M} e \mathbf{K} são as matrizes de massa e rigidez do sistema respectivamente; $\mathbf{r}^T = [\theta \quad \mathbf{q}_v^T]$; θ representa o deslocamento angular do sistema como corpo rígido; \mathbf{q}_v é o vetor de coordenadas modais da vibração elástica para os modos supostos na viga; J_r é a inércia rotacional do sistema todo como corpo rígido; \mathbf{M}_v é a matriz de inércia da viga flexível incluindo a inércia da massa concentrada na extremidade; \mathbf{p} representa o acoplamento entre o movimento de rotação como corpo rígido e a vibração como corpo flexível; \mathbf{D} é a matriz de amortecimento do sistema sendo D_r o atrito viscoso no eixo do motor e \mathbf{D}_v é a matriz de amortecimento na viga; \mathbf{K}_v é a matriz de rigidez da viga; \mathbf{B}_T é a

matriz de influência para o torque do motor $T_m = (K_m / R_m) V$, onde V , K_m , R_m são a voltagem, a constante elétrica e a resistência elétrica do motor respectivamente; e B_f é a matriz de influência para a força de contato/impacto f a qual atua na extremidade da viga flexível. A seguir, uma transformação modal para a Eq. (2) foi feita, obtendo-se:

$$\ddot{q} + 2 \Lambda \Omega \dot{q} + \Omega^2 q = B_T^m T_m + B_f^m f \tag{3}$$

onde q são agora as coordenadas modais do sistema; Λ é a matriz de amortecimento modal do sistema; e Ω é uma matriz diagonal com as frequências naturais do sistema. É importante notar que a frequência zero está incluída nesta última matriz devido ao modo de rotação como corpo rígido do manipulador. Finalmente, esta última equação foi convertida para a equação de estado com a finalidade de poder integrá-la numericamente.

2.2 Modelo de contato/impacto

Na literatura encontramos basicamente dois métodos para modelar impacto: o método de balanço de momentum e o método do elemento de contato. O primeiro foi desenvolvido baseado nas equações de impulso-momentum para corpos rígidos e através dele é obtido um sistema algébrico de equações o qual tem que ser resolvido para obter as velocidades após o impacto. As equações usam o coeficiente de restituição para tomar em conta as perdas de energia (Cataldo, 1999). A simplicidade e eficiência computacional fazem este método muito atrativo e mesmo sendo desenvolvido para corpos rígidos, encontrou-se na literatura trabalhos nos quais este método tem sido aplicado na modelagem de impacto em corpos flexíveis (Margaritu, 1999). No entanto, neste método supõe-se que os impactos aconteçam instantaneamente e portanto não é possível prever forças de contato nem o comportamento do sistema durante o impacto. Já no segundo método, durante o impacto, um elemento que pode estar formado por uma mola e um amortecedor é considerado conectando os nós associados com as superfícies impactantes. Através das deformações deste elemento é possível determinar as forças de contato/impacto atuante entre os corpos (Rajalingham and Rakheja, 2000). Uma vez que no presente trabalho um dos objetivos é estudar a dinâmica durante o impacto esta última metodologia foi a escolhida.

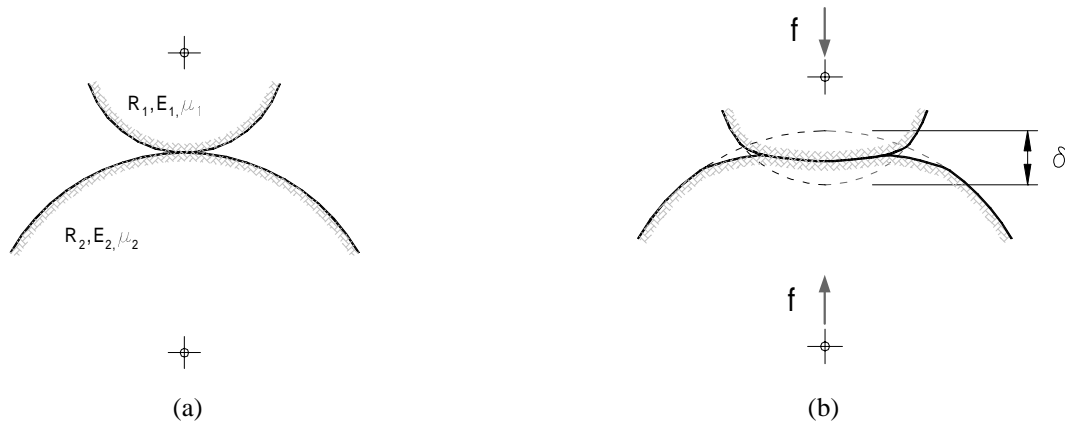


Figura 6. Corpos em contato e a Lei de Hertz. (a) Superfícies justamente antes do contato. (b) Durante o contato.

A base do método do elemento de contato é uma lei que relacione a deformação local nas superfícies em contato e as forças existentes entre elas. Neste sentido, a lei de contato de Hertz, a qual foi obtida considerando campos de esforços e deformações na região de contato, provê tal relação (Johnson, 1985). Com relação à Fig. (6), que apresenta o comportamento de dois corpos justamente antes e durante o contato, a lei de Hertz é a seguinte:

$$f = K_H \delta^{3/2} \tag{4}$$

onde f é a força de contato/impacto; δ é a deformação local como apresentado na Fig. (6)-b; e K_H é a constante de Hertz que depende da geometria e propriedades das superfícies em contato. Por exemplo para o caso de duas superfícies esféricas de raios R_1 , R_2 e módulos de Young E_1 , E_2 e coeficientes de Poisson μ_1 , μ_2 , temos:

$$K_H = \frac{4}{3} E^* R^* \tag{5}$$

$$E^* = \left[\frac{1-\mu_1^2}{E_1} + \frac{1-\mu_2^2}{E_2} \right]^{-1}, \quad R^* = \left[\frac{1}{R_1} + \frac{1}{R_2} \right]^{-1} \quad (6)$$

A lei de contato de Hertz é de natureza estática mas existe evidência experimental de que quando aplicada a situações de impacto com moderada velocidade (até 100 m/s) os resultados validam a equação. Isto é devido ao fato que as ondas de esforços atravessam muitas vezes a região de contato antes da separação dos corpos (Johnson, 1985). No entanto, uma limitação da lei de Hertz é que considera apenas deformações puramente elásticas e portanto não toma em conta as perdas de energia durante o impacto. Uma alternativa é considerar uma modelo de amortecedor em paralelo com a mola não-linear que representa a lei de Hertz dada pela Eq. (4), como no trabalho de Lankarani (1994). Entretanto, mesmo que este modelo consiga considerar as perdas de energia, ele não é realista pois o verdadeiro mecanismo de dissipação de energia não é um amortecedor e sim as deformações plásticas. No presente trabalho será adotada uma modificação da lei de Hertz como aquela apresentada no trabalho de Thornton (1997), a qual é uma combinação da lei de Hertz clássica com a teoria de deformação elasto-plástica e foi usada na referência para determinar analiticamente o coeficiente de restituição no impacto de duas esferas com colisão central. Esta nova lei de contato é representada na Fig. (7) e nas Eqs. (7)-(10), onde y representa o ponto de escoamento do material.

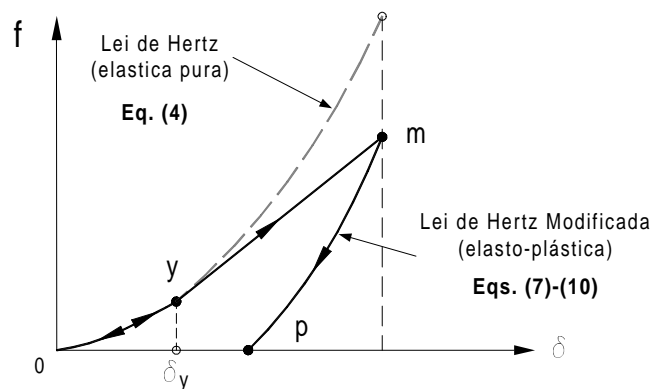


Figura 7. Lei de Hertz modificada para considerar deformações plásticas.

Fase 1. Carregamento Elástico ($0 \rightarrow y$):

$$f = K_H \delta^{3/2}; \quad \dot{\delta} > 0 \quad \text{e} \quad 0 \leq \delta \leq \delta_y \quad (7)$$

Fase 2. Carregamento Elasto-Plástico ($y \rightarrow m$):

$$f = K_H \delta_y^{3/2} + K_y (\delta - \delta_y); \quad \dot{\delta} > 0 \quad \text{e} \quad \delta_y \leq \delta \leq \delta_m \quad (8)$$

Fase 3. Descarregamento Elástico: Restituição ($m \rightarrow p$):

$$f = K_H \delta_y^{3/2} + K_y (\delta_m - \delta_y) + K_H (\delta^{3/2} - \delta_m^{3/2}); \quad \dot{\delta} < 0 \quad \text{e} \quad f \geq 0 \quad (9)$$

onde,

$$K_y = \left[\frac{\partial f}{\partial \delta} \right]_{\delta=\delta_y} = \frac{3}{2} K_H \delta_y^{1/2} \quad (10)$$

3. Simulação do manipulador sujeito a impacto

O modelo para o manipulador flexível e a lei de contato de Hertz modificada já descritos na seção anterior foram implementados num programa em MatLab[®]. Os valores numéricos dos parâmetros usados para simulação são apresentados na Tab. (1). Quatro modos flexíveis foram suficientes para a modelagem da viga e com um aumento do número de modos não houve modificação considerável nos resultados da simulação. Como condição inicial foi

considerado o sistema em repouso numa posição correspondente a $\theta = 45^\circ$ segundo a Fig. (5). Na simulação, o sistema foi acionado por um pulso retangular de voltagem no motor atuando durante os primeiros 0,1 segundos. Logo após o acionamento foi desligado e o manipulador se movimentou livremente até colidir com a parede. Durante a simulação um algoritmo especial foi necessário para poder capturar o começo de cada impacto com uma boa precisão e detectar os prováveis impactos múltiplos que podem acontecer devido à excitação dos modos de altas frequências. Basicamente este algoritmo consiste em integrar numericamente as equações de movimento com um passo razoável Δt até que num certo instante t seja detectado um contato (deformação $\delta(t - \Delta t) = 0$ e $\delta(t) > 0$). Neste ponto, o algoritmo faz que o integrador regresse até o passo anterior $t - \Delta t$ e refaça os cálculos com um novo passo diminuído. Se com este passo a nova deformação $\delta(t)$ é maior que uma tolerância pre-estabelecida, o integrador terá que voltar ao passo anterior e tentar com um passo ainda menor e assim sucessivamente até conseguir um valor menor que a tolerância. O algoritmo para detectar o instante no qual o contato acaba é análogo.

Tabela 1. Valores numéricos dos principais parâmetros usados para a simulação do manipulador flexível.

variável	descrição	valor
L_v	comprimento da viga flexível	300 mm
$E_v I_v$	rigidez à flexão da viga	12 N · m ²
m_v	massa por unidade de comprimento da viga	0,6 kg / m
n	número de modos supostos na viga	4
ζ_i	coeficiente de amortecimento dos modos flexíveis da viga	1ro. modo: 0,01 ; 2do. modo: 0,05 0,1 para os demais modos
m_p	massa na extremidade do manipulador	30 gr
K_H	constante de Hertz (rigidez de contato)	$7,3 \times 10^9$ N / m
S_y	resistência ao escoamento das superfícies em contato	255 MPa (alumínio)
K_m	constante elétrica do motor	0,1 N - m/A
R_m	resistência elétrica no motor	7 Ω

Os resultados da simulação são apresentados nas Figs. (8)-(13) onde com a finalidade de avaliar a influência da flexibilidade no comportamento dinâmico do manipulador sujeito a impacto foi adicionada uma simulação considerando um manipulador com a viga rígida. No caso do manipulador flexível foram detectados impactos múltiplos: um impacto principal (Fig. (8)) e três sub-impactos (Fig. (9)). Isto é peculiar em impacto de corpos flexíveis devido à excitação dos modos de altas frequências. Com referência à Fig. (8) que apresenta a evolução da força durante o primeiro impacto do manipulador flexível comparada com a evolução da força no caso rígido há uma observação importante: o impacto no manipulador flexível ocorreu com uma menor intensidade e também menor duração da força de impacto com relação ao caso rígido. Ou seja, a flexibilidade diminuiu a severidade do impacto. Isto pode ser verificado também na Tab. (2). O coeficiente de restituição também foi calculado para cada impacto e apresentado na Tab. (2). Pode-se observar que flexibilidade influenciou neste parâmetro (veja também a Fig. (10)). Já na Fig. (11) pode-se observar como a lei de contato adotada neste trabalho (Fig. (7)) foi reproduzida durante a simulação.

Tabela 2. Características dos impactos detectados na simulação.

<i>manipulador flexível</i>				
No.	força max. (N)	duração (μs)	veloc. antes (m/s)	coef. restit. *
1	989	108	0,95	0,62
2	293	143	0,26	0,86
3	400	138	0,32	0,80
4	196	174	0,14	0,96
<i>manipulador rígido</i>				
No.	força max. (N)	duração (μs)	veloc. antes (m/s)	coef. restit. *
1	1478	152	0,95	0,57

* coeficiente de restituição $\equiv v^n_{após} / v^n_{antes}$

Na Fig. (12) apresenta-se o fenômeno da troca de energia entre os modos de vibração durante o primeiro impacto do manipulador flexível. Antes do primeiro impacto, toda a energia é armazenada basicamente no modo de rotação de corpo rígido, mas durante o impacto existe uma excitação dos modos flexíveis. Isto deve-se ao acoplamento dos modos através da matriz B_f^m na Eq. (3). Após o impacto, a energia total diminuiu devido à natureza plástica da lei de contato. A energia armazenada no modo de rotação como corpo rígido também diminuiu, uma vez que uma parte desta foi

transferida aos modos flexíveis. Isto faz que o manipulador flexível, após a colisão, consiga retornar com uma menor velocidade angular comparado com o manipulador rígido. Veja Fig. (13).

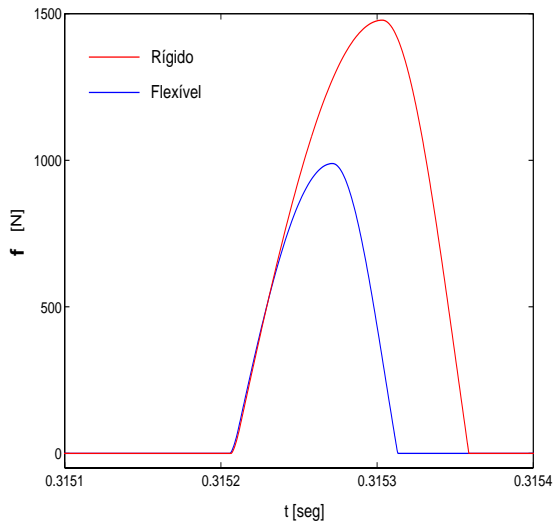


Figura 8. Força do impacto para os casos do manipulador rígido e flexível

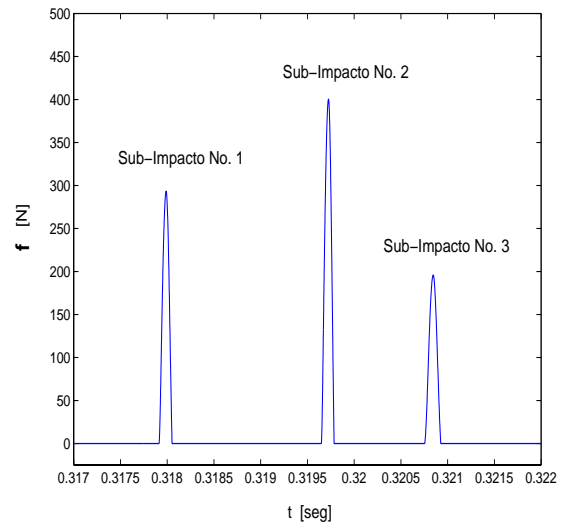


Figura 9. Sub-impactos no caso do manipulador flexível

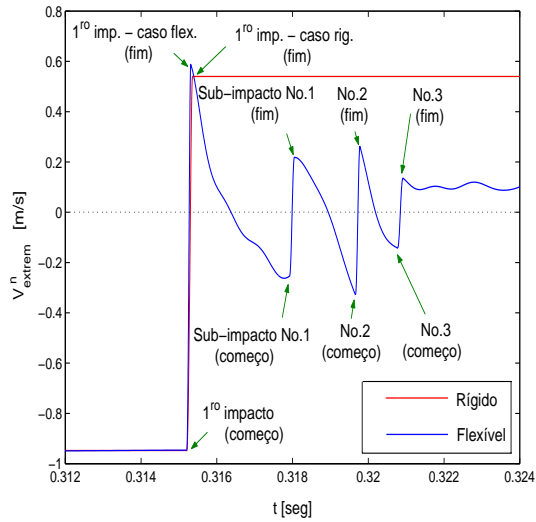


Figura 10. Velocidade normal à parede durante os impactos dos manipuladores rígido e flexível

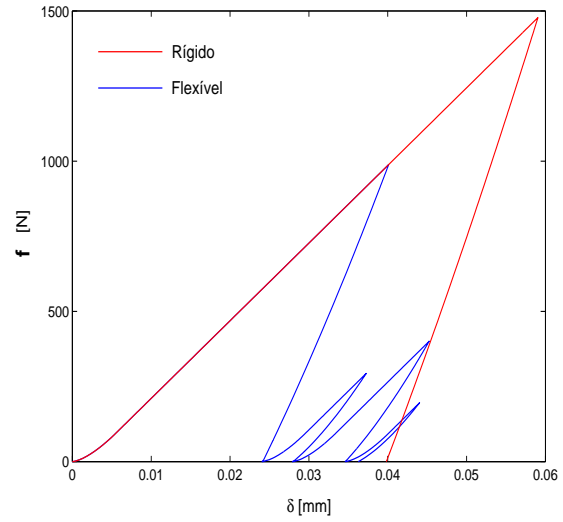


Figura 11. Força contra deformação local para os casos rígido e flexível

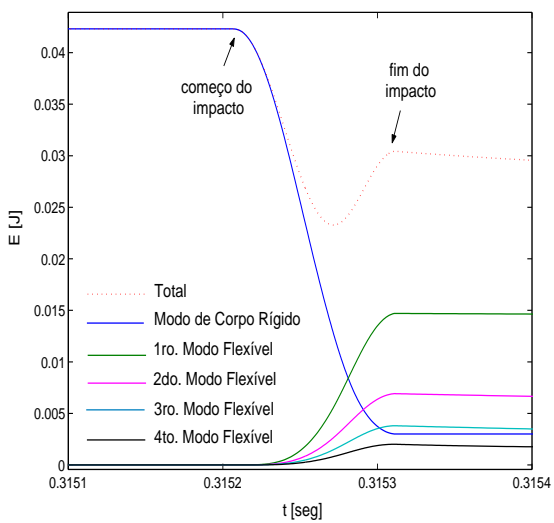


Figura 12. Distribuição de energia por modo de vibração durante o primeiro impacto .

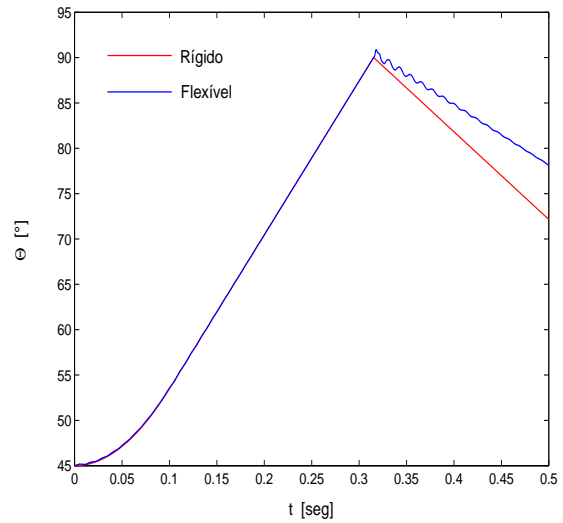


Figura 13. Deslocamento angular para os casos dos manipuladores rígido e flexível

4. Controle de força

Nesta seção é considerada a implementação de um controle de força no manipulador flexível em estudo. O controle da força de contato é necessário quando é requerido que o manipulador mantenha contato estável com o meio de trabalho, evitando assim que forças excessivas sejam exercidas no manipulador e regulando a força desejada no contato. Na literatura encontramos algumas destas técnicas como o método do controle de impedância, controle híbrido de posição/força e o controle integral (Sciavicco and Siciliano, 1996). As duas primeiras técnicas são desenvolvidas apenas para controladores rígidos e a última não tem restrições. No caso em estudo temos um manipulador flexível, por tal motivo foi escolhido o controle de força com ação integral. É importante fazer notar aqui que todas essas técnicas de controle de força foram projetadas considerando que não há impacto (ou seja quando o manipulador atinge a superfície do meio de trabalho com velocidade relativa nula na direção normal). No entanto, uma vez que não existem técnicas específicas para o controle de força em presença de impacto, e devido a que na prática em presença de impacto é o controlador de força que tem que lidar com o fenômeno transiente até conseguir que o manipulador exerça forças de contato estavelmente, o objetivo de esta seção será estudar e avaliar o desempenho de um controlador de força quando o manipulador está sujeito a impacto.

A estrutura de controlador utilizada neste trabalho está apresentada na Fig. (14). Como pode se observar, o erro da força de contato, definido como a diferença entre o valor desejado e o valor real, é realimentado de maneira integral. Quanto maior o ganho K_I , mais rápida será a resposta do sistema. No entanto nas primeiras simulações o sistema apresentou instabilidade. Por tal motivo, um ganho $K_{\dot{\theta}}$ para realimentação proporcional à velocidade de rotação do eixo do motor foi acrescentado no sistema de controle (Fig. (14)). Com isto a margem de instabilidade do sistema aumentou e foi possível aumentar o ganho K_I sem haver instabilidade. Portanto, a lei de controle adotada foi:

$$u = K_I \int_0^t (f_d - f) dt - K_{\dot{\theta}} \dot{\theta} \tag{11}$$

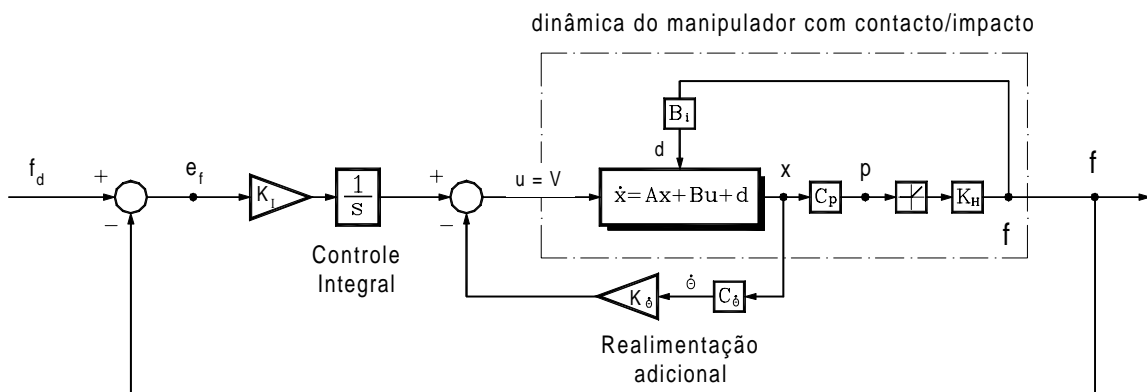


Figura 14. Controle de força com ação integral para um manipulador sujeito a impacto

Após algumas tentativas foram escolhidos os valores $K_I = 100$ e $K_{\dot{\theta}} = 50$ para os ganhos do controlador e $f_d = 5 N$ para a força de contato desejada. A primeira simulação foi feita sem considerar impacto (ou seja, considerando que a extremidade do manipulador atinge a parede com velocidade nula). Os resultados são apresentados na Fig. (15) tanto para o caso rígido como para o caso flexível. Pode-se observar que em ambos os casos o controle de força mostrou-se efetivo atingindo a força de contato desejada após de aproximadamente 1 segundo.

Finalmente, foi feita uma simulação com o controle de força implementado sendo agora em presença de impacto (foi considerado que o manipulador atinge a superfície de contato com uma velocidade angular $\dot{\theta} = 5 \text{ rad / seg}$). Os resultados são apresentados na Fig. (16) para o caso do manipulador flexível e na Fig. (17) para o caso do manipulador rígido. Pode se observar que em ambos os casos existe uma fase transiente durante a qual o contato não é estabelecido de forma contínua e sim de forma intermitente. Terminada esta fase o contato estável é obtido. Essa transição poderia ser inaceitável em algumas operações robóticas. No entanto a duração da fase transiente foi menor no caso flexível comparado com o caso rígido. Além disso, a severidade do impacto foi maior no caso rígido, por tanto podemos dizer que a flexibilidade contribuiu favoravelmente durante o controle de força em presença de impacto.

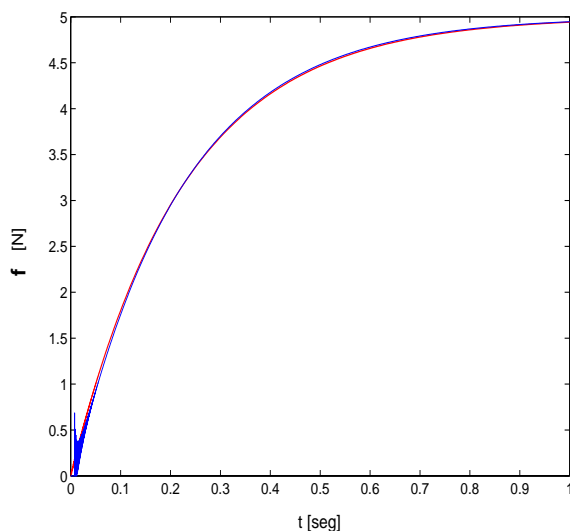


Figura 15. Força de contato para o caso do manipulador rígido e flexível com o controle de força integral implementado. A velocidade de aproximação é zero (sem impacto)

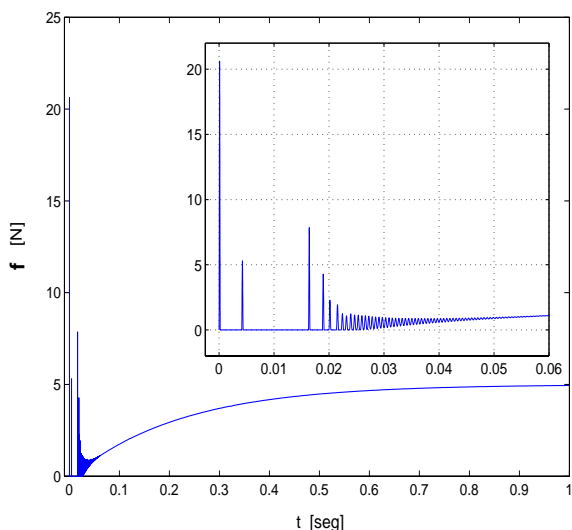


Figura 16. Força de contato/impacto para o caso de manipulador flexível com um controle de força integral. Velocidade de aproximação não-nula (sujeito a impacto)

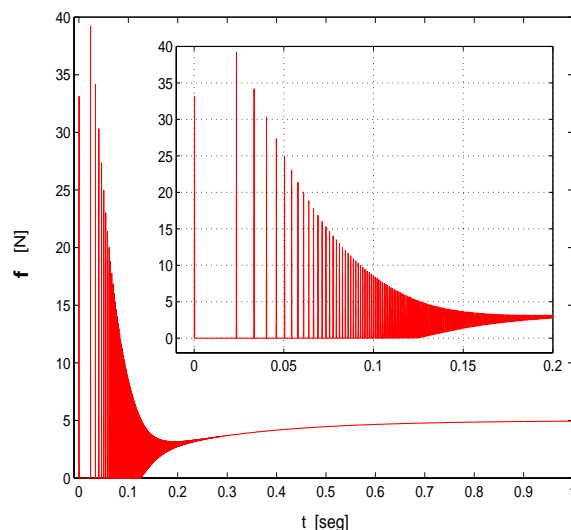


Figura 17. Força de contato/impacto para o caso de manipulador rígido com um controle de força integral. Velocidade de aproximação não-nula (sujeito a impacto)

5. Conclusões

A resposta ao impacto de um manipulador com um membro rígido/flexível foi estudada. Como modelo de impacto foi usada uma lei de Hertz modificada que considera deformações elásticas e plásticas. Um algoritmo especial foi necessário para capturar com precisão os impactos. A diferença de trabalhos anteriores, as simulações feitas aqui permitiram estudar a evolução no tempo das forças de impacto durante o período de contato, tanto para o caso de um manipulador flexível como para um manipulador rígido. Foi mostrado neste trabalho que a flexibilidade afeta significativamente tanto a severidade do impacto como a resposta pós-impacto quantificada pelo coeficiente de restituição. Foi também implementado um controle de força tipo integral para os manipuladores rígido e flexível sujeitos a impacto e o desempenho do controlador foi avaliado através de simulações. No regime permanente, nos casos rígido e flexível, o controlador de força conseguiu manter a força de contato desejada. No entanto numa fase transitente, durante a qual o contato tenta se estabelecer algumas diferenças foram detectadas. Esta fase transitente foi de menor duração e menor severidade para o caso do manipulador flexível.

6. Referências

- Cataldo, E. and Sampaio, R., 2001, "A Review of Rigid Bodies Collision Models in the Plane", 11^o Workshop on Dynamics and Control, sponsored by the NSF, CNPq and FAPERJ, Rio de Janeiro, Brazil.
- Cross, R., 1999, "Impact of a Ball with a Bat or Racket", American Journal of Physics, Vol. 67, No. 8, pp. 692-702.

- Cyril, X., Misra, A. K., Ingham, M. and Jaar, G. J., 2000, "Post-Capture Dynamics of a Spacecraft-Manipulator-Payload System", *Journal of Guidance Control and Dynamics*, Vol. 23, No. 1, pp. 95-100.
- Johnson, K. L., 1985, "Contact Mechanics", Cambridge University Press.
- Lankarani, H.M. and Nikravesh, P. E., 1994, "Continuous Contact Force Models for Impact Analysis in Multibody Systems", *Non Linear Dynamics*, Vol. 5, No. 2, pp. 193-207.
- Margaritu, D.B., 1999, "Control Techniques for Impacting Flexible Systems", *Archive of Applied Mechanics*, Vol. 69, pp. 55-568
- Margaritu, D. B., 1997, "The Impact of Flexible Links with Solid Lubrication", *Journal of Sound and Vibration*, Vol. 205, No. 5, pp. 712-720.
- Meirovitch, L., 1986, "Elements of Vibration Analysis", McGraw-Hill.
- Rajalinghan, C. and Rakheja, S., 2000, "Analysis of Impact Force Variations During Collision of Two Bodies Using a Single-Degree-of-Freedom System Model", *Journal of Sound and Vibration*, Vol. 229, No. 4, pp. 823-835.
- Sampaio, R. and Tavares, H., 1998, "Colisão em Sistemas de Multicorpos", *RBCM- J. of the Brazilian Soc. Mechanical Sciences*, Vol. 20, No. 1, pp. 116-125.
- Sciavicco, L. and Siciliano, B., 1996, "Modeling and Control of Robot Manipulators", McGraw-Hill, Inc.
- Thornton, C., 1997, "Coefficient of Restitution for Collinear Collisions of Elastic-Perfectly Plastic Spheres", *Transactions of the ASME: Journal of Applied Mechanics*, Vol. 64, pp. 383-386.
- Vukobratovic, M. K. and Potkonjak, V., 1999, "Dynamics of Contact Tasks in Robotics. Part I: General Model of Robot Interacting with Environment", *Mechanism and Machine Theory*, Vol. 34, pp. 923-942.

MODELING, SIMULATION AND FORCE CONTROL OF A FLEXIBLE ROBOTIC ARM IMPACTING ITS ENVIROMENT

Carlos Eduardo Ingar Valer

Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio
 Department of Mechanical Engineering
 cingar@mec.puc-rio.br

Rubens Sampaio

Pontifícia Universidade Católica do Rio de Janeiro, PUC-Rio
 Department of Mechanical Engineering
 rsampaio@mec.puc-rio.br

Abstract. *In this paper the impact response of a single-link flexible manipulator is studied. The flexible manipulator was modeled by using the assumed modes method and the Lagrange equations. An realistic elastic-plastic contact law based on the classic Hertz law was used to model the impact between the bodies. This contact law allows continuous transitions between contact and non-contact phases and it is capable of predicting impact forces histories. The effect of the flexibility was studied by comparing the impact response of the flexible manipulator with a rigid manipulator case. Some substantial differences was founded. Moreover, a force control strategy was implemented in the manipulator with the purpose of investigate the performance of a force controlled manipulator in presence of impact. It has been shown that the flexibility significantly affects the severity of impact which can be characterized by small impulses. This work will be also a base for future researches.*

Keywords: *Impact, Flexible structures, Robotics, Force control.*

Position Control of a Milling Module using Proportional Hydraulic

Fernando Buendgens Schneider

Indústrias Schneider S.A.
fernando@issa.com.br

Victor Juliano De Negri

Hydraulic and Pneumatic System Laboratory (LASHIP) – Federal University of Santa Catarina (UFSC)
victor@emc.ufsc.br

Mauro André Barbosa Cunha

Pelotas Federal Center for Technological Education
mauro@cefetrs.tche.br

Abstract: This paper addresses the design and construction of a milling module for tooling of mechanical components named injectors, which are coupled to hydrodynamic pumps for fluid pumping. Due to precision requirements, efforts and displacement profile, a proportional hydraulic system was chosen and mathematically modeled. Through linear simulation, the use of a PI controller proved to satisfy the dynamic and static requirements. All tests applied to determine the parameters of the hydraulic components followed the ISO 6404 recommended procedures. This allows the tests to be easily repeated during the construction of other similar systems. The results show that the design specifications were achieved.

Keywords: miling module, position control, hydraulic actuator, parameter identification

1 Introduction

The injector, Fig. (1), a cast iron piece, is used together with some pumps to improve the extraction of water from deep wells. These injectors come from the foundry rough and need to be milled, in order to be used in the assembled pump. Prior milling was a complicated process. It depended on heavy presses to lock the injector in place, which was a slow and unproductive process, demanding large amounts of injectors to be milled at the same time, to justify the setup time needed. The goal of this project was to design and build a machine capable of milling the injector, within tolerances of ± 0.2 mm, and with a quick and easy setup, which will allow the production of small lots to become economically viable.

At Indústrias Schneider S.A, three types of injectors, called I0, I1 and I2, are produced. They basically have the same shape, differing only in dimension and volumetric capacity. This characteristic allows one same tool, a milling head with 4 cutting areas, to be used in all of them. Based on this fact, a milling module was designed and built using proportional hydraulics.

The use of proportional hydraulic to mill this parts family was because of the excellent torque/dimension ratio that the hydraulic actuators present, developing great forces necessary for milling in little physical space. However, the hydraulic actuators have some characteristics that difficult their closed loop control. Among others, they are highly non-linear systems, present lightly damped dynamics, and some of its parameters are of difficult measurement.

In this work, the modeling, identification and control of the hydraulic actuator used for the milling machine position control are presented. Section 2 describes the milling machine. In section 3, the hydraulic actuator mathematical model is presented. Section 4 describes the parameter identification. In section 5, simulation results are presented. Section 6 presents the experimental implementation. Section 7 shows conclusions and perspectives.



Figure 1 – Injector.

2 The Milling Machine

To construct the milling module the following equipment was used: a steel base to fix a double action hydraulic cylinder, 63.5mm diameter with 220 mm of stroke, and a proportional Parker valve D1FX AJ ($\pm 10V$), all used to move and control the position of the milling tool; also for position control, a resistive position transducer, with 225 mm of stroke, and precision of $1\mu m$ according to the manufacturer; a motor-reduction assembly, responsible for the tool's rotation and the extraction of shavings from the injector; a special coupling that prevents the rotation produced by the motor to be transmitted to the hydraulic cylinder, once this last one is not constructed to support its rod rotation; a hydraulic power unit and a clamp to lock the injector during the milling. Figure (2) shows the whole milling machine on an iron base structure under operating condition.



Figure 2 – The Milling Machine

3 Modeling

The hydraulic system model is based on physical laws, which describe the behavior of the proportional valve and the hydraulic cylinder.

Applying the flowrate orifice equation to the valve and linearizing it around an operation point, one obtains (Stringer, 1976):

$$Q_c = Kq_0 \cdot xv - Kc_0 \cdot p_c, \quad (1)$$

where Q_c is the flowrate, Kq_0 is the flowrate gain, xv is a voltage signal corresponding to the spool position, Kc_0 is the pressure-flowrate gain and p_c is the load pressure difference.

The valve dynamics can be represented as a first order system (Yao et al., 1998, Alleyne, 1996):

$$xv = \frac{1}{\tau s + 1} \cdot u, \quad (2)$$

where τ is the valve time constant and u is the input signal.

Remark 1 – In the static conditions one has “ $xv = u$ ”.

Using the continuity equation, the flowrate is given by (De Negri, 1988):

$$Q_c = Ap \cdot \frac{dy}{dt} + \frac{Vt}{4\beta} \cdot \frac{dp_c}{dt} + C_{in} \cdot p_c, \quad (3)$$

where Ap is the piston area, y is the piston position, Vt is the total volume, β is the fluid bulk modulus and C_{in} is the valve internal leakage coefficient.

Applying the Newton's second law to the cylinder and load, one finds:

$$F_a = A_p \cdot p_c = M_t \cdot \frac{d^2 y}{dt^2} + (B_a + B_c) \cdot \frac{dy}{dt} + F_d, \tag{4}$$

where F_a is the force developed by the actuator, M_t is the total actuator mass, B_a is the actuator viscous damping coefficient, B_c is the load viscous damping coefficient and F_d is the dry friction force.

Some parameters are easily measurable or inferrable, as for example the piston area (A_p) or the actuator mass (M_a). Some other parameters can be neglected such as the valve internal leakage coefficient (C_{in}). However, others are much more complicated to be inferred and cannot be neglected, like the flowrate gain (K_{q0}). To solve these problems, a series of tests were performed in order to evaluate such parameters.

4 Parameter Identification

The tests, executed in order to identify all the necessary parameters, were done at the Laboratory for Pneumatic and Hydraulic Systems (LASHIP), located at the Mechanical Engineering Department of the Federal University of Santa Catarina (UFSC), following ISO 6404 (ISO, 1985). This norm determines the procedure to evaluate several parameters of continuous control valves, as servo and proportional valves. For more details about parameter physical identification of a hydraulic actuator with proportional valves see Andrigheto (1996).

A test bench was used to fix all the components; a PC computer with a real time data acquisition and a control board (dS 1102 – dSPACE Germany) was used perform the control algorithm; a Hydraulic Computer Controlled Power Unit (UPCH) provided oil to the system with controlled pressure, flow and temperature. The UPCH is basically composed of a 500 liter oil reservoir, two piston pumps, each is able to supply 210 bar at a 50 l/min flowrate, and a temperature control, which maintains the temperature within a one Celsius degree range. Furthermore, three pressure transducers were used: one for measuring the supply pressure and the other two for measuring the cylinder chambers pressure. One flow transducer was installed between the control ports of the proportional valve. Figure (3) shows the base with the cylinder, valve and pressure transducers locked on the test bench.

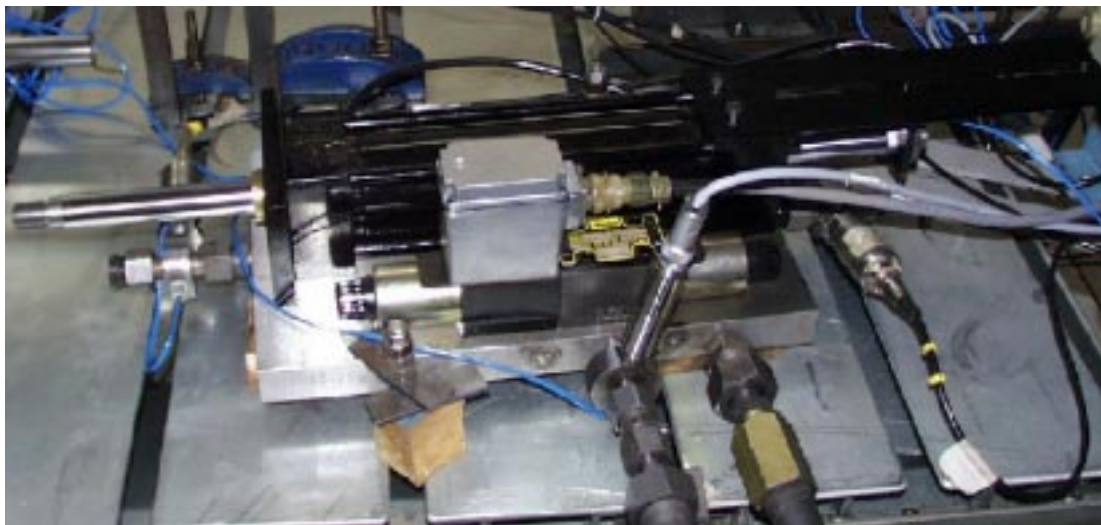


Figure 3 – Base interconnecting valve and cylinder at the test bench

The transducers used during the tests were all calibrated and certified, Tab. (1) shows their uncertainties, which placed them on the Error Class B, established by ISO 6404 (ISO, 1985). This Error Class admits errors up to $\pm 1.5\%$ of the final scale of the input signal, also $\pm 1.5\%$ for flow and pressure transducers and $\pm 1\text{ }^\circ\text{C}$ as the oil temperature variation.

Transducer	Uncertainties (% VFE)
500 bar Pressure Transducer	0.40
200 bar Pressure Transducer	0.20
350 bar Pressure Transducer	0.30
225 mm Position Transducer	0.06
100 l/min Flow Transducer	1.20

Table 1 – Transducer Uncertainties

The transducers' electric connections followed a basic pattern. All the transducers' cables were connected to the signal amplifier. The signals received by the amplifier (millivolts) were amplified to the desired range and then retransmitted to the data acquisition board. This was used for the pressure transducers as well as for the flow transducer. Figure 4 presents a diagram with the connections.

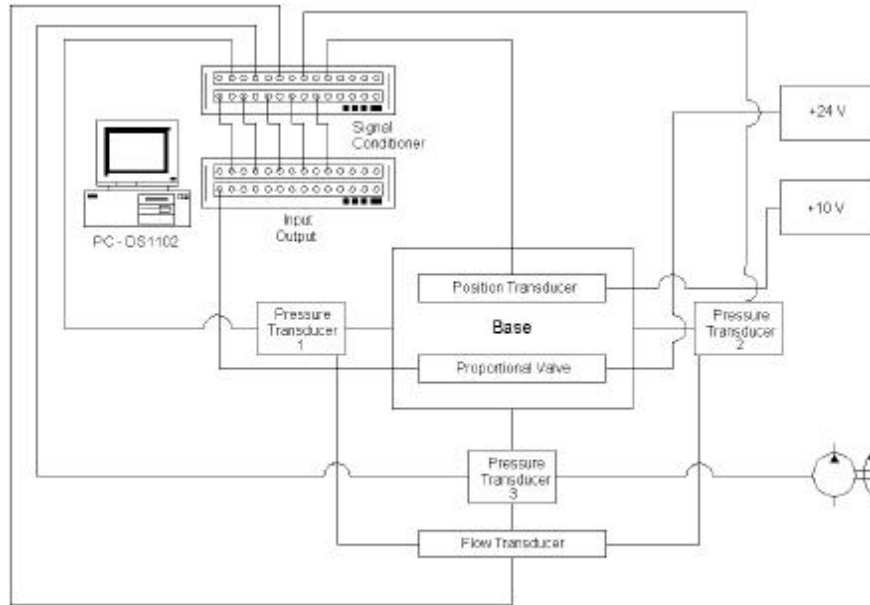


Figure 4 – Electrical Diagram

The presence of two 10 μm filters in the system guarantees the oil's filtration. Before installing the hydraulic components, such as the base with the valve and cylinder, a 30-hour flushing was done to clean all the pipes and hoses used for the components connections.

4.1 Pressure Gain

This test allows the Pressure Gain K_{p0} to be inferred. K_{p0} is necessary to calculate the flowrate-pressure coefficient K_{c0} . In order to undergo this test, the hydraulic cylinder was removed from the steel base and both its chambers were sealed. The cylinder wasn't used on the test because K_{p0} is a physical characteristic of the proportional valve itself. Figure (5) shows that test hydraulic circuit diagram.

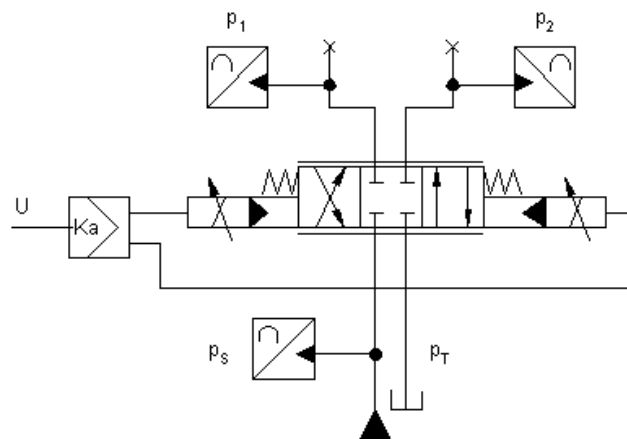


Figure 5 – Hydraulic Diagram to determine K_{p0}

Initially, must be verified if the proportional valve maintains equal pressure in both work lines (1 and 2) when a 0V signal is applied. In case this condition is not verified, then one can be assured that the valve spool is not centered. The problem is solved by applying an offset signal to the valve, in a way that the spool is at one position that provides equal pressure in the lines. All further applied signals must have this same offset.

For the pressure gain test, a variable signal is applied to the valve, making the spool move, creating a pressure differential between the lines. Three pressures were measured, supply pressure (P_s), chamber 1 pressure (P_1) and

chamber 2 pressure (P_2). The applied signal was a sine wave of $\pm 10V$ and 1Hz, with a $+0.8V$ offset. Figure (6) shows a graph with the input signal and the response output pressures in both chambers, as well as the supply pressure.

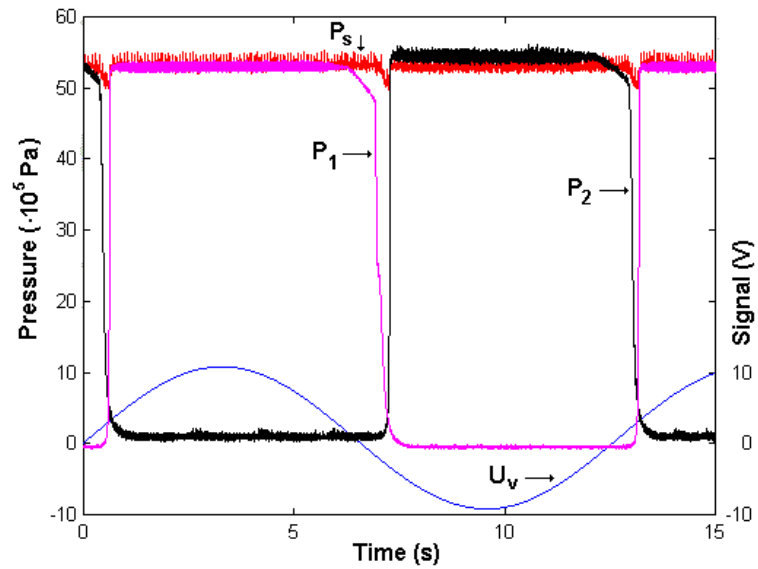


Figure 6 – Input voltage signal and the output pressure signals

The pressure gain in the chambers is analyzed at the beginning of the pressure curve. Using only one chamber as an example, Fig. (7). Points A and B are the point references to calculate the pressure gain.

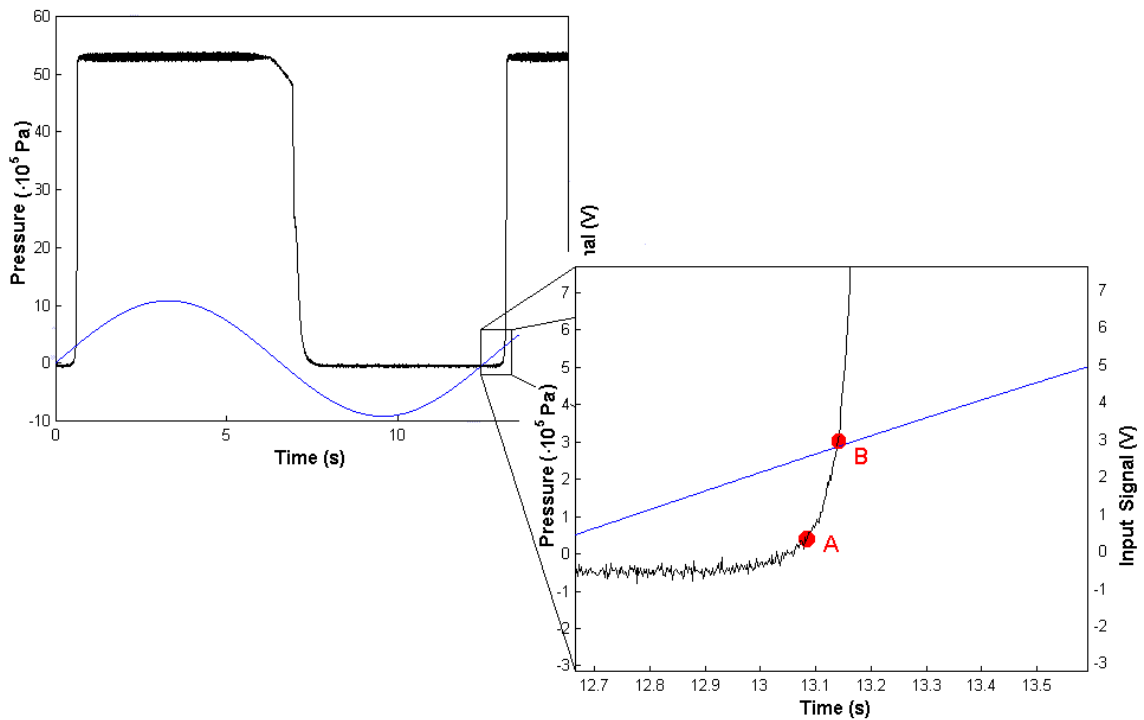


Figure 7 – Pressure Gain calculation

$$Kp_1 = \frac{P_{1B}(t_B) - P_{1A}(t_A)}{u(t_B) - u(t_A)} = 2.4328 \cdot 10^7 \text{ Pa} / \text{V} . \tag{5}$$

The same principle was used in chamber 2:

$$Kp_2 = \frac{P_{2B}(t_B) - P_{2A}(t_A)}{u(t_B) - u(t_A)} = 2.6964 \cdot 10^7 \text{ Pa} / \text{V} . \tag{6}$$

The Kp_0 adopted value was the average between Kp_1 and Kp_2 :

$$Kp_0 = \frac{Kp_1 + Kp_2}{2} = 2.5646 \cdot 10^7 \text{ Pa/V} \quad (7)$$

4.2 Dead Zone

The purpose of this test was to confirm the valve's dead zone information given by the catalogue because discrepancies are a normal fact.

The mechanical configuration was changed for this test. The hydraulic cylinder was still kept out because the dead zone is a characteristic of the valve. The two ports were interconnected, and the flow transducer was placed in between. A very slow sine wave of $\pm 10\text{V}$ and 0.004Hz was applied to the valve, so that the movable parts of the valve had not had a sudden position change. To find the dead zone value, one must observe the input signal and the flowrate transducer output signal. The values obtained for the dead zone test were -3.5 V and $+3.3 \text{ V}$. Those values are considered high and out of the catalogue specifications ($\pm 1.0\text{V}$).

4.3 Flow Gain

Using the same configuration of the dead zone test, a test to evaluate Kq_0 was executed. This parameter was obtained by taking into account the relation between the flowrate measured by the flowrate transducer and the input signal "u". Using the flowrate plot and obtaining the flowrate gain at two different points, Kq_0 was calculated as:

$$Kq_0 = \frac{Kq_1 + Kq_2}{2} = 6.489 \cdot 10^{-5} \text{ m}^3 / \text{s} \cdot \text{V} \cdot \quad (8)$$

After obtaining the pressure and flowrate gains, the pressure-flow gain (Kc_0) is given by:

$$Kc_0 = \frac{Kq_0}{Kp_0} = 2.5302 \cdot 10^{-12} \text{ m}^3 / \text{s} \cdot \text{Pa} \cdot \quad (9)$$

4.4 Friction Test

To obtain the friction parameters, the hydraulic cylinder was again assembled on the base and the flow transducer was removed, while the pressure transducers were kept. The position transducer was installed and connected to the cylinder. Beyond that, it was connected to the data acquisition board as well. Figure (8) shows the configuration for this test.

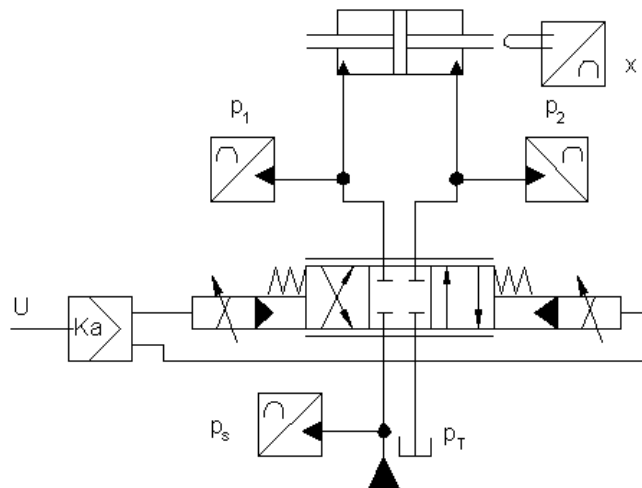


Figure 8 - Hydraulic Diagram to determine the cylinder's friction

The dry friction force can be represented as:

$$F_d = \text{sgn}(\dot{y}) \cdot \left[F_C + (F_S - F_C) \cdot e^{-\frac{|\dot{y}|}{v_S}} \right] \text{ for } \dot{y} \neq 0, \quad (10)$$

where F_S is the static friction force, F_C is the Coulomb friction force and v_S is the Stribeck velocity.

To obtain the static friction, considering the piston at rest, a crescent input signal was applied, and when the piston started to move, the pressure difference was capture. The product of the pressure difference by the piston area was the static friction force. This test was done in both directions and the average $F_S = 636 \text{ N}$ was used in the simulation model.

To obtain the Coulomb friction force the piston was moved at a constant speed. When the piston was at the halfway point, the pressures on both chambers were measured, and the Coulomb friction force was obtained. Following this procedure at several different speeds, a dynamic friction curve was drawn. Combining static, Coulomb and viscous friction forces, an approximate friction curve was obtained. Figure (9) shows the experimental curve and simulated curve for the friction force.

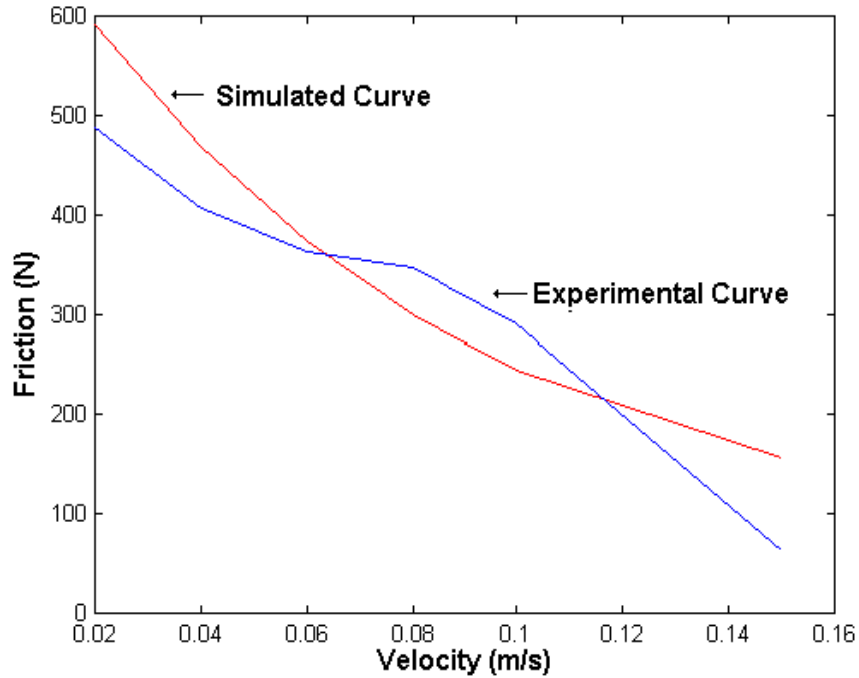


Figure 9 – Friction Curves

4.5 Proportional Valve’s Dynamic Characteristic

The valve constant was obtained by applying a step input signal and measuring the valve flowrate as shown in Fig. (10). The response was approximated by a first order system with a time constant equal to 0.0278s.

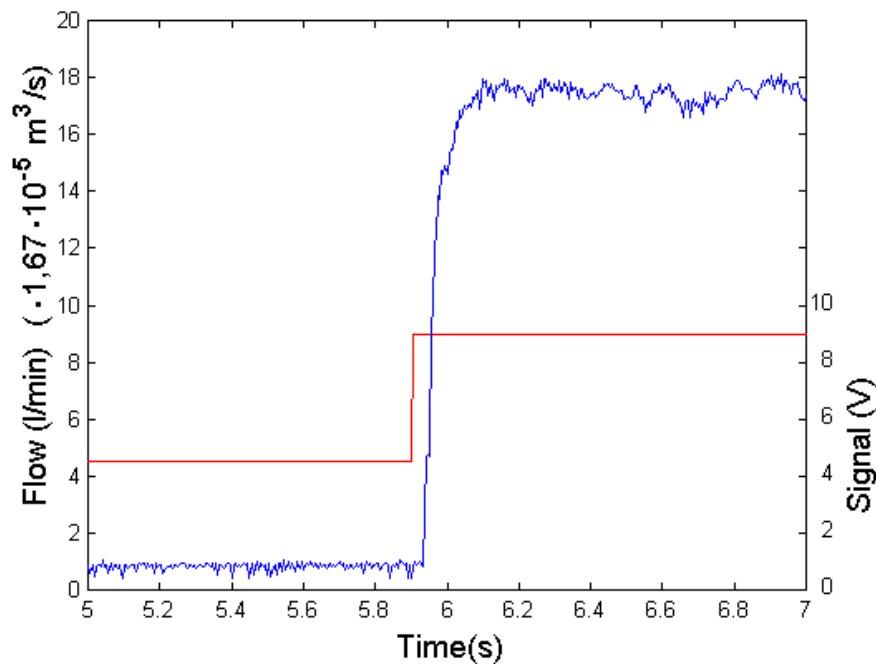


Figure 10 – Valve response to a step input signal.

5 Simulation

Using the identified parameters and the equations presented in Section 3, simulations were performed and compared with the experimental results in order to validate the model. Selecting a PID controller, one of the most used in industries today, allowed the use of some known techniques to adjust its parameters. The Ziegler-Nichols technique (Franklin et al., 1994), which uses a step signal at the system in open loop, was chosen. When evaluating the system response, the initial P, I and D values could be calculated. Trial adjustments could then be done in order to find the best system response to any input signal, using this PID controller, always focusing on the maximum position error granted, in this case 0.2 mm.

A trajectory composed of 3^d degree equations, was selected to compare the model obtained and the real actuator response to the comand signal. Taking the P,I and D values obtained from the Ziegler-Nichols technique as a initial point, simulations with the system in close loop were performed. Some adjustments in the controller parameters had to be done, and a PI controller was found as being the best performance controller. All the tests so far were performed without load. Figure (11) brings the comparison between desired trajectory, response trajectory with the PI controller with both the real actuator and the simulated model of it. One can note the acuity of the model, which represents the real system very well.

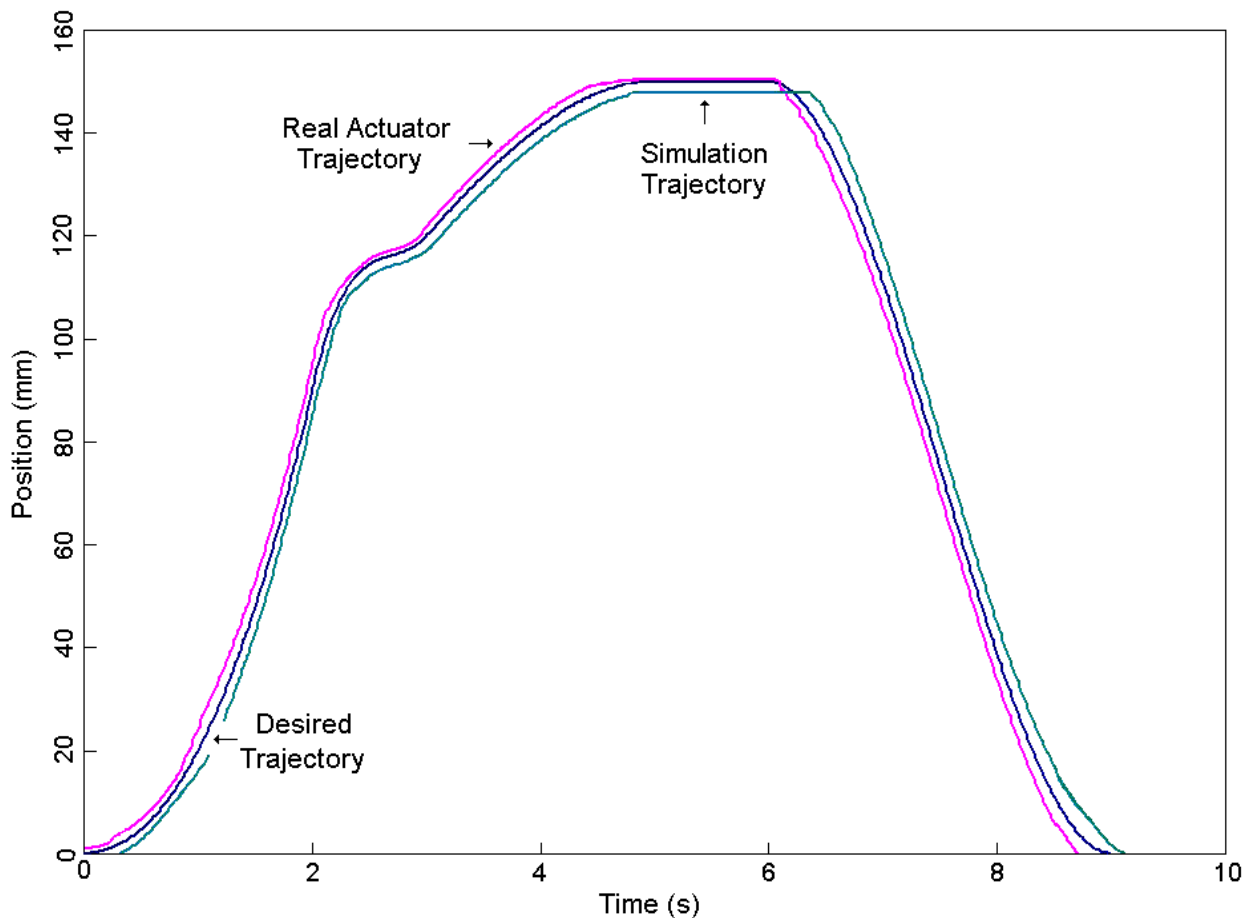


Figure 11 – Trajectory comparison

6 System Implementation

In order to mill the injector within the required tolerance, the start and endpoints of the milling had to be set. To have better control over the milling, a trajectory was created, so it would be followed by the tool. It consists basically of three parts: the tool aproximation to the injector, the milling at controled speed and the tool's return.

This trajectory will make the tool aproximate the injector quickly, then it will decrease its velocity to the milling speed, keeping it constant until the last 10 mm, when it will start to follow a 2^d order equation. This will gradually reduce its speed, which will be zero at the end of the milling process. Then the tool will retreat, slowly at first, while still inside the injector, then it will accelerate and, near the original starting point its speed will again decrease to zero. Figure (12) presents the trajectory used.

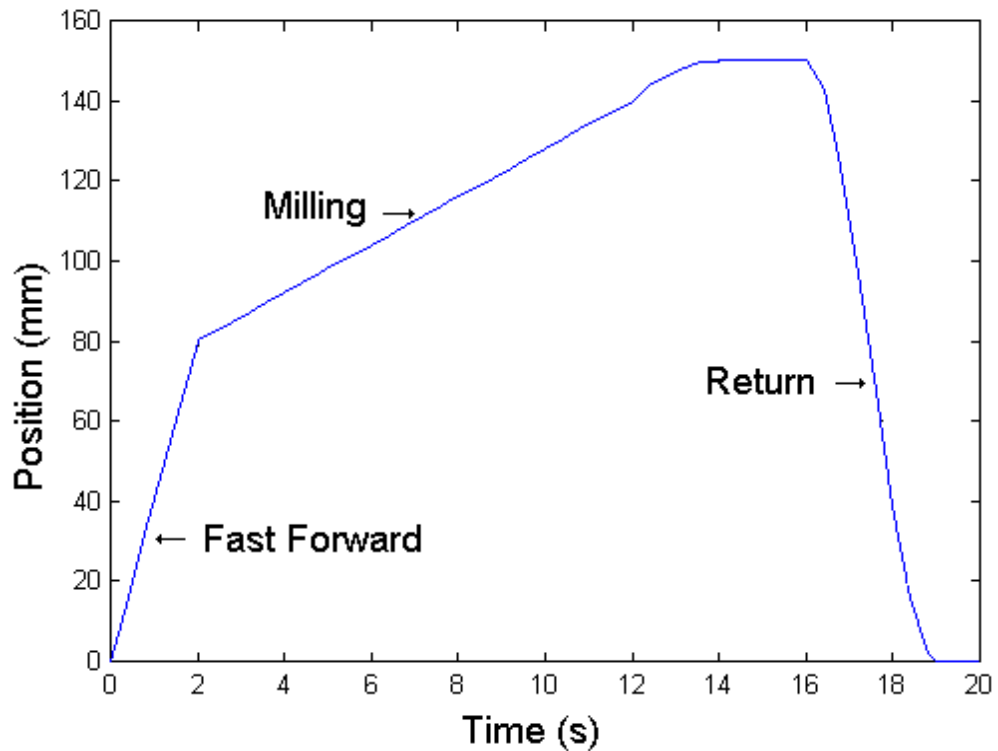


Figure 12 – Tool path

Some practical milling tests were done with real injectors, taking the controller developed in the simulation as a start. The fine-tuning of the proportional and integrative gain was done.

The large dead-zone and the static friction caused a substantial error in the desired final position. To decrease this error, a dead-zone compensation, based on the dead-zone inverse, was implemented. This dead-zone compensation basically consists of adding a positive offset to the control signal when it is positive, or a negative offset when the control signal is negative with a small smoothness around origin. When the real dead-zone value is overcompensated, this kind of compensation is also compensating dry friction as explained in Cunha (2001).

This compensation significantly decreased the effects of the dead zone on the final positioning. After several adjustments, where a PID was also tested, a PI controller proved to be the one with better results. The value of the proportional part was 1.7 while the integrative part was 0.07. Figure (13) shows the results using the PI controller. One can observe that the final position error was half of the limit imposed by the company.

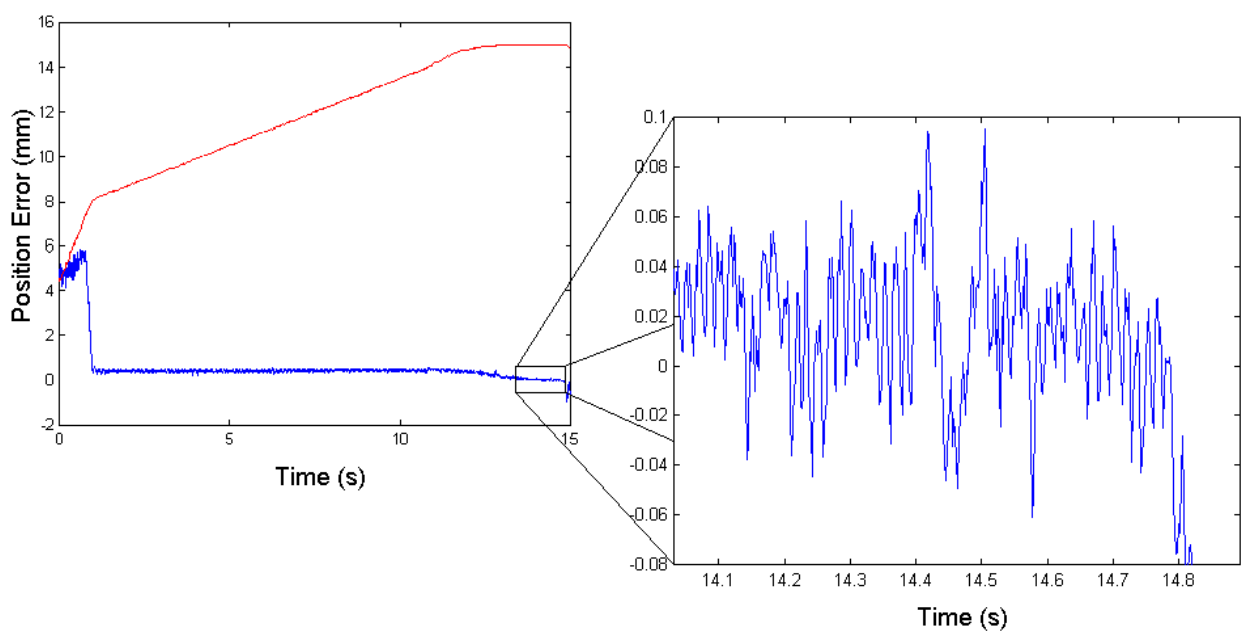


Figure 13 – Position Error

7 Conclusions

In this paper, the project and the construction of a milling module for the tooling of components such as injectors, used for fluid pumping, was addressed. The major attention was given to the identification, modeling and control of the hydraulic positioning system.

According to the obtained results, one can perceive that using a correct model with reliable values to all parameters, and with the application of fundamental control theory concepts, the construction of proportional hydraulic systems, with response characteristics compatible to the necessities of many industrial applications is possible using regular components, without high sophistication. However, it is important to outline that the use of classical controller in high performance closed-loop applications, as do robot systems, is limited.

The hydraulic actuator nonlinearities, such as dead zone and friction, showed to produce large final positioning errors. The use of a dead-zone inverse could effectively reduce such errors. This lack of knowledge, knowing the system as one, is a common fact on industrial equipment construction, taking a considerably large startup time on machines and processes. This work showed that the creation process can be quite simple and time can easily be saved, if some procedures are followed.

8 References

- Alleyne, A., 1996, "Nonlinear Force Control of an Electro-Hydraulic Actuator", Proceedings of Japan/USA Symposium on Flexible Automation, Boston, vol.1, pp. 193-200.
- Andrigheto, P.L., 1996, "Electro hydraulic Positioning System Controlled by a Directional Proportional Valve", In Portuguese, Master Thesis, Federal University of Santa Catarina, 170p.
- Cunha, M.A.B., 2001, "Cascade Control of a Hydraulic Actuator: Theoretical and Experimental Contributions", In Portuguese, Ph.D. Thesis, Federal University of Santa Catarina, 177p.
- De Negri, V. J., 1988, "Study of the Hydraulic Systems Dynamic Behavior", In Portuguese, Internal Report, Federal University of Santa Catarina.
- Franklin, G.F.; Powell, J.D.; Emami-Naeini, A. "Feedback Control of Dynamic Systems". 3ed., Addison Wesley, 1994.
- ISO, International Organization for Standardization. ISO 6404 - Fluid Power Servovalves – Test Methods. Switzerland, 1985. 23 p.
- Schneider, F.B., 2000, "Position Control of a Milling Module using Proportional Hydraulic", In Portuguese, Course Final Project, Automation and System Department, Federal University of Santa Catarina.
- Stringer, J.D., 1976, "Hydraulic Systems Analysis", 1 ed., The Macmillan Press Ltd..
- Yao, B.; Bu, F.; Chiu, G.T.C., 1998, "Nonlinear Adaptive Robust Control of Electro-hydraulic Servo Systems with Discontinuous Projections", *Proceedings of 37th IEEE Conference on Decision & Control*, Florida, pp. 2265-2270.

A GLOBAL VISION SYSTEM FOR MOBILE MINI-ROBOTS

Kelson Rômulo Teixeira Aires

Universidade Federal do Rio Grande do Norte, LECA – CT – UFRN, Lagoa Nova, Natal – RN, Cep.: 59072-970
kelson@engineer.com

Pablo Javier Alsina

Universidade Federal do Rio Grande do Norte, LECA – CT – UFRN, Lagoa Nova, Natal – RN, Cep.: 59072-970
pablo@leca.ufrn.br

Adelardo Adelino Dantas de Medeiros

Universidade Federal do Rio Grande do Norte, LECA – CT – UFRN, Lagoa Nova, Natal – RN, Cep.: 59072-970
adelardo@leca.ufrn.br

Abstract. *The navigation based on visual feedback for a robot team working in a closed environment can be attained through an onboard camera in each robot. A global vision system is a cheapest solution for this problem. This paper presents the implementation and experimental results of a global vision system for a mobile robot team. The proposed experimental system consists of a top camera, a frame grabber, a PC for image processing, and a team of six mobile robots. The PC is responsible for the team motion control. In order to the system be able to univocally recognize each robot, each one has two circular labels on its top. The arithmetic mean of their centroid coordinates gives the actual position of the robot. The vector lying both centroids is utilized to obtain the robot orientation. A great problem of the proposed system was the classification of each pixel color of the robot labels images in real time and under time-variant illumination conditions. To overcome this problem, automatic camera calibration software, based on clustering K-means algorithm, was implemented. This method guarantees that similar pixels will be clustered around a unique color class. The obtained experimental results show that the implemented system updates the robot with acceptable accuracy (position error less than 0.5 cm). The developed system presents a great robustness under large illumination changes.*

Keywords: *robot vision, image processing, k-means.*

1. Introduction

With the advent of the Artificial Intelligence, researchers began to think about the possibility of endowing machines with the capacity to take more complex decisions, so as to adapt the machines to unexpected variations in their environment. It was the first step for the development of Autonomous Robot Systems. An important property of these systems is the fact that they can interact, in an intelligent way, with their work space, feeling what happens around them and acting in their environment in agreement with the felt information. For this, several sensors can be installed in the robot for the acquisition of every kind of information data.

In order to increase the robot autonomy, different sensorial systems were utilized in practice. A Vision System can be considered as another sensor, between the several ones that can be coupled to a robot system, that allow it to sense the world around. Digital image processing is a vast field, with several applications in the science and engineering. Image Processing assures the possibility of development of last generation machines, capable of executing several functions of the human vision (Jain, 1989). Vision Systems are capable of supplying a large amount of useful information about the robot workspace and computer vision techniques are utilized to extract, in the most possible efficient way, a great amount of useful information from the sensed images.

In cooperative robotic systems, the information about robot position and orientation is indispensable. In order to obtain this information, a vision system can be used. The vision system should be able to sense the dynamic environment where the robot team navigates and to supply the necessary information for the robot control system. An efficient and robust vision system should be capable to act in different environments, with time-variant illumination and color conditions, maintaining, at the same time, processing speed and accuracy.

This paper describes the implementation details of a global vision utilized for the navigation of a mobile robot team. The proposed experimental system consists of a top camera covering all the robots workspace, a frame grabber, a host PC for image processing, and a robot team, whose images are captured by the camera. The PC is responsible for the robot team motion control based on visual feedback, sending commands to them through a radio link. In order to each robot be univocally recognized by the vision system, each one has a label on its top, consisting of two colored circles. One of the most complex tasks on the system implementation was to classify, in real time, each image pixel color of the robot labels under time-variant illumination conditions. In order to overcome this problem, automatic camera calibration software, based on clustering K-means algorithm (Alsabti et al, 1997), was implemented. This method guarantees that similar pixels will be clustered around unique color class. In this paper, the general architecture of the proposed vision system is described in Section 2. Section 3 presents several issues related to the choice of the adequate color model to be adopted for image processing in the proposed vision system. The implemented calibration method is detailed in Section 4. The adopted robot localization and identification algorithms are described in Section 5. Section 6 presents a discussion about the experimental results obtained from the implemented vision system. Concluding remarks are presented in Section 7.

2. Vision system architecture

The proposed vision system can be divided in three modules, as shown in the Fig. (1). The acquisition module obtains the image of the workspace through a CCD camera and a frame grabber. The Image Processing module computes, from the acquired image, the position and orientation of each robot. As results of the data processing, position, orientation and identification of the objects in the workspace are supplied for robot motion control purposes. In order to adequate the vision system to local illumination conditions, some color calibration is need before any image processing. The calibration of the system can be made off-line or, when necessary, in real-time. The following subsections detail the modules of the proposed system.

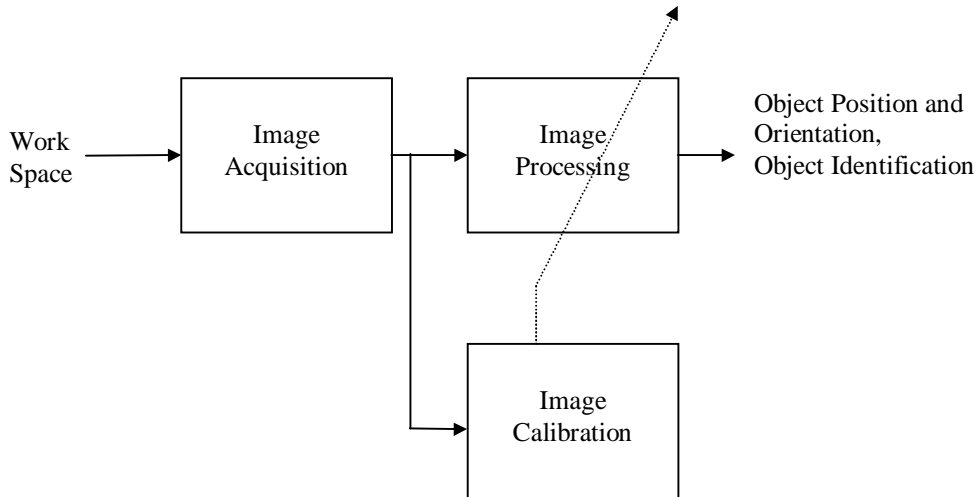


Figure 1. Vision system architecture.

2.1. Image acquisition

There are two primordial elements in a digital image acquisition system (Gonzales and Woods, 2000). The first is a physical device sensitive to a band of the electromagnetic spectrum and that produces an electric signal proportional to the energy level of the sensed radiation. The second element, called frame grabber, is a device for conversion of the electric signal into a digital value. In this implementation, it was utilized a CCD (Charged Coupled Device) colored camera located 2 m above the workspace, and a frame grabber, which furnishes the acquired digital image to a PC (Personal Computer), which is responsible by the rest of the image processing. The frame grabber supplies data in three matrixes of color components, where each of their elements is associated to the correspondent color component of each image pixel. The standard color models furnished by the utilized frame grabber are RGB and HSL.

2.2. Color calibration

The calibration stage frequently is performed off-line, previously to the image processing. Its goal is the tuning of the image processing module, in order to optimize the object detection. After the choice of an appropriate color model, thresholds for classification of the image colored objects are defined for each of their color components. The calibration process can be made manually, by the operator, or in an automatic way, through a calibration algorithm. The calibration consists, basically, in the adjustment of these thresholds, sometimes including other processing parameters used in the vision system, such as: brightness, saturation, contrast and image resolution. These data are supplied as initialization parameters for the Image Processing Module.

2.3. Image processing

After a good calibration, the acquired image is ready for processing. The localization and identification routines should be performed. The localization algorithm supplies the position and the orientation of all robots in scene. The identification process consists on determinate who is each object located in the image, i.e., the identification algorithm should recognize the detected objects, associating them to well-known models. The used algorithm should act in a fast and efficient way, analyzing the largest possible amount of pictures per second.

3. The choice of the color model

3.1. Color models

The choice of an appropriate color model to codify the image information is the first step in the design of a vision system. A color model determines a subspace of a three-dimensional coordinate system, where each point represents a

color. The main objective of a Color models is to facilitate the representation of the colors in some acceptable standard form. They can be classified in two types: hardware guided models (RGB, YUV and CMY) and users guided models (HSV, HSL and HVC), (Foley et al, 1990).

The frame grabber utilized in the implemented system furnishes both RGB and HSL video output. It is possible to use another color model through an additional conversion process, which can be easily implemented by software, although with a larger computational cost.

3.2 RGB color model

RGB is the most used color model. It is based on a cartesian coordinate system, where each color is represented by its primary components: **Red**, **Green** and **Blue**. The mixture of these three primary colors with full intensity results in the white color and its absence results in the black color. Any possible color is composed as a mixture of specific amount of red, green and blue. RGB color model can be represented, with normalized values in the interval $[0,1]$, in agreement with the subspace of a three-dimensional coordinate system, as we can see in the Fig. (2).

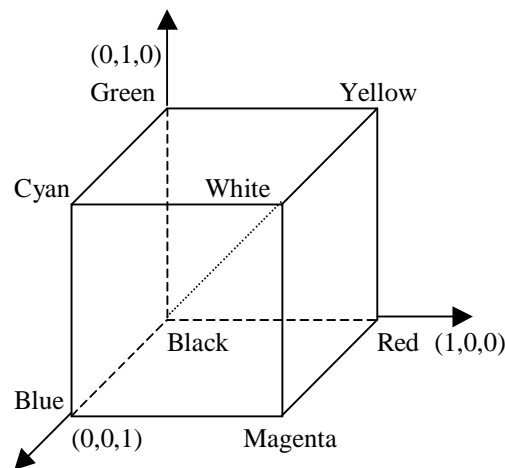


Figure 2. RGB color cube.

The origin of the coordinate system corresponds to the black color, being the white represented by the vertex $(1,1,1)$ of the color cube. In this cube, the gray scale varies from black to white, along the straight line that joins these two vertexes. The primary colors (red, green and blue) are represented by the vertexes of the cube on the axes of the coordinates system. The remaining vertexes represent the complementary colors (yellow, cyan and magenta) (Foley et al, 1990).

3.3. HSL color model

HSL (Hue, Saturation, and Lightness) color model is based on intuitive color parameters, being derived from the RGB color cube. It is represented by a double hexagonal pyramid as can be seeing in Fig. (3) (Hearn et al, 1994). Hue (H) specifies an angle about the vertical axis of the pyramid, varying from 0° , that corresponds to the red, to 360° . The hexagon vertexes correspond to the primary colors and its complementary ones. Each vertex of complementary color is shifted 180° from its respective primary color vertex. The parameter H possesses indefinite value for the gray scale which varies from black to white. Saturation (S) is measured along the horizontal radius of the pyramid and specifies the relative purity of the color. This parameter varies from 0 (gray scale) to 1 (pure colors). Lightness (L), measured along the vertical axis, possesses value 0 for black and 1 for white. It specifies the amount of light in the color.

3.4. Color model choice

The choice of the color model to be utilized in image processing is very important. In order to localize and to identify colored objects in an image, a component or a relationship among components of the color model must be chosen to allow the best possible differentiation among the colors of the different elements in the image. This allows the definition of specific thresholds that will be utilized for the classification of the colored objects in the scene. An adequate choice of these thresholds is fundamental to guarantee the robustness of the vision system in the presence of light changes.

Besides the robustness to illumination variations, a very important factor to be considered in the project of the vision system is the execution time of the image processing routines. In this context, the chosen color model must supply the largest amount of color information, in the smallest amount of data, in order to turn more efficient the process of classification of the colors of each object. HSL model presents a great separation between color (component

H) and lightness (component L) characteristics. Therefore, light variations in the image interfere directly in the component L, having little influence in the values of H (Gonzales and Woods, 2000). As the separation of color characteristics and luminance is not complete, a safety margin must be considered when working with HSL model. A disadvantage of the HSL model is that the parameter H is undefined in the gray scale. This is the case in the present implementation, where the robot team navigates on a black floor. This is the reason because the RGB model was chosen for our vision system. On the other hand, RGB model necessarily uses the three components to define the colors and the amount of light in the image, which implies that, when using the RGB model, all the three components, or a relationship among them, must be utilized, increasing the computational effort.

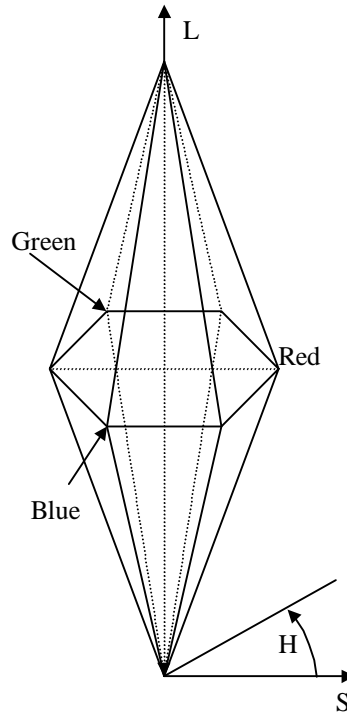


Figure 3. HSL double hexagonal pyramid.

4. Automatic calibration method

Color calibration one of the most important stages of the vision system. The main objective of the calibration is to facilitate the identification and the classification, with respect to the color, of all image pixels (MIN98). The calibration stage consists, basically, in an analysis of the data of an image model under certain light conditions in order to define the optimal parameters utilized by the vision system for pixel classification. This stage is very important, because the system must be calibrated in agreement with the available illumination. This work proposes a form of calibrate the vision system automatically based on the K-means clustering algorithm.

4.1. K-means clustering algorithm

Several clustering techniques have frequently been recommended as tools for pattern classification. Clustering is a partitioning process of a set of patterns in separated groups (clusters), or classes. Patterns with similar characteristics are classified in the same group, patterns with different characteristics are classified in disjoint groups (Alsabti et al, 1997). The final purpose of all clustering techniques is to find natural groupings through the database analysis. The most popular clustering algorithm is probably the K-means algorithm.

K-means algorithm produces a partition of n objects in K clusters, generally optimizing an objective function. Some advantages exist: the possibility of changes on the pertinence of objects in relation to a group during the cluster formation process and the possibility of working with great database. This method requests computation time of order $O(n)$, where n is the number of objects in the database (Costa, 1999). K-means algorithm is sensitive to the choice of initial classes. The number of classes K must be known *a priori*, and different solutions could be generated depending on the choice of number K . This is a great disadvantage of K-means. The problem of a wrong choice of K is that the method will impose an arbitrary structure to the data instead of looking for an inherent data structure (Alsabti et al, 1997). Each class C_i is represented by a point (center) w_i and, initially, that class is constituted only by this point. After having chosen the centers, each data is marked as belonging to the class whose center is closer to it. After processing all data according this method, the mean of each class is calculated and used as its new center. Then, the data is processed again. This process is repeated until no changes occur in the found centers or a user defined number of iterations is attained. The whole process is described in the Algorithm I.

Algorithm I

1. Initialize the K centers of the classes (w_1, \dots, w_k)
2. Associate each class C_i to a center w_i
3. Include each input data p_j to the class C_i whose center is closer to p_j .
4. For each class C_i , actualize its center w_i as the arithmetic mean of all input data $p_j \in C_i$.
5. Compute de error function:

$$E = \sum_{i=1}^k \sum_{p_j \in C_i} (p_j - w_i)^2$$

6. Repeat steps 3, 4 and 5 until error E do not change significantly.

4.2. System calibration

The tuning of some image parameters, (such as brightness, contrast, saturation and resolution), was necessary before color calibration. In The implemented system, this process was manually executed, through the software that accompanies the frame grabber. Another important parameter is the height of the camera above the robot team, which was previously adjusted, because it directly influences the values of the other mentioned parameters.

Color calibration was made using K -means algorithm. In order to classify each color as belonging to a center, a number of classes K was chosen larger than two times the number the main colors present in the scene, being the extra centers responsible by the classification of image spurious pixels. This was made in order to avoid the imposition of an erroneous data structure, containing spurious pixels in a same class of object color pixels. Initially, 20 centers were chosen, in a random way, among the group of pixels that forms the image to be processed.

After the choice of the centers, the pixel cluster processing was initiated. Due to real time constrains, the established stop condition was a program iteration number equal to 100. This criterion guaranteed a reasonable error in the pixel classification process, small enough for this application. At the end of the 100 iterations, 20 centers were selected representing 20 pixels groupings, where each object color class is represented by a center, and the remaining centers represent undesirable pixels. The choice of the correct groupings that classify, among the 20 final groupings, the desired colors, was realized taking into account the number of pixels belonging to each grouping. For this, a range of values can be established analyzing image resolution, label size and amount of each color label in the image. In order to relate each class with its respective color, its normalized component values were utilized. Each image pixel was classified as belonging to the color class with the nearest center. After system calibration, all colored objects in the image can be classified with respect to the color, with good precision, allowing a more efficient execution of the localization and identification routine.

5. Localization and identification**5.1. Labels**

Labels were attached on the top of each robot to permit its identification and the computation of its position and orientation. There are several possible robot labeling methods. The chosen labeling method is shown in the Fig. (4). Two circular labels were placed in the diagonal line of the robot top. This label consists of a main blue or yellow circle and a secondary green, pink or cyan circle, both placed on a black background. In this way, up to six robots can be univocally labeled: blue-green, blue-pink, blue-cyan, yellow-green, yellow-pink and yellow-cyan. A sufficiently large black background margin is provided to separate labels belonging to different robots when two or more are in collision. This avoids that a pair of circles, each one belonging to a different robot in a collision, could be identified as belonging to a same robot. Thus this strategy avoids a false robot's identification.

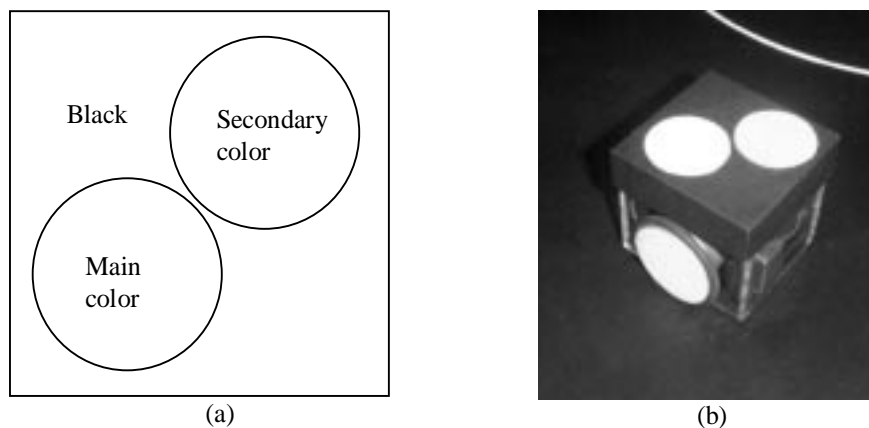


Figure 4. Robot label. a) Robot labeling method. b) Image of a real robot.

5.2. The localization and identification algorithm

The adopted algorithm for robot localization and identification is a modification of the algorithm proposed by Bianchi et al (2000). The image pixels are analyzed with sampling step smaller than the half of the smallest object diameter (in pixel) in the scene. When a main color pixel (blue or yellow) is found, a centroid calculation algorithm of circular objects is applied.

The centroid calculation algorithm is quite simple. After being found a pixel P_{ini} with a desired color, a diagonal segment is traced until finding the border points P_1 and P_2 of the colored circle. The center P_c of the previous segment is calculated and, starting from it, a perpendicular segment is traced until reaching the border points P_3 and P_4 . The second segment center, $P_{central}$, corresponds to the centroid of the colored circular label, as depicted in the Fig. (5).

After finding a main color circle, the secondary color circle is searched around the first, at a distance equal to the circle diameter, analyzing the neighborhood using a sampling step smaller than the half of center diameter. When a secondary color pixel is found, the same centroid calculation algorithm is used. The arithmetic average of the circle centroids coordinates supplies the robot position, as depicted in Fig. (6). The segment that links the main circle centroid to the secondary circle centroid is utilized to obtain the robot's orientation.

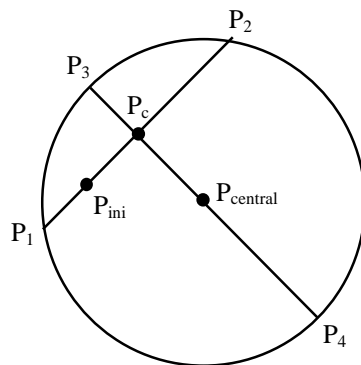


Figure 5. Centroid calculation algorithm of circular objects.

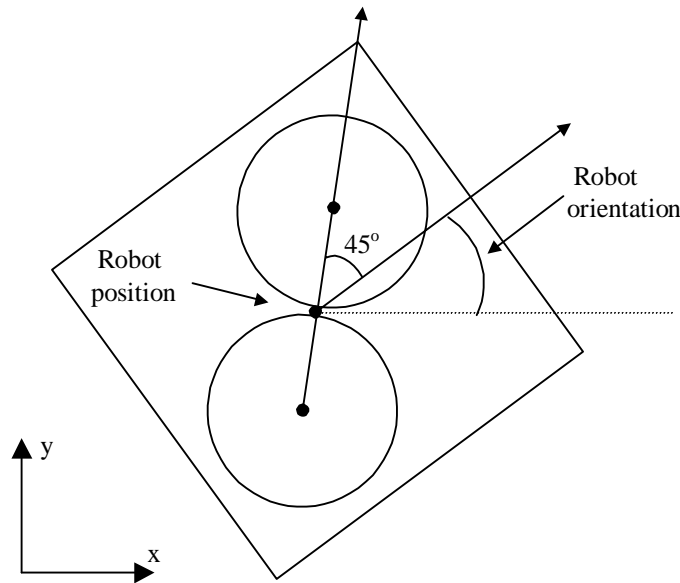


Figure 6. Robot position and orientation.

6. Experimental results

In order to validate the proposed vision system, some tests were performed with the calibration and localization modules. Color calibration was realized in RGB and HSL robots' images, under several illumination conditions. Twenty initial centers were randomly chosen and the algorithm stop condition established as 100 iterations. In all tested situations, the system was calibrated in less than 2 minutes for definition of the 20 groups and less than 1 minute for correct choice of the groups that classify the colors in use (blue, yellow, cyan, rose and green).

Calibration based on the HSL color model was very difficult, due to white and black colors of the floor, which are predominant in the image, resulting in undefined values for the H parameter. K-means algorithm imposes an erroneous structure to HSL pixels. The RGB color model was shown more efficient for this kind of calibration. All used colors were correctly clustered and the RGB color model was chosen as the more adapted for this calibration application.

In order to verify the precision of the proposed localization system, 100 stopped robot images were acquired and its position and orientation was computed by the implemented algorithm. The computed results were compared with the more precise values, obtained by direct measure, (by means of a millimetric ruler), of position and orientation. These results are shown in the Tab. (1), where SD means standard deviation and ME is the absolute mean error. Robot position was obtained with precision less than 2 mm and robot's orientation with precision less than 4°.

Another experiment was realized in order to verify the system robustness in the presence of illumination variations and possible optical distortions of the camera along the workspace. A robot was placed in different locations of the workspace and its position and orientation was computed by means of the vision system. The results for this test are shown in Tab. (2). The results were similar for all the tested positions, proving the robustness of the proposed vision system to small illumination changes due to irregular spatial lights distribution.

A third experiment was realized to verify the robustness of the proposed vision system under severe changes in the illumination. The vision system was calibrated under 50% of the standard illumination. After this, the vision system was operated under standard illumination (1000 lux). The results obtained from this test are shown in Tab. (3). According to these results, the measured position and orientation errors don't differ significantly of those of the other experiments, showing that the proposed system is robust to great illumination changes.

Table 1. Experimental results for test 1 – Measure of position and orientation error.

Robot	Position (x,y) (cm)				Orientation (°)			
	Real	Mean	SD	ME	Real	Mean	SD	ME
Yellow-cyan	(12,-1)	(11.98,-1.13)	0.13	0.17	145	143.66	2.39	2.47
Yellow-green	(-19.3,32.5)	(-19.26,32.38)	0.16	0.19	-125	-125.28	2.89	2.52
Blue-pink	(-30,25.5)	(-29.94,25.34)	0.12	0.20	160	156.97	2.43	3.46

Table 2. Experimental results for test 2 (blue-cyan robot) – Robustness to spatial illumination changes.

Workspace position	Position (x,y) (cm)				Orientation (°)			
	Real	Mean	SD	ME	Real	Mean	SD	ME
1 st quadrant	(21,20)	(20.83,20.32)	0.18	0.37	-165	-163.79	4.08	2.78
2 nd quadrant	(-50,45)	(-50.58,45.94)	0.26	0.48	125	120.53	4.84	5.27
3 rd quadrant	(-35,-35)	(-35.99,-34.65)	0.14	0.32	-30	-28.44	4.17	3.58
4 th quadrant	(21,-18)	(21.93,-18.16)	0.14	0.25	-105	-106.97	1.44	2.12
Center	(0,0)	(0.08,0.16)	0.13	0.20	0	1.38	2.80	2.30

Table 3. Experimental results for test 3 (yellow-pink robot) – Robustness to 100% of illumination changes.

real	Position (x,y) (cm)			Orientation (°)			
	Mean	SD	ME	Real	mean	SD	ME
(24,22)	(24.78,21.98)	0.15	0.38	127	125.98	2.58	2.33

In all performed tests, the position was obtained with precision always less than 5mm. The error in the computed orientation was always less than 5 degrees in all experiments. Both robot's position and orientation were updated in real time, analyzing 30 frames per second. In all cases, the localization routine computed the position and orientation, and identified all robots, approximately in 14 milliseconds, which is smaller than the image acquisition time interval of the frame grabber (33 milliseconds). This leaves enough computation time for the additional robot control routines.

7. Conclusions

Analyzing the experimental results of the proposed global vision system, we can present some features of our vision system:

- *Easy and fast off-line calibration.* The system was quickly calibrated in less than three minutes.
- *Fast processing.* Our system determinate position and orientation all robots in real time, analyzing 30 images per second.
- *Reliability.* The proposed vision system can determinate the robot position and orientation without previous frame information. If any robot position can not be determined in the current frame, this will be possible in the next frame but, until now, in all tests, all robot positions and orientations were found successfully.
- *Robustness.* The vision system was robust to the time-varying illumination. All colors were separated and clustered, also under severe illumination changes.
- *Precision.* The robot's localization was calculated with precision less than 5mm for position and 5° for orientation.

One problem faced in K-means algorithm is the choice of the initial classes. Depending oh this choice, the final values can converge to a local minimum. In spite of this, in our implementation, this kind of problem did not occur.

The proposed calibration technique can be easily adapted to other robot vision applications. The proposed calibration technique can be utilized in other applications involving classification of pixels according its color. At the moment, the use of this calibration system in visual servo controller for robot manipulator is under study.

8. References

- Alsabti, K., Ranka, S., and Singh, V., 1997, "An Efficient K-Menas Clustering Algorithm", disponible at <http://www.cise.ufl.edu/~ranka>.
- Bianchi, R. A. C. and Costa, A. H. R., 2000, "O Sistema de Visão Computacional do Time FUTEPOLI de Futebol de Robôs", Proceedings of XIII Congresso Brasileiro de Automática, Florianópolis, Brazil, pp 2156-2161.
- Campos, M., Anício, M., Carvalho, M., Dias, R., Hartman, A., Nagem, D., Oliveira, V., Oliveira, E., Pereira, G., Ribeiro, A., Sanches, F., and Silveira, M. A., 1998, "MIneiROSOT – The Development of a Centralized Control Set of Soccer-playing Micro-robots", Proceedings of FIRA Robot World Cup, Paris, França, pp 57-62.
- Costa, J. A. F., 1999, "Classificação Automática e Análise de Dados por Redes Neurais Auto-Organizáveis", PhD Thesis, University of Campinas – UNICAMP.
- Foley, Van Dam, Feiver and Hughes, 1990, "Computer Graphics: Principles and Praticce", Addison-Wesley.
- Gonzales, R. C. and Woods, R. E., 2000, "Processamento de Imagens Digitais", Editora Edgard Blücher LTDA.
- Hearn, Donald and Baker, M. Pauline, 1994, "Computer Graphics: Principles and Practice", Prentice Hall.
- Jain, A. K., 1989, "Fundamentals of Digital Image Processing", Prentice Hall.

ESPAÇO DE TRABALHO DE UM ROBÔ QUADRÚPEDE CONSIDERANDO SEU MECANISMO DE PERNA

Tarcísio de Assunção Pizziolo

Universidade Vale do Rio Doce – UNIVALE
Departamento de Ciência da Computação - CECET
Campus Antônio Rodrigues Coelho
Rua Israel Pinheiro, n. 2000 – Campus Universitário
Governador Valadares – MG / BR
CEP – 35020-220
pizziolo@net.em.com.br

Peterson de Resende

Universidade Federal de Minas Gerais – UFMG
Centro de Pesquisas e Desenvolvimento em Engenharia Elétrica - CPDEE
Av. Antônio Carlos, n. 6627 – Pampulha
Belo Horizonte – MG / BR
CEP – 31270-901
pr@cpdee.ufmg.br

José Luiz Silvino

Universidade Federal de Minas Gerais – UFMG
Centro de Pesquisas e Desenvolvimento em Engenharia Elétrica - CPDEE
Av. Antônio Carlos, n. 6627 – Pampulha
Belo Horizonte – MG / BR
CEP – 31270-901
silvino@cpdee.ufmg.br

Luiz S. Martins Filho

Universidade Federal de Ouro Preto – UFOP
Departamento de Ciência da Computação – DECOM/ICEB
Campus Universitário – Morro do Cruzeiro
Ouro Preto – MG/BR
CEP – 35400-000
luizm@iceb.ufop.br

Resumo. O espaço de trabalho para uma dada configuração das pernas de um robô apresenta todas as possíveis translações e rotações do corpo relativas a uma dada posição e orientação, as quais não violam as restrições impostas por este espaço e os limites das juntas. Utilizando-se um modelo desenvolvido para um mecanismo de perna possuindo três graus de liberdade, e através de um algoritmo computacional, determinou-se o espaço de trabalho cinemático e estático para um robô quadrúpede. O espaço de trabalho no qual o robô quadrúpede utiliza para a movimentação de seus pés, é a interseção dos espaços de trabalho cinemático e estático. O conhecimento prévio do espaço de trabalho de um robô é usualmente empregado na otimização de sua geometria construtiva, na determinação do número e da seleção de suas juntas ativas, na determinação das forças e torques das juntas ativas e na distribuição de forças nas pernas. Através de simulações foram determinados resultados gráficos os quais apresentam, utilizando-se planos de níveis, áreas de trabalho para o posicionamento dos pés do robô quadrúpede proposto. Um protótipo do mecanismo de perna modelado foi desenvolvido em laboratório e é também apresentado neste trabalho.

Palavras chave: robótica, máquinas caminhantes, espaço de trabalho.

1. Introdução

Robôs podem ser utilizados em trabalhos sucessivos onde a atuação humana torna-se inviável principalmente sob certas condições ambientais: subaquática, em plantas nucleares e em outros ambientes hostis. Robôs móveis podem ser divididos em dois tipos de acordo com o seu mecanismo de acionamento: tipo sobre rodas e tipo sobre pernas. Para o caso de robôs com locomoção sobre pernas as análises estruturais da geometria do mecanismo de perna, bem como seus movimentos são de suma importância para sua construção. (Cho et al., 1995). Uma forma de estabelecer condições para tais análises é a determinação do espaço de trabalho do robô. O conhecimento prévio do espaço de trabalho de um robô é usualmente empregado na otimização de sua geometria construtiva, na determinação do número e da seleção de suas

juntas ativas, na determinação das forças e torques das juntas ativas e na distribuição de forças nas pernas (Zhang et al., 1996a).

2. O protótipo

O protótipo do mecanismo de perna constitui-se basicamente de uma caixa metálica envoltória às engrenagens que acionam a perna pantográfica. A relação do comprimento das barras horizontais e verticais do protótipo foi adotada conforme literatura com uma razão de 0,67 (Song et al., 1989). A Fig. (1) apresenta os esquemas do protótipo desenvolvido para o mecanismo de perna e do robô quadrúpede proposto.

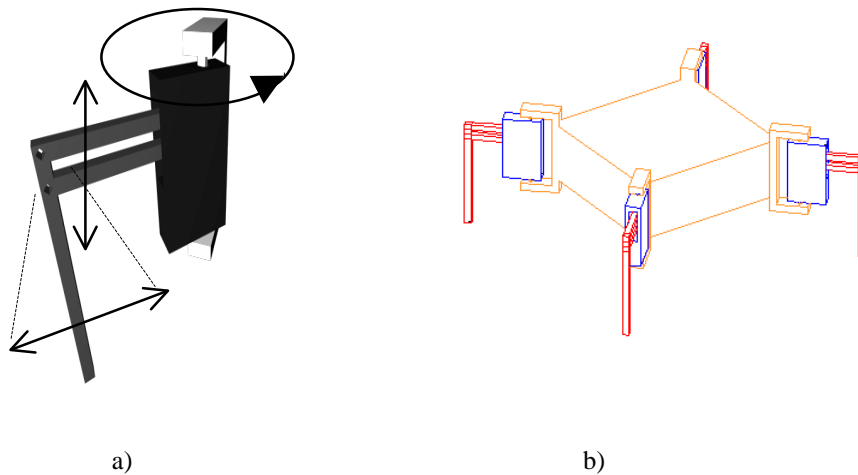


Figura 1. a) Esquema do protótipo do mecanismo de perna. b) Esquema do robô quadrúpede proposto.

As especificações do protótipo do mecanismo de perna proposto foram definidas considerando três graus de liberdade no movimento e são dadas na Tab. (1).

Tabela 1. Especificações do protótipo.

Extensão do passo	0,10 m.
Deslocamentos (m)	1. Longitudinal: 0,08 2. Vertical: 0,06 3. Lateral: 0,05
Cargas (kg)	1. Vertical: 2,5. 2. Horizontal: 1,0 3. Lateral: 0,5
Características Estruturais (m)	1. Altura: 0,12 2. Largura: 0,08 3. Espessura: 0,02
Peso (kg)	1

3. Modelagem cinemática

A modelagem cinemática do mecanismo envolveu transformações de sistemas de coordenadas. O método utilizado para tais transformações foi o Método de Denavit-Hartenberg (DHM) (Koren, 1985). A Fig. (2) apresenta o modelo proposto.

Para a modelagem cinemática direta utilizaram-se parâmetros das juntas os quais são apresentados na Tab. (2). A matriz de deslocamento entre o referencial O_3 e a base O_0 é estabelecida utilizando a cadeia de operação para matrizes:

$${}^3_0T = {}^1_0T \cdot {}^2_1T \cdot {}^3_2T \tag{1}$$

A posição do ponto P expressa em relação ao referencial O_0 é dada pelo vetor de translação:

$$X_P = (a_3c_1c_2c_3 - a_3c_1s_2s_3 + a_2c_1c_2) \tag{2}$$

$$Y_p = (a_3 s_1 c_2 c_3 - a_3 s_1 s_2 s_3 + a_2 s_1 c_2) \tag{3}$$

$$Z_p = (a_3 s_2 c_3 + a_3 c_2 s_3 + a_2 s_2 + d_1) \tag{4}$$

Onde: $c_i = \cos \theta_i$ e $s_i = \sin \theta_i$.

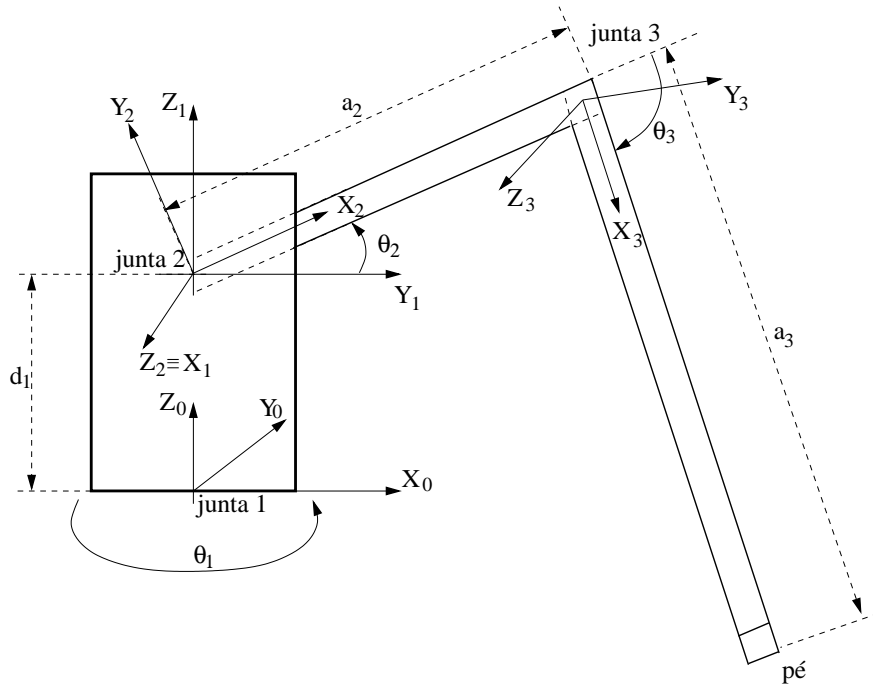


Figura 2. Esquema do modelo proposto para o mecanismo de perna.

Tabela 2. Parâmetros DHM para as juntas do mecanismo.

<i>Parâmetros das juntas do mecanismo de perna</i>				
Junta	θ_i	d_i	a_i	α_i
1	θ_1	2	0	$+\pi/2$
2	θ_2	0	8	0
3	θ_3	0	12	0

Para a modelagem cinemática inversa utilizaram-se soluções geométricas aplicadas ao modelo da perna. Os cálculos para encontrar os valores dos ângulos de posicionamento do pé, dadas as coordenadas espaciais do mesmo, são efetuados através das equações apresentadas a seguir considerando os sistemas de coordenadas da Fig. (3).

$$\beta_1 = \cos^{-1}((r_{1p}^2 - a_2^2 - a_3^2) / (-2a_2a_3)) \tag{5}$$

$$\beta_2 = \cos^{-1}((a_3^2 - r_{1p}^2 - a_2^2) / (-2r_{1p}a_2)) \tag{6}$$

$$\beta_3 = \sin^{-1}(z_3 / r_{1p}) \tag{7}$$

$$\theta_1 = \text{tg}^{-1}(x_1 / y_1) \tag{8}$$

$$\theta_2 = (\beta_2 + \beta_3) \tag{9}$$

$$\theta_3 = -(180^\circ - \beta_1) \tag{10}$$

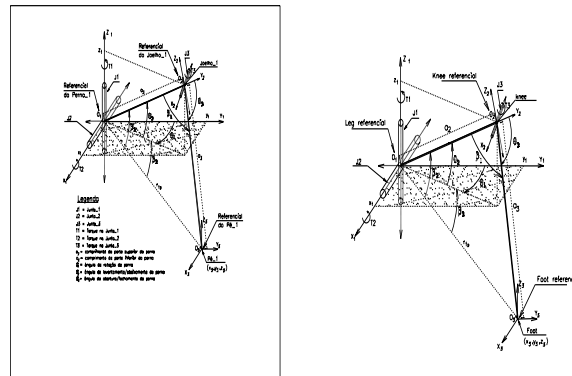


Figura 3. Sistemas de coordenadas do modelo de mecanismo de perna proposto.

4. Modelagem estática

Considerou-se que o mecanismo seja constituído por atuadores mecânicos (juntas e elos de ligações) e que todas as forças e momentos que agem sobre este, com exceção das forças gravitacionais e inerciais, são denominadas forças generalizadas não conservativas e representadas através de um vetor Q_i . Estas forças são calculadas a partir dos torques τ de controle das juntas (Craig, 1986; Zhang et al., 1996b).

$$Q_i = \tau + J \cdot F_e \tag{11}$$

Onde:

- τ = vetor torque nas juntas ativas.
- F_e = vetor de forças externas.
- J = matriz Jacobiana.

As reações do solo no pé do mecanismo de perna dependem das combinações das juntas ativas que estão sendo acionadas. As reações entre o pé e o solo são expressas em termos das componentes de força independente $[f_x \ f_y \ f_z]^T$ (Zhang et al., 1996b). A Fig. (4) mostra um esquema para a determinação das forças de reações do solo no pé do robô.

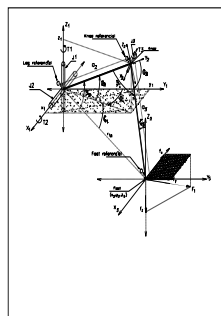


Figura 4. Esquema para a determinação das forças de reações do solo no pé do robô.

A componente f_x é formada pela contribuição do torque τ_1 referente à junta-1 ativa e é dada por:

$$f_x = (\tau_1 / r_{1p}) \cdot \cos(\theta_1) \quad (12)$$

A componente f_y é formada pela contribuição do torque τ_1 referente à junta-1 ativa, do torque τ_2 referente à junta-2 ativa, do torque τ_3 referente à junta-3 ativa e do peso do robô, e é dada por:

$$f_y = (\tau_1 / r_{1p}) \cdot \sin(\theta_1) + (\tau_2 / r_{1p}) \cdot \sin(\zeta_1) + (\tau_3 / a_3) \cdot \sin(\zeta_2) + F_{\text{Peso}} \cdot \cos(\alpha_1) \quad (13)$$

$$\zeta_1 = \sin^{-1}(z_p / r_{1p}) \quad (14)$$

$$\zeta_2 = \cos^{-1}((z_3 - z_1) / a_3) \quad (15)$$

$$\beta_1 = \cos^{-1}((r_{1p}^2 - a_2^2 - a_3^2) / (-2a_2a_3)) \quad (16)$$

A componente f_z é formada pela contribuição do torque τ_2 referente à junta-2 ativa, do torque τ_3 referente à junta-3 ativa e do peso do robô, e é dada por:

$$f_z = (\tau_2 / r_{1p}) \cdot \cos(\zeta_1) + (\tau_3 / a_3) \cdot \cos(\zeta_2) + F_{\text{Peso}} \cdot \sin(\beta_1) \quad (17)$$

Onde:

r_{1p} = comprimento do vetor de origem na junta-2 e extremidade no pé.

θ_1 = ângulo de rotação da perna.

F_{Peso} = força do peso do robô.

z_3 = coordenada do pé no espaço referida ao sistema inercial.

z_1 = coordenada do pé no espaço referida ao sistema O_1 .

a_2 = comprimento da “parte superior” da perna.

a_3 = comprimento da “parte inferior” da perna.

A reação total do solo no pé do mecanismo de perna será então:

$$\mathbf{F}_t = [f_x \quad f_y \quad f_z]^T \quad (18)$$

5. Espaço de trabalho

O espaço de trabalho para uma dada configuração das pernas de um robô apresenta todas as possíveis translações e rotações do corpo relativas a uma dada posição e orientação, as quais não violam as restrições impostas por este espaço. O espaço de trabalho é a interseção dos espaços de trabalho cinemático e estático, o qual é usualmente empregado para otimizar a construção do robô (Klein et al., 1990).

Devido ao desempenho de modernos atuadores e ao desenvolvimento de estudos sobre a geometria do corpo principal de um robô, considera-se que as limitações das coordenadas de juntas tornam-se mais relevantes para os estudos e análises cinemáticos em relação às limitações de velocidade e aceleração da perna bem como a interferência geométrica desta. Daí estas restrições angulares poderem ser expressas por:

$$\theta_{\text{mín}} \leq \theta_{ij} \leq \theta_{\text{máx}} \quad (19)$$

Onde θ_{ij} é o ângulo da junta j pertencente à perna i .

Escrevendo sob forma de um vetor de restrição cinemática:

$$\mathbf{q}_{\text{mín}} = [\theta_{1\text{mín}} \quad \theta_{2\text{mín}} \quad \theta_{3\text{mín}}]^T \quad (20)$$

$$\mathbf{q}_{\text{máx}} = [\theta_{1\text{máx}} \quad \theta_{2\text{máx}} \quad \theta_{3\text{máx}}]^T \quad (21)$$

Para a determinação do espaço de trabalho cinemático calcula-se as coordenadas de juntas θ_{i1} , θ_{i2} e θ_{i3} ($i = n^\circ$ de pernas) que representam a posição do pé do robô no espaço de trabalho. A partir destas coordenadas constrói-se o vetor $\mathbf{q}_i = [\theta_{i1} \quad \theta_{i2} \quad \theta_{i3}]$. Tomando-se como referência os vetores de restrição, subtrai-se \mathbf{q}_i de $\mathbf{q}_{\text{máx}}$ e \mathbf{q}_i de $\mathbf{q}_{\text{mín}}$. Os resultados são escritos como $\Delta\mathbf{q}_{\text{máx}}$ e $\Delta\mathbf{q}_{\text{mín}}$, respectivamente. Quando qualquer componente de $\Delta\mathbf{q}_{\text{máx}}$ ou de $\Delta\mathbf{q}_{\text{mín}}$ for zero, um ponto da fronteira do espaço de trabalho cinemático foi encontrado. Este ponto então é considerado como limite máximo ou mínimo do espaço de trabalho cinemático do robô, ou seja, o pé do robô somente poderá ser posicionado dentro destes

limites. Aplicando-se uma varredura em todos os componentes de \mathbf{q}_i obtém-se o espaço de trabalho cinemático do mecanismo de perna.

Para o espaço de trabalho estático utiliza-se:

$$F_{\min} \leq F_{ij} \leq F_{\max} \text{ (Força de acionamento na junta } j \text{ da perna } i) \quad (22)$$

$$T_{\min} \leq T_{ij} \leq T_{\max} \text{ (Torque de acionamento na junta } j \text{ da perna } i) \quad (23)$$

$$f_{\min} \leq f_{ij} \leq f_{\max} \text{ (Força de reação do solo no pé da perna } i) \quad (24)$$

$$\mu_t = f_{xy}/f_z = (f_x^2 + f_y^2)^{1/2}/f_z \leq \mu \quad (25)$$

(μ = Coeficiente de atrito estático entre o pé e o solo)

O vetor de restrição estática é dado por:

$$\mathbf{q}_{stmax} = [F_{\max}, T_{\max}, f_{\max}, \mu] \quad (26)$$

$$\mathbf{q}_{stmin} = [F_{\min}, T_{\min}, f_{\min}, \mu] \quad (27)$$

Para obter-se espaço de trabalho estático, aplicou-se um raciocínio análogo ao utilizado para espaço de trabalho cinemático (Zhang et al., 1996a; Zhang et al., 1996b).

6. Resultados das simulações

A seguir são apresentados os espaços de trabalho cinemático e estático para o robô quadrúpede proposto. Considerou-se nestes resultados que o robô realiza movimentos somente no plano. Planos de níveis foram determinados a partir dos espaços de trabalho para que fosse possível a visualização das interseções entre estes. Como amostra são apresentados os espaços de trabalho cinemático e estático e os planos de nível para a cota $z=-5$.

6.1. Espaço de trabalho cinemático

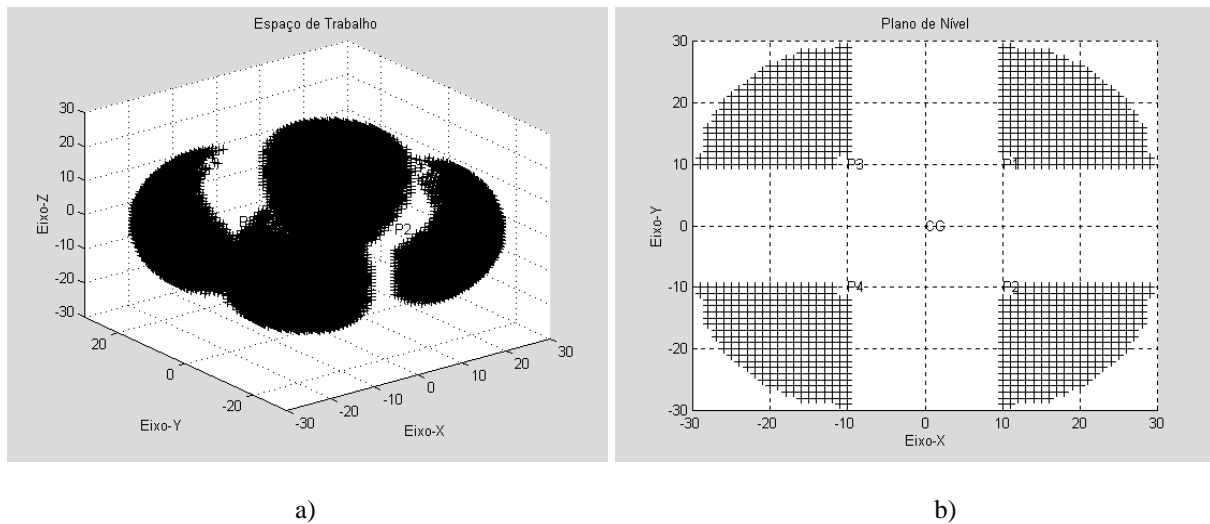
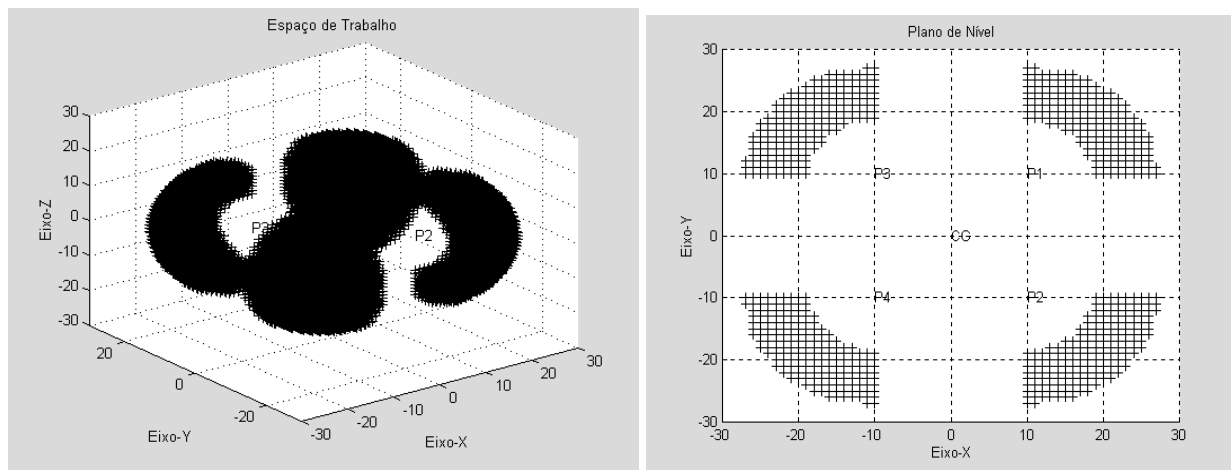


Figura 5. a) Espaço de trabalho cinemático. b) Plano de nível para $z = -5$.

6.2. Espaço de trabalho estático



a) Espaço de trabalho estático. b) Plano de nível para $z = -5$.

7. Conclusões

Neste trabalho foi apresentada uma sistemática de investigação do espaço de trabalho de um robô quadrúpede. Foram também discutidas as análises das cinemáticas direta e inversa do modelo, as restrições cinemáticas e estáticas e um algoritmo para a determinação do espaço de trabalho para o robô quadrúpede foi proposto. Os planos de nível apresentados para $z=-5$ fornecem informações de que existem pontos onde as restrições estáticas não permitem o posicionamento dos pés do robô em relação às restrições cinemáticas. Isto implica que na interseção dos espaços de trabalho de um robô quadrúpede, em determinadas condições, as restrições de um espaço de trabalho específico, cinemático ou estático, poderão interferir no projeto do sistema de locomoção do mesmo. Assim, para o caso de robôs com locomoção sobre pernas, a combinação destas análises entre os dois espaços de trabalho é de suma importância para o projeto da estrutura da geometria do mecanismo de perna, bem como seus movimentos para sua construção.

8. Agradecimentos

Os autores agradecem aos suportes financeiros da FAPEMIG, CAPES e CNPq.

9. Referências

- Cho D. J., Kim J. H. and Gweon D. G., 1995, "Optimal turning gait of a quadruped walking robot". *Robotic*, Vol. 13, pp. 559-564.
- Craig, J. J., 1986, "Introduction to Robotics Mechanics and Control", Ed. Addison Wesley Publishing Company, Stanford University, USA.
- Klein, C. A. and Kittivatcharapong, S., 1990, "Optimal force distribution for legs of a walking machine with friction cone constraints", *IEEE Trans. On Robotics and Automation*, Vol. 6, N.1.
- Koren Y., 1985, "Robotics for engineers", Ed. McGraw Hill.
- Song, S.M., Waldron, K.J., *Machines That Walk*, 1989, MIT Press.
- Zhang, S. J., Sanger, D. J. and Howard, D., 1996a, "Workspaces of a walking machine and their graphical representation. Part I: Kinematic Workspaces", *Robótica*, Vol. 14, pp. 71-79, Cambridge University Press, USA.
- Zhang, S. J., Sanger, D. J. and Howard, D., 1996b, "Workspaces of a walking machine and their graphical representation. Part II: Static Workspaces", *Robótica*, Vol. 14, pp. 219-226, Cambridge University Press, USA.

WORKSPACE FOR A QUADRUPED ROBOT CONSIDERING YOUR LEG MECHANISM

Tarcísio de Assunção Pizziolo

Universidade Vale do Rio Doce – UNIVALE

Departamento de Ciência da Computação - CECET

Campus Antônio Rodrigues Coelho

Rua Israel Pinheiro, n. 2000 – Campus Universitário

Governador Valadares – MG / BR

CEP – 35020-220

pizziolo@net.em.com.br

Peterson de Resende

Universidade Federal de Minas Gerais – UFMG
Centro de Pesquisas e Desenvolvimento em Engenharia Elétrica - CPDEE
Av. Antônio Carlos, n. 6627 – Pampulha
Belo Horizonte – MG / BR
CEP – 31270-901
pr@cpdee.ufmg.br

José Luiz Silvino

Universidade Federal de Minas Gerais – UFMG
Centro de Pesquisas e Desenvolvimento em Engenharia Elétrica - CPDEE
Av. Antônio Carlos, n. 6627 – Pampulha
Belo Horizonte – MG / BR
CEP – 31270-901
silvino@cpdee.ufmg.br

Luiz S. Martins-Filho

Universidade Federal de Ouro Preto – UFOP
Departamento de Ciência da Computação – DECOM/ICEB
Campus Universitário – Morro do Cruzeiro
Ouro Preto – MG/BR
CEP – 35400-000
luizm@iceb.ufop.br

***Abstract.** This paper discusses the analysis of the leg workspace mechanisms for a quadruped. The kinematic and static workspaces for a quadruped robot are defined and an algorithm for investigation is presented. The leg workspace for a given leg configuration considers all possible body translations and rotations, the initial position and orientation, and the joints limits. The resultant workspace is the intersection of the kinematic and static workspaces, where the kinematic workspace is based upon the kinematic constraints and the static workspace is based upon the static constraints. The analysis of the workspace is very important to attempt to optimise the machine's geometry, the number and selection of active joints, the active joint forces/torques and leg force distribution. Graphical results are obtained by numerical simulation of the kinematics and dynamics leg models.*

Key words: robotics, walking machines, workspace.

UM BARRAMENTO DE CAMPO CONTROLLER AREA NETWORK PARA UM DIRIGÍVEL AUTÔNOMO NÃO TRIPULADO

Faria, Bruno Guedes*

* Instituto Nacional de Tecnologia da Informação – ITI . Laboratório de Robótica e Visão – LRV - Rod. D. Pedro Km 143,6 - Caixa Postal 6162 - CEP 13089-500 Campinas, SP – Brasil

Bruno.Faria@iti.br

Maeta , Silvio Mano*

Silvio.Maeta@iti.br

Ramos , Josué Jr. Guimarães*

Josué.Ramos@iti.br

Bergerman , Marcel*

Marcel.Bergerman@iti.br

Bueno , Samuel Siqueira*

Samuel.Bueno@iti.br

Resumo. Muitas instituições vêm desenvolvendo veículos aéreos autônomos objetivando-se a execução de diversas tarefas como exploração, vigilância ou como plataformas para demonstrar novas tecnologias. Neste contexto o Instituto Nacional de Tecnologia da Informação propôs o projeto AURORA (Autonomous Unmanned Remote Monitoring Robotic Airship). A meta principal do projeto é oferecer um veículo de baixa velocidade e baixa altitude para missões de inspeção aérea. Este artigo trata da implementação de uma solução para integração dos sensores e atuadores presentes no dirigível . Essa solução tem como objetivo substituir a solução adotada anteriormente, que foi conectar esses dispositivos à CPU por meio de portas seriais. Apesar destas serem eficientes tanto em velocidade de transmissão quanto em facilidade de configuração, o número limitado de portas disponíveis restringe o número de componentes que podem ser acoplados à CPU. Tendo em vista esse problema, foram estudadas várias possibilidades de integração. Dentre elas a que se mostrou mais adequada foi a utilização de um barramento CAN (Controller Area Network). Este tipo de barramento é muito difundido na indústria automobilística, sendo utilizado, principalmente na conexão de dispositivos que necessitem de velocidade e imunidade a falhas.

Palavras chave: Contoller Area Network, CAN, Dirigível, Robótica, Barramento de Campo

1. Introdução

Muitas instituições vêm desenvolvendo veículos aéreos autônomos e semi-autônomos com o objetivo de executar diversas tarefas como exploração, vigilância, inspeção aérea, ou como plataformas para demonstrar novas tecnologias. Neste contexto o Instituto Nacional de Tecnologia da Informação propôs o projeto AURORA (Autonomous Unmanned Remote Monitoring Robotic Airship), onde estratégias de navegação autônoma estão sendo desenvolvidas para o controle de dirigíveis não tripulados.

O projeto AURORA visa o desenvolvimento de protótipos de veículos aéreos tele-operados até a obtenção de veículos telemonitorados, através do desenvolvimento de sistemas com graus de autonomia crescentes. A meta principal do projeto é oferecer um veículo de baixa velocidade e baixa altitude para missões aéreas de inspeção e monitoramento ambiental. A escolha do dirigível se deve às suas características superiores aos aviões e helicópteros para este tipo de missão, principalmente a sua capacidade de flutuar, o que o torna menos intrusivo, e possibilidade de permanecer por longos períodos no ar. A Figura (1a) apresenta a conceituação do Projeto Aurora e a Fig.(1b) mostra o dirigível AS800 utilizado no projeto.

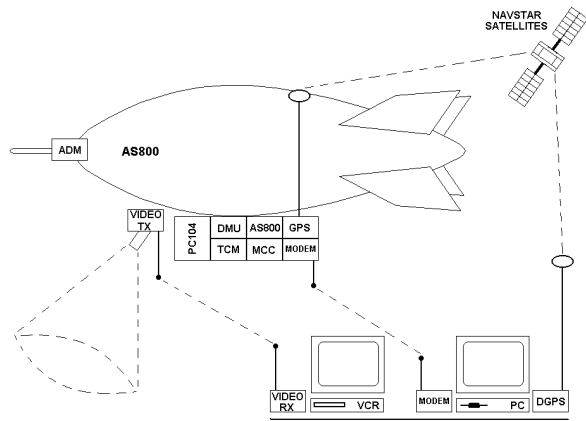


Figura 1a: Conceituação do Projeto AURORA.

Figura 1b: Dirigível do Projeto AURORA

Os principais componentes do AURORA são: o dirigível em si, o sistema de controle e navegação a bordo com seu computador; sensores internos, atuadores e sensores externos, caracterizando o sistema embarcado; uma estação de terra, móvel, que provê a interface para a operação do dirigível; e o sistema de comunicação entre a infra-estrutura embarcada e a estação de terra. Os sensores externos, também situados a bordo do dirigível, são aqueles necessários para o cumprimento das missões descritas acima, e, naturalmente, variam de missão para missão.

Dentro desse contexto surgiu a necessidade do acoplamento de diversos dispositivos no sistema de navegação e controle. Inicialmente a solução adotada foi a conexão por meio das interfaces seriais RS232 do computador embarcado. Entretanto esta solução não é adequada, pois limita muito tanto a quantidade de sensores quanto a capacidade de comunicação entre dispositivos. A solução para possibilitar uma maior flexibilidade na instalação de dispositivos (sensores, atuadores) e aumentar a interação entre eles e o computador de bordo foi a adoção de um barramento de campo. O barramento escolhido para esse propósito foi o barramento CAN - Controller Area Network.

Este artigo trata exatamente dessa questão, abordando os principais tópicos relativos ao desenvolvimento de uma arquitetura de interconexão de dispositivos em um veículo aéreo não tripulado sobre o barramento CAN. O artigo está organizado da seguinte forma: a segunda seção apresenta a motivação da utilização do CAN como barramento de campo do Projeto AURORA, a terceira seção define a solução CAN embarcada, a quarta seção avalia o desempenho geral da solução CAN, a quinta seção propõe a definição de um protocolo de alto nível proprietário para CAN a ser implementado futuramente no barramento de campo do dirigível do Projeto AURORA, a sexta seção conclui o artigo, a sétima seção refere-se aos agradecimentos e a oitava seção é composta pela lista de referências.

2 . Motivação da utilização do CAN como barramento de campo do Projeto AURORA

Conforme mencionado acima, no sistema embarcado do dirigível existe uma grande variedade de componentes que devem ser acoplados ao computador embarcado, no caso um computador PC-104, doravante denominado CPU. Inicialmente essa conexão era feita por meio de portas seriais. Apesar destas portas serem eficientes tanto em velocidade de transmissão quanto em facilidade de configuração, o número de portas disponíveis é limitado, impondo assim um número máximo de componentes que podem ser acoplados simultaneamente à CPU.

Tendo em vista esse problema, foram estudadas várias formas alternativas de integração desses componentes. O que se mostrou mais adequado foi o barramento CAN. No barramento CAN a nível físico a transmissão é feita por dois fios com sinal diferencial, o que confere grande imunidade a ruídos, e o envio de mensagens é feito via *broadcast*. Uma larga gama de microcontroladores com interfaces CAN permite o desenvolvimento de sensores e atuadores inteligentes. No caso do projeto AURORA estas são exatamente as situações em que essa tecnologia está sendo aplicada.

Para a conexão de um dispositivo ao barramento CAN é necessário que um terceiro elemento atue como interface entre ambos. No Projeto AURORA, a escolha adotada para esta interface foi a utilização de módulos equipados com microcontroladores capazes de atender aos requisitos necessários para a conexão. Dessa forma podem estar ligados ao barramento todos os dispositivos antes conectados a portas seriais. Para isso é necessário desenvolver um software para o microcontrolador referente a um ou mais sensores de maneira que ele tenha como entrada os dados desse(s) sensor(es) e como saída as informações empacotadas no formato CAN ou então comprar módulos disponíveis comercialmente que convertem de CAN para diferentes tipos de interface como serial, ethernet, etc. Outra vantagem dessa arquitetura de conexão dos dispositivos baseada em módulos com microcontroladores é que estes dispositivos vêm equipados com diversos recursos que facilitam a conexão de sensores (mecanismo de interrupções externas, conversores analógico-digital, saídas digitais seriais, paralelas, etc).

Nas Figuras(2a,2b,2c) podem ser observadas diferentes possibilidades de arquiteturas de conexão dos dispositivos. Elas são, respectivamente, a maneira convencional de acoplamento dos sensores em estrela, uma solução utilizando-se CAN e um sistema híbrido, este último atualmente em funcionamento no Projeto AURORA.

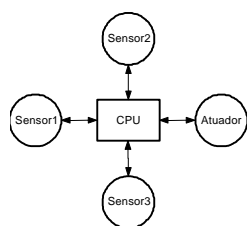


Figura 2.a: Sistema com portas seriais RS232

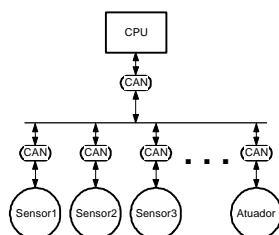


Figura 2.b: Sistema embarcado Utilizando o CAN

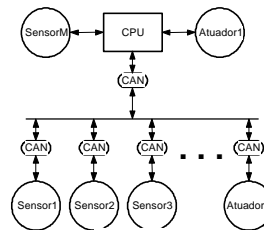


Figura 2.c: Sistema híbrido (Atual)

O barramento CAN com elementos microcontrolados possibilita também que haja interação entre os outros elementos da rede sem participação de um elemento intermediário. Com o recurso de prioridades (já implementado pelo hardware CAN), pode-se definir uma mensagem com alto nível de prioridade (caracterizando uma mensagem emergencial) que, ao ser recebida por um determinado módulo atuador, não espere comandos diretos da CPU para atuar no sistema. Para que se possa tornar possível a integração dos módulos nesse nível é necessário que esse comportamento esteja previsto no programa presente no microcontrolador dos módulos em questão. Essa característica é muito útil em situações de falha da CPU, pois, se for implementada, permite que o barramento tome algumas atitudes de maneira independente da CPU. Um exemplo dessa situação é o desligamento imediato dos motores se detectada uma acentuada proximidade do solo em um pouso forçado.

3 . Solução CAN embarcada

Nesta seção são apresentadas as características de uma solução de integração dos dispositivos ao barramento CAN. Embora o número de dispositivos conectados seja reduzido, o processo é análogo para um número maior deles.

Uma topologia possível para o barramento de campo é ilustrada na Fig.(2b). Ligados ao barramento estão os dispositivos de interface, que, associados aos elementos do sistema embarcado(sensores, atuadores e CPU), são denominados módulos. Cada módulo é considerado um nó da rede. Dentro desse contexto, distinguem-se dois tipos de nós: nós escravos e nó central. Os nós escravos são os módulos sensores e módulos atuadores, enquanto o nó central é o módulo que integra a CPU ao barramento. As Figuras 3a e 3b mostram os dois dispositivos de interface utilizados no barramento. O dispositivo CAN4EPP (Fig.(3a)) faz parte do módulo da CPU(nó central), enquanto o dispositivo miniMODUL (Fig.(3b)) é utilizado nos outros módulos (nós escravos). A seguir serão listados os componentes de hardware do sistema de conexão dos sensores e a descrição dos métodos utilizados para integrá-los ao sistema embarcado.



Figura 3a: CAN4EPP da Zanthic

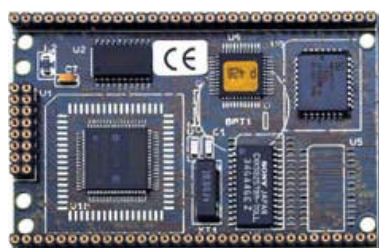


Figura 3b: miniMODUL 515C da Phyttec

3.1 Interface de conexão da CPU com o barramento CAN

O dispositivo CAN4EPP, mostrado na Fig.(3a), está acoplado à CPU embarcada do dirigível pela porta paralela EPP. Para possibilitar o acoplamento desse dispositivo à CPU foi gerado um driver para RTLinux (Real Time Linux, sistema operacional utilizado na CPU embarcada) baseado no driver para Windows 95 fornecido pelo fabricante. Esse driver possuiu apenas a função de atuar como uma interface entre o sistema de gerenciamento das mensagens e o dispositivo CAN4EPP, não possuindo nenhuma função relativa ao protocolo.

Além do referido driver, foi implementado um programa com as seguintes características: definir uma política de utilização dos objetos de mensagens e outros recursos disponíveis no CAN4EPP; estabelecer comunicação com o sistema embarcado fornecendo para o mesmo as informações obtidas pelos sensores do barramento CAN e atender a

solicitações efetuadas pelo sistema embarcado passando-as para os dispositivos do barramento da maneira correta. Todos os programas foram escritos em linguagem C++ de acordo com o paradigma de programação orientada a objetos.

3.2 Módulo do sensor de rotação dos motores

Este módulo usa, para interpretar os dados sensoriais e prover a interface com o barramento CAN, o miniMODUL mostrado na Fig.(3b).

A informação de rotação dos motores é extremamente útil na fase de avaliação do modelo do dirigível e na fase de controle automático, pois permite relacionar a rotação das hélices com a velocidade relativa alcançada pelo dirigível e também efetuar um controle mais preciso da atuação em cada motor.

Para a obtenção dessa informação foi projetado um sensor infravermelho de rotação. O funcionamento do sensor é baseado na emissão de um pulso lógico cada vez que a hélice se coloca entre o emissor e o receptor interrompendo o feixe infravermelho. A rotação é dada pelo cálculo do intervalo de tempo entre a borda de subida de dois pulsos consecutivos. A Figura(4) apresenta um esquema ilustrativo da montagem do sensor no dirigível e sua conexão ao barramento CAN.

Para efetuar essa medida foi utilizado o recurso de interrupção externa do microcontrolador que integra o módulo miniMODUL da Phytec, o C515C da Siemens. O mecanismo utilizado foi a aquisição do valor de um contador contínuo interno ao C515C para cada pulso gerado, por meio de uma rotina de tratamento de interrupção. A partir da medida do intervalo de tempo são efetuados os cálculos cabíveis e enviados para o barramento CAN as informações de rotação de ambos os motores.

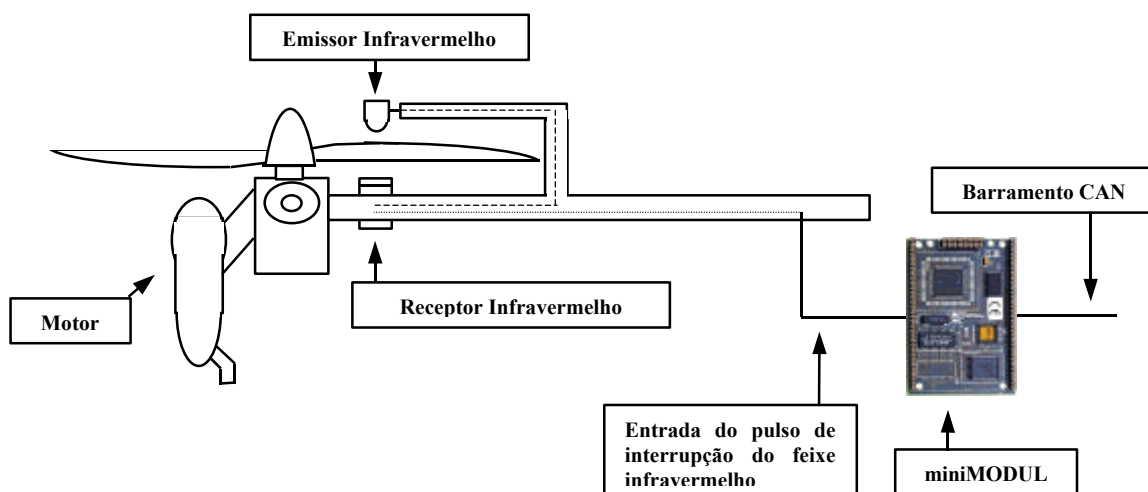


Figura 4: Esquema de montagem do sensor de rotação no dirigível.

Além desses componentes, no software do microcontrolador foi incorporado todo o mecanismo de comunicação CAN. A versão atual não implementa nenhum tipo de gerenciamento da rede, somente são enviados os dados ao barramento sob um identificador definido arbitrariamente. Posteriormente neste documento será proposto um protocolo de alto nível para CAN a ser implementado.

3.3 Módulo do sensor ADM

Outras necessidades importantes para controle de um veículo aéreo são as informações de direções e velocidades relativas ao vento e o valor da altitude absoluta. Para gerar estes dados foi incorporada ao sistema uma sonda de vento, a ADM (Air Data Measurement).

Essa sonda, ilustrada na Fig.(5), funciona através de sensores piezzo-elétricos que medem a diferença de pressão criada quando um fluxo de ar percorre o corpo da sonda. Através de um conjunto de furos situados de maneira simétrica na extremidade da sonda, é possível obter diferenças de pressão pelo princípio do tubo de Pitot. A partir desses dados pode-se obter as seguintes informações (saídas de tensão analógica): ângulo de ataque vertical, ângulo de ataque horizontal, velocidade do vento e altitude barométrica.

Como esses dados são analógicos é necessário convertê-los para digital para que possam ser transferidos pelo barramento CAN para o sistema embarcado. O conversor analógico-digital utilizado foi o existente no próprio microcontrolador do miniMODUL. Entretanto a conversão não pode ser executada diretamente, pois o sinal de altitude absoluta, por exemplo, exigiria uma precisão muito maior que a existente no microcontrolador C515C. Assim os sinais precisam ser tratados por um circuito de pré-processamento. Esse circuito é formado por componentes analógicos e

digitais que ajustam o nível de saída dos sinais de modo a tornar possível a sua conversão pelo conversor A/D do microcontrolador C515C. Os valores de tensão máximos e mínimos dos sinais de saída são estabelecidos por meio de sinais de configuração enviados pelo microcontrolador do módulo. Esses sinais digitais de configuração do circuito de pré-processamento são mostrados na Fig.(5).

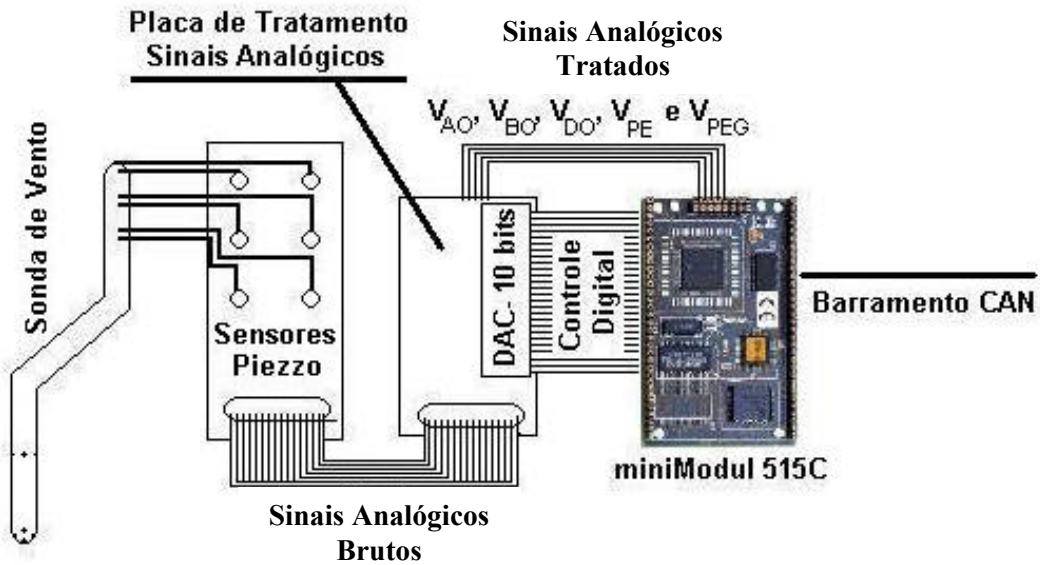


Figura 5: Esquema de conexão da sonda de vento ao barramento CAN

O software do módulo para o caso do ADM possui uma estrutura bem similar ao usado no sensor de rotação dos motores com a ressalva que, ao invés de medir o intervalo de tempo entre interrupções, ele monitora continuamente as entradas de conversão analógico-digital. O método de envio para o barramento CAN também é o mesmo, com a diferença do identificador da mensagem, pois trata-se de uma informação diferente que está sendo enviada.

Vale ressaltar que, nesse caso, foi a solução CAN que possibilitou a inclusão desse sensor de forma adequada no sistema embarcado. Como o sensor está vinculado ao comportamento do fluxo de ar na extremidade da proa do veículo, onde ele é localizado fisicamente, há uma distância considerável até a central de hardware do sistema embarcado, localizada próxima ao centro da parte inferior do corpo do dirigível. Uma conexão física para transportar um dado analógico seria sujeita a interferências eletromagnéticas, degradando consideravelmente o sinal (o que foi efetivamente constatado).

As Figuras(6a,6b) apresentam os resultados de aquisição de dados de altitude e pressão estática para um vôo do dirigível.

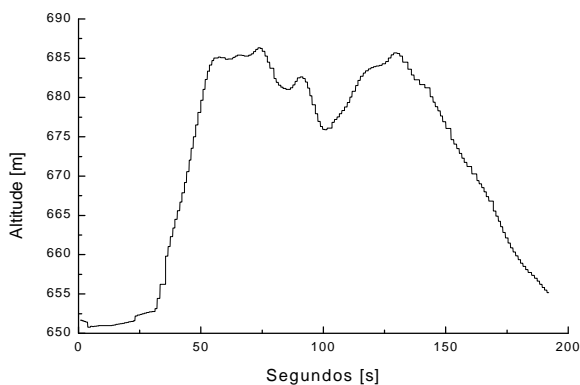


Figura 6a: Altitude dada pelo GPS durante um vôo experimental

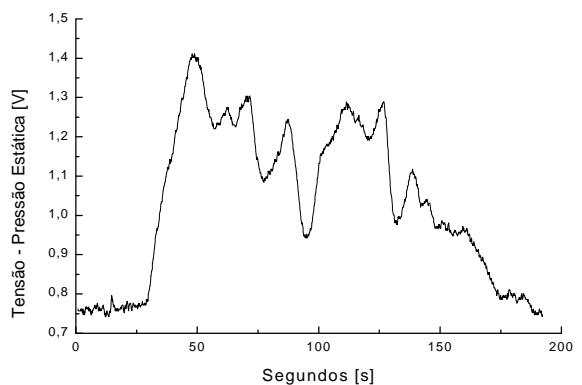


Figura 6b: Variação de tensão dada pela sonda vento durante o mesmo vôo da figura anterior

4 . Avaliação geral de desempenho da solução CAN

Foram efetuados vários testes com a arquitetura descrita anteriormente tanto em bancada quanto em ensaios práticos de telemetria em vôo, sendo observado o funcionamento correto do barramento CAN e de todos os dispositivos a ele acoplados, o que compreende os módulos de rotação dos motores, o ADM e a CPU.

Estes testes permitem estabelecer um cenário para a evolução da rede CAN, que deve contemplar na sua expansão os seguintes módulos sensoriais e de atuação:

GPS – (Global Position System) – Sistema de localização

DMU – (Dynamic Measurement Unit) – Sistema de medição inercial

TCM – (Electronic Compass Module) – Bússola eletrônica

ADM – (Air Data Measurement) – Sonda de vento

AS800 – Conjunto de sensores do corpo do dirigível (somente o medidor de rotação está operacional atualmente).

8 servomotores – Conjunto de atuadores do sistema embarcado.

Uma mensagem CAN (CAN Versão 2.0A) é composta por vários campos. De maneira simplificada, existem dois tipos principais: os dados a serem transmitidos e as informações da mensagem CAN. O tamanho máximo de uma mensagem CAN é de 110 bits, sendo 64 bits o valor máximo de dados que podem ser transmitidos e 46 bits fixos que compõe o cabeçalho da mensagem. A Figura (7) mostra os campos que compõe uma mensagem CAN.

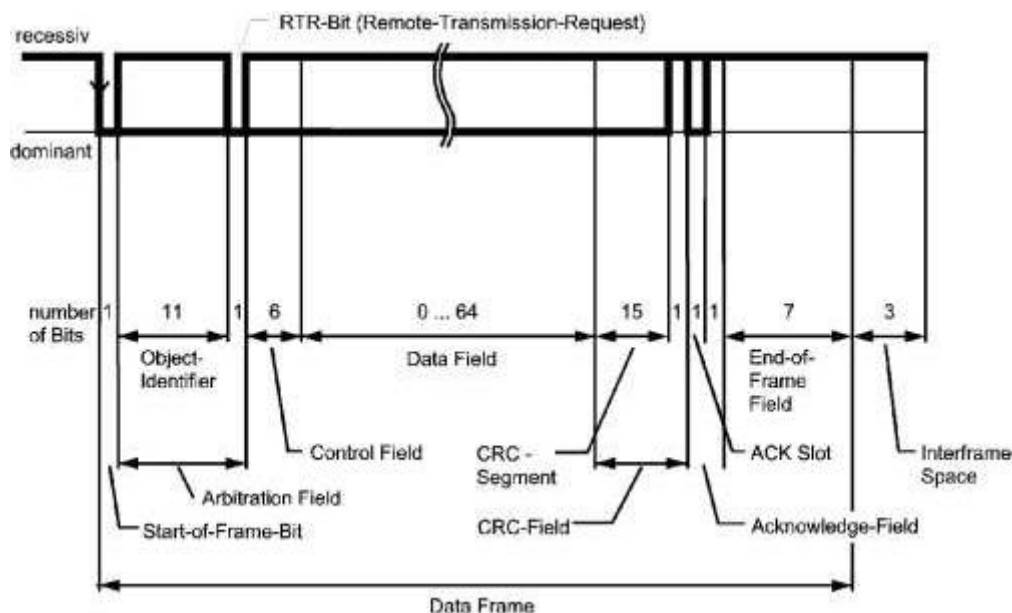


Figura 7: Composição de uma mensagem CAN

Assim, para possibilitar a transmissão de dados maiores que 8 bytes, ou deverão ser inseridos mecanismos de protocolos de alto nível para controlar o envio, o que influencia diretamente no tamanho total de dados transmitidos por dispositivo ou atribuir identificadores múltiplos relativos a um mesmo sensor. Como nesta seção o objetivo é somente o de comprovar a eficiência do mecanismo, foi adotada essa última solução para o envio de mensagens maiores de 8 bytes relativas a um mesmo dispositivo.

Os resultados obtidos na avaliação de desempenho do barramento CAN na sua primeira implementação e nas projeções futuras de utilização do barramento são apresentados a seguir:

Atualmente somente os sensores ADM e o AS800 estão conectados ao sistema embarcado via CAN. Para verificar a situação atual da arquitetura de sensores foi feito, primeiramente, um levantamento da utilização do barramento pelos dispositivos já operacionais. Assim, a situação atual do tamanho total das mensagens enviadas pelos sensores é apresentada na Tabela (1).

Tabela 1: Tamanho em bytes dos dados enviados por cada sensor ao barramento (dispositivos atualmente instalados). O cálculo do tamanho total das mensagens contabiliza todos os campos das mensagens necessários para transmitir de maneira completa a informação do dispositivo.

Dispositivo	Bytes por dispositivo	Pacotes de dados por dispositivo	Tamanho total das mensagens por dispositivo (bits)
ADM	16	2	220
AS800	6	1	94
Total	22	3	314

A sonda de vento (ADM) transmite 4 dados (ângulo de ataque vertical, ângulo de ataque horizontal, velocidade do vento e altitude barométrica) de 4 bytes cada. Dessa forma o sensor transfere 16 bytes de dados a cada ciclo de amostra. Como em cada mensagem CAN somente é possível enviar no máximo 8 bytes os dados foram divididos em duas mensagens com identificadores distintos, isto é, o módulo ADM envia duas mensagens distintas a cada ciclo, sendo que cada uma delas contém duas determinadas informações da sonda. Como são enviadas duas mensagens CAN completamente preenchidas (8 bytes de dados cada) o número total de bits transmitidos pelo barramento por ciclo é de 220.

A frequência de amostragem dos dados dos sensores é de 10 Hz. Esse valor é adequado às características dinâmicas do AS800. De posse desse valor é possível estimar a velocidade necessária para o tráfego de dados na rede:

$$314 \text{ bits a } 10 \text{ HZ} = 3140 \text{ bps};$$

A taxa utilizada atualmente é de 20 kbits/s. Nessas condições a taxa de ocupação é de cerca de 15%. Conforme observado, dentro do escopo do projeto e do nível de complexidade atual do conjunto de sensores, o barramento CAN é uma solução adequada e supre com uma considerável margem de segurança os quesitos exigidos pela arquitetura de conexões dos sensores embarcados.

Para comprovar a eficiência desta solução para a configuração futura de conexão (todos os dispositivos citados conectados ao barramento) foram efetuados cálculos relativos à ocupação do barramento levando em conta todos os dispositivos disponíveis. A Tabela (2) mostra uma projeção dos tamanhos das mensagens necessárias para cada sensor.

Tabela 2: Tamanho em bytes dos dados enviados por cada sensor ao barramento (previsão futura para os sensores conectados ao barramento CAN).

Dispositivo	Bytes por dispositivo	Pacotes de dados por dispositivo	Tamanho total das mensagens por dispositivo (bits)
GPS	53	7	746
DMU	52	7	738
TCM	20	3	298
ADM	16	2	220
AS800	6	1	94
MCC	36	5	518
8 Servos Motores	16	2	220
Total	199	27	2834

Os valores utilizados na Tab.(2) foram obtidos de maneira análoga aos da Tab.(1). Essa solução, embora viável, não apresenta uma estrutura expansível e de fácil manutenção, entretanto é válida para comprovar o conceito.

Considerando a frequência de 10 Hz, o tráfego da rede é calculado da seguinte forma:

$$2834 \text{ bits a } 10 \text{ Hz} = 28340 \text{ bps (bits por segundo)}.$$

Considerando uma taxa de erro de 10 % (elevada) o número total de bits por segundo é de 31174. Utilizando uma taxa de 50kbits/s (valor relativamente baixo para os valores possíveis de uma rede CAN, que pode chegar a 1Mbps) temos ocupação de cerca de 63% da capacidade de transmissão (cerca de 1/3 do tempo o barramento está livre).

5. Proposta de definição do protocolo de alto nível proprietário para CAN a ser implementado futuramente no barramento de campo do dirigível do Projeto AURORA

Nesta seção apresenta-se uma proposta para o protocolo de alto nível a ser utilizado no barramento CAN do dirigível do projeto AURORA.

Durante o desenvolvimento da solução do barramento de sensores, foi observado que é essencial que exista algum tipo de controle de mais alto nível sobre o protocolo nativo CAN. O monitoramento do funcionamento dos nós, o envio de mensagens mais longas e funções de comandos para os dispositivos são algumas das características desejáveis no barramento CAN e que somente são possíveis com a implementação de um protocolo de alto nível.

A partir dos estudos realizados sobre os protocolos de alto nível existentes na literatura (CANopen, CAN Kingdom, DeviceNet, CAN CAL) e sobre os requisitos desejados de um protocolo CAN foi gerada a especificação de um protocolo próprio a ser utilizado no projeto AURORA. O esboço dessa especificação é apresentada nessa seção.

A parte do cabeçalho que identifica a mensagem CAN é o identificador. Para a CAN Versão 2.0A esse identificador é composto de 11 bits. A maneira como é alocada a informação nesses 11 bits é essencial para o bom funcionamento de um barramento CAN, pois o esquema de prioridades de acesso ao barramento entre as mensagens é arbitrado pelo valor do identificador da mensagem. Levando em conta este conceito surgiram na literatura inúmeras maneiras de utilizar o identificador das mensagens para especificar qual o propósito da mensagem, qual sua origem e qual seu destino. Usualmente o identificador era dividido em partes, sendo que cada parte teria um significado de acordo com a especificação do protocolo. A exemplo do sistema utilizado no protocolo CANopen, foi especificado que o identificador fosse dividido em duas partes, uma especificando a função associada à mensagem (que tem uma maior prioridade) e outra associada à identificação do dispositivo (de menor prioridade).

Os bits do identificador das mensagens foram divididos da seguinte forma:

- Bits 7 a 10 - Identificador de função (de 0h até Fh)
- Bits 0 a 6 - Identificador de dispositivo (de 01h até 8Fh)

Dessa forma pode-se observar que foram reservadas 16 possíveis funções, podendo ser conectados até 127 nós (o endereço 00h para dispositivo é reservado). Na Tabela3 pode-se observar a relação das funções definidas, seus respectivos códigos e as direções do fluxo de dados.

Tabela 3: Funções das mensagens, seus respectivos códigos e direções. Quanto menor o valor do código maior a prioridade da função.

Código da Função	Descrição da Função	Direção da Mensagem*1
0 0 0 0	Reservado	-
0 0 0 1	BOOT	UP
0 0 1 0	BOOT ACK	DOWN
0 0 1 1	RESET	DOWN
0 1 0 0	NET ERROR	UP
0 1 0 1	COMMANDS	UP / DOWN
0 1 1 0	DATA 0	UP / DOWN
0 1 1 1	DATA 1	UP / DOWN
1 0 0 0	DATA 2	UP / DOWN
1 0 0 1	DATA 3	UP / DOWN
1 0 1 0	DATA 4	UP / DOWN
1 0 1 1	DATA 5	UP / DOWN
1 1 0 0	DATA 6	UP / DOWN
1 1 0 1	DATA 7	UP / DOWN
1 1 1 0	HEARTBEAT DOWN	DOWN
1 1 1 1	HEARTBEAT UP	UP

*1 - Direção do ponto de vista do dispositivo

As funções descritas na Tab.(3) são resultado da especificação e análise de requisitos do sistema embarcado. As mensagens de BOOT e BOOT ACK são utilizadas na fase de inicialização do sistema. A mensagem de RESET é utilizada pelo nó mestre para tentar recuperar um dispositivo de uma falha ativando remotamente a reinicialização do dispositivo. A mensagem NET ERROR é enviada por um nó escravo que percebe alguma irregularidade na rede. Essa mensagem não tem como objetivo inicial a indicação do erro, apenas a sinalização. A mensagem COMMANDS é utilizada para o envio e a recepção de comandos remotos de caráter genérico, determinados na fase de desenvolvimento pelo dispositivos que devem se comunicar. As mensagens DATA são utilizadas para a recepção e o envio de dados. Por fim as mensagens de HEARTBEAT são utilizadas pelos nós para monitorar o estado dos outros nós, tendo uma função essencial no combate a falhas de dispositivos.

A parte do identificador de mensagens relativa à identificação dos nós (bits 0 a 6) pode ser preenchida de maneira arbitrária dentro do espaço de endereçamento especificado (01h a 80h). A atribuição de identificadores de nós é feita de

maneira estática, isto é, o servidor e o nó devem ser configurados previamente para que seja possível a integração correta do dispositivo ao sistema. Da mesma forma se comportam as mensagens de controle no nível da aplicação trocadas entre eles.

6 . Conclusões

Este artigo mostrou a viabilidade da aplicação da rede CAN como barramento de campo de um dirigível autônomo não tripulado. Esta viabilidade pode ser constatada através de estimativa do tráfego gerado tanto pela configuração atual -uma configuração híbrida- como pela futura configuração, onde todos os sensores atuais possuem interface CAN. Também são apresentados os resultados práticos da utilização desta solução num voo real do dirigível. Para concluir apresenta-se uma proposta de um protocolo proprietário simples de alto nível, adequado ao caso do dirigível.

7 . Agradecimentos

Este trabalho teve o financiamento da FAPESP (processos nº 00/01000-4, 97/13384-7 e 99/04645-7) e programa PIBIC do CNPq.

8 . Referências

- Azinheira, J. R.; Oliveira, H. V.; Rocha, B. F. - "Sistema de Medições Aerodinâmicas para um Dirigível do Projeto AURORA" - Relatório de Calibração e Manual do Utilizador. Instituto de Engenharia Mecânica, Instituto Superior Técnico, Lisboa, Portugal.
- C515C , 8-Bit CMOS Microcontroller – Data Sheet 12.97 - Siemens
- CAN application Layer for Industrial Applications, CiA DS 201-207, Version 1.1, 1996
- CAN4EPP – Controller Area Network interface for Enhanced Parallel Port – Zanthic Technologies Inc. User’s Manual. Jan 4, 1999.
- CANopen, Communication Profile for Industrial System based on CAN.
- DeviceNet Specifications, release 2.0 1997, Vol I:Communication Model and protocol, Vol II: Device Profiles and Object Library.
- Elfes, A.; Bueno, S.S.; Bergerman, M.; Ramos, J.J.G. “A semi-autonomous robotic airship for environmental monitoring missions.” IEEE International Conference on Robotics and Automation, Leuven, Bélgica, Maio de 1998, pp. 3449-3455.
- Gomes, S.B.V.; Ramos, J.J.G. “Airship dynamic modeling for autonomous operation.” IEEE International Conference on Robotics and Automation, Leuven, Belgium, Maio de 1998, pp. 3462-3467.
- Lars-Berno, Fredriksson - A CAN Kingdom Ver 3.01
- miniMODUL-515C – Phytex. Hardware-Manual. January 1999
- Ramos, J.J.G.; Bueno, S.S.; Maeta, S. M.; Nascimento, L. G.; Bergerman, M.; Elfes, A.; Asanuma, K.; Paiva, E.; Beiral, J. A. “Project AURORA: Autonomous Unmanned Remote Monitoring Robotic Airship.” 2nd International Airship Convention and Exhibition, Bedford, Inglaterra, Junho de 1998.
- Robert Bosch GmbH. CAN Specification Version 2.0, September 1991.
- Victor Yodaiken and Michael Barabanov – I Real Time Linux Workshop, December 1999. Vienna, Austria.

A CONTROLLER AREA NETWORK FIELD BUS FOR AN UNMANNED ROBOTIC AIRSHIP

Faria, Bruno Guedes *

* Nacional Information Technology Institute – ITI . Robotics and Vision Laboratory – LRV - Rod. D. Pedro Km 143,6
- Caixa Postal 6162 - CEP 13089-500 Campinas, SP – Brasil

[Bruno.Faria@iti.br](mailto: Bruno.Faria@iti.br)

Maeta , Silvio Mano*

[Silvio.Maeta@iti.br](mailto: Silvio.Maeta@iti.br)

Ramos, Josué Jr. Guimarães*

[Josué.Ramos@iti.br](mailto: Josué.Ramos@iti.br)

Bergerman, Marcel*

[Marcel.Bergerman@iti.br](mailto: Marcel.Bergerman@iti.br)

Bueno, Samuel Siqueira*

[Josué.Ramos@iti.br](mailto: Josué.Ramos@iti.br)

Abstract. *There is a growing interest in the robotics community in the development of autonomous unmanned aerial vehicles for the execution of tasks such as inspection, exploration, surveillance, transportation, and demonstration of novel technologies. In Brazil, in particular, such vehicles may be utilized for environmental, biodiversity, and climate research and monitoring. It's in this context that the authors develop Project AURORA (Autonomous Unmanned Remote mOnitoring Robotic Airship), which aims at the deployment of low-speed, low-altitude airships geared mostly towards aerial inspection for environmental purposes. This article deals specifically with the sensor/actuator field bus architecture designed and developed within Project AURORA, as part of the onboard airship hardware. The architecture is based on the Controller Area Network (CAN) bus, largely utilized in the automotive industry. CAN was selected over serial communication due to the aspects of number of components that can be connected at any one time, transmission speed, ease of reconfigurability, and fault tolerance. The CAN solution developed is presented in detail, including the chipset selection and custom sensor/actuator connection hardware developed in-house. The article also presents experimental results from the utilization of the CAN bus on the measurement of actual sensor data, which is fundamental for automatic control and navigation purposes.*

Keywords. *Controller Area Network, CAN, Airship, Robotics, Fieldbus.*

Controle de um Manipulador Flexível Utilizando uma Rede Neural Artificial para Identificação e Compensação de Atritos não Lineares

Sebastião Cícero Pinheiro Gomes

Fundação Universidade Federal do Rio Grande
Av. Itália, Km 8,
96201-900, Rio Grande, RS
dmtscpg@super.furg.br

Vitor Irigon Gervini

Fundação Universidade Federal do Rio Grande
ec2vig@super.furg.br

Celiane Costa Machado

Fundação Universidade Federal do Rio Grande
mmacm@super.furg.br

Resumo. *O desenvolvimento de controle ativo para manipuladores flexíveis permanece um problema em aberto devido à existência de atritos não lineares inerentes aos atuadores. Neste artigo propõe-se um mecanismo de compensação de atritos para ser utilizado em paralelo a uma lei de controle linear. Idealizou-se uma estratégia de treinamento e uma estrutura de rede neural artificial (RNA) para aprender o torque de atrito de um atuador robótico do tipo moto-reductor. Para treinar a rede foi utilizado um modelo de atrito proposto na literatura (publicado em 1995). Considerou-se o torque motor e a velocidade angular do rotor como entrada na RNA, sendo o torque de atrito a única saída, a qual foi utilizada na proposição do referido mecanismo de compensação. Os resultados atestaram a eficiência da estimação e compensação neural de atrito, tendo-se obtido um bom desempenho no controle do manipulador flexível.*

Palavras chave. Rede neural artificial, manipulador flexível, controle, atrito, compensação.

1. Introdução

Controlar robôs com elos flexíveis permanece um problema em aberto em razão da existência de atritos não lineares inerentes aos atuadores robóticos. O que acontece na realidade é o seguinte: os torques calculados por leis de controle são efetivamente aplicados nos rotores dos atuadores, mas não passam integralmente para a saída dos redutores e portanto, não são transmitidos à estrutura para a devida atenuação das vibrações. Diversos trabalhos têm surgido nos últimos vinte anos sobre o controle de manipuladores flexíveis. Porém, a grande maioria aborda o problema do ponto de vista teórico de síntese de controle, considerando portanto que os atuadores são ideais e perfeitamente lineares. Em (Machado, 1999) há uma interessante descrição do estado da arte sobre o controle de estruturas flexíveis. No presente artigo, descreve-se o resultado de uma pesquisa realizada com o objetivo de identificar os atritos não lineares e compensá-los via *software*, na própria lei de controle, utilizando-se para tanto, uma rede neural artificial (RNA).

Existem atualmente diversas aplicações de redes neurais artificiais (RNA), nos mais variados domínios da ciência e tecnologia (Kaynak and Ertugru, 1997; Jung and Hsia, 1998). Trata-se de um assunto que tem merecido grande atenção por parte da comunidade científica. A principal motivação para se treinar *off line* uma RNA e posteriormente rejeitar os atritos não lineares estimados pela própria RNA, reside na dificuldade de se encontrar um modelo matemático muito realista e preditivo para reproduzir os atritos não lineares internos aos atuadores (Armstrong, 1988). Esta aplicação (identificação de atrito via RNA) começou a ser mais estudada apenas recentemente (Dapper and Zanh, 1999; Selmic and Lewis, 2000), tratando-se, portanto, de um assunto bastante atual. Uma vez identificado o atrito não linear, propõe-se um mecanismo de compensação deste a fim de linearizar o atuador. A estratégia de identificação e a proposição do mecanismo de compensação de atritos, o qual é utilizado em paralelo à lei de controle do manipulador flexível, constituem a principal contribuição deste trabalho.

Como não se dispõe, até o presente momento, de um atuador robótico isolado em uma bancada a fim de se realizar experimentos, utilizou-se um modelo proposto na literatura (Gomes, 95). Os resultados deste modelo serviram de referência como padrões de treinamento da RNA para a estimação do torque de atrito não linear. Como o modelo utilizado foi muito testado experimentalmente, acredita-se que todo o formalismo de identificação e compensação proposto deva proporcionar bons resultados na prática.

2. O Atuador Robótico

Um atuador robótico do tipo moto-redutor pode ser visualizado como um elemento transmissor de movimento contendo uma elasticidade interna de constante K , conforme representado na Fig. (1). O torque motor T_m é aplicado ao rotor de inércia I_r . Atritos não lineares estão sempre presentes neste tipo de dinâmica, fazendo com que o torque motor não seja transmitido integralmente à inércia I_s acoplada à saída do redutor. As equações (1) descrevem a dinâmica deste sistema (Gomes and Chrétien, 1992). T_{at} corresponde ao torque de atrito não linear, o qual será obtido via RNA. θ_r e θ_s são, respectivamente, os ângulos do rotor e do eixo de saída do redutor.

$$\begin{aligned} I_r \ddot{\theta}_r + \frac{K}{n^2}(\theta_r - n\theta_s) &= T_m - T_{at} \\ I_s \ddot{\theta}_s - K\left(\frac{\theta_r}{n} - \theta_s\right) &= 0 \end{aligned} \tag{1}$$

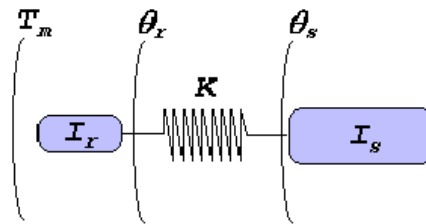


Figura 1. Representação de um atuador robótico do tipo moto-redutor.

O modelo para o torque de atrito utilizado foi publicado originalmente em Gomes (1995), tendo apresentado bons resultados quando feitas comparações entre experimentos e simulações. Este modelo reproduziu bem o fenômeno de *stick-slip*, notável por aparecer com frequência na prática. Este fenômeno foi definido em Gomes (1995) como tendo a seguinte origem: quando o torque resultante através do atuador está menor do que o atrito seco e ainda, a velocidade do rotor menor do que um certo valor limite que define a zona de *stiction* (intervalo de velocidade em torno do zero no qual ocorre o fenômeno), então a velocidade rotor vai a zero e assim permanece até que o torque resultante volte a ser superior ao atrito estático. Conforme será visto na sequência deste trabalho, o modelo adotado reproduziu este fenômeno e a RNA o aprendeu bem.

3. O Mecanismo de Compensação

Apesar de ter se mostrado bastante eficiente, o mecanismo de compensação de atrito proposto no presente trabalho é simples, consistindo na rejeição direta do atrito não linear identificado pela RNA. Simulou-se o caso mais real de controle discreto, com um passo de discretização de 0.001 segundos. O torque motor no instante $k+1$ assume a forma:

$$T_m(k+1) = T_c(k) + \hat{T}_{at}(k) \tag{2}$$

sendo $T_c(k)$ o torque de controle, o qual em malha fechada depende da observação do sistema no instante k . $\hat{T}_{at}(k)$ é o atrito não linear estimado pela RNA (saída), tendo como entrada nesta a velocidade rotor $\dot{\theta}_r(k)$ e o torque motor $T_m(k)$.

4. Arquitetura da RNA

A arquitetura de rede adotada para a identificação do torque de atrito foi a *perceptrons* de múltiplas camadas (MLP, *multilayer perceptron*) (Beale and Jackson, 1990), utilizando-se como algoritmo de treinamento o *backpropagation* com *momentum* (Fausett, 1994). A camada de entrada foi composta de dois neurônios (torque motor e velocidade do rotor) e a camada de saída por um neurônio (torque de atrito).

Foi utilizada somente uma camada intermediária, tendo sido necessários apenas quatro neurônios nesta para aprender os padrões apresentados à rede. Foi testado um número maior de neurônios (8, 10, 16, 40 e outros), na camada intermediária, mas os resultados foram muito similares ao obtido com apenas 4 neurônios. Já que a arquitetura da RNA é muito pequena, isto a torna rápida e eficiente, com um alto poder de generalização, permitindo assim uma compensação *on line* do torque de atrito.

Na Fig. (2) abaixo é mostrada a arquitetura da rede utilizada:

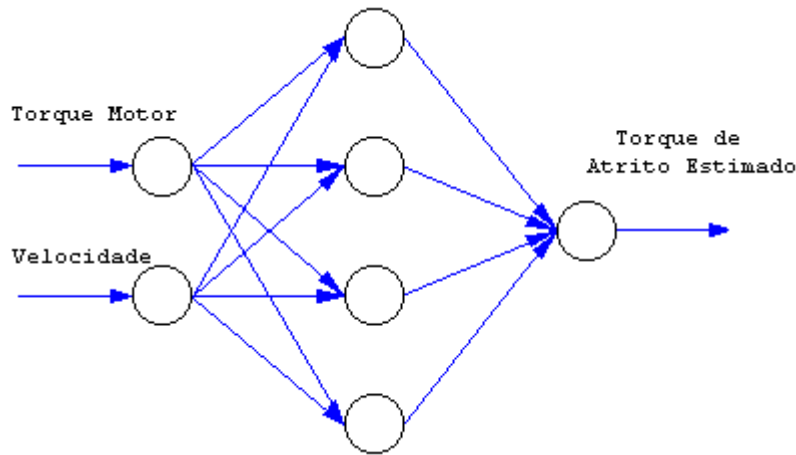


Figura 2. Arquitetura da RNA.

A arquitetura dos neurônios é mostrada na Fig. (3), tendo-se utilizado uma função de ativação sigmoidal (tanh). Após o treinamento, a rede apresentou uma taxa de aprendizado de 98% dos padrões apresentados.

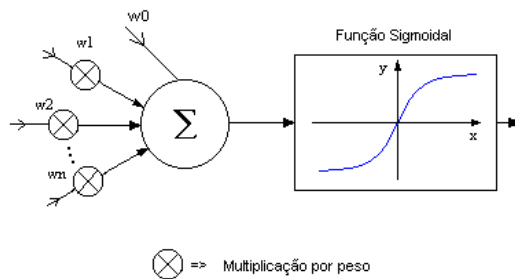


Figura 3. Arquitetura dos neurônios utilizada.

Na Fig. (4) mostram-se o torque motor e a velocidade rotor (entrada da rede), o torque de atrito do modelo (vermelho) e o torque de atrito reconhecido pela rede (preto). Pode-se perceber então que de fato, a taxa de aprendizado foi elevada. Outro dado interessante é que se escolheu o torque motor de entrada formado a partir de seguimentos de senoides e degraus em malha aberta, além de alguns resultados em malha fechada obtidos com um controle proporcional e derivativo (últimos seguimentos em torque motor da Fig. (4)). Esta diversificação do torque na estratégia de treinamento foi importante para a generalização da RNA.

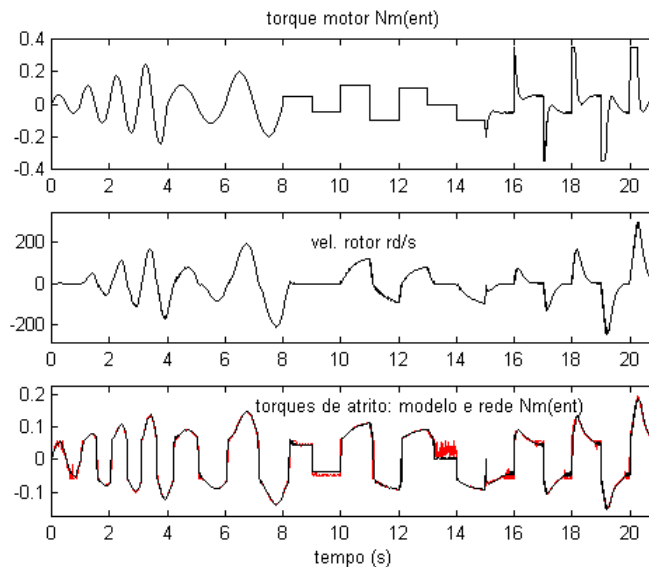


Figura 4. Padrões de treinamento da Rede.

No presente artigo, a RNA aprendeu o torque de atrito proveniente de um modelo, conforme já explicado anteriormente. Na prática, este torque de atrito pode ser identificado experimentalmente. A mesma seqüência de torques motor presente na Fig. 4 pode ser aplicada ao atuador, registrando-se os dados da velocidade rotor, a qual deve ser filtrada *off line* a fim de eliminar ruídos de alta freqüência. A aceleração do rotor $\ddot{\theta}_r$ é obtida derivando-se no tempo a velocidade filtrada. Considerando-se uma aproximação rígida para o atuador, o torque de atrito poderia então ser identificado experimentalmente a partir da seguinte equação:

$$\left(I_r + \frac{I_s}{n^2} \right) \ddot{\theta}_r = T_m - T_{at} \tag{3}$$

A maioria dos atuadores do tipo moto-redutores possui apenas um sensor de posição ou velocidade solidário geralmente ao rotor do atuador, sendo esta a razão da utilização da equação (3) para a identificação experimental do torque de atrito. Caso se dispusesse também de um sensor na saída do redutor de velocidades, a equação (1) poderia ser utilizada. Entretanto, os moto-redutores em geral possuem rigidez alta o suficiente para permitirem a utilização da equação (3) como uma boa aproximação para efeito da identificação experimental do torque de atrito.

5. Modelo do Manipulador Flexível Utilizando o Formalismo Discreto

O procedimento de modelagem descrito a seguir é denominado formalismo discreto (*lumped mass approach*), o qual representa a flexibilidade contínua por uma aproximação discreta, a partir da introdução de articulações fictícias (Pereira, 1999). Utilizou-se o processo de modelagem proposto por (Machado *et al.*, 2000), o qual consiste num aprimoramento do formalismo discreto objetivando obter um modelo com modos de vibração possuindo as mesmas freqüências previstas analiticamente. A flexibilidade contínua do elo é imaginariamente substituída por uma discreta, a partir da introdução de articulações elásticas fictícias. Considerou-se três destas articulações, conforme representado na Fig. (5). Como o modelo do atuador também possui uma elasticidade interna, a dinâmica global (atuador mais estrutura) terá quatro modos flexíveis. Em Machado *et al.* (2000), há a descrição detalhada de um algoritmo que, conhecendo-se as freqüências analíticas, permite que se obtenha o valor de cada constante elástica das articulações fictícias, as distâncias entre elas, devolvendo ao final as matrizes do modelo dinâmico escrito na forma:

$$[I]\ddot{\theta} + [C_{at}]\dot{\theta} + [K_{el}]\theta = BT_m \tag{4}$$

onde $\theta = [\theta_r \ \theta_s \ \theta_1 \ \theta_2 \ \theta_3]^T$ é o vetor com as posições angulares, sendo θ_r e θ_s relativas ao rotor e à saída do redutor respectivamente, enquanto $\theta_i, i = 1$ a 3 correspondem aos ângulos nas articulações fictícias. $[I], [C_{at}]$ e $[K_{el}]$ são as matrizes de inércia, atritos e constantes elásticas, sendo $B = [1 \ 0 \ 0 \ 0 \ 0]^T$ e T_m o torque motor de controle. As freqüências analíticas são obtidas a partir do conhecimento da massa da extremidade, das dimensões, massa específica e o módulo de elasticidade do elo (Pereira, 1999).

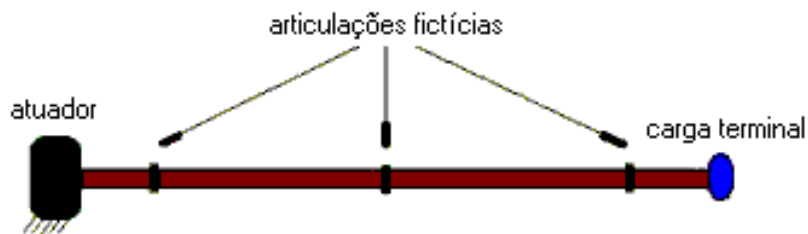


Figura 5. Representação do manipulador com um único elo flexível, considerando-se três articulações fictícias.

Os parâmetros adotados no modelo estão especificados na tabela seguinte, englobando tanto parâmetros relativos ao atuador quanto à estrutura flexível. As terminações (*ent*) e (*sai*) designam valores expressos na entrada ou saída do redutor de velocidades.

Tabela 1. Parâmetros adotados no modelo dinâmico.

Parâmetros relativos ao atuador	Parâmetros relativos ao elo flexível
$K = 3400 \frac{Nm}{rd} (ent) \Rightarrow$ elasticidade interna do atuador;	$l = 1.61m \Rightarrow$ comprimento do elo flexível;
$I_r = 5.87e - 5Kgm^2 (ent) \Rightarrow$ inércia do rotor;	$m_e = 2.1735Kg \Rightarrow$ massa do elo;
$I_s = 0.9Kgm^2 (sai) \Rightarrow$ inércia acoplada à saída do redutor, para os testes com a carga rígida utilizados no aprendizado da RNA;	$m_c = 6.1Kg \Rightarrow$ massa da carga terminal;
$c_s = 0.057Nm(ent) \Rightarrow$ atrito seco, aproximadamente igual ao estático;	$EI = 77Nm^2 \Rightarrow$ módulo de elasticidade vezes a inércia da seção reta do elo;
$c_v = 0.0004 \frac{Nms}{rd} (ent) \Rightarrow$ coeficiente de atrito viscoso; $n = 80 \Rightarrow$ índice de redução	<p>freqüências analíticas</p> $\left. \begin{aligned} w_1 &= 17.64rd / s \\ w_2 &= 48.71rd / s \\ w_3 &= 135.8rd / s \end{aligned} \right\} \Rightarrow \text{modos estruturais}$ $w_4 = 1019.5rd / s \} \Rightarrow \text{modo rotor}$

6. Resultados de Simulações

Caso o atuador tivesse um comportamento linear, não haveriam os atritos estático e seco (*dry friction*), mas apenas o atrito viscoso, proporcional à velocidade angular do rotor. O modelo dinâmico global (atuador mais estrutura flexível) da equação (3), pode então ser posto na forma de estado $\dot{\vec{X}} = F\vec{X} + GT_m$, onde \vec{X} é o vetor de estado formado pelas coordenadas de posição e velocidade angulares. Considerando-se os parâmetros do atuador e da estrutura adotados (Tabela 1), os pólos da dinâmica em malha aberta (autovalores de F) são:

$$0 \pm 0i \quad -0.7 \pm 1019.5i \quad -0.43 \pm 135.8i \quad -1.401 \pm 48.71i \quad -1.738 \pm 17.645i .$$

O par de pólos de mais alta freqüência corresponde na realidade ao modo rotor, ou seja, trata-se da contribuição da elasticidade interna do atuador. Para essa dinâmica linear por aproximação em malha aberta, poder-se-ia, a princípio, projetar qualquer controle linear, como é o caso dos controladores baseados na estrutura LQG (Linear Quadrático Gaussiano). Porém, como o principal objetivo do presente trabalho é verificar se um compensador neural de atritos pode ser eficiente na compensação dos atritos não lineares do atuador, projetou-se um controlador simples, por alocação de pólos, o qual utiliza um retorno de estado: $T_m = -K_c \vec{X}$. Determinou-se então K_c (Franklin *at al.*, 1995), a fim de que os pólos em malha fechada tenham os seguintes valores:

$$-0.75 \pm 1019.7i \quad -6.07 \pm 135.8i \quad -4.64 \pm 96.88i \quad -2.87 \pm 24.036i \quad -4 \pm 0.7i .$$

Todas as simulações foram realizadas com a rotina ode45 do MATLAB, tendo-se programado de forma a se trabalhar com 0.001s como passo de discretização do controle.

A Fig. (6) apresenta os resultados de duas simulações. Os gráficos da esquerda correspondem à simulação totalmente em malha aberta. Aplicou-se, nos primeiros 0.1s, um torque motor de 0.08Nm, de -0.08Nm entre 0.1 e 0.2s, sendo nulo o torque motor no restante do tempo. Simulação semelhante é mostrada nos dois gráficos da direita, com a diferença que, após 0.4s, aplicou-se a lei de controle na forma: $T_m(t_{k+1}) = -K_c \vec{X}(t_k)$, sendo $t_{k+1} - t_k$ equivalente ao passo de discretização de 0.001s. As retas horizontais visíveis no gráfico do torque motor com o controle correspondem aos níveis de atrito seco, o qual possui um valor ligeiramente menor do que o atrito estático. Estes resultados da Fig. (6) revelam um dos maiores problemas do controle de estruturas flexíveis a partir de atuadores não lineares, ou seja, o torque motor de controle necessário à atenuação das vibrações é, na maior parte do tempo, menor em módulo do que o atrito estático ou seco por aproximação. Este torque de controle permanece na conhecida zona morta em torque, não passando portanto efetivamente para a estrutura, a qual permanece vibrando como se fora uma viga, engastada em uma extremidade (a velocidade angular do rotor, apesar de não ter sido mostrada no gráfico, é nula quando o torque motor está na zona morta) e livre na outra.

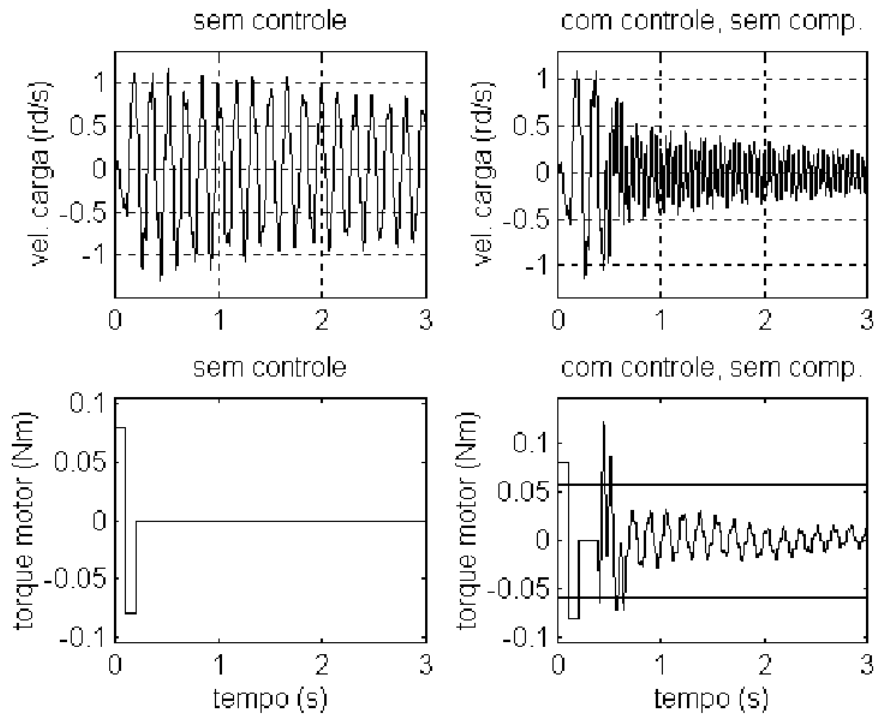


Figura 6. Simulações com e sem controle e em ambos os casos, a dinâmica do atuador é não linear e não há compensação neural de atritos.

A Fig. (7) contém os resultados de duas simulações, ambas com a aplicação do controle. Os gráficos da esquerda foram obtidos considerando-se o atuador linear e o controle sem a compensação neural de atrito. Analisando-se a velocidade da carga, nota-se que as vibrações foram atenuadas em aproximadamente 2s. Os dois gráficos da direita da Fig. (7) são resultados de uma simulação considerando-se o atuador não linear, sendo que foi utilizado o controle com a compensação neural de atrito especificada na equação (2). Pode-se perceber que o controlador, com a compensação neural de atrito, teve um excelente desempenho uma vez que a atenuação das vibrações foi semelhante ao caso linear. Observa-se ainda que a compensação de atrito tende a fazer com que o torque motor tenha atuação fora da zona morta em torque (nos limites das retas horizontais do gráfico do torque motor). Deve ser observado que a mesma ativação dos modos em malha aberta da Fig. (6) também esteve presente na Fig. (7) e que, apesar do controle em malha fechada ser ativado após 0.4s, a compensação de atrito foi acionada desde o início (mesmo na fase de malha aberta), como é visível no gráfico do torque motor da direita da Fig. (7).

O gráfico presente na Fig. (8) clarifica a razão do bom desempenho do controle com a compensação neural de atrito. Nele vê-se o torque de atrito do modelo, sobreposto ao torque de atrito estimado pela RNA (em vermelho). Percebe-se que a RNA efetuou uma boa estimativa do torque de atrito e portanto, como o mecanismo de compensação utiliza esta estimativa, o mesmo possibilitou um bom desempenho ao controlador, conforme já verificado nos gráficos da direita da Fig. (7).

8. Conclusões

No presente artigo propôs-se uma solução para o principal problema imposto ao controle de estruturas flexíveis, ou seja, atenuar as vibrações utilizando um atuador do tipo moto-reductor, o qual possui atritos não lineares que inviabilizam o controle uma vez que o torque motor calculado pela lei de controle encontra-se dentro da zona morta em torque, delimitada pelos atritos estáticos nos sentidos positivo e negativo. Existem duas possíveis alternativas para contornar esse problema: uma é melhorar o atuador do ponto de vista mecânico, trabalhando-se quando possível, sem reductor de velocidade (*direct-drive*) e reduzindo assim a amplitude em módulo da zona morta em torque; outra é tentar resolver o problema via *software*, na própria lei de controle. A primeira das alternativas parece inviável quando a estrutura a ser controlada tem uma inércia importante, impossibilitando assim o uso de *direct-drive*. A segunda alternativa tem sido desprezada pela comunidade que pesquisa este assunto por uma razão simples: tenta-se compensar o atrito não linear utilizando-se modelos baseados no modelo de Coulomb, o qual preconiza de forma simplificada que o atrito não linear é uma constante multiplicada pelo sinal da velocidade angular do rotor. Este mecanismo de compensação, além de impreciso, freqüentemente acarreta variações de alta freqüência no torque motor (*shattering*) em razão de comutações do sinal da velocidade rotor, para valores de velocidade próximos do zero. O mecanismo de compensação de atrito proposto no presente trabalho, o qual estima o torque de atrito *on line* via RNA, resolveu bem este problema. É importante lembrar que, para o aprendizado da rede, considerou-se uma inércia rígida acoplada à saída do reductor, embora a rede já treinada tenha sido utilizada para compensar o torque de atrito não linear,

trabalhando-se com uma inércia flexível acoplada à saída do redutor. Outro fato significativo é que a RNA proposta e sua estratégia de treinamento, possibilitaram uma estrutura simples e com um alto poder de generalização, sendo que esta simplicidade da rede significa um pequeno custo computacional numa estratégia de compensação em tempo real. O fato do compensador neural ter evitado o *shattering* constitui uma importante vantagem com relação à compensação de atritos baseada em modelos. A próxima etapa da pesquisa será a construção de um manipulador com um elo flexível, a fim de testar experimentalmente os presentes resultados obtidos em simulação. Será também estudada a possibilidade de adaptação *on line* da rede, com o objetivo de identificar possíveis variações no torque de atrito em razão de desgastes com o tempo e variações nas condições de trabalho do manipulador.

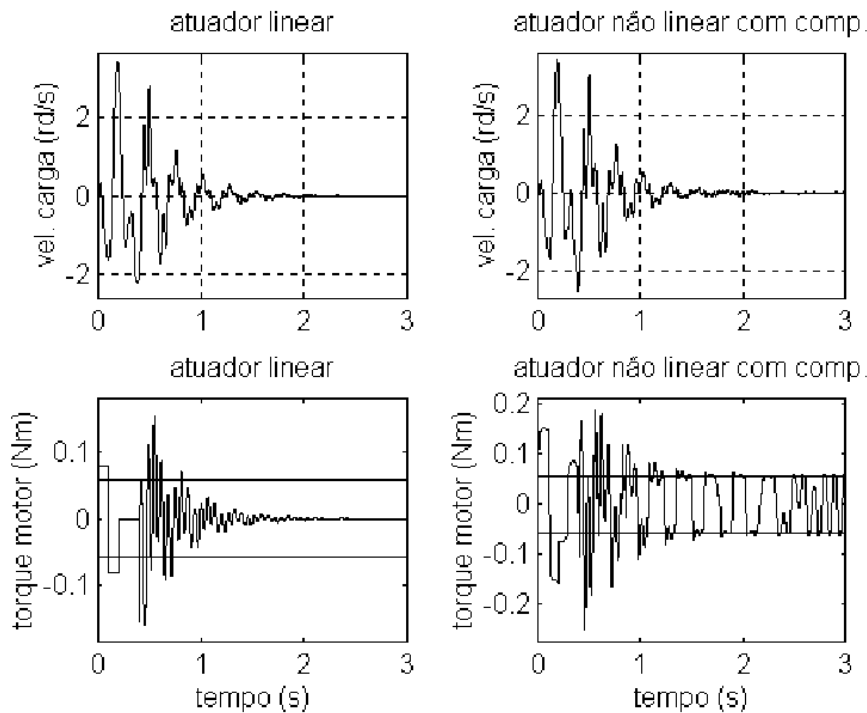


Figura 7. Simulações com a utilização do controle nos casos: atuador linear e controle sem compensação de atrito (gráficos da esquerda); atuador não linear e controle com compensação de atritos (gráficos da direita).

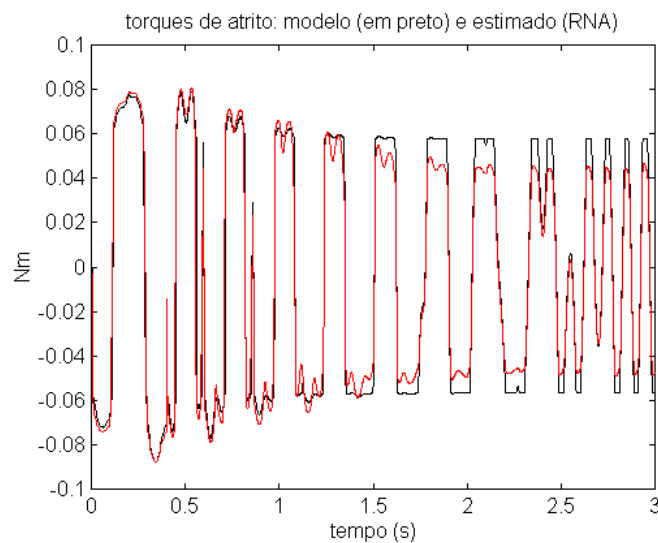


Figura 8. Torques de atrito do modelo e estimado pela RNA (em vermelho), resultados colhidos da simulação que gerou os gráficos da direita da Fig. (7) .

9. Referências

- Armstrong, B. S. R., 1988, "Dynamics for robot control: friction modeling and ensuring excitation during parameter identification", PhD thesis, Stanford University.
- Beale, R. and Jackson, T., 1990, "Neural computing: an introduction", Adam Higler Bristol.
- Dapper, M., Zanh, V., Maass, R. and Ekmiller, R., 1999, "How to compensate stick-slip friction in neural velocity force control (NVFC) for industrial manipulators", In IEEE Robotic and Automation Conference, Detroit, USA, May.
- Fausett, L., 1994, "Fundamentals of neural networks", Prentice Hall, New Jersey.
- Franklin, G. F., Powell, J. D. and Naeni, A. E., 1995, "Feedback control of dynamic systems", Addison-Wesley, USA.
- Gomes, S. C. P. and Chrétien, J. P., 1992, "Dynamic modeling and friction compensated control of a robot manipulator joint", In IEEE Robotic and Automation Conference, Nice, France, May.
- Gomes, S. C. P., 1995, "Modelagem de atritos internos às articulações de robôs manipuladores", In XIII COBEM, CD-ROM, Belo Horizonte.
- Jung, S. Hsia, T. C., 1998, "Analysis of non linear neural network impedance force control for robot manipulator". In IEEE Robotic and Automation Conference, Leuven, Belgium, May.
- Kaynak, O. and Ertugru, M., 1997, "Neural network adaptive Sliding Mode Control and its application to SCARA type robot manipulator". In IEEE Robotic and Automation Conference, Albuquerque, New Mexico, USA, April.
- Machado, C. C., 1999, "Um Estudo sobre Controle de Estruturas Flexíveis Ativadas por Escoamento de Fluidos", *Dissertação de Mestrado*, UFRGS, Porto Alegre, Brasil.
- Machado, C. C., Pereira, A. E. L., Gomes, S. C. P. e De Bortoli, A. L., 2000, "Um novo algoritmo para a modelagem dinâmica de manipuladores flexíveis", *Anais do CBA-2000*, Florianópolis, SC.
- Miller III, W.T., Sutton, R.S., Werbos, P.J., 1995, "Neural networks for control". MIT Press.
- Pereira, A. E. L., 1999, "Um Estudo Sobre Modelagem Matemática de Estruturas Flexíveis", *Dissertação de Mestrado*, UFRGS, Porto Alegre, Brasil.
- Selmic, R. R. and Lewis, F. L., 2000, "Dead zone compensation in motion control systems using Neural Networks", *IEEE Transactions on Automatic Control*, Vol 45, April.

Title: Control of a flexible manipulator using a neural network for non linear friction identification and compensation.

Abstract. Active control law for flexible manipulators is still an open problem due to non-linear friction in the actuators. In this work we propose a friction compensation mechanism to use in parallel with a linear control law. So, we present a training strategy and propose a structure of Neural Network (NN) to learn the friction torque of a motor drive joint robotic actuator with gear. To train the NN it was used a friction model proposed in the literature (published in 1995). It was considered the motor torque and the rotor angular velocity as input in the NN, while the friction torque was the only output, which was used in the proposition of a non-linear friction compensation mechanism. The results attested the efficiency of the NN friction estimate and compensation with the proposed mechanism, and good results were obtained in the control of the flexible manipulator.

Keywords. Neural Networks, flexible manipulator, control, friction, compensation.

10. Copyright Notice

The authors is the only responsible for the printed material included in his paper.

HIERARCHICAL SINGULARITY ANALYSIS OF AN ARTICULATED ROBOT

Daniel Martins**Raul Guenther****Henrique Simas**

Departamento de Engenharia Mecânica, Universidade Federal de Santa Catarina, Caixa Postal 476, 88040 900 - Florianópolis, SC, Brazil

E-mails: dmartins, guenther, simas@lcmi.ufsc.br

Abstract: Singularity avoidance is a current research topic. Singularities in general are analytically computed by equating the determinant of the Jacobian matrix to zero. However each singularity has a specific influence on the end-effector behaviour. This paper analyses the Jacobian matrix selected for a specific manipulator. The chosen manipulator is a non redundant six degree of freedom manipulator. It is a simplified version of the Roboturb manipulator that was specially designed for recovering turbine blades submitted to cavitation processes. The method is based on graph theory. Singularities are ranked according to the partial order of the associated graph. The results to the manipulator are discussed at the end of the paper.

Keywords: Jacobian matrix, Singularity avoidance, Robot Analysis Screw Theory; Graph Theory.

1. Introduction

Kinematic analysis derives the relative movements among various links of a given mechanism, for instance a robot manipulator. Typically the end-effector of a robot is programmed to follow a set of desired positions and orientations in space. The first derivatives (velocities) are often imposed as extra conditions on the path tracking. The *inverse kinematics problem* then is to find all sets of actuated joint variables and their corresponding time derivatives which will bring the end-effector to the desired motion characteristics.

The Jacobian matrix, Jacobian for short, is probably the most important matrix related to the kinematics of serial manipulators. Inverse kinematics problem can be solved using closed form expressions (a *global* solution) for position inverse kinematics. This approach has several drawbacks (Leahy Jr et al., 1987) like computational time and uncertainties when a singularity is reached. The inverse kinematics problem is generally solved *locally* using the Jacobian. The Jacobian also maps the movement of the end-effector in the Cartesian space to the joint space where robot control takes place. Several attempts have been made to improve the velocity of the Jacobian computation via parallelisation (Zomaya et al., 1999) or via a favorable choice of auxiliary variables (Leahy Jr et al., 1987).

The Jacobian $J = J(q)$ is a map between velocities at the actuators joints and at the end-effector. Each element of this matrix depends upon position parameters, q , of the manipulator. A singularity occurs whenever the Jacobian drops rank. Singularities are normally detected by the determinant of the Jacobian matrix. This determinant ($\det J$) is dependent upon geometrical and position parameters. Equation $\det J = 0$ is a necessary and sufficient condition for the Jacobian be rank deficient. Factoring $\det J$ as a product of minimal terms

$$\det J = \prod_{i=1}^m \Phi_i = 0 \quad (1)$$

Terms Φ_i group variables responsible for the i -th singularity. The order of these terms is arbitrary. Terms Φ_i are minimal in the sense that they cannot be splitted up as a product of smaller, non unity, terms. The manipulator has, in general, m different singularities.

A common understanding is that all singularities equally affect, for worst, the manipulator response. This perception possibly comes out from eq. (1). There the order of the products Φ_i is immaterial. Furthermore $\Phi_i = 0 \rightarrow \det J = 0$ no matter which i -th term is becoming zero. General singularity avoidance algorithms detect the proximity of a singularity but do not specify which *particular* singularity is being reached.

Following developments presented in (Martins and Guenther, 2001), this paper emphasizes that singularities are NOT equally important to the behaviour of the manipulator. The robot kinematic analysis method, proposed in (Martins and Guenther, 2001) is applied to a new manipulator. This method is based on the structure of the Jacobian matrix. The sparsity of this matrix will be enhanced and geometrically explored using screw theory. Afterwards graph theory techniques are applied to extract qualitative relationships among input and output variables. The Jacobian matrix

is permuted to a block triangular form and the diagonal blocks are used to generate a hierarchy of singularities. Singularities are ranked according to the control they exert on the kinematics of the robot. Each type of manipulator has its own hierarchy of singularities. The hierarchy is based exclusively on the Jacobian matrix. The Roboturb manipulator is here analysed by this method.

This paper outlines the hierarchy of singularities of a non redundant version of the Roboturb manipulator. The Roboturb manipulator was specifically designed for recovering turbine blades submitted to cavitation processes. For singularity avoidance it is redundant and has seven joints. The first joint is a prismatic kinematic pair that is a guide to will be considered fixed for this paper. The remaining six joints are revolutes and are responsible for the internal singularity, sometimes called shoulder and wrist singularities (Tsai, 1999) that are more cumbersome to avoid online. The simplified version of the Roboturb manipulator presented here embrace all internal singularities of the manipulator.

The organization of this paper is described here. Section 2 describes the Jacobian matrix and its relationships with a serial robot. The structure of the Denavit-Hartenberg Jacobian of the Roboturb manipulator is also presented. Section 2.1 describes the screw theory that aims at simplifying the Jacobian. Section 3 presents the digraph method to obtain the hierarchy of singularities by a combinatorial rearrangement of matrices. It culminates with Algorithm 1 that lists the steps to obtain a block triangular form of a sparse matrix. Section 4 describes the concept of kinematic structure matrix that overcomes the limitations of the digraph method. Section 5 applies the method to the Roboturb manipulator. Some comments are presented at section 6.

2. Theoretical background

Jacobians can be obtained using Denavit-Hartenberg parameters (Asada and Slotine, 1986)

$$\dot{x} = \mathcal{S}_e = \begin{bmatrix} \omega_e \\ v_e \end{bmatrix} = J\dot{q} \quad \text{where } J \triangleq \begin{bmatrix} J_{R_1} & \cdots & J_{R_n} \\ J_{L_1} & \cdots & J_{L_n} \end{bmatrix} \text{ and } \dot{q} \triangleq [\dot{q}_1 \dot{q}_2 \cdots \dot{q}_n]^T \quad (2)$$

where: ω_e is the angular velocity of the end-effector, v_e is the velocity of a point in the end-effector, J_{R_i} and J_{L_i} are 3×1 vectors corresponding to the rotational and linear effects of the i -th joint. Velocity \dot{q}_i is rotational or translational depending upon the joint type.

The Jacobian is generally sparse due to the particular geometrical configuration of the manipulator. The sparsity of a matrix can be presented as a structure matrix (Röbenack and Reinschke, 1997).

The Denavit-Hartenberg Jacobian of the Roboturb manipulator has the following structure

$$\langle J \rangle = \begin{bmatrix} \times & \times & \times & 0 & 0 & 0 \\ \times & \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & 0 & 0 & 0 \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ \times & 0 & 0 & \times & \times & \times \end{bmatrix} \quad (3)$$

where “0” means a *always null* element while “ \times ” means a structurally non-null element *i.e.* a term *generally* not zero. More precisely \times indicates a non null function of geometrical and position parameters of the robot. The symbol $\langle \cdot \rangle$ means the structure matrix of a given real matrix.

The lower right 3×3 submatrix of J is null and corresponds to the spherical wrist (Wang and Waldron, 1987). Only 14 of the 36 elements are structurally null in eq. (3). The coordinate system at the base turns the elements of this matrix relatively complex. Kinematic algorithms are simplified with adequately chosen auxiliary variables (Leahy Jr et al., 1987).

2.1. Screw Theory

Equation (3) has limited sparsity. Denavit-Hartenberg convention (Denavit and Hartenberg, 1955) imposes a unique coordinate system. Denavit-Hartenberg convention also creates some unnecessary rotation angles (Hunt, 1987). Graph theory techniques entail sparse matrices. The sparsity of the Jacobian matrices must be explored and enhanced as much as possible. Coordinate systems and position variables well chosen improve the sparsity of the Jacobian. Screw theory concepts based on rigid body movements (Ball, 1900) can lead to an appropriate choice of coordinate system.

Rigid body instantaneous movements can be “carried out by” a screw. In eq. (2) $\mathcal{S}_e (= \dot{x})$ is a non normalised screw representing the velocity state of the end-effector. The screw has an axis (line), a pitch and a magnitude in velocity dimensions (rad/s or m/s). The instantaneous movement of the end-effector \mathcal{S}_e is the sum of the instantaneous movements, \mathcal{S}_i , of the links composing the serial chain.

$$\dot{x} = \$_e = \sum_{i=1}^n \$_i = \sum_{i=1}^n \begin{bmatrix} J_{R_i} \\ J_{L_i} \end{bmatrix} \dot{q}_i = \sum_{i=1}^n \hat{\$_i} \dot{q}_i \quad (4)$$

Each column of the Jacobian in eq. (2) is a *normalised* screw $\hat{\$_i}$ of eq. (4) corresponding to the i -th joint. Details about this topic can be found in (Hunt, 1987) and references therein.

Screw theory has no pre-specified coordinate system. Denavit-Hartenberg convention, on the other hand, also imposes a unique choice of angles. The screw transformation matrix, here in ray order and from coordinate system i to coordinate system j , is

$$\mathbf{T}_{i \rightarrow j}^{ray} \triangleq \begin{bmatrix} R_{i \rightarrow j} & 0_{3 \times 3} \\ R_{i \rightarrow j} S & R_{i \rightarrow j} \end{bmatrix} \text{ where } S \triangleq \begin{bmatrix} 0 & -x & y \\ x & 0 & -z \\ -y & z & 0 \end{bmatrix} \quad (5)$$

Matrix $\mathbf{T}_{i \rightarrow j}^{ray}$ computes the Jacobian matrix in any coordinate system (Hunt, 1987) (Tischler et al., 2000). In eq. (5) $R_{i \rightarrow j}$ is the 3×3 rotation matrix from coordinate system i to coordinate system j , and (x, y, z) are the coordinates of the origin of the coordinate system j in the coordinate system i .

Some geometrical insight is required to select the coordinate system j to represent the kinematics of the manipuloan. A good choice of the coordinate system improves the sparsity of the Jacobian (Pissanetsky, 1984).

In the following section the graph approach to obtain the hierarchy of singularities by a combinatorial rearrangement of matrices is described. Firstly the graph theory concepts are shown. Sections 3.1 to 3.3 describe Boolean operations on the incidence matrices related to those graphs. Finally algorithm 1 lists the steps necessary to rearrange a matrix to block triangular form.

3. Digraph Approach applied to the Analysis of Jacobian Matrices

Jacobians are usually sparse matrices. The structure of a sparse matrix, also called structure matrix, is the relative position in the matrix of the null and non-null elements. The structure matrix can be modelled by directed graphs (Jantzen, 1996). The graphical representation of the structure matrix is also called the incidence matrix of the associated graph. Figure 1 shows a simple directed graph and its incidence matrix, eq. (7).

The incidence matrix $\bar{A} = [a_{ji}]$ is a graph theoretical concept. Their elements are defined by

$$a_{ji} \triangleq \begin{cases} 1 & \text{if there is an edge from vertex } i \text{ to vertex } j \\ 0 & \text{if, otherwise, there is no such edge.} \end{cases} \quad (6)$$

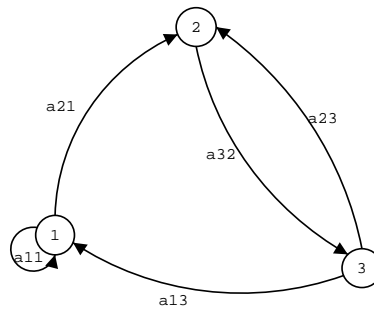


Figure 1: A simple directed graph (digraph)

The corresponding incidence matrix of the graph in Fig. 1 is

$$\bar{A} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \quad (7)$$

Boolean arithmetics satisfy the common properties of the natural numbers relative to sum (+ or \vee , logic OR) and product (\cdot or \wedge , logic AND). The most significant difference between Boolean addition from real numbers addition is $1 + 1 = 1$. It is however derived from the logic equation $1 \vee 1 = 1$. A *Boolean matrix* is a matrix whose elements

operate according to the Boolean rules. A *full Boolean matrix* is a Boolean matrix where all elements are “1”. The incidence matrix $\bar{A} = [a_{ji}]$, eq. (6), is a Boolean matrix but it is not full. Boolean matrices are here designated by Roman capital with an overbar like $\bar{A}, \bar{R}, \bar{T}, \dots$. A graph related to a incidence matrix A is designated by $G(A)$.

A specific labelling of vertices in $G(A)$ is irrelevant to the properties of the graph itself. Relabelling the vertices has no impact on the directed graph behaviour but produces a simultaneous row and column permutation on the corresponding incidence matrix. Such permutations can be used to reorder the matrix. A good reordering can lead to a block triangular form using the concept of strongly connected sets.

Strongly connected sets (or subgraphs) is a maximal set of vertices that can reach and be reached one another (Reinschke, 1988; Jantzen, 1996). Let $K(i) \subset G$ be, by definition, the strongly connected subgraph that contains vertex i . A vertex $i \in G$ in a strongly connected set $K(i) \subset G$ is the initial vertex of a path that can reach any other vertex $j \in K(i)$. To maintain a coherent notation $K(i) = K(j)$ if $i, j \in K(i)$.

3.1. Product of Boolean matrices

Let $\bar{F}_{m \times n} = \bar{D}_{m \times r} \bar{E}_{r \times n}$ be a Boolean matrix \bar{F} generated by the product of two generic Boolean matrices \bar{D} and \bar{E} . Each element of \bar{F} is a number 0 or 1 and is generated by the summation

$$\bar{F}_{i,j} = \sum_{k=1}^r \bar{D}_{i,k} \cdot \bar{E}_{k,j} \quad (8)$$

An element $\bar{F}_{i,j}$ is null only if *all* the terms in the summation are zero *i.e.*

$$\bar{D}_{i,k} \cdot \bar{E}_{k,j} = 0 \quad \forall k = 1, \dots, r \quad (9)$$

Equations (8) and (9) can be rewritten in logical (Boolean) terms as

$$\bar{F}_{i,j} = 0 \iff \bar{D}_{i,k} = 0 \vee \bar{E}_{k,j} = 0 \quad \forall k = 1, \dots, r \quad (10)$$

In other words: $\bar{F}_{i,j} = 1$ unless the internal product of row i of first matrix \bar{D} with column j of second matrix \bar{E} is null.

3.2. Matrix powers

An important product of two Boolean matrices is the matrix power *i.e.* when $\bar{D} = \bar{E}$ in eq. (8). Here both matrices represent the *same* digraph and then eq. (10) indicates that there is no path of length 2 from vertex j to vertex i in the graph represented by the incidence matrix \bar{D} ($= \bar{E}$).

An incidence matrix \bar{A} is the first power of itself: $\bar{A} = \bar{A}^1$. An “1” value in an element of the matrix \bar{A} imply the existence of a path of length 1. A simple example will illustrate this result.

Let $\bar{A}^2 = \bar{A} \cdot \bar{A}$ using the example of Figs. 1 and eq. (7)

$$\bar{A}^2 = \bar{A} \cdot \bar{A} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \quad (11)$$

There is only one null element on \bar{A}^2 , eq. (11), *i.e.* $a_{32}^{(2)} = 0$. The superscript “(2)” in $a_{32}^{(2)}$ indicates the power of matrix $\bar{A} = [a_{ij}]$. Starting from vertex 2 is impossible to reach vertex 3 with a path of length exactly 2. This is the only impossible path $i \rightarrow j$ with length 2 between any arbitrary pair of vertices (i, j) in the graph represented by Fig. 1. However there are paths $2 \rightarrow 3$ with lengths different from 2 as *e.g.* the single edge (length 1 path) $2 \rightarrow 3$.

Two useful lemmas related to matrix powers are ((Reinschke, 1988))

Lemma 1 Let B^k be a k -power of the incidence matrix B of a directed graph and $b_{ij}^{(k)}$ an element of B^k then $b_{ij}^{(k)} = 1$ only if there is a path with length k starting from vertex j and ending at vertex i in the graph represented by the incidence matrix B , otherwise $b_{ij}^{(k)} = 0$.

Lemma 2 Let B^k be a k -power of the incidence matrix $B_{n \times n}$ of a directed graph with n vertices. If $k \geq n$ then B^k will be a full matrix or will repeat cyclically the patterns of the lower powers B^1, B^2, \dots, B^{n-1} .

3.3. Related matrices $\bar{R}, \bar{S}, \bar{T}$

The reachability matrix \bar{R} expresses the capability of a vertex of the graph be accessed by any other vertex. Based on Lemma 1 \bar{R} can be computed by

$$\bar{R} = \sum_{i=1}^{n-1} \bar{A}^k \quad (12)$$

Lemma 2 justifies the use of powers up to $n - 1$. As matrix \bar{A} is Boolean eq. (12) has multiplication replaced by the Boolean **and** (\wedge) and sum replaced by the Boolean **or** (\vee). The same principle is valid for matrices \bar{S} and \bar{T} described below.

Matrix $\bar{R} = [r_{ij}]$ can also be defined element by element as

$$r_{ij} = \begin{cases} 1 & \text{if a path leads from vertex } j \text{ to vertex } i \\ 0 & \text{if, otherwise, vertex } i \text{ cannot be reached starting from vertex } j \end{cases} \quad (13)$$

Any vertex can be accessed by any other vertex in the graph if the reachability matrix is full. Let i and j be two vertices randomly chosen. If the reachability matrix is full then a path starting at vertex i and ending at vertex j can *always* be found. The interdependency of all variables is complete. In matricial terms this means that there is no simultaneous permutation of rows and columns that lead the matrix to a block triangular form. Singularities in this particular case cannot be ranked.

However if the reachability matrix has some zeros the block triangular form can *always* be reached. Two other Boolean matrices, \bar{S} and \bar{T} , are required for rearrange the original matrix \bar{A} in this case.

The *symmetrical structure Boolean matrix* \bar{S} searches strongly connected subgraphs from the graph represented by the incidence matrix \bar{A} .

$$\bar{S} \triangleq \bar{R} \wedge \bar{R}^T \quad (14)$$

i.e.

$$s_{ij} = r_{ij} \cdot r_{ji} \quad (15)$$

Matrix \bar{S} maps all strongly connected sets of vertices in a directed graph. A column (vertex) j of \bar{S} has “connections” with all vertices i whose elements $a_{ij} \neq 0, i = 1, n$. Strongly connected subgraphs $K(j)$ are formed by all vertices i such that $a_{ij} \neq 0$. Two subgraphs with the same cardinality (number of elements) are disjoint or congruent (*i.e.* are the same strongly connected subgraph).

The symmetrical structure matrix $\bar{S} = [s_{ij}]$ can be alternatively defined by

$$s_{ij} = \begin{cases} 1 & \text{if vertices } i \text{ and } j \text{ are strongly connected.} \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

The *asymmetrical structure Boolean matrix* \bar{T} differentiates strongly connected subgraphs one another.

$$\bar{T} \triangleq \bar{R} \wedge \sim \bar{R}^T \quad (17)$$

where the operator $\sim [\cdot]$ is the Boolean NOT applied to all elements of the Boolean matrix. Matrix $\sim \bar{R}$ is simply a change $0 \rightleftharpoons 1$ in all elements of the original matrix \bar{R} .

In other words

$$t_{ij} = r_{ij} \cdot \sim r_{ji} \quad (18)$$

Matrix $\bar{T} = [t_{ij}]$ can be similarly defined as

$$t_{ij} = \begin{cases} 1 & \text{if a path leads from vertex } j \text{ to vertex } i \\ & \text{but there is no return path } i \rightarrow j \\ 0 & \text{otherwise.} \end{cases} \quad (19)$$

This interpretation of matrix \bar{T} leads to the definition of the asymmetry vector $\vec{w} = [w_1, w_2, \dots, w_n]$ where

$$w_j \triangleq \sum_{i=1}^n t_{ij} \quad (20)$$

So w_j is the number of vertices k that can be reached *starting from* vertex j but no return path from k to j exists. Vector $\{w_j\}$ is formed by the sum of all elements in column j of matrix \bar{T} . This is NOT Boolean.

Trivially if two vertices i and j are of the same strongly connected subgraph then $w_i = w_j$. The reciprocal condition however is not true. If $w_i = w_j$, same degree of asymmetry, it is *not* guaranteed that they are on the same strongly connected subgraph. Sets of vertices with the same degree of asymmetry, $w_i = \text{const}$, can be possibly splitted up in smaller strongly connected subgraphs using the symmetrical structure matrix \bar{S} .

The elements of \vec{w} must be rearranged according to their magnitude (in crescent order) keeping record of the original indices. The vertices with the same value of w_j are arranged in crescent order. Inside these groups all vertices of a same strongly connected subgraph must be kept in sequence.

The rearranged vector $\tilde{w} = P\vec{w}$ is a permutation of the original vector \vec{w} . Permutation matrix P converts the incidence matrix \bar{A} to $\tilde{A} = PAP^T$ where \tilde{A} is block triangular (Reinschke, 1988).

Algorithm 1 describes the steps in order that must be followed to obtain the block triangular form of the kinematic structure matrix. Notice that the row permutation (P) applied to Q_{kin} is the same column permutation (P^T) in itens 7 and 8 of algorithm 1.

Algorithm 1: Graph partitioning based on Boolean matrices Reinschke (1988)

- Input data: Q_{kin} or \bar{Q}_{kin}
 - Output data: reordered matrix \check{Q}_{kin} or \tilde{Q}_{kin}
1. Structure matrix: $A = \bar{Q}_{kin}$
 2. Reachability matrix: $R = \sum_{i=1}^{n-1} A^i$
 3. Symmetrical structure matrix: $S \triangleq R \wedge R^T$
 4. Asymmetrical structure matrix: $T \triangleq R \wedge \bar{R}^T$
 5. Asymmetry vector: $\vec{w} : w_j = \sum_{i=1}^n t_{ij}$
 6. Reordering of the asymmetry vector $\vec{w} \rightarrow P\vec{w}$ where P is a permutation matrix
 7. Reordering of the kinematic structure matrix $\check{Q}_{kin} = PQ_{kin}P^T$
 8. Alternatively $\tilde{Q}_{kin} = P\bar{Q}_{kin}P^T$

4. Hierarchical Jacobian reordering

Simultaneous rows and columns permutations are too restrictive to obtain a block triangular form of a generally sparse Jacobian matrix. For this reason directed graph techniques when directly applied to the Jacobian matrix have poor results. The kinematic structure matrix Q_{kin} defined as

$$Q_{kin} \triangleq \begin{pmatrix} 0 & J \\ J^{-1} & 0 \end{pmatrix} \implies \begin{bmatrix} \dot{x} \\ \dot{q} \end{bmatrix} = Q_{kin} \begin{bmatrix} \dot{x} \\ \dot{q} \end{bmatrix} \quad (21)$$

overcomes this deficiency. Matrix Q_{kin} is part of a linear system with identical input and output vectors.

Each i -th column and i -th row are associated to the same variable in \dot{x} or \dot{q} . This property turns the restrictive simultaneous rows and columns permutations on \bar{Q}_{kin} be as general as independent rows and columns permutations on J or, equivalently, independent rows and columns permutations on J^{-1} .

Matrix Q_{kin} can be reordered following the procedures described in (Jantzen, 1996) and (Martins, 2000). These procedures basically rely on Boolean matrices products and sums. Any numerical software can perform these computations. The procedures of this paper were simulated using the software Octave (Eaton, 1996).

The reordering of Q_{kin} , eq. (21), is more properly a simultaneous reordering of both matrices J and J^{-1} . The Roboturb robot in section 5 illustrates the use of these concepts. Matrix Q_{kin} need not to be *a priori* known but only

its structure matrix

$$\overline{Q}_{kin} \triangleq \begin{pmatrix} 0 & \overline{J} \\ \overline{J}^{-1} & 0 \end{pmatrix} \quad (22)$$

where \overline{J} and \overline{J}^{-1} are structure matrices of the Jacobian and inverse Jacobian matrices, respectively. A method for extracting the structure matrix of the inverse is presented in (Martins, 2000).

5. Hierarchy of the singularities of the Roboturb robot

The Roboturb robot is a serial manipulator with seven degrees of freedom. The first joint is prismatic. The remaining six joints are rotative kinematic pairs. The last three joint axes intersect one another in a single point forming a so-called spherical wrist. Figure 2 illustrates the simplified version of the manipulator that will be analysed in this paper.

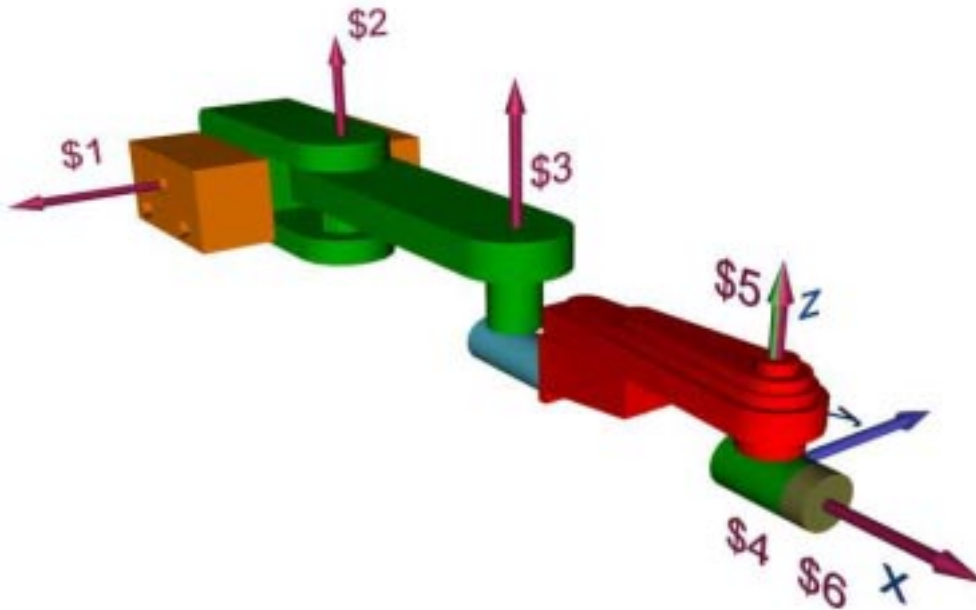


Figure 2: The Roboturb manipulator

Figure 3 shows a sketch of the manipulator configuration and the chosen coordinate system where screw theory methods are applied. This Figure remarks the initial configuration of the robot.

Position angles q_i at the rotative joints are shown in Figures 2 and 3. Screws $\$i$ are aligned with the joint axes and drawn as conical arrows. The chosen coordinate system is represented by the tryad x, y, z . Figure 3 shows the coordinate system chosen and is at the position where all joint angles are at zero position *i.e.* $q_i = 0$ for $i = 1, 2, \dots, 6$.

The Jacobian corresponding to Fig. 2 is

$$J = \begin{bmatrix} -s_{23} & 0 & 0 & 1 & 0 & c_5 \\ -c_{23} & 0 & 0 & 0 & -s_4 & c_4 s_5 \\ 0 & 1 & 1 & 0 & c_4 & s_4 s_5 \\ d_4 c_{23} & a_2 s_3 & 0 & 0 & 0 & 0 \\ -d_4 s_{23} & x_{24} & a_3 & 0 & 0 & 0 \\ x_{14} & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (23)$$

where $s_i = \sin q_i$; $s_{ij} = \sin(q_i + q_j)$; $c_i = \cos q_i$; $c_{ij} = \cos(q_i + q_j)$, etc . . . The symbols a_1, a_2, a_3 are distances shown in Fig. 2 and 3.

Here $x_{14} = a_1 + a_2 c_2 + a_3 c_{23}$ and $x_{24} = a_3 + a_2 c_3$ represent distances along the x -axis from the spherical wrist centre to joints 1 and 2, respectively. Hunt (1987) uses a similar notation to the Puma robot.

The structure of the inverse Jacobian is obtained numerically using algorithm described in (Martins, 2000)

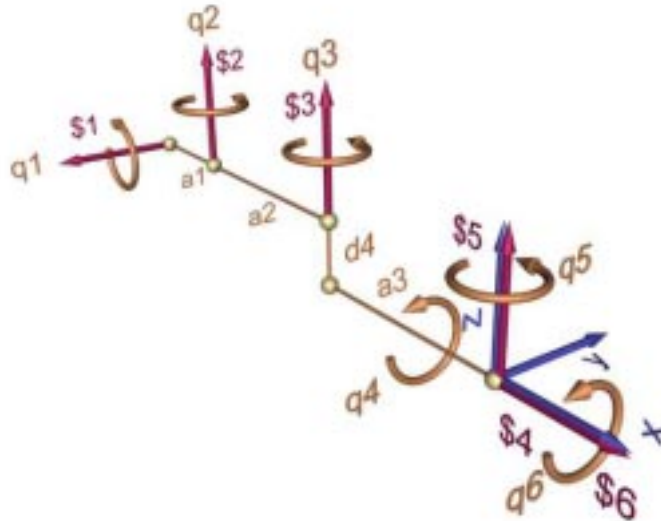


Figure 3: Coordinate system chosen to the Roboturb highlighting the joint variables q_i at zero position.

$$\overline{J^{-1}} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \end{bmatrix} \quad (24)$$

$$z = \begin{bmatrix} \dot{x} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \\ \mathcal{P} \\ \mathcal{Q} \\ \mathcal{R} \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & \times \\ 0 & 0 & 0 & \times & 0 & \times \\ 0 & 0 & 0 & \times & \times & \times \\ \times & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \\ 0 & \times & \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \times & 0 & 0 & \times & 0 & \times \\ \times & 0 & 0 & 0 & \times & \times \\ 0 & \times & \times & 0 & \times & \times \\ \times & \times & 0 & 0 & 0 & 0 \\ \times & \times & \times & 0 & 0 & 0 \\ \times & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathcal{L} \\ \mathcal{M} \\ \mathcal{N} \\ \mathcal{P} \\ \mathcal{Q} \\ \mathcal{R} \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = Q_{Roboturb} z \quad (25)$$

where the end-effector screw $\dot{x} = \mathcal{S}_e = [\mathcal{L}, \mathcal{M}, \mathcal{N}; \mathcal{P}, \mathcal{Q}, \mathcal{R}]^T$ is represented in coordinate system 4, see Fig. 2. This coordinate system is different from that used in eq. (2).

The hierarchically arranged linear system corresponding to eq. (24) is obtained using the techniques described in section 3.

The choice of the coordinate system based on screw theory and the screw transformation matrix, eq. (5) have a crucial role in the method. Without a proper choice of the coordinate system a clear picture of this decoupling process would not appear. The Denavit-Hartenberg convention appears to be not satisfactory due to the sparsity of the Jacobian be almost always lower than screw theory based Jacobian.

Interior singularities of the Roboturb manipulator are analysed using a simplified (non redundant) version of the Roboturb shown in Figs. 2 and 3. The obtained results are similar to those obtained in (Martins and Guenther, 2001) for the Puma robot.

7. Acknowledgments

This work was partially supported by 'Fundação Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES)', Brazil. The first author is in debt to the Melbourne University Robotics Group more specifically to Dr Craig Tischler and Prof Kenneth Hunt for their patience on discussing screw theory.

References

- Asada, H. H. and Slotine, J.-J. E. ,1986, "Robot Analysis and Control". John Wiley and Sons, New York, N.Y.
- Ball, R. S. ,1900, "A Treatise on the Theory of Screws". Cambridge University Press, Cambridge.
- Denavit, J. and Hartenberg, R. S. ,1955, A kinematic notation for lower-pair mechanisms based on matrices, "Trans. ASME E, Journal of Applied Mechanics", 22:215–221.
- Eaton, J. W. ,1996, Free Software Foundation, second edition, for Octave Version 1.94.
- Hunt, K. H. ,1987, Robot kinematics—a compact analytic inverse solution for velocities, "Trans. ASME, Journal of Mechanisms, Transmissions and Automation in Design", 109:42–49.
- Jantzen, J. ,1996, "Digraph Analyses of Linear Control Systems". Technical University of Denmark, Dept of Automation, publ. 96-H-838 (lecture notes).
- Leahy Jr, M. B., Nugent, L. M., and Saridis, G. N. ,1987, Efficient PUMA manipulator Jacobian calculation and inversion, "Journal of Robotic Systems", 4(2):185–197.
- Martins, D. ,2000, Geometrical and combinatorial approaches to robot manipulators kinematics, Research report, Universidade Federal de Santa Catarina, Florianópolis, SC.
- Martins, D. and Guenther, R. ,2001, Hierarchical singularity analysis of robots, In Espindola, J. J., Lopes, E. M. O., and Bazan, F. S. V., editors, "Proceedings of the Ninth International Symposium on Dynamic Problems in Mechanics -IX DINAME", pages 219–224, Florianopolis. Brazilian Society of Mechanical Sciences.
- Pissanetsky, S. ,1984, "Sparse Matrix Technology". Academic Press, New York, NY, USA.
- Reinschke, K. J. ,1988, "Multivariable control : a graph-theoretic approach". Springer-Verlag.
- Röbenack, K. and Reinschke, K. J. ,1997, Graph-theoretically determined Jordan-block-size structure of regular matrix pencils, "Linear Algebra Appl.", 263:333–348.
- Tischler, C. R., Lucas, S. R., Downing, D. M., and Martins, D. ,2000, Screw geometry and Ball's inertia, In Lipkin, H. and Duffy, J., editors, "Proceedings of a Symposium Commemorating the Legacy, Works and Life of Sir Robert Stawell Ball Upon the 100th Anniversary of A Treatise on the Theory of the Screws", pages 1–14:Ball2000–28.pdf, Trinity College. University of Cambridge, University of Cambridge CDROM.
- Tsai, L.-W. ,1999, "Robot Analysis: the Mechanics of serial and parallel manipulators". John Wiley & Sons, New York.
- Wang, S.-L. and Waldron, K. J. ,1987, A study of the singular configurations of serial manipulators, "Journal of Mechanisms, Transmissions, and Automation in Design", 109:14–20.
- Zomaya, A. Y., Smith, H., and Olariu, S. ,1999, Computing robot Jacobian on meshes with multiple buses, "Microprocessors and Microsystems", 23:309–324.

Compensação de Atritos em Atuadores Robóticos do Tipo Moto-Redutores Via Redes Neurais Artificiais

Vitor Irigon Gervini

Fundação Universidade Federal do Rio Grande
Av. Itália, Km 8,
96201-900, Rio Grande, RS
ec2vig@super.furg.br

Sebastião Cícero Pinheiro Gomes

Fundação Universidade Federal do Rio Grande
Av. Itália, Km 8,
96201-900, Rio Grande, RS
dmtscpg@super.furg.br

Resumo. *Atritos não lineares existentes em atuadores do tipo moto-redutores contribuem para diminuir o desempenho de leis de controle, principalmente nos casos da robótica de grande precisão ou manipuladores com elos flexíveis. Neste artigo propõe-se uma estratégia de treinamento e uma estrutura de Rede Neural Artificial (RNA) para aprender o torque de atrito de um atuador do tipo moto-redutor. Para treinar a RNA foi utilizado o torque de atrito obtido a partir de um modelo proposto na literatura (publicado em 1995). Considerou-se o torque motor e a velocidade angular do rotor como entradas na RNA, enquanto o torque de atrito foi a única saída, a qual foi usada na proposição de um mecanismo de compensação do torque não linear de atrito. Os resultados atestaram a eficiência da identificação e posterior compensação com o mecanismo proposto.*

Palavras Chave. Rede Neural Artificial, robótica, atrito, compensação.

1. Introdução

Existem atualmente diversas aplicações de redes neurais artificiais (RNA), nos mais variados domínios da ciência e tecnologia (Kaynak and Ertugru, 1997; Jung and Hsia, 1998). Trata-se de um assunto que tem merecido grande atenção por parte da comunidade científica. Em (Miller, 1995) tem uma importante descrição sobre o histórico das redes neurais artificiais.

Neste artigo investiga-se a possibilidade de identificação do torque de atrito de um atuador robótico do tipo moto-redutor, utilizando-se uma Rede Neural Artificial (RNA). A principal motivação reside na dificuldade de se encontrar um modelo dinâmico muito realista e preditivo para atuadores robóticos, dificuldade esta devida principalmente aos atritos não lineares internos aos atuadores (Armstrong, 1988). Esta aplicação (identificação de atrito via RNA) começou a ser mais estudada apenas recentemente (Dapper and Zanh, 1999; Selmic and Lewis, 2000), tratando-se, portanto, de um assunto bastante atual. Uma vez identificado o atrito não linear, propõe-se um mecanismo de compensação deste a fim de linearizar o atuador. A estratégia de identificação, utilizada conjuntamente ao mecanismo de compensação, constitui a principal contribuição deste trabalho.

Como não se dispõe, até o presente momento, de um atuador robótico isolado em uma bancada a fim de se realizar experimentos, utilizou-se um modelo proposto na literatura (Gomes, 95). Os resultados deste modelo serviram de referência como padrões de treinamento da RNA. A seqüência natural do presente trabalho será adquirir um moto-redutor do tipo *harmonic-drive* para a obtenção dos resultados experimentais, a fim de validar definitivamente o mecanismo de compensação de atritos proposto e testado em simulação. Como o modelo utilizado foi muito testado experimentalmente, acredita-se que todo o formalismo de identificação e compensação proposto deva proporcionar bons resultados na prática.

2. O Atuador Robótico

Um atuador robótico do tipo moto-redutor pode ser visualizado como um elemento transmissor de movimento contendo uma elasticidade interna de constante K , conforme representado na Fig. (1). O torque motor T_m é aplicado ao rotor de inércia I_r . Atritos não lineares estão sempre presentes neste tipo de dinâmica, fazendo com que o torque motor não seja transmitido integralmente à inércia I_s acoplada à saída do redutor. As equações (1) descrevem a dinâmica deste sistema (Gomes and Chrétien, 1992). T_{at} corresponde ao torque de atrito não linear, o qual será obtido via RNA. θ_r e θ_s são, respectivamente, os ângulos do rotor e do eixo de saída do redutor.

$$\begin{aligned}
 I_r \ddot{\theta}_r + \frac{K}{n^2} (\theta_r - n\theta_s) &= T_m - T_{at} \\
 I_s \ddot{\theta}_s - K \left(\frac{\theta_r}{n} - \theta_s \right) &= 0
 \end{aligned}
 \tag{1}$$

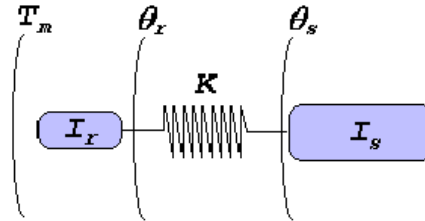


Figura 1. Representação de um atuador robótico do tipo moto-reductor.

O modelo para o torque de atrito utilizado foi publicado originalmente em (Gomes, 1995), tendo apresentado bons resultados quando feitas comparações entre experimentos e simulações. Este modelo reproduziu bem o fenômeno de *stick-slip*, notável por aparecer com frequência na prática. Este fenômeno foi definido em (Gomes, 1995) como tendo a seguinte origem: quando o torque resultante através do atuador está menor do que o atrito seco e ainda, a velocidade do rotor menor do que um certo valor limite que define a zona de *stiction* (intervalo de velocidade em torno do zero no qual ocorre o fenômeno), então a velocidade rotor vai a zero e assim permanece até que o torque resultante volte a ser superior ao atrito estático. Conforme será visto na seqüência deste trabalho, o modelo adotado reproduziu este fenômeno e a RNA o aprendeu bem.

3. O Mecanismo de Compensação

Apesar de ter se mostrado bastante eficiente, o mecanismo de compensação de atrito proposto no presente trabalho é simples, consistindo na rejeição direta do atrito não linear identificado pela RNA. Simulou-se o caso mais real de controle discreto, com um passo de discretização de 0.005 segundos. O torque motor no instante \$k+1\$ assume a forma:

$$T_m(k+1) = T_c(k) + \hat{T}_{at}(k)
 \tag{2}$$

sendo \$T_c(k)\$ o torque de controle, o qual em malha fechada depende da observação do sistema no instante \$k\$. \$\hat{T}_{at}(k)\$ é o atrito não linear estimado pela RNA (saída), tendo como entrada nesta a velocidade rotor \$\dot{\theta}_r(k)\$ e o torque motor \$T_m(k)\$.

4. Arquitetura da RNA

A arquitetura de rede adotada para a identificação do torque de atrito foi uma rede *backpropagation (multilayer, feedforward network, fully connected)* (Beale and Jackson, 1990). Utilizou-se, como regra de treinamento, o *backpropagation com momentum* (Fausett, 1994). A camada de entrada foi composta de dois neurônios (torque motor e velocidade do rotor) e a camada de saída por um neurônio (torque de atrito).

Foi utilizada somente uma camada intermediária, tendo sido necessários apenas quatro neurônios nesta para aprender os padrões apresentados à rede. Foi testado um número maior de neurônios (8, 10, 16, 40 e outros), na camada intermediária, mas os resultados foram muito similares ao obtido com apenas 4 neurônios. Já que a arquitetura da RNA é muito pequena, isto a torna rápida e eficiente com um alto poder de generalização, permitindo assim uma compensação *on line* do torque de atrito.

Na Fig. (2) abaixo é mostrada a arquitetura da rede utilizada:

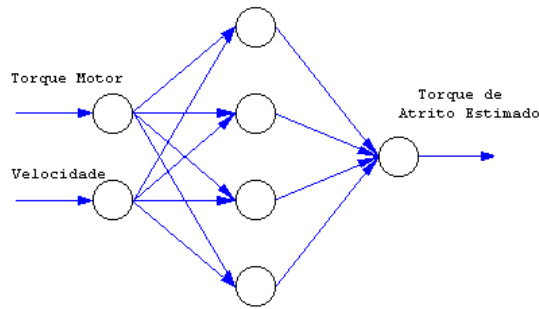
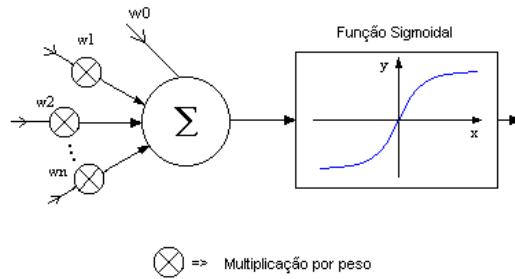


Figura 2. Arquitetura da RNA.

A arquitetura dos neurônios é mostrada na Fig. (3), tendo-se utilizado uma função de ativação sigmoidal (tanh). Após o treinamento, a rede apresentou uma taxa de aprendizado de 98% dos padrões apresentados.



$\otimes \Rightarrow$ Multiplicação por peso

Figura 3. Arquitetura dos neurônios utilizada.

Na Fig. (4) mostram-se o torque motor e a velocidade rotor (entrada da rede), o torque de atrito do modelo (vermelho) e o torque de atrito reconhecido pela rede (preto). Pode-se perceber então que de fato, a taxa de aprendizado foi realmente elevada. Outro dado interessante é que se escolheu o torque motor de entrada formado a partir de seguimentos de senoides e degraus em malha aberta, além de alguns resultados em malha fechada, obtidos com um controle proporcional e derivativo (últimos seguimentos em torque motor da Fig. (4)). Esta diversificação do torque na estratégia de treinamento foi importante para a generalização da RNA.

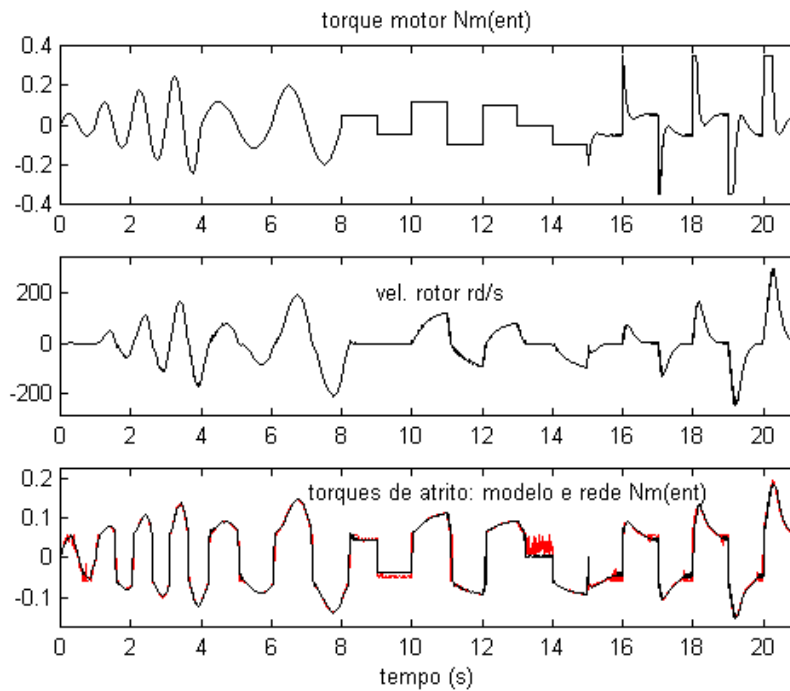


Figura 4. Padrões de treinamento da Rede.

5. Resultados de Simulações

Foram realizadas diversas simulações com dois objetivos principais: confrontar os resultados da saída da RNA com o modelo do torque de atrito não linear; testar a eficiência da estratégia de compensação do torque não linear de atrito a partir da identificação do mesmo via RNA.

O gráfico superior da Fig. (5) mostra o torque motor aplicado em malha aberta, composto de diversos seguimentos de senoides. A parte inferior mostra os torques de atrito do modelo e da RNA (em vermelho). Percebe-se que, de fato, a saída da RNA é praticamente coincidente com a curva obtida com o modelo.

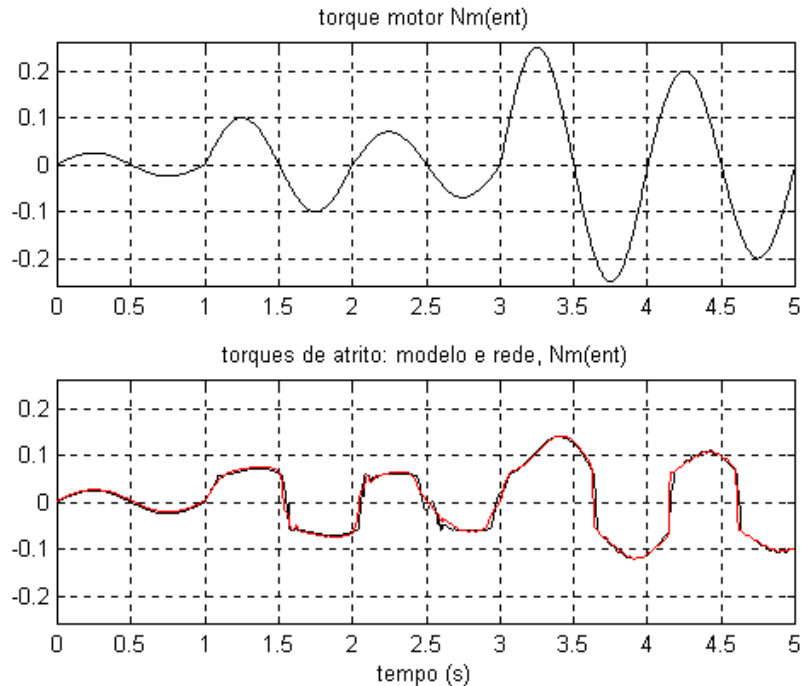


Figura 5. Torques, motor e de atritos.

Na Fig. (6) mostram-se os resultados da posição angular da saída do redutor, obtidas em três simulações distintas, mas com o mesmo torque motor da Fig. (5): o caso linear (considerou-se apenas o atrito linear no modelo dinâmico (curva em vermelho)); os casos não linear com e sem a estratégia de compensação do torque de atrito. Pode-se perceber que a estratégia de compensação de atrito via RNA tende a linearizar a articulação, evitando de forma eficiente o fenômeno de *stick-slip* (Gomes, 1995), o qual é bastante visível na simulação com o atrito não linear e sem a compensação. Observar que há situações nas quais apesar da velocidade, na simulação com o atrito linear, estar diferente de zero, a velocidade na simulação com o atrito não linear permanece zero. Isto acontece porque, durante estes intervalos de tempo, o torque motor foi inferior ao atrito estático. Este é precisamente o conhecido modo *stick* (Armstrong, 1988): a articulação se bloqueia e não há transmissão de movimento neste estado, apesar do torque motor ser diferente de zero.

Resultados semelhantes foram obtidos também em malha aberta para respostas ao degrau em torque (Figs. (7) e (8)). Assim como para o caso das senoides das Figs. (5) e (6), utilizou-se amplitudes variadas e diferentes das que formaram os padrões de aprendizado da RNA. As pequenas diferenças entre as simulações linear e não linear com compensação de atrito atestam a eficiência da estratégia de compensação proposta.

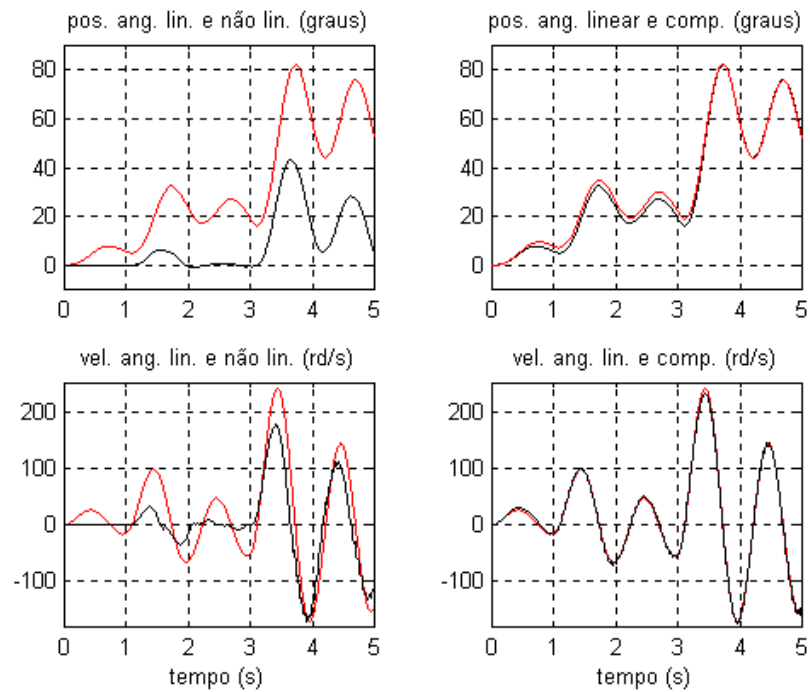


Figura 6. Posições e velocidades angulares (simulação com o atrito linear em vermelho).

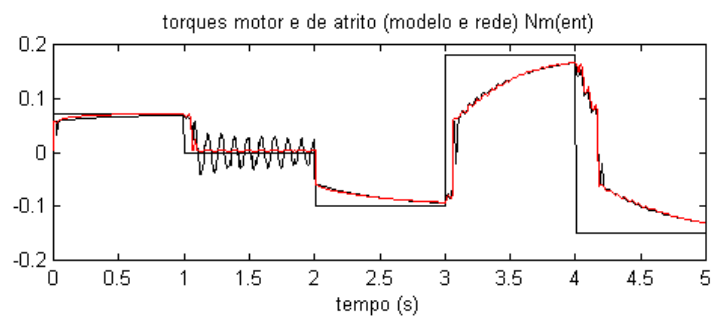


Figura 7. Torques, motor e de atrito (atrito identificado pela RNA em vermelho).

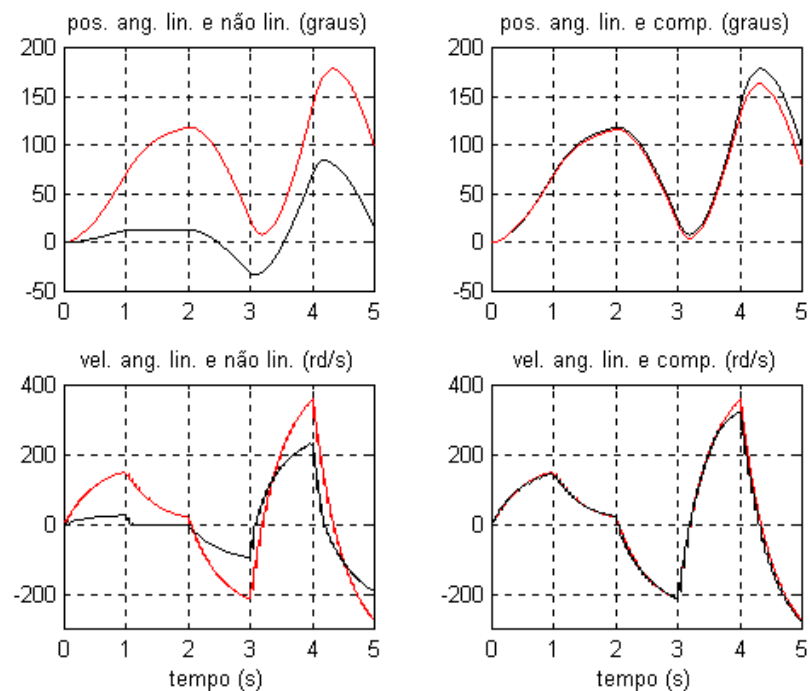


Figura 8. Posições e velocidades angulares (simulação com o atrito linear em vermelho, torque motor em graus).

Alguns resultados em malha fechada foram também obtidos. Utilizou-se um controle proporcional e derivativo (PD), nos casos com e sem o mecanismo de compensação. A Fig. (9) mostra o torque motor com a compensação de atrito, para degraus em posição em malha fechada, bem como os torques de atrito do modelo e da RNA. De forma semelhante aos casos em malha aberta, a identificação do atrito via RNA foi muito eficiente.

A Fig. (10) apresenta as respostas (posição e velocidade) do controle em malha fechada, para os casos do torque de atrito linear e não linear com e sem o mecanismo de compensação. No caso não linear sem a compensação, percebe-se que, para as pequenas amplitudes em degrau na entrada de referência, o torque motor de controle é menor do que o atrito estático, configurando-se assim o modo *stick*, não havendo portanto movimento. O caso não linear com a compensação de atrito se assemelha ao caso linear, mostrando desse modo a eficiência da identificação via RNA do atrito e da compensação deste a partir do mecanismo proposto.

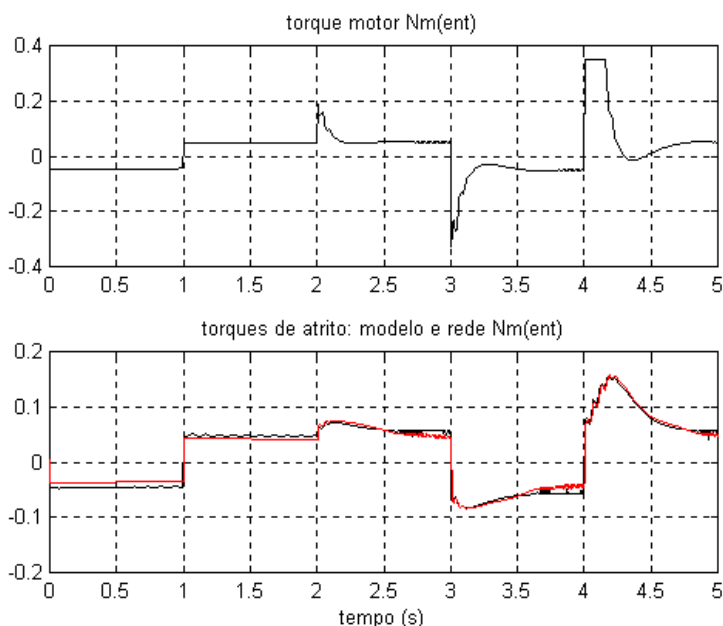


Figura 9. Torque motor com a compensação de atrito e torques de atrito (atrito da RNA em vermelho).

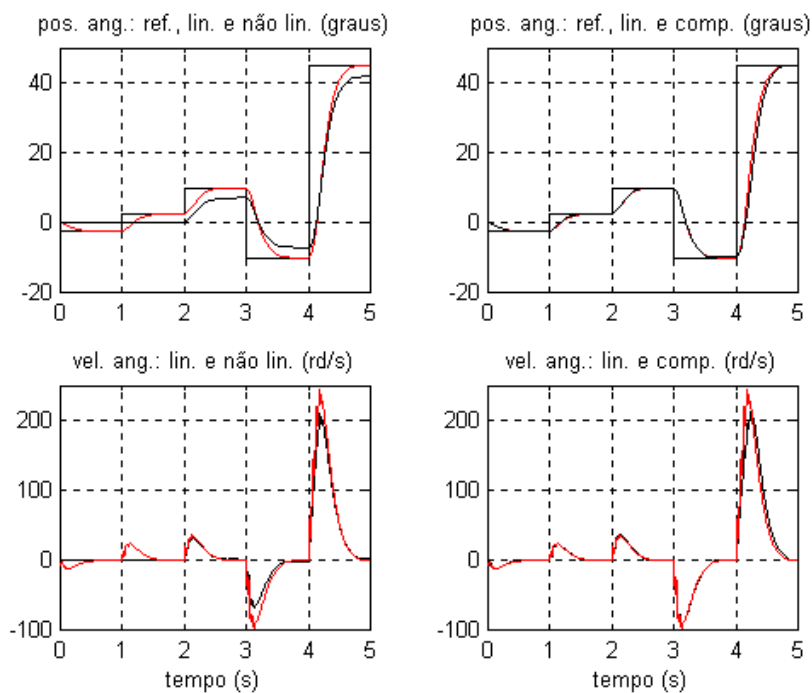


Figura 10. Posições e velocidades angulares em malha fechada (simulações com o atrito linear em vermelho).

6. Conclusões

Pode-se extrair, como principal conclusão deste trabalho, que uma RNA de configuração simples pode ser eficiente na identificação do torque de atrito não linear de um atuador robótico do tipo moto-reductor. O mecanismo proposto de compensação do atrito não linear, o qual utiliza a identificação (saída da RNA), mostrou-se muito eficiente. Como o modelo de atrito que serviu de base para formar os padrões de treinamento da RNA foi bastante testado experimentalmente, acredita-se que a estratégia de compensação de atrito proposta deva funcionar na prática para atuadores do tipo moto-redutores em geral, respeitando-se todas as etapas aqui desenvolvidas, inclusive a mesma estratégia de treinamento da RNA, a qual utilizou resultados em malhas aberta e fechada. Como continuação do presente trabalho, pretende-se adquirir um moto-reductor do tipo *harmonic-drive* a fim de se obter os resultados experimentais que comprovem em definitivo o que foi demonstrado em simulação.

7. Referências

- Armstrong, B. S. R., 1988. "Dynamics for robot control: friction modeling and ensuring excitation during parameter identification". PhD thesis, Stanford University, USA.
- Beale, R. and Jackson, T. 1990. "Neural computing: an introduction". Adam Higler Bristol.
- Dapper, M., Zanh, V., Maass, R. and Ekmiller, R. 1999. "How to compensate stick-slip friction in neural velocity force control (NVFC) for industrial manipulators". In IEEE Robotic and Automation Conference, Detroit, USA, May.
- Fausett, L. 1994. "Fundamentals of neural networks". Prentice Hall, New Jersey.
- Gomes, S. C. P. and Chrétien, J. P., 1992. "Dynamic modeling and friction compensated control of a robot manipulator joint". In IEEE Robotic and Automation Conference, Nice, France, May.
- Gomes, S. C. P., 1995. "Modelagem de atritos internos às articulações de robôs manipuladores". In XIII COBEM, CD-ROM, Belo Horizonte.
- Jung, S. Hsia, T. C., 1998. "Analysis of non linear neural network impedance force control for robot manipulator". In IEEE Robotic and Automation Conference, Leuven, Belgium, May.
- Kaynak, O. and Ertugru, M., 1997. "Neural network adaptive Sliding Mode Control and its application to SCARA type robot manipulator". In IEEE Robotic and Automation Conference, Albuquerque, New Mexico, USA, April.
- Miller III, W.T., Sutton, R.S., Werbos, P.J., 1995. "Neural networks for control". MIT Press.
- Selmic, R. R. and Lewis, F. L., 2000. "Dead zone compensation in motion control systems using Neural Networks". IEEE Transactions on Automatic Control, Vol 45, April.

Title: Friction compensation of robotic actuators through Neural Network

Abstract. The knowledge of realistic dynamic models to robotic actuators would be of great aid in the conception of control laws for robot manipulators, mainly in the cases of the great precision robotic or still for manipulators with flexible links. In this paper we present a training strategy and propose a structure of Neural Network (NN) to learn the friction torque of a motor drive joint robotic actuator with gear. To train the NN it was used a friction model proposed in the literature (published in 1995). It was considered the motor torque and the rotor angular velocity as input in the NN, while the friction torque was the only output, which was used in the proposition of a non-linear friction compensation mechanism. The results attested the efficiency of the NN friction estimate and compensation with the proposed mechanism.

Keyword: Neural Networks, robotics, friction, compensation.

8. Copyright Notice

The authors are the only responsible for the printed material included in his paper.

IMPLEMENTAÇÃO DE OPERAÇÕES BOOLEANAS USANDO ARITMÉTICA INTERVALAR EM UM MODELADOR DE SÓLIDOS B-REP

Marcelo Shimada

Escola Politécnica – Universidade de São Paulo
Departamento de Engenharia Mecatrônica
Av. Prof. Mello Moraes, 2231
shimada@usp.br

Marcos de Sales Guerra Tsuzuki

Escola Politécnica – Universidade de São Paulo
Departamento de Engenharia Mecatrônica
Av. Prof. Mello Moraes, 2231
mtsuzuki@usp.br

Resumo: Neste trabalho estaremos considerando a utilização de aritmética intervalar e dimensões relativas para aumentar a robustez dos algoritmos de classificação geométrica utilizados em Operações Booleanas em Sistemas de Modelagem de Sólidos. A aritmética intervalar incorpora os erros de truncamento, eliminando a necessidade de definir uma tolerância fixa para realizar a comparação entre números de ponto flutuante. Entretanto, não é possível converter diretamente os algoritmos usando ponto flutuante para algoritmos usando aritmética intervalar, então, é necessário uma total reformulação. Apresentaremos os conceitos básicos de um modelador de sólidos B-Rep, a aritmética intervalar e as dimensões relativas.

Palavras chave: Modelagem de Sólidos, Operações Booleanas, Aritmética Intervalar, Dimensões relativas

1. Introdução

Atualmente, os sistemas de modelagem de sólidos são amplamente utilizados. Exemplo disto é o seu uso no desenvolvimento de novos produtos, no processo de fabricação para armazenar as informações sobre as peças, durante o projeto e análise de componentes mecânicos através de sistemas de análise por elementos finitos. Um componente principal dos modeladores de sólidos é a máquina de operações booleanas. Estas operações facilitam a criação de sólidos complexos a partir de sólidos simples. Por uma análise das publicações na literatura é possível observar que o desenvolvimento de algoritmos para implementar operações booleanas envolve a definição de algoritmos de classificação geométrica entre os elementos primitivos (vértice, aresta e face). É necessário determinar se um vértice está posicionado sobre algum vértice do outro sólido (incidência), ou se uma aresta intersecciona uma aresta do outro sólido em um vértice ou se estas duas arestas são colineares. É comum implementar algoritmos que, por meio de manipulação numérica, determinam esta classificação. Entretanto, a manipulação numérica de ponto flutuante é susceptível a erros.

Neste trabalho estaremos considerando a utilização de aritmética intervalar, apresentada por Hu et al (1996) para aumentar a robustez dos algoritmos de classificação geométrica utilizados em Operações Booleanas. A aritmética intervalar incorpora os erros de truncamento, eliminando a necessidade de definir uma tolerância para realizar a comparação entre números de ponto flutuante.

Outro item importante desta proposta é a utilização do conceito de dimensão relativa entre faces. Esta dimensão fornece a distância relativa entre duas faces paralelas. Esta informação será utilizada, principalmente, nas operações booleanas, para simplificar a classificação de incidência entre os elementos dos sólidos envolvidos na operação.

Logo, este trabalho propõe o uso da aritmética intervalar e da dimensão relativa entre faces em um modelador de sólido “B-rep” para aumentar a robustez das Operações Booleanas. A aritmética intervalar lidará com os cálculos em ponto flutuante, aumentando a robustez, validade e coerência dos resultados, enquanto que a dimensão relativa fornecerá informações de amarramento entre os sólidos.

Na seção seguinte, são apresentadas informações sobre modeladores de sólido B-rep. Na seção 3, são apresentados os Operadores do Modelador de Sólidos. A aritmética intervalar é apresentada na seção 4 e a dimensão relativa, na seção 5. Na seção 6 é apresentada a implementação atual. Finalmente, a seção 7 apresenta as conclusões iniciais sobre o estudo.

2. Modelador de Sólidos B-Rep

Segundo Tsuzuki (1991), é possível dividir o modelador de sólidos em três níveis:

- Nível Superior: neste nível, o usuário interage com o sistema com o auxílio de ferramentas para construir, modificar e armazenar os sólidos. Os operadores locais e operações booleanas compõem este nível.

- Nível Intermediário: onde são desenvolvidas ferramentas para implementar as ferramentas do nível superior. Este nível é principalmente composto pelos Operadores de Euler.
- Nível Inferior: onde é desenvolvida a estrutura de dados que forma uma representação apropriada para a manipulação computacional.

2.1. Relações de Adjacência

A representação B-rep armazena detalhes de como as faces, arestas e vértices se unem para representar um sólido. Um sólido modelado pela representação B-rep deve possuir, por exemplo, a capacidade de descrever como cada face está conectada às suas faces adjacentes, de maneira que um volume totalmente fechado seja definido. Em uma representação B-rep, esta informação está disponível explicitamente.

Esta informação de adjacência é, geralmente, referenciada informalmente como topologia do sólido modelado. As informações topológicas criam um conjunto de vigas, no qual as informações geométricas são apoiadas. As informações topológicas e as informações geométricas não podem ser tratadas independentemente, pois elas estão profundamente relacionadas. Entretanto, a topologia é consequência da geometria e não vice-versa.

2.2. Estruturas B-rep

A representação B-rep descreve um sólido por superfícies orientadas através de uma estrutura de dados composta por vértices, arestas e faces, segundo Hoffmann (1989). A convenção de orientação permite-nos determinar o interior e exterior do sólido de forma não ambígua. A Fig. (1) ilustra um exemplo.

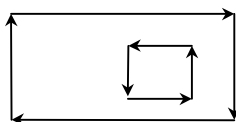


Figura 1. Laço externo representado por um conjunto de arestas orientados formando a borda da face e um furo interno com orientação contrária .

Dentre as estruturas de dados, baseadas na aresta como elemento de referência, encontradas na literatura, destacam-se a “winged-edge” e a “halfedge”.

A estrutura “winged-edge” mantém as informações de adjacência por meio de ponteiros a vários elementos adjacentes à aresta de referência: duas faces, dois vértices e quatro arestas. O conjunto de arestas adjacentes é dividido em dois grupos, cada um associado ao circuito de arestas ao redor das faces adjacentes à aresta de referência. Em outras palavras, a estrutura “winged-edge” possui ponteiros para as quatro arestas diretamente anteriores e posteriores à aresta de referência, nos ciclos de aresta que contornam as duas faces adjacentes à aresta de referência.

Na estrutura “winged-edge”, as arestas assumem duas principais funções: dividir o contorno direcional das faces e definir a conectividade entre os elementos primitivos por meio de informações de adjacência da aresta de referência. Porém, do ponto de vista computacional, este é o ponto mais negativo da estrutura “winged-edge”. Esta deficiência é clara, em particular, quando todas as arestas de uma face devem ser referenciadas pelo procedimento que percorre sequencialmente o circuito direcional de cada face. A necessidade deste algoritmo surge com muita frequência em operações gráficas e geométricas aplicadas ao objeto representado.

Percorrer o circuito de arestas de uma face é uma operação unidirecional, devido à própria orientação do circuito de arestas. Entretanto, na estrutura “winged-edge” cada aresta é utilizada para delimitar duas faces; portanto, cada aresta participa em dois circuitos orientados. Segundo a lei Möebius, os dois circuitos possuem orientações opostas entre si, portanto, cada aresta é percorrida exatamente duas vezes e em direções opostas. Em outras palavras, cada aresta pode ser percorrida em duas direções opostas, dependendo da face cujo circuito está sendo analisado. É necessário, entretanto, verificar e determinar a direção em que cada aresta está sendo referenciada a cada passo do percurso, um processo que aumenta consideravelmente o custo do procedimento.

Para resolver esta deficiência, a estrutura “halfedge” foi proposta; onde as duas principais funções da aresta foram separadas. Esta separação foi obtida pela divisão de cada “winged-edge” em duas metades. A conectividade entre ambas as metades é mantida por um ponteiro que referencia a metade oposta.

Na estrutura “halfedge”, cada metade da aresta participa em apenas um circuito de arestas, portanto, cada metade possui apenas uma única orientação. Globalmente, cada aresta de referência é referenciada duas vezes em direções opostas pelos circuitos de arestas que contornam as duas faces adjacentes. A Fig. (2) ilustra a representação “halfedge”.

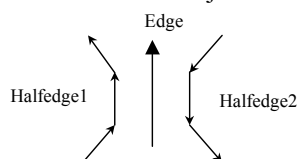


Figura 2. Representação de uma “halfedge” formada por três elementos básicos: “edge”, “halfedge1” e “halfedge2”.

Outros dois elementos devem ser mencionados: o “*shell*” e o “*region*”. O elemento “*shell*” consiste na delimitação formada pelas faces, como um conjunto fechado; e o elemento “*region*” representa o interior delimitado pelo elemento “*shell*”. Em termos simples, o “*shell*” representa uma “casca” (que pode ser interior ou exterior) e “*region*” o interior desta “casca”. Um exemplo de um sólido com dois “*shells*” e apenas um “*region*”, é mostrado na Fig. (3), que consistem em um bloco que possui um vazio em seu interior.

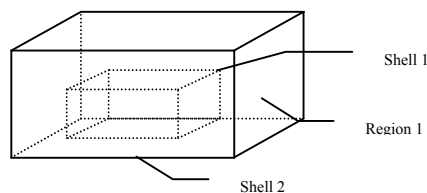


Figura 3. Exemplo de sólido que possui dois “*shells*” e apenas um “*region*”, que a parte limitada pelos “*shells*”.

3. Operadores do Modelador de Sólidos

Os operadores são ferramentas utilizadas como interface entre o nível inferior (estrutura de dados) com o usuário do Modelador de Sólido. No nível intermediário, estão os operadores de Euler e no nível superior estão os operadores locais e as operações booleanas.

3.1. Operadores de Euler

Conceitualmente, Operadores de Euler podem ser considerados como criadores e modificadores da topologia de superfícies de objetos “*manifolds*”, segundo Hoffmann (1989). Uma superfície “*manifold*” é aquela em que ao redor de todos os pontos dela existe uma vizinhança que é homeomórfica ao plano, isto é, pode-se deformar a superfície “*manifold*” localmente em um plano e não será possível identificar pontos vizinhos separados. São usados como uma linguagem intermediária em sistemas de modelagem de sólido. São capazes, por exemplo, de criar faces/superfícies, adicionar/retirar vértices/arestas, simplificando a formação do sólido e mantendo a consistência topológica.

O principal objetivo dos Operadores de Euler é simplificar a manipulação das complicadas estruturas B-rep. A idéia principal é que a construção dos modelos possa ser realizada passo a passo pelo uso de um conjunto de operadores que manipule a estrutura de dados B-rep e que efetivamente esconda detalhes de implementação da representação.

Os operadores de Euler tornam possível a construção incremental de objetos em uma, duas ou três dimensões de maneira semelhante a desenhar linha a linha. Eles também facilitam alterações locais da forma, característica que é muito conveniente e eficiente para projetistas de modeladores de sólido.

Tradicionalmente, denota-se os operadores de Euler por um conjunto de letras formados pelas iniciais do que o operador está realizando. Exemplo: *make edge and vertex*, torna-se o operador MEV. Para cada operador de Euler existe um inverso.

- MVSF/KSVF (*Make/Kill Vertex Solid Face*) – operador necessário para iniciar a construção de um sólido. Cria um novo sólido, uma nova “*region*”, um novo “*shell*”, uma nova face, um novo laço, um novo vértice (dado de entrada) e uma nova “*halfedge*”. Não cria arestas.

- MEV/KEV (*Make/Kill Edge Vertex*) – cria uma aresta que liga um vértice (fornecido) com um novo vértice que será criado.

- MEF/KEF (*Make/Kill Edge Face*) – cria uma aresta que ligará dois vértices (fornecidos), criando uma nova face.

- KEMR/MEKR (*Kill/Make edge Make/Kill Ring*) – divide um laço em duas partes. Ou seja, ao apagar uma aresta, ela deixa um vértice separado dos demais que formavam o laço inicial. E neste vértice separado, é criado o novo laço. No exemplo apresentado na Fig. (4), o laço externo possui, inicialmente, cinco vértices. Após o uso de KEMR, uma aresta é eliminada, um novo laço é criado, e uma “*halfedge*” é eliminada.

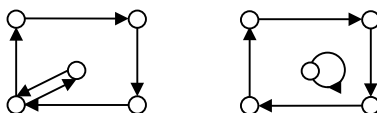


Figura 4. Exemplo de utilização do KEMR em que um laço formado por 6 “*halfedges*” é quebrado em 2 laços.

- KFMRH/MFKRH (*Kill/Make Face Make/Kill Ring Hole*) – transforma duas faces em uma única face, colocando o laço de uma como laço interno da outra. Utilizado em casos em que se utilizou o KEMR para iniciar um furo e adicionou-se vértices que chegam até a face oposta, e deseja-se que este furo seja passante. De modo simples, apaga-se a face (que chegou até a face oposta do sólido) e ela é adicionada como um laço interno da face oposta. Um exemplo é apresentado na Fig. (5).

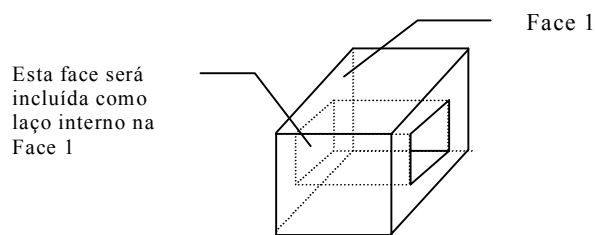


Figura 5. Exemplo de utilização de KFMRH em que um furo passante é realizado.

- KSFMR/MSFKR (*Kill/Make Shell Face Make/Kill Ring*) – utilizado na situação em que se possui dois “shells” e duas faces que estão no mesmo plano e deseja-se torná-los em um único “shell”. Na Fig. (6) são apresentados dois exemplos do uso do KSFMR. No Exemplo (a), os dados do “shell 2” (vértices, faces, arestas) são colocados no “shell 1” (e o “shell 2” é então apagado) e a face que coincidia com outra do “shell 1”, torna-se um laço interno da face do “shell 1”, deste modo criando um furo não passante do “shell 1”. Já o exemplo (b), o “shell” resultante não apresenta furo, mas apresenta ainda uma face com laço interno.

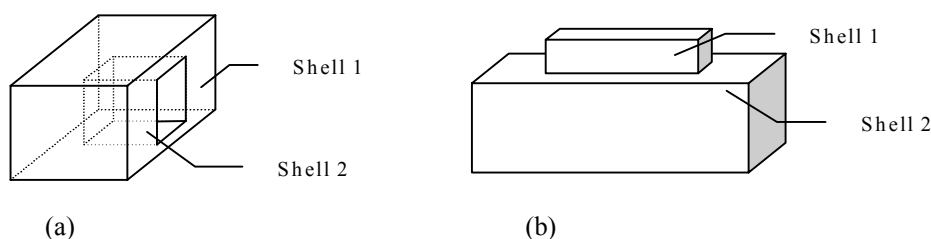


Figura 6. Exemplos de utilização de KSFMR.

Durante a construção de modelos com Operadores de Euler, a topologia é mantida válida segundo a equação de Euler – Poincaré, mostrada na Eq. (1):

$$v - e + f = 2 * (s - h) + r \quad (1)$$

Sendo: s peças desconexas, f faces, e arestas, v vértices, h furos, l laços e r anéis ($r = l - f$).

Ao final de uma seqüência de Operadores de Euler assume-se que a geometria do sólido está correta, mas durante os estágios intermediários não há como manter a topologia e a geometria consistentes devido à presença de faces não planares; que, freqüentemente, não são possíveis de serem representadas por nenhuma forma matemática. Logo, os operadores de Euler não são operadores seguros por si, mas devem ser colecionados em seqüências que forneçam um significado.

3.2. Operadores Locais

Os operadores locais são ferramentas de alto nível que permitem a criação de modelos práticos de objetos de engenharia. Entretanto, eles podem ser utilizados para criar a descrição de peças de uma maneira adequada à sua forma. Operadores locais oferecem meios para alterar o modelo de modo a satisfazer um refinamento incremental do modelo inicial.

Eles fornecem os meios para criar modificações diretas sobre os objetos. Os operadores locais não necessitam de verificações trabalhosas, pois eles não realizam alterações em áreas extensas à localidade das características a que elas se aplicam. Como exemplos de operadores locais encontrados na literatura podemos citar: suavizar, chanfrar, grudar, revolver e encompridar.

3.3. Operações Booleanas

Algoritmos para determinar a união, intersecção ou diferença de dois objetos podem ser utilizados por modeladores de representação B-rep. Conceitualmente, estes algoritmos não são difíceis, mas sua implementação requer um substancial trabalho por várias razões. Considerar as várias posições espaciais de incidência das primitivas geométricas em três dimensões pode ser trabalhoso.

A não redundância das informações geométricas reduz a possibilidade de contradição dos dados e portanto aumenta a robustez. As coordenadas dos vértices e as equações dos planos dos modelos poliedrais são exemplos de redundâncias; pois, geralmente, os vértices não estão exatamente sobre os planos, mas podem estar deslocados por uma quantidade muito insignificante. Logo a forma e o contorno das faces podem não ser consistentes.

Para a implementação das Operações Booleanas a manipulação dos modelos dos objetos pode ser dividida em duas fases distintas: cálculos numéricos e modificações na estruturas B-rep. A fase de cálculos numéricos é representada por

dois passos: cálculo de todos os pontos de intersecção entre os dois objetos e análise da vizinhança dos pontos de intersecção. A fase de modificação da estrutura B-rep é representada por quatro passos: geração de arestas nulas, análise dos pontos de intersecção e criação dos circuitos de arestas, recorte das arestas nulas e união dos devidos componentes para criar o objeto resultado da operação booleana.

Da literatura, duas formas de análise da vizinhança de intersecção são comuns. Uma é baseada na classificação dos vértices vizinhos ao vértice de intersecção, onde é necessário classificar os setores convergentes ao vértice de intersecção, de modo a determinar como a aresta nula deve ser inserida. A implementação de Mäntylä (1988) é um exemplo desta forma de análise. A segunda forma de análise, é através da classificação do posicionamento relativo entre setores dos diedros, de modo a definir os elementos primitivos de um sólido que são internos ao outro, e vice-versa. Exemplos desta forma de análise são as implementações de Chiyokura (1988) e de Glaeser e Stachel (1998).

4. Aritmética Intervalar Arredondada

Segundo Hu, um intervalo convencional consiste em um conjunto de números reais definido por:

$$[a, b] = \{x \mid a \leq x \leq b\} \quad (2)$$

O intervalo $[a, b]$ é considerado degenerado se $a = b$. Dois intervalos $[a, b]$ e $[c, d]$ são considerados iguais se $a = c$ e $b = d$. A intersecção de dois intervalos é vazia se $a > d$ ou $c > b$. Do contrário, $[a, b] \cap [c, d] = [\max(a, c), \min(b, d)]$. A união de dois intervalos que possuem intersecção é obtida por: $[a, b] \cup [c, d] = [\min(a, c), \max(b, d)]$. Uma ordem de intervalos é definida por $[a, b] < [c, d]$ se e somente se $b < c$.

As comparações em aritmética intervalar devem ser repensadas. Um intervalo pode ser possivelmente igual ou certamente igual a outro intervalo.

Logo, as operações utilizando aritmética intervalar são definidas como:

$$[a, b] + [c, d] \equiv [a + c - \epsilon, b + d + \epsilon] \quad (3)$$

$$[a, b] - [c, d] \equiv [a - d - \epsilon, b - c + \epsilon] \quad (4)$$

$$[a, b] \cdot [c, d] \equiv [\min(ac, ad, bc, bd) - \epsilon, \max(ac, ad, bc, bd) + \epsilon] \quad (5)$$

$$[a, b] / [c, d] \equiv [\min(a/c, a/d, b/c, b/d) - \epsilon, \max(a/c, a/d, b/c, b/d) + \epsilon] \quad (6)$$

Para a aritmética intervalar define-se o número ϵ , sendo ϵ obtido por: $\epsilon = 2^{\text{exp} - 53}$, onde exp é obtido para cada limite superior e inferior computado (exemplo: para a soma: calcula-se $\epsilon(a+c)$ e $\epsilon(b+d)$). ϵ é chamado de unidade de último lugar e é denotado por ulp ("unit of last place").

Em aritmética intervalar, será considerado que um intervalo $[a, b]$ é possivelmente igual a um intervalo $[c, d]$, se a intersecção entre eles é não nula.

Pontos intervalares são definidos pelas componentes das coordenadas intervalares. Um ponto 3D é representado por: $([x_{\text{low}}, x_{\text{upper}}], [y_{\text{low}}, y_{\text{upper}}], [z_{\text{low}}, z_{\text{upper}}])$. Uma linha intervalar que passa por dois pontos intervalares $[P_0]$ e $[P_1]$ é definido como a combinação linear dos dois pontos intervalares. No caso 2D, é um polígono, enquanto que no caso 3D é formado um poliedro. A Fig. (7) ilustra um ponto intervalar 3D e uma linha intervalar 2D.

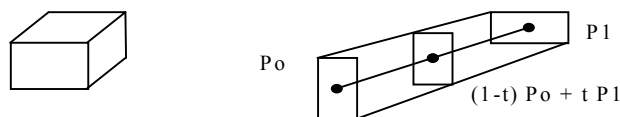


Figura 7. Ponto 3D intervalar e linha 2D intervalar.

Segundo Hu e Patrikalaris (1986), a simetria de incidência e a transitividade de incidência são exemplos de situações onde nota-se claramente a robustez superior da aritmética intervalar contra a aritmética de ponto flutuante. O problema da simetria de incidência no contexto da aritmética do ponto flutuante significa que um ponto pode ser incidente a outro mas não vice versa. Em aritmética intervalar, uma vez que o ponto é considerado incidente a outro, este também deve ser incidente ao primeiro.

No problema de transitividade de incidência, devido ao uso da tolerância pré-fixada em aritmética de ponto flutuante, um ponto A pode ser incidente a um ponto B, e este pode ser incidente a um ponto C. Mas o ponto C pode não ser incidente ao ponto A, pois a tolerância não é grande o suficiente para alcançar o ponto A. Em aritmética intervalar, neste caso, os pontos incidentes A e B são repostos por um novo ponto que cobre estes dois pontos, ou seja, o intervalo aumenta. Deste modo, o ponto C torna-se incidente a este novo ponto. A Fig. (8) ilustra o problema da transitividade de incidência.

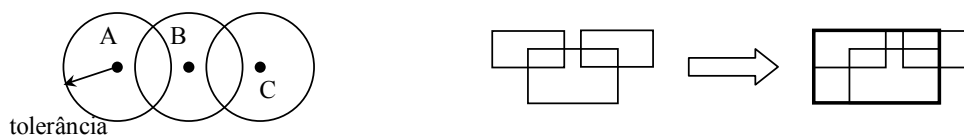


Figura 8. Transitividade de incidência com aritmética do ponto flutuante e aritmética intervalar

5. Dimensão Relativa

A dimensão relativa relaciona elementos primitivos de dois sólidos distintos. Considere o exemplo ilustrado na Fig. (9) onde temos os sólidos A e B distanciados de dez unidades.

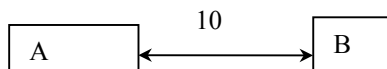


Figura 9. Dimensão relativa entre sólido A e B.

O usuário poderá modificar este valor para zero e neste caso um dos sólidos deve ser movimentado para que a distância entre eles seja nula. Entretanto, por problemas numéricos, possivelmente estas duas faces podem não ficar paralelas coincidentes. A idéia deste trabalho é utilizar esta informação definida pelo usuário na definição das Operações Booleanas.

6. Implementação

O modelador de sólidos possui como núcleo a estrutura de dados, e deseja-se comparar um modelador de sólidos convencional implementado em aritmética de ponto flutuante e um modelador de sólidos utilizando aritmética intervalar. Para que não fossem implementados dois modeladores de sólidos, foram estudados os “*templates*” e o STL (“*Standard Template Library*”) que permitem parametrizar o conteúdo das estruturas de dados (principalmente lista ligada). O uso da aritmética intervalar visa diminuir o erro causado por problemas de precisão numérica (aritmética de ponto flutuante) e instabilidades de cálculos, segundo Hoffman (1989).

O protótipo foi implementado utilizando OpenGL para saída gráfica (Woo, 1998). Foram utilizadas algumas ferramentas para o OpenGL como GLUT (“OpenGL Utility Toolkit”) de Kilgard (2000) e o GLUI (“User Interface Library to OpenGL”) de Rademacher (1999).

6.1. Estrutura de dados

Uma nova estrutura (Fig. (10)), baseada na estrutura “*halfedge*”, descrita por Mäntylä, foi implementada.

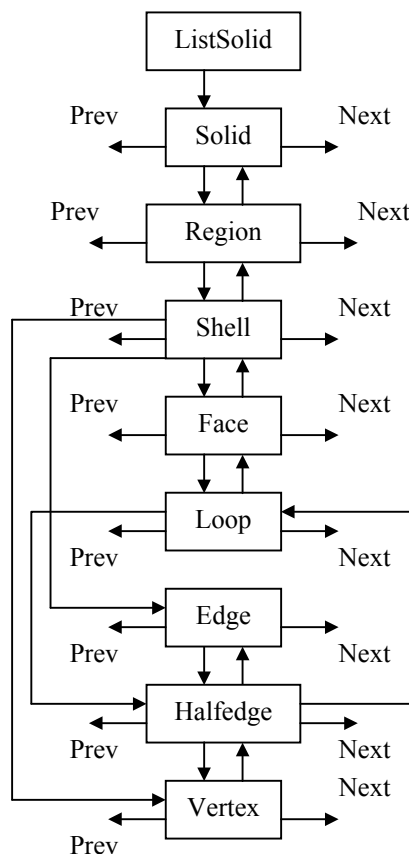


Figura 10. Hierarquia dos elementos da estrutura de dados.

Uma breve descrição de cada elemento é apresentada a seguir.

A classe “*ListSolid*” é a que contém a lista dos sólidos. Por existir apenas uma lista de sólidos no modelador de sólidos, esta classe foi implementada utilizando-se o recurso de objeto “*singleton*” de Gabrilovich (1999), que utiliza “*auto pointer*” (Sutter, 1999). O “*auto pointer*” é um ponteiro que ao sair de contexto automaticamente libera a memória para onde ela aponta. Ele não permite que existam dois ou mais ponteiros apontando para a mesma região de memória, sendo permitido apenas um “*auto pointer*” apontar para uma região de memória. Este recurso de programação C++ não irá permitir a existência de não mais que uma instância, e manterá a consistência da estrutura de dados, quando ela estiver sendo removida. Este mesmo recurso também foi utilizado para implementar os Operadores de Euler, pelo mesmo motivo.

A classe “*Solid*” foi implementada de modo convencional, pois podem existir várias instâncias dela e ela não possui ponteiro de retorno pois só existe uma instância da “*ListSolid*”. A classe “*Region*” possui implementação semelhante a da classe “*Solid*” com mais um método para retornar o ponteiro para o sólido que possui o “*Region*”. A classe “*Vertex*” possui a coordenada do vértice (*x, y, z, w*), seu identificador e um ponteiro para uma “*halfedge*”. A classe “*Edge*” possui seu identificador e ponteiros para duas “*halfedges*”.

A classe “*Halfedge*” não é incluída em “*container list*”, sendo assim, necessita de ponteiros para a próxima e a anterior “*halfedge*”. Possui o seguinte formato:

```
template <class T>
class THalfedge {
public:
    THalfedge() // constructor
    ~THalfedge() // destructor
    THalfedge<T>* mate(void) // retorna "halfedge" oposta
    TLoop<T>* Wloop() // retorna o ponteiro para o laço
    TEdge<T>* Edg() // retorna o ponteiro para o aresta
    THalfedge<T>* Nxt() // retorna o ponteiro para a próxima "halfedge"
    THalfedge<T>* Prv() // retorna o ponteiro para a anterior "halfedge"
    TVertex<T>* Vtx() // retorna o ponteiro para o vértice
    int ou void No (void ou int) // retorna ou seleciona valor para o identificador do sólido
private:
    TLoop<T> *wloop; // ponteiro para o laço
    TEdge<T> *edg; // ponteiro para a aresta
    THalfedge<T> *nxt; // ponteiro para a próxima "halfedge"
    THalfedge<T> *prv; // ponteiro para a anterior "halfedge"
    TVertex<T> *vtx; // ponteiro para o vértice
    int halfedgeno; // identificador da "halfedge"
}
```

A classe “*face*” possui uma lista de “*loop*” e um ponteiro específico para indicar qual o laço externo que delimita a face. A classe “*loop*” possui um ponteiro de retorno para a face e um para uma “*halfedge*”, que é início de um ciclo de “*halfedge*” que forma o laço.

A aritmética intervalar é utilizada nesta parte da implementação. Um exemplo simples do uso da aritmética intervalar é caso da divisão do número um por três. Sabemos que este valor é uma dízima 0.333333..., mas o microcomputador apresentará o seguinte valor (usando tipo “*double*”):

Ex.: $1/3 = 0.33333333333333331000$

Ao se utilizar a aritmética intervalar, esse valor fica armazenado como:

Ex.: $1/3 = [0.33333333333333326000 \quad 0.3333333333333337000]$

Podemos notar que é coerente o valor, pois o valor real é interno ao intervalo.

Logo, a comparação entre elementos geométricos (como vértice A é igual ao vértice B, ou vértice A está contido no plano C) é mais robusta, sem depender de uma tolerância fixa pré definida.

O próximo exemplo, ilustra a comparação entre a abordagem para verificar se um dado vértice está na mesma posição que outro vértice. A primeira, fornecida por Mäntylä, utiliza a distância Euclidiana. Se esta distância for menor que a tolerância, os vértices são considerados coincidentes. Enquanto que a segunda, mais simples, retorna incidente quando os vértices forem possivelmente iguais (existe intersecção nos intervalos). Uma ilustração da comparação, no espaço 2D, pode ser vista na Fig. (11).



Figura 11. Comparação de Vértices (a) utilizando distância euclidiana e comparação com tolerância pré definida; (b) utilizando aritmética intervalar

6.2. Operadores

Além da estrutura de dados para os elementos geométricos, existe uma outra estrutura para lidar com os Operadores de Euler. Todos os operadores de Euler foram implementados segundo a idéia de “*Singleton*”, para que existisse apenas uma instância.

Um exemplo de um operador local implementado com a ajuda dos operadores de Euler é encompridar (ou “*extrusão*” ou “*sweep*” linear). Dada uma face, cria-se um conjunto de vértices no sentido (dx, dy, dz) fornecido, de modo a extrudar a face na direção desejada. Deste modo obtém-se um conjunto de faces. Como exemplo, na Fig. (12) podemos ver um sólido onde esta ferramenta foi utilizada duas vezes. Primeiro, criou-se uma face de três vértices e em seguida aplicou-se o operador local “*sweep*” linear na direção $(dx, 0, 0)$. Em seguida, criou-se um furo seguindo os passos de utilização do operador KEMR. O passo seguinte foi criar uma face de cinco vértices no interior da face de três vértices. Executou-se o “*sweep*” linear com a face de cinco vértices na direção $(dx, 0, 0)$ e obteve-se quase um furo. O passo final é o operador KFMRH, que elimina a face redundante. A listagem abaixo ilustra a implementação deste operador:

```
template <class T>
int MSD_lowMakeSweep(TFace<T> *fac, T dx, T dy, T dz) {
    THalfedge<T> *first; // halfedge
    THalfedge<T> *scan;
    TVertex<T> *v;

    int maxv = TListSolid<T>::instance().getMaxId(Vertex), // obter maior identificador de vértice
        maxf = TListSolid<T>::instance().getMaxId(Face); // obter maior identificador de face

    list<TLoop<T>> >::iterator it = fac->FLoops_it(); // primeiro laço da face
    for ( ; it!=fac->FLead_it(); it++) { // criar vértices para todos os laços da face
        first = it->LEdg(); // primeira "halfedge" do laço "it"
        scan = first->Nxt(); // próxima "halfedge"
        v = scan->Vtx(); // vértice correspondente
        TMEV<T>::instance().low(scan, scan, maxv++, v->VCoord()[0]+dx,
            v->VCoord()[1]+dy, v->VCoord()[2]+dz);
        // cria novo vértice e aresta conforme a direção dada
        while (scan!=first) { // até encontrar a primeira "halfedge" deste laço
            // criar vértices, arestas e faces na direção dada
            v = scan->Nxt()->Vtx();
            TMEV<T>::instance().low(scan->Nxt(), scan->Nxt(), maxv++, v->VCoord()[0]+dx,
                v->VCoord()[1]+dy, v->VCoord()[2]+dz);
            TMEF<T>::instance().low(scan->Prv(), scan->Nxt()->Nxt(), maxf++);
            scan = scan->Nxt()->mate()->Nxt();
        }
        // criar a última face, de modo a fechar o conjunto
        TMEF<T>::instance().low(scan->Prv(), scan->Nxt()->Nxt(), maxf++);
    }
    return (SUCESS);
}
```

Quanto as operações booleanas, está programado o uso das dimensões relativas. No caso da dimensão relativa entre uma face A a uma face B, para verificar se elas estão em planos paralelos, é muito comum o uso do vetor normal ao plano. O uso da aritmética intervalar poderia ajudar nesta verificação de forma robusta, criando um vetor normal com um tolerância coerentemente gerada das operações aritméticas envolvidas para determinar o próprio vetor normal. Porém, de forma mais robusta ainda, será o uso da dimensão relativa, pois será informado que a face B é e deve ser paralela à face A.

Através da imposição do valor da dimensão relativa, pode-se eliminar a necessidade de certas verificações e cálculos, por exemplo: dado um bloco A, que possui as seguintes dimensões relativas: face F1 e face F2 paralelas de uma distância $d1$: $(F1/F2 d1)$, $(F3/F4 d2)$, $(F5/F6 d3)$; e um bloco B com $(F7/F8 d4)$, $(F9/F10 d5)$, $(F11/F12 d6)$; podemos informar ou impor dimensões relativas entre estes dois sólidos: $(F1/F7 d7)$, $(F3/F9 d8)$, $(F5/F12 d9)$. Deve-se executar algum processo de translação ou rotação com um dos sólidos até que as dimensões relativas $d7$, $d8$ e $d9$ sejam atendidas. Quando as dimensões relativas $d7$, $d8$ e $d9$ forem atendidas, significa que a operação booleana pode ser executada. E neste momento temos informações que nos permitem eliminar a comparação entre as normais de algumas faces. Facilmente pode-se notar que, a situação em que $d7 = d8 = d9 = 0.0$, descarta de início a verificação de incidência de três faces e um vértice.

Outro caso onde ocorrerá uma simplificação da análise, é a determinação dos pontos de intersecção entre faces. Se as faces de sólidos distintos possuírem uma dimensão relativa não nula, significa que não deve existir ponto de intersecção; portanto, não é realizada verificação e procura de pontos de intersecção para este par de faces.

O sólido gerado de uma operação também terá suas próprias dimensões relativas, que serão baseadas nas dimensões relativas dos sólidos criadores.

Estes são apenas alguns exemplos iniciais de simplificações de verificação de incidência. Mais estudos mostrarão outras simplificações e propriedades que irão aumentar a consistência e validade dos sólidos gerados pelos operadores que trabalham em aritmética intervalar com o auxílio de dimensões relativas.

A Fig. (12) ilustra um exemplo de sólido modelado em “*wireframe*” ou com “*shading*”. Na Fig. (13) está uma tomada da tela para apresentar como está a interface gráfica.

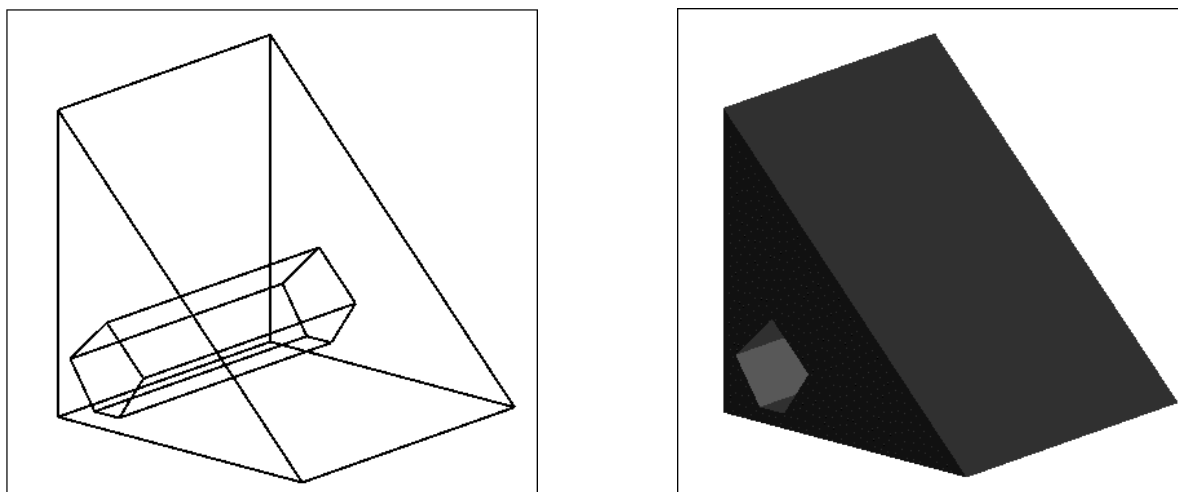


Figura 12. Exemplo de “*wireframe*” e “*shading*” obtidos da implementação atual.

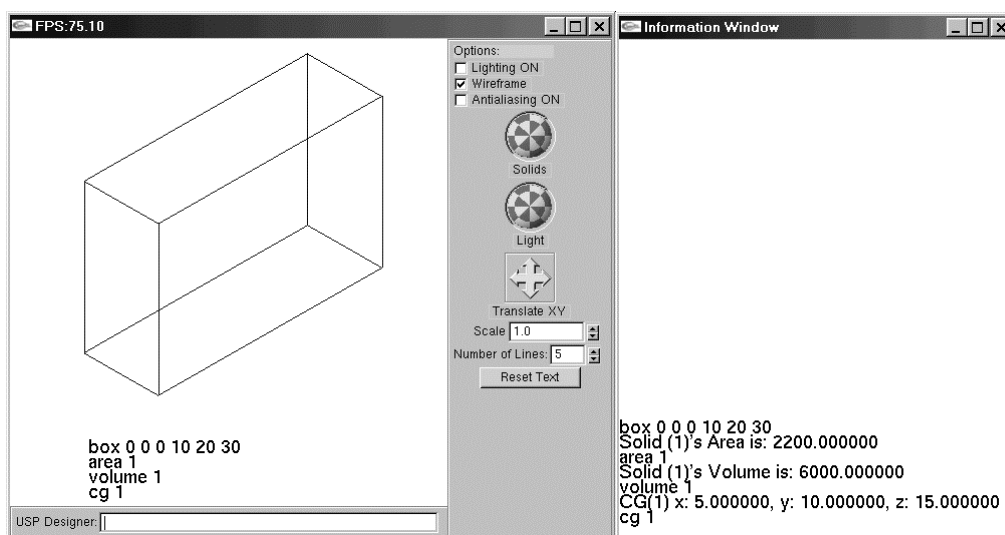


Figura 12. Interface Gráfica.

Como a estrutura de dados foi escrita com o uso de “*templates*”, pode ser realizado a rápida conversão do tipo de dados, ou seja, substituir o tipo, de “*double*” para aritmética intervalar.

7. Conclusões

Foram apresentados os conceitos básicos de um modelador de sólidos B-rep.

A aritmética intervalar aumentará a robustez dos cálculos, ao gerar uma tolerância que se adapta conforme a quantidade de operações aritméticas realizadas. Apresentou-se como as dimensões relativas poderão auxiliar nas operações booleanas, através do uso de informações que simplificarão a verificação dos casos de incidência, simplificando as próprias operações booleanas. Os sólidos criados com o uso da operações booleanas serão mais consistentes e possuirão dimensões relativas coerentes com o posicionamento relativo dos sólidos criadores, não ocorrendo perda de informação.

Foi apresentado o protótipo de um modelador de sólido B-Rep que utiliza a estrutura de dados “*halfedge*” para manipular os elementos geométricos e que utilizará aritmética intervalar para lidar com os erros de precisão numérica gerados por cálculos em ponto flutuante. Os testes iniciais já demonstram um aumento na robustez dos cálculos envolvidos.

Referências

Ammeraal, L., 1996, “STL for C++ programmers”, Wiley.

- Chiyokura, H., 1998, "Solid Modeling with DESIGNBASE: Theory and Implementation", Addison Wesley Publishing Company.
- Gabrilovich, E., 1999, "Controlling the Destruction Order of Singleton Objects", C/C++ Users Journal, Outubro, pp. 57- 61.
- Glaeser, G. & Stachel, H., 1998, "Open Geometry: OpenGL + Advanced Geometry", Springer Verlag.
- Hoffmann, C.M., 1989, "Geometric and Solid Modeling: An Introduction", Morgan & Kaufmann.
- Hu, C.Y.; Patrikalaris, N.M., Ye, X., 1996, "Robust Interval Solid Modelling Part I: Representations" Computer Aided Design 28(10), pp. 807 - 817.
- Hu, C.Y.; Patrikalaris, N.M., Ye, X., 1996, "Robust Interval Solid Modelling Part II: Boundary Evaluation" Computer Aided Design 28(10), pp. 819-830.
- Kilgard, M.J., 2000, "GLUT – OpenGL Utility Toolkit", <http://reality.sgi.com/mjk/glut3/>, versão 3.7 beta
- Mäntylä, M., 1988, "An Introduction to Solid Modeling", Computer Science Press.
- Rademacher, Paul, 1999, "GLUI – User Interface to OpenGL", versão 2.0 beta, <http://www.cs.unc.edu/~rademach/glui/>
- Sutter, H., 1999, "Using auto_ptr Effectively", C/C++ Users Journal, Outubro, pp. 63 - 67.
- Tsuzuki, M.S.G., 1991, "Modelagem de Sólidos: Representação por Fronteira (B-rep)", XI Congresso Brasileiro de Engenharia Mecânica, dezembro, pp. 611 – 614.
- Tsuzuki, M.S.G., 1995, "Contribution to the Representation of Parametric Features Applied to CAD/CAE/CAM Systems", Boletim Técnico da Escola Politécnica da USP
- Woo, Mason et al, 1998, "OpenGL Programming Guide", Addison – Wesley Publishing Company,

IMPLEMENTATION OF BOOLEAN OPERATION USING INTERVAL ARITHMETIC IN B-REP SOLID MODELLING

Marcelo Shimada

Polytechnic School – University of São Paulo
 Mechatronics Systems Department
 Av. Prof. Mello Moraes, 2231
shimada@usp.br

Marcos de Sales Guerra Tsuzuki

Polytechnic School – University of São Paulo
 Mechatronics Systems Department
 Av. Prof. Mello Moraes, 2231
mtsuzuki@usp.br

Abstract. In this work, we will consider the use of interval arithmetic and relative dimensions to increase robustness of geometric classification algorithms. The interval arithmetic incorporates truncate errors, so that, eliminates the need of defining a fixed tolerance to do the comparison between float point numbers. However, it is not possible to convert directly the algorithms using float point to algorithms using interval arithmetic implementation, so that, there is a need of total reformulation. So, this work will show the basic concepts of a B-Rep Solid Modelling, the interval arithmetic and the relative dimensions.

Keywords. *Solid Modeling, Boolean Operations, Interval Arithmetic, relative dimensions*

DETERMINAÇÃO DE CARACTERÍSTICAS DE FUNCIONAMENTO DE UM MOTOR PIEZELÉTRICO PARA DESLOCAMENTOS LINEARES

Hugo Leonardo Paiva Rodrigues

Departamento de Engenharia. Mecatrônica e de Sistemas Mecânicos da Escola Politécnica da USP, Av. Prof. Mello Moraes 2231, CEP:05508-900, São Paulo, SP
e-mail: hugo.rodrigues@poli.usp.br

Ricardo Cury Ibrahim

Departamento de Engenharia. Mecatrônica e de Sistemas Mecânicos da Escola Politécnica da USP, Av. Prof. Mello Moraes 2231, CEP:05508-900, São Paulo, SP
rci@usp.br

Resumo: Motores ultra-sônicos (ou piezelétricos) estão se tornando muito comuns em sistemas de posicionamento de precisão. O presente trabalho é parte de um grande projeto que visa o estudo deste tipo de motor no Brasil. O estudo consistiu na montagem de um protótipo de um motor ultra-sônico de deslocamento linear baseado em um modelo preexistente e a determinação de suas características de funcionamento. O protótipo foi interfaceado com um microcomputador e com diversos instrumentos tais como geradores de funções, osciloscópios, amplificadores, indicadores de posição, etc. para acionamento e medição de diversas propriedades. O desempenho do motor foi influenciado pelo tipo de sinal de excitação, sua frequência, tensão, e detalhes de montagem, como o contato entre estator e o rotor, materiais, etc.

Palavras chave: motor ultra-sônico; motor piezelétrico; posicionamento de precisão.

1. Introdução

Motores piezelétricos (ou ultra-sônicos) começaram a ser estudados em meados dos anos 80, principalmente no Japão, para uso em dispositivos de precisão tais como: mesas posicionadoras, sistemas de auto-foco de câmeras fotográficas, acionadores de vidros elétricos para automóveis, pequenos manipuladores (robôs), acionadores para diversos dispositivos usados em computadores, etc.

São atuadores baseados na propriedade piezelétrica inversa, ou seja, a aplicação de tensões elétricas variáveis em materiais piezelétricos convenientemente dispostos gera oscilações na superfície de um estator de maneira a deslocar outro corpo (o deslizador) por atrito (Sashida & Kenjo, 1993; Ueha & Tomikawa, 1993). A maioria dos motores piezelétricos atuais consiste de conjuntos de cerâmicas piezelétricas atuados por sinais senoidais (ou de outro tipo) defasados de forma a gerar uma onda progressiva elíptica (“travelling waves”) na superfície do estator, causando o deslocamento de um deslizador em contato (por atrito), conforme esquematizado na Fig. (1). O sistema deve operar próximo de sua frequência de ressonância de forma a se obter oscilações de maior amplitude no estator.

Este trabalho descreve as primeiras etapas de um projeto de pesquisa amplo que visa implantar este tema no Brasil e desenvolver novas tecnologias de construção desse tipo de motor. Esta etapa consistiu na construção de um primeiro protótipo simples para estudos iniciais dos fenômenos envolvidos, e na montagem de um sistema para executar diversos tipos de medições de características de funcionamento de motores piezelétricos.

Serão relatados alguns detalhes da construção do protótipo e do sistema de medição, bem como alguns resultados obtidos e problemas encontrados.

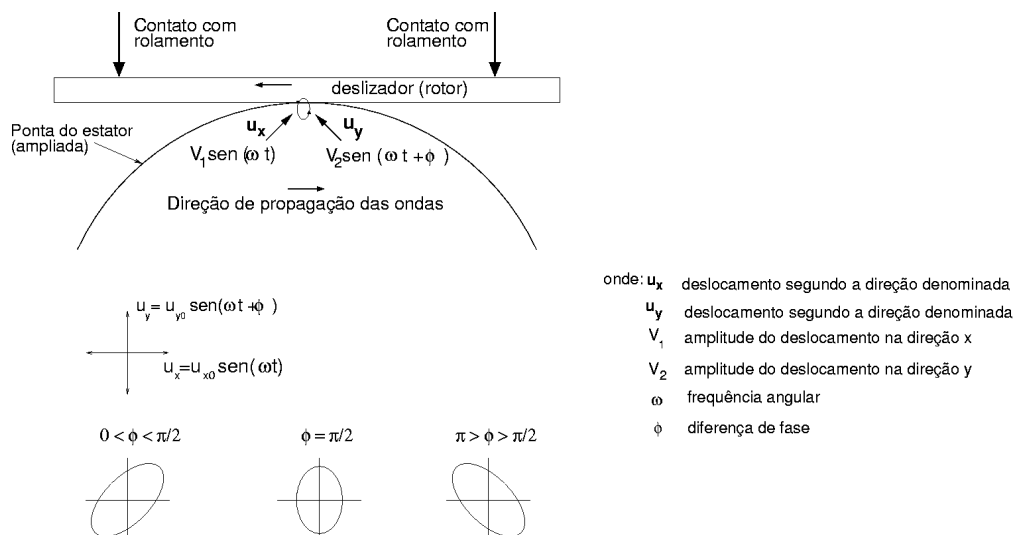


Figura 1. Formação de ondas progressivas elípticas na superfície do estator.

2. Configuração de um motor piezelétrico para deslocamentos lineares

Dentre as diversas configurações relatadas de motores piezelétricos para deslocamentos lineares procurou-se uma que apresentasse boas características de funcionamento e uma certa flexibilidade para alterações visando a facilitar inovações de projeto. Com isto, espera-se poder criar uma nova variação otimizada de motor piezelétrico num futuro próximo.

O trabalho aqui relatado trata da construção e medição de características de funcionamento de um protótipo de motor baseado no modelo de Kurosawa et al (1998). O princípio de funcionamento pode ser entendido acompanhando a Fig. (2). Dois conjuntos atuadores de cerâmicas piezelétricas dispostos ortogonalmente, atuados com tensões senoidais com diferença de fase não nula, produzem uma onda progressiva elíptica na ponta do estator. Dessa forma, uma barra servindo como deslizador, pressionada contra o estator por meio de rolamentos, deverá se mover por atrito. É importante notar que se não houver diferença de fase entre as tensões de acionamento não poderá haver a formação de ondas elípticas. O sentido do movimento do deslizador pode ser invertido com a inversão da diferença de fase das tensões de acionamento.

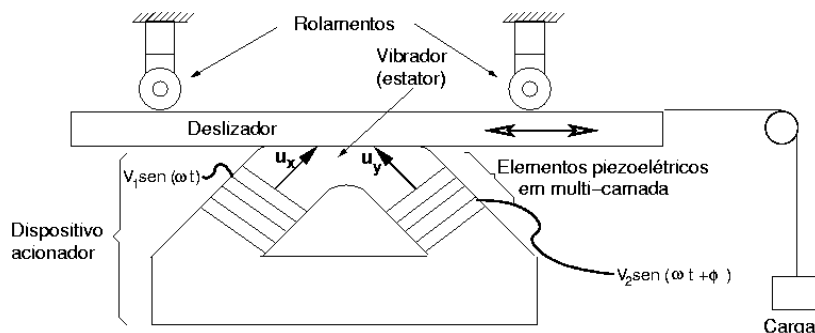


Figura 2. Diagrama de acionamento da configuração adotada de motor piezelétrico para deslocamentos lineares.

O protótipo foi montado numa base especialmente projetada para dar bastante flexibilidade para ajustes e variações de diversos parâmetros da estrutura, de forma a permitir a determinação experimental de diversas características de funcionamento. A configuração da montagem pode ser vista na Fig. (3).

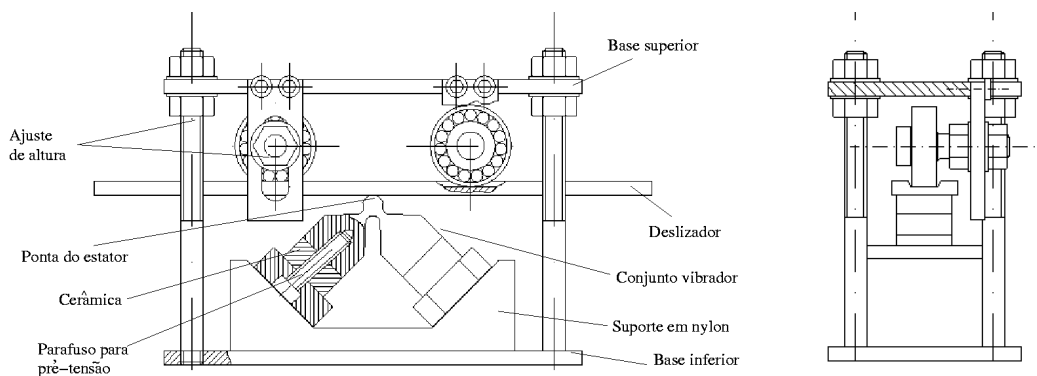


Figura 3. Vista detalhada da montagem do motor piezelétrico.

O conjunto do atuador contém cerâmicas piezelétricas em forma de discos com furo no centro para gerar as oscilações na ponta do estator. É desejável que a frequência de ressonância das cerâmicas seja igual ou bem próxima à frequência de ressonância do sistema completo, ou seja, de toda a estrutura do motor. Isto para que se obtenha uma maior amplitude de vibração na ponta do estator e se tenha maior eficiência. Desta forma, as cerâmicas devem ser ativadas por tensões com frequência próxima a de ressonância da estrutura completa.

3. Montagem do protótipo e instrumentação auxiliar

O conjunto acionador foi usinado em aço 1020, enquanto que o deslizador foi usinado em alumínio. Os dois rolamentos utilizados são do tipo 6200Z, de uma carreira de esferas, com furo interno de diâmetro 10 mm e diâmetro externo de 30 mm.

O sistema projetado permite aplicar uma pré-tensão de compressão nas cerâmicas por meio de um parafuso (Fig. (3)). Isto é muito conveniente, já que este tipo de cerâmica apresenta resistência a compressão muito maior que a tração.

Com a aplicação de pré-tensão de compressão, a região de atuação da cerâmica (movimento periódico alternando extensão e contração para geração das ondas) aumenta consideravelmente, além de se tornar mais simétrica. Entretanto, a pré-tensão aplicada em cada um dos dois conjuntos atuadores deve a mesma, devido à simetria do sistema.

É mais conveniente usar em cada conjunto várias cerâmicas empilhadas, pois isto aumenta a amplitude das oscilações e diminui a tensão de alimentação necessária. Entretanto, nesta primeira etapa da pesquisa foi utilizada uma única cerâmica em cada conjunto. Como a frequência de ressonância é inversamente proporcional à espessura do material em forma de disco (Ikeda, 1990), foi utilizada uma cerâmica relativamente espessa, de 10 mm de espessura, que estava disponível em nosso laboratório. O diâmetro externo é de 25 mm, enquanto que o furo central é de aproximadamente 8 mm de diâmetro.

Como a estrutura é metálica, é importante isolar bem as cerâmicas já que serão acionadas por tensões relativamente altas. Utilizou-se uma chapa fina de cobre coberta com isolante em uma das faces para facilitar as conexões com os diversos instrumentos, Fig. (4).

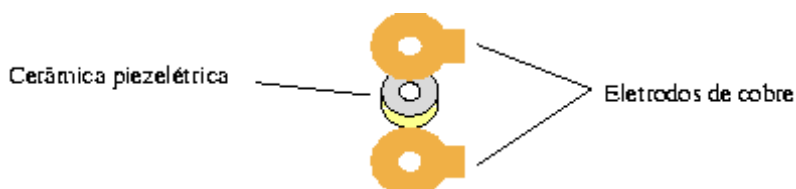


Figura 4. Disposição da cerâmica piezoeletrica com eletrodos auxiliares de cobre para fixação de cabos de conexão com instrumentos.

As dimensões do conjunto atuador/estator seguem as dimensões propostas por Kurosawa et al (1998).

No processo de montagem, é importante observar que este tipo de motor tem condições de operação bastante estritas, sendo necessária uma boa ajustagem de seus componentes. Pode-se notar da Fig. (3) que a pressão de contato entre o deslizador e a ponta do estator pode ser facilmente modificada variando-se a altura dos rolamentos e/ou a altura entre as bases inferior e superior.

O suporte de nylon é utilizado como um encaixe para fixação do conjunto acionador. A seleção do material teve como meta restringir as oscilações apenas na ponta do estator.

A configuração final pode ser vista na Fig. (5).

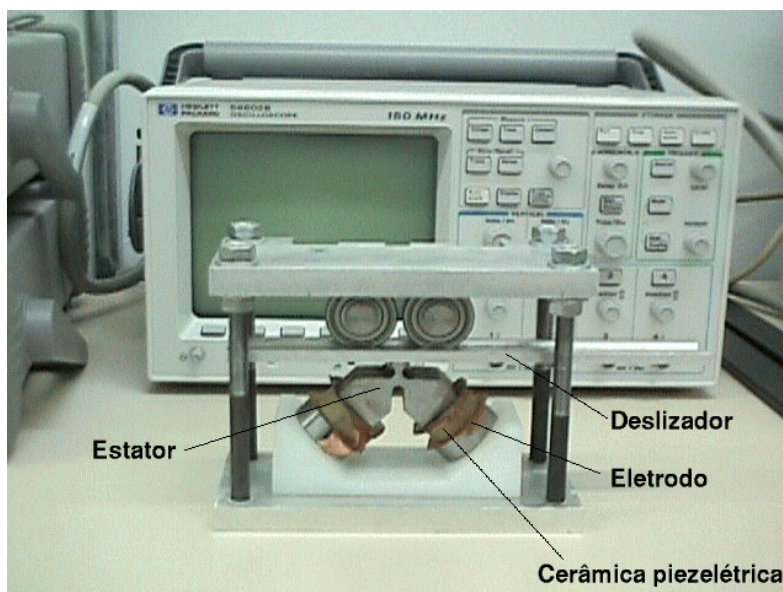


Figura 5. Foto do motor em sua configuração final.

Foi montado um sistema com diversos instrumentos para acionamento, testes e aquisição de dados do protótipo. O sistema é constituído dos seguintes instrumentos:

- Dois geradores de funções programáveis Agilent 33120A com conexão serial e GPIB, com acessório Agilent Opção 001 que permite a sincronização da base de tempo destes dois instrumentos. Dessa forma, com o auxílio de um osciloscópio, é possível enviar sinais senoidais defasados de 90° aos dois conjuntos vibradores do estator. A frequência dos sinais pode ser ajustada com grande precisão, sendo bastante adequado para encontrar a frequência de ressonância do motor.
- Osciloscópio digital de 4 canais HP 54602B com conexão GPIB e módulo matemático.
- Relógio comparador digital da Mitutoyo, com capacidade de medição de até 50 mm e conexão serial.

- Microcomputador com placa GPIB e software HP VEE, que é uma interface gráfica amigável para programação e aquisição de dados dos equipamentos.

Uma visão geral do sistema em operação pode ser obtida da Fig. (6). Outro trabalho (Simões, 2001) trata do desenvolvimento de aplicativos para interfaceamento dos diversos instrumentos e do motor com um microcomputador

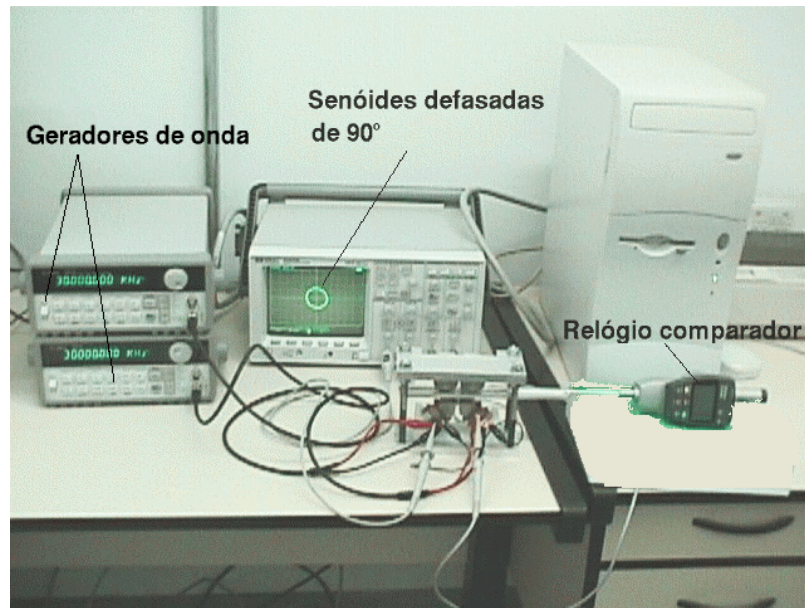


Figura 6. Instrumentação para acionamento e medição de propriedades de funcionamento do motor piezelétrico.

4. Resultados

O aperto de pré-tensão foi feito de forma que cada um dos conjuntos tivesse mesma freqüência de ressonância, medida por meio de um analisador de impedâncias da Hewlett Packard modelo HP 4191. As curvas de impedância e de fase para um dos conjuntos podem ser vistas na Fig. (7).

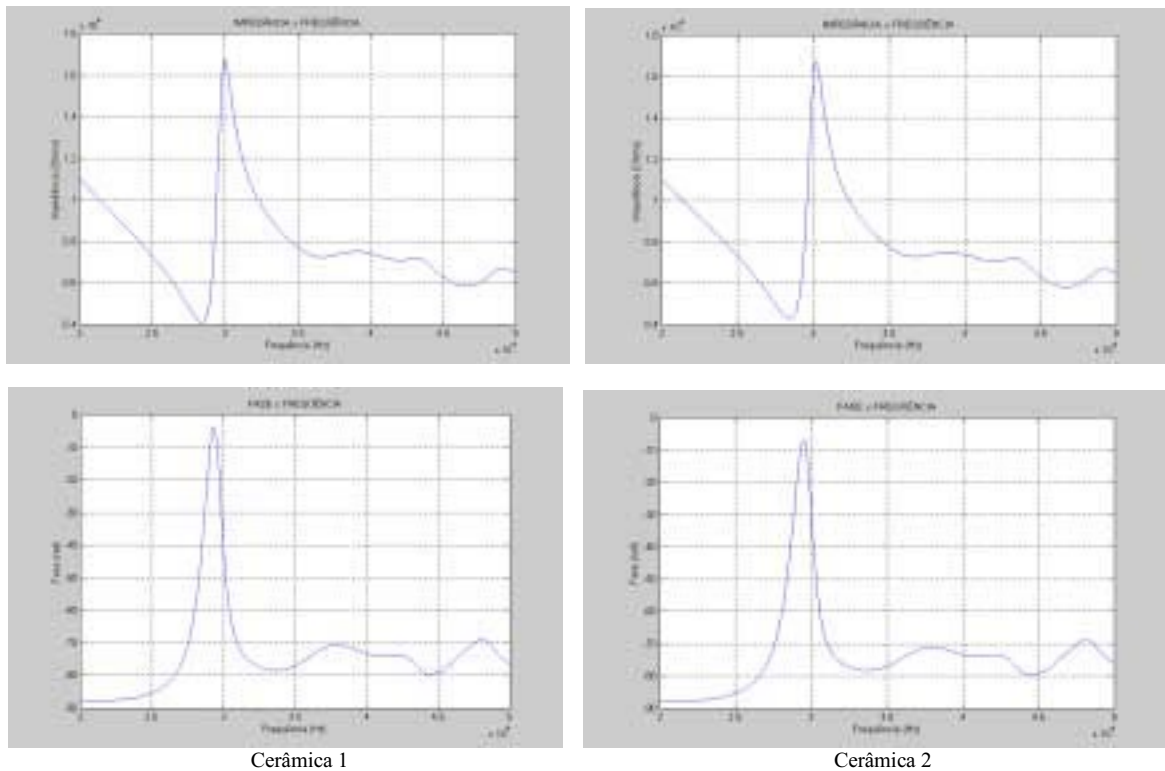


Figura 7. Curvas de impedância e de fase das duas cerâmicas após a aplicação de pré-tensão de compressão.

Das curvas de impedância, é possível inferir que os dois conjuntos foram bem ajustados e que a frequência de ressonância para o sistema atuador é próximo de 30 kHz. Esse valor é o mesmo obtido por Kurosawa et al (1998), e é foi valor adotado para atuação do protótipo. Na Figura (7) foi apresentado apenas o modo de vibração longitudinal (axial), que é o mais interessante para esta configuração de motor. Foram detectados outros modos de vibração em frequências mais altas.

Como foi utilizado um único disco espesso de cerâmica piezelétrica para cada conjunto atuador, ao invés de pilhas de discos mais finos, a tensão de atuação para que fosse notado movimento do deslizador foi superior a 400 V. Essa tensão não pode ser alcançada apenas com o gerador de função (tensão máxima de saída de 10 V), sendo necessário o uso de amplificadores sofisticados que trabalham em frequências altas, Fig. (8). Infelizmente, o uso dos amplificadores disponíveis torna o sistema um tanto instável e provoca um certo desequilíbrio nas frequências de vibração das cerâmicas e no ângulo de defasagem. No estudo, foram utilizados amplificadores da Amplifier Research modelo 150A100A.

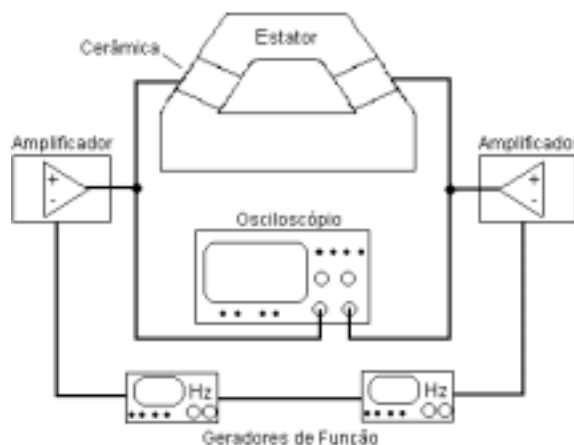


Figura 8. Esquema de acionamento das cerâmicas com o uso de amplificadores.

Desta forma, ainda é necessário projetar um sistema amplificador próprio para este tipo de aplicação, de pequenas dimensões, e que possa estar integrado ao corpo do motor. E também deverão ser utilizados discos de cerâmicas finos empilhados para tornar o motor mais eficiente e mais simples seu acionamento (tensão de acionamento mais baixa).

Portanto, o movimento do deslizador pôde ser constatado para frequências de acionamento próximas de 30 kHz, mas o sistema de acionamento ainda precisa ser melhor adaptado para que se possa utilizar o motor de forma estável e controlada.

O movimento do deslizador foi bastante influenciado pela força de contato entre o deslizador e o estator. Se o contato for leve, a transmissão da força gerada pelos conjuntos vibradores é baixa podendo não ser suficiente para vencer o atrito estático e a inércia do deslizador. Por outro lado, se o contato for excessivo, a força gerada pelos conjuntos vibradores e transmitida ao deslizador não será suficiente para vencer o atrito. Portanto, o contato entre o deslizador e o estator deve ser bem ajustado.

Ainda com relação ao contato estator/deslizador, notou-se uma certa flexão do deslizador quando era aplicada uma força de contato excessivamente grande. Essa flexão se deve ao fato de o deslizador estar apoiado em três pontos (na verdade, linhas), conforme a Fig. (9). Neste caso, o movimento do deslizador fica bastante comprometido. Uma possível solução é fabricar um deslizador com espessura maior; ou pode-se utilizar um material mais rígido que o alumínio.

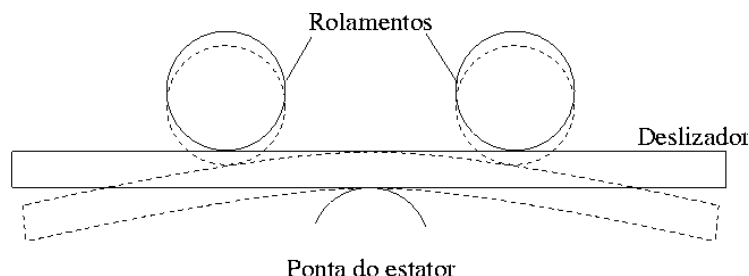


Figura 9. Flexão no deslizador devido ao ajuste excessivo dos rolamentos contra o estator.

5. Comentários e conclusões

O desempenho deste primeiro protótipo ainda é insuficiente devido ao sistema de acionamento por cerâmicas espessas. Foi constatada a necessidade de se utilizar cerâmicas empilhadas (em multicamada) para cada conjunto

atuador de modo a se reduzir a tensão de acionamento. Deve-se também desenvolver um sistema amplificador de tensão para o acionamento das cerâmicas piezelétricas. Essas dificuldades são devidas, em parte, ao tamanho relativamente grande do motor. Motores menores, para cargas mais leves, devem ser mais fáceis de construir.

Devido a uma certa complexidade da estrutura do estator, é necessário o uso do método de elementos finitos para a determinação da frequência de ressonância do sistema no projeto de novas estruturas. A seleção das cerâmicas piezelétricas dos conjuntos atuadores deve ser baseada nessa frequência.

Apesar das dificuldades encontradas, o projeto tem uma montagem flexível de modo a permitir que várias modificações sejam feitas para melhorar o sistema de acionamento no futuro.

Numa etapa posterior, espera-se estar em condições de se construir motores piezelétricos para movimentos rotativos e, também, desenvolver microatuadores baseados no mesmo princípio de funcionamento, utilizando filmes finos piezelétricos.

6. Agradecimentos

Os autores agradecem à FAPESP (Fundação de Amparo a Pesquisa do Estado de São Paulo) pelo apoio financeiro ao projeto de pesquisa sob o processo 99/05026-9, e à bolsa de estudos sob o processo 99/06028-5.

7.Referências

- Ikeda, T., 1990 “Fundamentals of Piezoelectricity”, Oxford University Press, UK
 Kurosawa, M.K.; Kodaira, O.; Tsuchitoi, Y.; Higuchi, T., 1998, “Transducer for High Speed and Large Thrust Ultrasonic Linear Motors Using Two Sandwich-Type Vibrators”, IEEE Trans. Ultrason., Ferroelect., Freq. Contr., vol. 45, pp. 1188 to 1195.
 Sashida, T.; Kenjo, T., 1993, “An Introduction to Ultrasonic Motors”, Oxford University Press, UK
 Simões, M. P., Ibrahim, R. C., 2001, “Desenvolvimento de Interface Gráfica Amigável para Aquisição de Dados e Controle de Motor Ultra-Sônico”, Anais do XVI COBEM, Uberlândia, M.G., sendo submetido.
 Ueha, S.; Tomikawa, Y., 1993, “Ultrasonic Motors”, Oxford University Press, UK

DETERMINATION OF OPERATION PERFORMANCE OF AN ULTRASONIC MOTOR FOR LINEAR DISPLACEMENT

Hugo Leonardo Paiva Rodrigues

Departamento de Engenharia. Mecatrônica e de Sistemas Mecânicos da Escola Politécnica da USP, Av. Prof. Mello Moraes 2231, CEP:05508-900, São Paulo, SP
 e-mail: hugo.rodrigues@poli.usp.br

Ricardo Cury Ibrahim

Departamento de Engenharia. Mecatrônica e de Sistemas Mecânicos da Escola Politécnica da USP, Av. Prof. Mello Moraes 2231, CEP:05508-900, São Paulo, SP
 rci@usp.br

Abstract: *Ultrasonic (piezoelectric) motors are becoming very common in precision positioning systems. The present work is part of large project aimed at designing this type of motor in Brazil. The study consisted on assembling a prototype of an ultrasonic motor for linear displacement based on a previous model, and determination of its operating features. The prototype was interfaced to a microcomputer and to several instruments such as function generators, oscilloscope, amplifiers, position indicator, etc. for driving and to measure several properties. The motor performance was influenced by the type of driving signal, its frequency, voltage, and by assembling matters such as contact between rotor and stator, materials, etc.*

Keywords. *ultrasonic motor; piezoelectric motor; precision positioning*

Multiple Neural Networks in Flexible Link Control Using Feedback-Error-Learning

Areolino de Almeida Neto

Universidade Federal do Maranhão, CEP 65.085-580 São Luís-MA, Brazil
areolino@mec.ita.br

Luiz Carlos S. Góes

Instituto Tecnológico de Aeronáutica, CEP 12228-900 São José dos Campos-SP, Brazil
goes@mec.ita.br

Cairo L. Nascimento Jr.

Instituto Tecnológico de Aeronáutica, CEP 12228-900 São José dos Campos-SP, Brazil
cairo@ita.cta.br

Abstract: In this paper two different control approaches using Feedback-Error-Learning are combined. These approaches are used for neural control of a flexible link in order to acquire the inverse dynamic model of the plant. Such systems are characterized as a non-minimum phase system, which is difficult to be controlled by most techniques and by only one neural network, hence this difficulty is overcome by the approaches discussed here: one redefines the output of the plant and the other modifies the reference input. And besides, these approaches are combined in order to obtain a better trained neural ensemble and so a better performance is achieved in neural control.

Keywords: multiple neural networks, combining neural networks, neural control, flexible link control, feedback-error-learning

1. Introduction

Flexible link systems are being used in many areas, for instance the aerospace industry, mainly due to the use of light-weight materials in large space structures and flexible space robots. The use of lightweight structures has some advantages, which results in an overall system with a lower energy consumption, faster operation and lower cost. However, these structures have disadvantages too and the principal one is the control of flexible systems is more difficult to implement than the control of rigid bodies, because the greater influence of unmodelled dynamics, coupling effects, structural non-linearities, and errors in the estimation of physical parameters.

In this paper, we present the combination of two approaches that use neural network (NN) techniques to control a flexible link system. The usage of first approach for controlling flexible link were proposed by Talebi et al. (1998) and the usage of second approach was proposed by Rios Neto et al. (1999)-1. These structures are studied, tested and compared in Almeida Neto et al. (2000)-1, Almeida Neto et al. (2000)-2 and Almeida Neto et al. (2001). All structures use a NN in a control strategy known as Feedback-Error-Learning (FEL). This strategy, described in Miyamoto et al. (1988) and Gomi et al. (1993), utilizes a conventional feedback controller (CFC) concurrently with a NN adapted on-line using the output of the CFC as its error signal. The CFC should at least be able to stabilize the system under control when used without the neural network. The Feedback-Error-Learning strategy aims to acquire the inverse dynamics model of the plant under control.

One great difficulty for most control strategies when dealing with flexible systems is the fact that they are in general non-minimum phase plants. In the first approach this problem is surpassed by feed-backing a redefined output of the flexible system, which is also used to generate the input signal for the NN controller. In the second approach the non-minimum phase characteristic of the plant is dealt by using as the input for the CFC the comparison between a delayed reference input signal and the actual output of the flexible system, while the NN controller receives the reference signal after it has been fed through a tapped-delay line, Nascimento Jr. et al. (1994).

Due to the difficulty in getting only one NN well trained, i.e., to find a good minimum of output error signal by one NN, both approaches are combined in order to make good use of each approach. The way to combine several NNs is object of recent researches, principally in control of dynamical systems. In this paper we propose a new way to combine the neural nets, which is similar to increasing topology scheme, that means, one NN is incorporated to control system after the others had already been incorporated to it. This procedure tries to find a global minimum using the contribution of the prior neural nets used and adding the improvement of the recent neural net incorporated to the system.

This paper is structured as follows: in the second section the nonlinear mathematical model of the flexible link system is introduced and the experimental setup is presented in the third section. In the fourth section output redefinition approach for tip position control is discussed, in the fifth section the input delayed approach is investigated, in the sixth section the procedure of random search used is detailed and some experimental results are presented, and in the last section conclusions are drawn.

2. Mathematical Modeling

Figure 1 shows a structure composed of a flexible beam where one end is fixed on a rigid rotating hub and the other

end is free. The flexible structure is excited by an actuator at the hub with a torque τ . The structure has a rigid body angular displacement θ and an elastic deformation $y(x,t)$, where x is the position of a point along the beam, such that $0 \leq x \leq l$, and l is the length of the beam. The variable to be controlled is the total angular displacement of the beam tip, v_{tip} written as $v_{tip} = \theta + y(l,t)/l$.

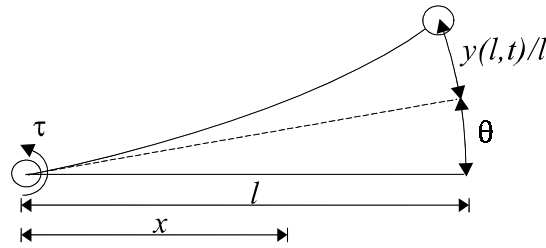


Figure 1 – Schema of the Flexible Link

Considering I_h the hub inertia, r the radius of the hub, ρ the mass per unit of length of one of the flexible appendage, l the length of the appendage, m_t the mass of the accelerometer at the tip of the flexible link, the line (') and dot (.) the partial derivatives with respect to space and time, respectively, the kinetic energy of the flexible system can be expressed in the following equation:

$$\begin{aligned}
 T = & \frac{1}{2} \rho \int_0^l (r+x)^2 dx \dot{\theta}^2 + \rho \int_0^l (r+x) \dot{y} dx \dot{\theta} + \\
 & \frac{1}{2} \rho \int_0^l y^2 dx + \frac{1}{2} m_t (r+l)^2 \dot{\theta}^2 + \frac{1}{2} I_h \dot{\theta}^2 + \\
 & m_t (r+l) \dot{y}(l) \dot{\theta} + \frac{1}{2} m_t \dot{y}^2(l) + \frac{1}{2} m_t y^2(l) \dot{\theta}^2 + \\
 & I_h \dot{\theta} \dot{y}'(0) + \frac{1}{2} I_h \dot{y}'^2(0) + \frac{1}{2} \rho \int_0^l \dot{y}^2 dx \dot{\theta}^2
 \end{aligned} \tag{1}$$

The elastic potential energy is:

$$V = \frac{1}{2} \int_0^l EI y''^2 dx \tag{2}$$

The parameter E is the modulus of elasticity and I is the cross-sectional area moment of inertia. The assumed modes method is used to transform the distributed system in a discrete one. The mode shape function $\phi_j(x)$ and the characteristic equation taken are described in Rios Neto (1998). Applying the generalized Lagrange equation, Eq. (3), where the Lagrangian is $L = T - V$, $q = [\theta \eta_1 \eta_2]^T$ (one rigid body angular displacement and two elastic modes), and $Q = [\tau \ 0 \ 0]^T$, results the matrix equation, Eq. (4), for the discrete flexible system:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = Q \tag{3}$$

$$M \ddot{q} + Kq + H[\eta \dot{\theta}] = F \tag{4}$$

$$F = [\tau \ 0 \ 0]^T \tag{5}$$

$$H = \begin{bmatrix} 2\dot{\eta}^T [MS] \\ -[MS] \dot{\theta} \end{bmatrix}_{(3 \times 2)} \tag{6}$$

$$M = \begin{bmatrix} I_t + \eta^T [MS] \eta & [\mu s]_1 & [\mu s]_2 \\ [\mu s]^T_1 & [MS]_{11} & [MS]_{12} \\ [\mu s]^T_2 & [MS]_{21} & [MS]_{22} \end{bmatrix}_{(3 \times 3)} \tag{7}$$

$$K = \begin{bmatrix} 0 & 0 & 0 \\ 0 & K_{\eta_{11}} & K_{\eta_{12}} \\ 0 & K_{\eta_{21}} & K_{\eta_{22}} \end{bmatrix}_{(3 \times 3)} \quad (8)$$

$$\begin{aligned} \eta &= [\eta_1 \eta_2]^T \\ [MS] &= \int_0^l \rho \phi(x) \phi^T(x) dx + m_l \phi(l) \phi^T(l) \\ [\mu s] &= \int_0^l \rho x \phi'(x) dx + m_l l \phi^T(l) \\ [K] &= \int_0^l EI \phi''(x) \phi''^T(x) dx \\ I_l &= I_h + m_l l^2 \end{aligned} \quad (9)$$

3. The Output Redefinition Approach

The output redefinition approach can be performed by two different schemes. Wang et al. (1989) proposed to use as output of the flexible system the so-called *reflected tip position* (RTP), v_a , defined as $v_a = h(\theta, y(l, t)) = \theta - y(l, t)/l$. Later on, Madhavan et al. (1991) proposed a generalized version where the output is redefined as, $v_a = \theta + \alpha y(l, t)/l$, where $-1 < \alpha < 1$. In this more general version, they showed that there is a critical value $0 < \alpha^* < 1$, such that for $\alpha > \alpha^*$ the zero dynamics related to this redefined output are unstable, and for $-1 < \alpha < \alpha^*$ the zero dynamics are stable. Hence, an inverse dynamics controller can be designed to control the tip position. The value of α^* depends on the payload, and it takes its smallest value when the payload is zero.

In Talebi et al. (1998), it was proposed the usage of two possible NN structures to control the tip position of the flexible link system using the redefined output v_a . The first structure was called Inverse Dynamic Model Learning (IDML) and is not used in this work. The second structure was called Nonlinear Regulator Learning (NRL) and is shown in Fig. 2. These structures are not new, but its use in flexible link control are.

In both structures the CFC control law is $u_{cfc} = K_2(\ddot{v}_r - \ddot{v}_a) + K_1(\dot{v}_r - \dot{v}_a) + K_0(v_r - v_a)$, the learning rule for the NN is $\dot{w} = \gamma u_{cfc} \partial u_n / \partial w$, where γ is the learning rate. The NN output is $u_n = \Phi(r_n, w)$ where r_n is the NN input and w is the NN weight matrix. In the NRL structure $r_n = f_n(\ddot{v}_r, \dot{v}_r, v_r, \dot{v}_a, v_a, \dot{\theta}, y, \dot{y})$. In both structures the term v_r is the reference input signal for tip position and the term v_a is the redefined output. A conventional feedback controller is used as an ordinary controller to guarantee asymptotic stability during the learning period and its output is fed to the NN as its output error signal. After the learning phase, the NN should acquire an arbitrarily close model of the inverse dynamics of the plant. Then one can say that the error equation will give by $K_2 \ddot{e} + K_1 \dot{e} + K_0 e \cong 0$, where $e = v_r - v_a$, which means that the tracking error e should converge to zero, Talebi et al. (1998).

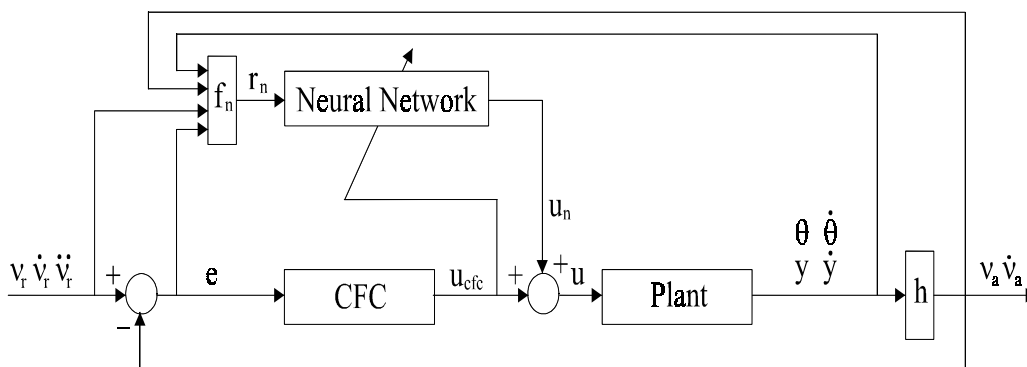


Figure 2 – Structure of the Nonlinear Regulator Learning (NRL)

4. The Input Delayed Approach

The structure of the adaptive control proposed by Rios Neto et al. (1999)-1 is shown in the Fig. 3. This structure originally proposed by Miyamoto et al. (1988) is modified by the introduction of two basic modifications: a) the high-order differentiators for the reference signal were substituted by a tapped delay line of L length; and b) the input

reference signal is delayed by M sampling periods, Nascimento Jr. et al. (1994), before it is compared with the **real** tip position of the beam, given by $v_{tip} = \theta + y(l,t)/l$.

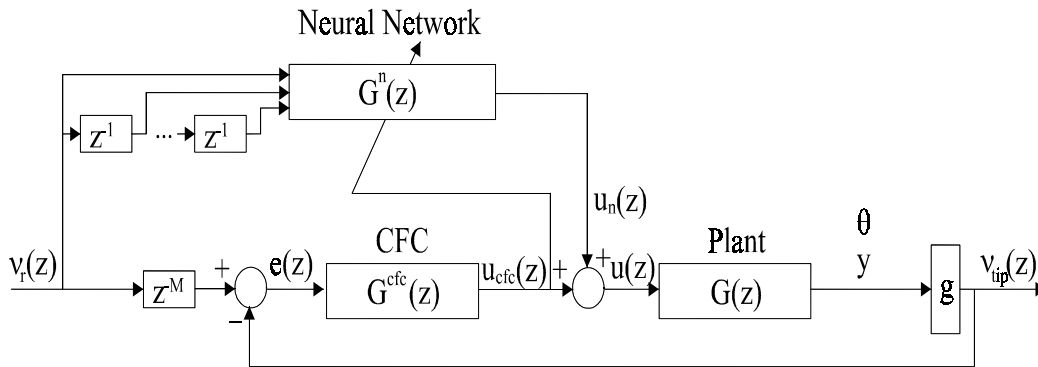


Figure 3 – Structure of Input Delayed Approach

Assuming that the plant is a stable SISO linear dynamic system with a transfer function $G(z)$, the linear NN transfer function is given by:

$$G^n(z) = \frac{u_n(z)}{v_r(z)} = \beta_0 + \beta_1 z^{-1} + \dots + \beta_L z^{-L} \quad (10)$$

Assuming that the closed loop control without the NN is stable, the polynomials $\delta(z)$ and $\psi(z)$ converge:

$$\delta(z) = \frac{G^{cfc}(z)}{1 + G(z)G^{cfc}(z)} = \sum_{j=0}^{\infty} \delta_j z^{-j} \quad (11)$$

$$\psi(z) = \frac{G(z)G^{cfc}(z)}{1 + G(z)G^{cfc}(z)} = \sum_{j=0}^{\infty} \psi_j z^{-j} \quad (12)$$

From Fig. 3, one finds the equation below such that if $G_n(z) = z^{-M}/G(z)$, then $v_{tip}(z)/v_r(z) = z^{-M}$ and $u_{cfc}(z)/v_r(z) = 0$:

$$u_{cfc}(z) = [z^{-M} \delta(z) - \psi(z)G^n(z)]v_r \quad (13)$$

The NN training problem can be seen as an optimization problem, where the objective is to obtain a NN which minimizes the cost function J as defined below:

$$J = \frac{1}{2} E \left[\left(z^{-M} \delta(z) v_r(z) - \psi(z) (\beta^*)^T v_r^*(z) \right)^2 \right] \quad (14)$$

$$\beta^* = [\beta_0 \quad \beta_1 \quad \dots \quad \beta_L]^T \quad (15)$$

$$v_r^*(z) = [v_r(z) \quad z^{-1}v_r(z) \quad \dots \quad z^{-L}v_r(z)]^T \quad (16)$$

The cost function, J , will have a unique minimum if the matrix $\partial^2 J / \partial \beta^{*2}$ is positive-definite, where:

$$\partial^2 J / \partial \beta^{*2} = E \left\{ [\psi(z) v_r^*(z)] [\psi(z) v_r^*(z)]^T \right\} \Delta F^l \quad (17)$$

The β_i parameters are found by solving the equations:

$$F^l \beta^* = F^r = E \left[\left(z^{-M} \delta(z) v_r^*(z) \right) (\psi(z) v_r^*(z)) \right] \quad (18)$$

These Parameters will have a unique solution if the input reference signal v_r has enough exciting properties, such that F^l is a positive-definite matrix. However, to calculate F^r and F^l it is necessary to know the plant transfer

function $G(z)$ and its associated polynomials $\delta(z)$ and $\psi(z)$. Hence as $G(z)$ is unknown, a learning algorithm with a simple gradient descent search can be used. In this case, the following learning rule for the parameters β_i is derived:

$$\Delta\beta_i = -\gamma \left[\frac{\partial J}{\partial \beta_i} \right] = \gamma u_{cfc,k} \sum_{j=0}^{\infty} \psi_j v_{r,k-i-j} \tag{19}$$

Since the polynomial $\psi(z)$ is unknown, Rios Neto et al. (1999)-2 proposed to use another (maybe guessed) polynomial as its approximation, such that Eq. 19 becomes:

$$\sigma(z) = \sum_{j=0}^{NL} \sigma_j z^{-j} \Rightarrow \Delta\beta_i = \gamma u_{cfc,k} \sum_{j=0}^{NL} \sigma_j v_{r,k-i-j} \tag{20}$$

5. The Combination of Two Approaches

To obtain an excellent neural system, that means, a neural system able to acquire the inverse model of the plant, it's not easy for only one NN. So one way is to change the NN used, either changing its parameters, like number of units, of layers, etc., or even changing the way to apply the NN in the control system. But it's possible to use another way to solve this problem: application of more than one NN. The usage of several NN is explored by many researches, but not so much in control systems and less in flexible link control.

In this work we combine the approaches above in order to obtain a better performance than if only one NN were used, and so only one approach were used. The figure below shows the scheme of these two approaches combined.

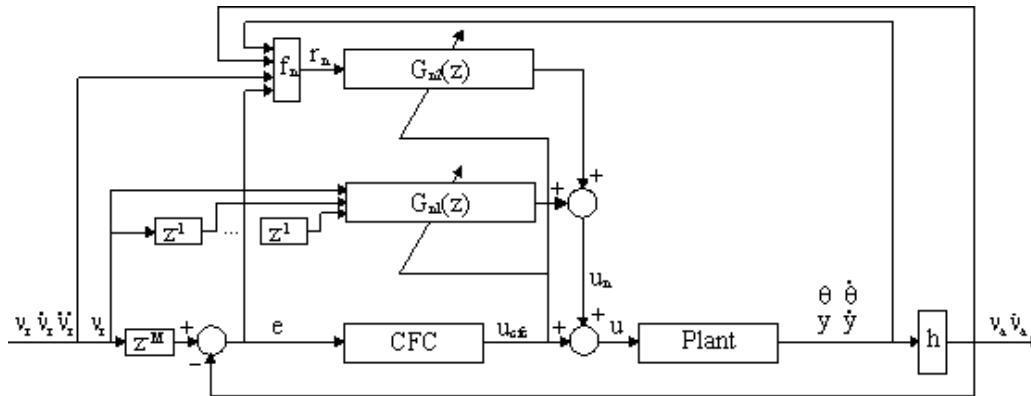


Figure 4 – Structure of Two NNs Combined Using Input Delayed and NRL Approaches

The main problem in using more than one NN is how to mix their output in order to produce the unique output of the neural system. Some problems permit several output, so each output can be provided by one NN, but in that problems which this is not possible, the output of the NNs should be mixed and the result of this mixing is considered the output of the neural system. Important works which treat this problem are 1, 2, 3 e Jacobs and Jordan (1993). The Hashem's works present a procedure to find the weights in a weighted sum of the output of the NN, while in the work of Jacobs another NN is used to provide the weighting of the output of the NNs. In essence these and others works are very similar: the outputs of the NNs should be weighted before producing the output of the neural system.

Here the output of the NNs are not weighted, because the training of one NN doesn't happen simultaneously with the training of another NN. In fact, in the beginning only one NN, let's say the G_{n1} in figure above, is used and trained to acquire the inverse model of the plant. Probably the learning of the inverse model will not occur, so this NN, G_{n1} , is not trained anymore, but it continues producing its output and feeding the plant with this output. So another NN, G_{n2} , is incorporated in the system in a such way that its output is zero in the beginning and step by step it will produce an output, which is added to the output of the previous NN. The reason for not be necessary a weight for each NN is because in the training of G_{n2} , the output of the previous trained NN continues to feed the plant and after the processing of this output by the plant and by the CFC, this value is used in the learning of the G_{n2} . The equations below show the error used by the first NN and the error used by the second NN, respectively Eq, 21 and 22, where $E(z)$ is the error, $R(z)$ is the reference and $G(z)$ is the transfer function of the plant.

$$E(z) = z^{-M} R(z) - G(U^{cfc}(z) + U^{n1}(z)) \tag{21}$$

$$E(z) = z^{-M} R(z) - G(U^{cfc}(z) + U^{n1}(z) + U^{n2}(z)) \tag{22}$$

The first advantage of this scheme is the learning of the inverse model is divided by two NNs, which is much simpler for each NN than to learn the complete inverse model. The second advantage is the dispense of the procedure to find the weights for each NN, which permits this scheme to be smaller than if were necessary the usage of some weight arrangement. The computational cost associated with the training of more than one NN can become the control unable in real implementation, because the processing of the training algorithm is much heavier than the processing of the output of the trained NN. One disadvantage of this scheme is the slow learning of the inverse model, but what is better a slow and right learning or a fast and wrong learning? Of course, the best is a fast and right learning, but this is too long.

6. Experimental Results

The results presented here were obtained from simulations. However even in this way, when two NNs were combined using the approaches described above, the performance was better. The simulations consisted of tests using the approaches above with the NNs trained by BP algorithm. Due to the nonlinear characteristics of the flexible system, the linear NN was not able to converge to a good approximation of the delayed inverse dynamics. So in this work we used a 3-layers Perceptrons (input, hidden e output layers). The weights between input and hidden layers are initialized randomly and the other weights were initialized with zero, in such way that output of all NN were zero: this procedure avoids to unstabilize the system at the beginning of training session. At all, the training goes in well-known procedure: the weights are initialized, the BP algorithm is ran and the results are obtained; if not a good result is obtained the procedure is repeated with some changes in the first NN. But after some experiences, the result is not improved only changing some parameter of the first NN, so when this occurs is time to incorporate another NN and run the BP again, but only over the second NN.

The variable controlled used in these tests was only the hub position with a sine wave reference of 90° of amplitude and a period of 6 s. For the input delayed approach L_{input} and L_{hidden} , the number of units in the input and hidden layers, were 20 and 50, respectively, while in the NRL structure $L_{input}=9$ and $L_{hidden}=50$ too. The learning rate parameters γ_{input} and γ_{hidden} used by the BP algorithm were 0.000001 and 0.000001 for all trainings. The delay for the inverse model of the system in the input delayed approach (M) was 1. For both approach, the CFC was only one PD controller with the proportional and derivative gains 45 and 22, respectively.

The figure below shows the output of the plant at the end of the training of the second NN. It's possible to see that the actual response practically accomplishes the reference signal.

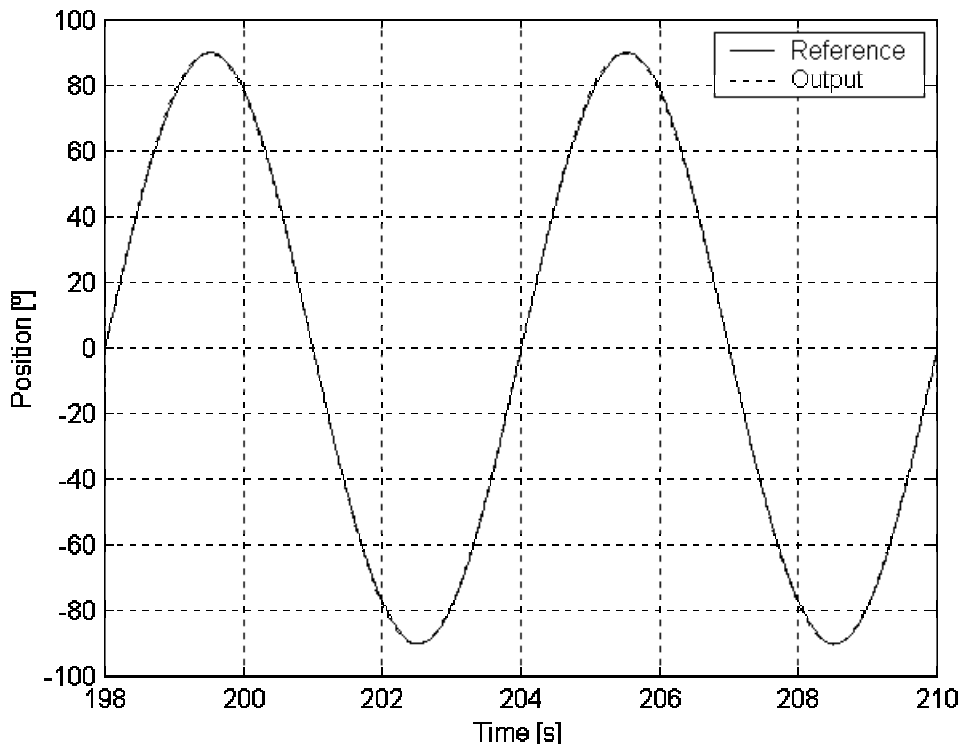


Figure 5 – Output of the System at the End of Training Using Both Structures

The next figure shows the output error during the complete training. From 0 s up to 120 s, only one NN is used with the Input Delayed Approach. As can be seen, after 50 s, the error stabilized and the NN could not improve the performance. At this point the maximum absolute error was 2.5°, that means, the maximum relative error of 2.8 %. When the second NN was incorporated in the system, the first NN continued to produce its output, so the second NN had only to improve the performance learning a part of the plant, not being necessary to learn everything about the

plant. Considering this, the second NN could increase the performance bringing the maximum absolute error to 1.25° or the equivalent relative error of 1.4 %.

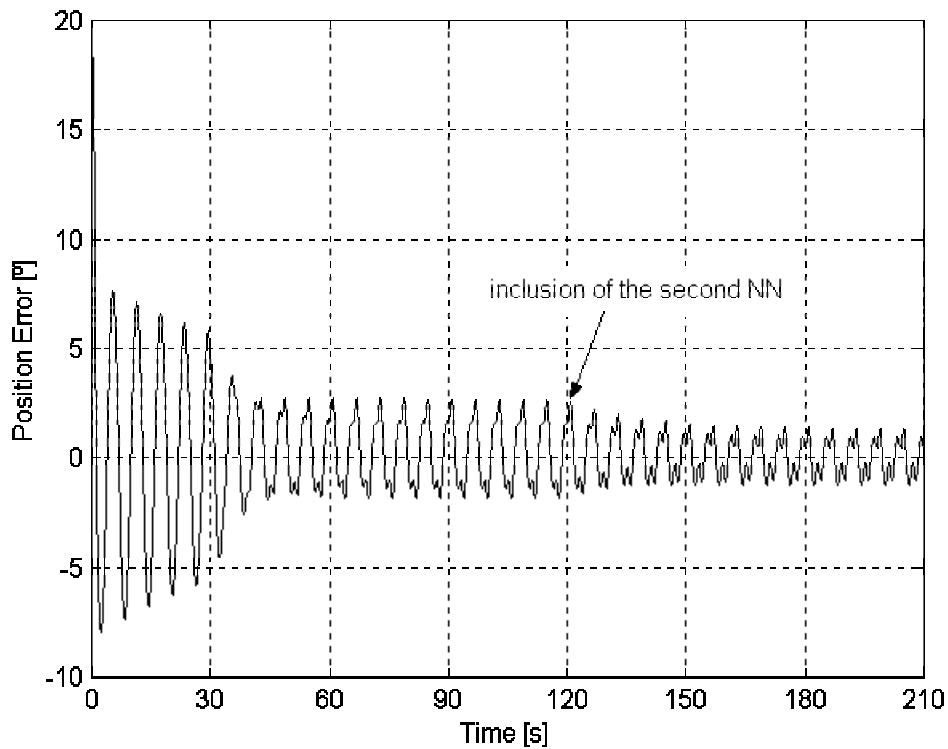


Figure 6 –Output Error for the Complete Training Using First Input Delayed Approach and after NRL Approach

In Fig. 7 it is showed the quadratic mean of hub position error during the training session. This graphic shows also the improvement due to the addition of the second NN.

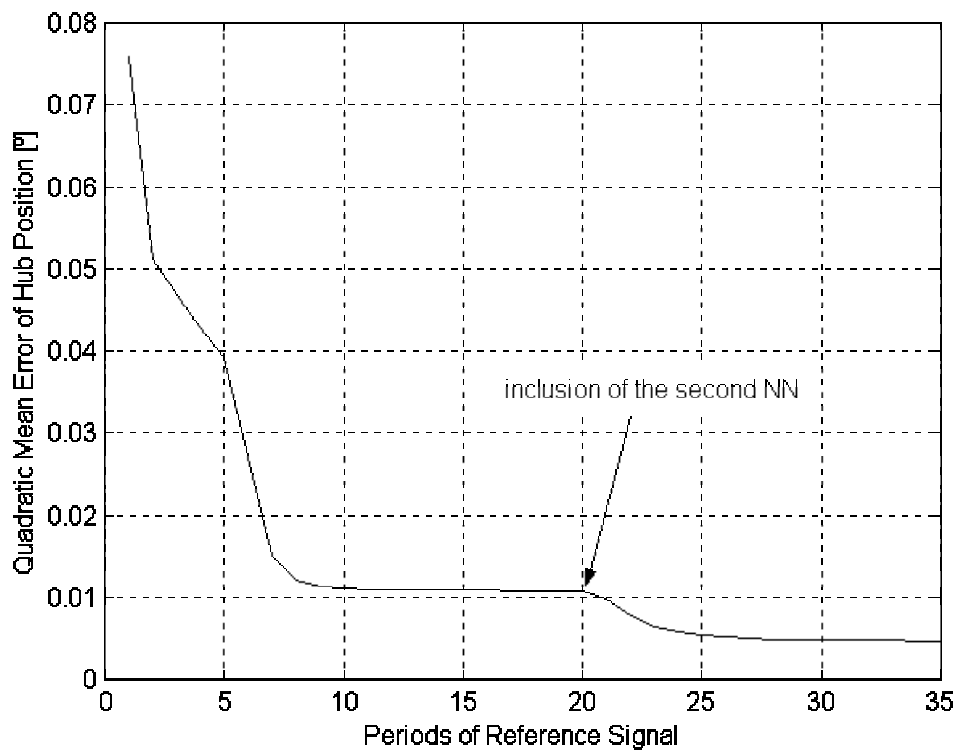


Figure 7 – Quadratic Mean Output Error for Hub Position during the Training Session

The next figure shows the quadratic mean value for the output of the controllers. In this graphic is showed the output of the CFC, the sum of the output of NNs and the sum of these signal, which is the true control signal sent to the

plant. In this figure the quadratic mean values (QMV) for the output of the CFC (U_{cfc}) is decreased, while the QMV for the sum of output of the NNs ($U_{n1}+U_{n2}$) is increased and the control law ($U = U_{cfc} + U_{n1} + U_{n2}$) is more influenced by the NNs at the end of training session than by the CFC. This evidences that in this simulation the inverse model obtained with the combination of NRL and Input Delayed structures were close to the desired model.

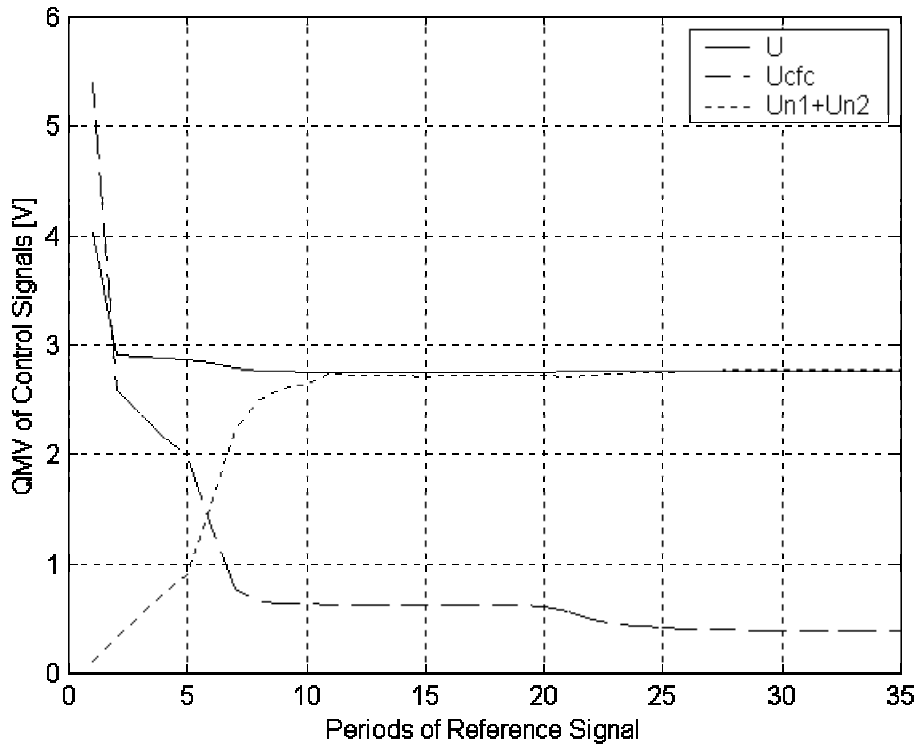


Figure 8 – Quadratic Mean Value for the CFC (U_{cfc}), NNs ($U_{n1}+U_{n2}$) and for the Control Law ($U=U_{cfc}+U_{n1}+U_{n2}$)

In Fig. 9 is showed the output of the controllers at the end of the training session. One can see that the first NN has a signal which is the principal part of the total control law, the second NN has a signal less than the output of the first NN, but greater than the output of the CFC. And the output of the CFC is bounded around the zero volt.

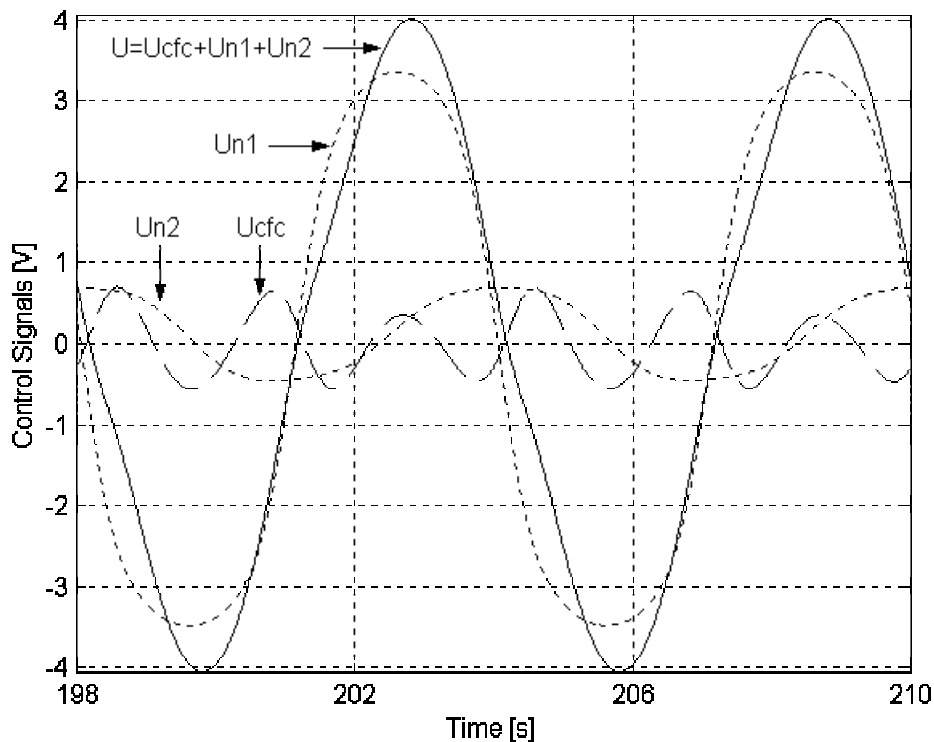


Figure 9 – Control Signals U , U_{cfc} and U_{n1} and U_{n2}

7. Conclusions

Two different structures for neural control of a flexible link were combined in a such way that only one feedback controller was used and two NNs, each one belonging to one approach, had the task of learning the inverse model of the flexible plant. In simulations this combination was successful and was showed that the possibility to increase the learning about an inverse model of a plant is much easier with more than one NN. The performance achieved with the combination was very good. More, it was presented also a way to combine the networks with no necessity of weighting procedure or weighting element, like another NN used in many others works.

8. References

- Almeida Neto, A., Rios Neto, W., Góes, L.C.S. and Nascimento Jr., C.L., 2000, "Neural Control of a Flexible Link Using Feedback-Error-Learning", Proceedings of the 3rd International Conference on Nonlinear Dynamics, Chaos, Control and Their Applications in Engineering Sciences, Vol. 6, Campos do Jordão, Brazil, pp. 242-251.
- Almeida Neto, A., Rios Neto, W., Góes, L.C.S. and Nascimento Jr., C.L., 2000, "Feedback-Error-Learning for Controlling a Flexible Link", Proceedings of the 4th Brazilian Symposium of Neural Network, Vol. 1, Rio de Janeiro, Brazil, pp. 273-278.
- Almeida Neto, A., Góes, L.C.S. and Nascimento Jr., C.L., 2001, "Flexible Link Control with Three Different Structures Using Feedback-Error-Learning", Proceedings of the 5th Brazilian Congress of Neural Network, Vol. 1, Rio de Janeiro, Brazil, pp. 637-642.
- Gomi, H. and Kawato, M., 1993, "Neural Network Control for a Closed-loop System using Feedback-Error-Learning", Neural Networks, Vol. 6., pp. 933-946.
- Hashem, S. et al., 1994, "Optimal Linear Combinations of Neural Networks: an Overview", Proceedings of IEEE Conference on Neural Networks, Vol. 3, pp. 1507-1512.
- Hashem, S., Schmeiser, B., 1995, "Improving Model Accuracy Using Optimal Linear Combinations of Trained Neural Networks", IEEE Transactions on Neural Networks, Vol. 6, N. 3, pp. 792-794.
- Hashem, S., 1997, "Algorithms for Optimal Linear Combinations of Neural Networks", Proceedings of International Conference on Neural Networks, Vol. 1, pp. 242-247.
- Jacobs, R. A., Jordan, M. I., (1993), "Learning Piecewise Control Strategies in a Modular Network Architecture", IEEE Transactions on Systems, Man and Cybernetics, Vol. 23, N. 2, pp. 337-345.
- Madhavan, S.K. and Singh, S.N., 1991, "Inverse Trajectory Control and Zero Dynamic Sensitivity of an Elastic Manipulator", Int. Journal of Robotic and Automation, pp. 179-191.
- Miyamoto, H., Kawato M., Setoyama T. and Suzuki, R., 1988, "Feedback Error-Learning Neural Network for Trajectory Control of a Robotic Manipulator", Neural Networks I, pp. 251-265.
- Nascimento Jr., C.L., 1994, "Artificial Neural Networks in Control and Optimization", PhD Thesis, Control Systems Centre, UMIST, UK.
- Rios Neto, W., 1998, "Control of a System with Flexible Appendages using Neural Networks", MSc Thesis, Technological Institute of Aeronautics (in portuguese).
- Rios Neto, W., Góes, L.C.S. and Nascimento Jr., C.L., 1999, "Experimental Neural Control of an Unconstrained Multibody System with Flexible Appendages", Proceedings of the 15th Brazilian Congress of Mechanical Engineering, Águas de Lindóia-SP, Brazil.
- Rios Neto, W., Nascimento Jr., C.L. and Góes, L.C.S., 1999, "Positional Control of a Flexible Structure Using Neural Networks", Proceedings of the 4th Brazilian Congress of Neural Networks, Vol.1, S. José dos Campos-SP, Brazil, pp. 378-383.
- Talebi, H.A., Khorasani, K. and Patel, R.V., 1998, "Neural Network Based Control Schemes for Flexible Link Manipulators: Simulations and Experiments", Neural Networks II, pp. 1357-1377.
- Wang, D. and Vidyasagar, M., 1989, "Transfer Function for a Single Flexible Link", Proceedings of the IEEE International Conference on Robotics and Automation, pp. 1042-1047.

CONTROLE DE VIBRAÇÕES UTILIZANDO LÓGICA FUZZY IMPLEMENTADA EM MICROCONTROLADORES

Israel Jorge Cárdenas Nuñez

Universidade Federal de Uberlândia – FEMEC – Faculdade de Engenharia Mecânica
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
icardenas@mecanica.ufu.br

Prof. Dr. José Francisco Ribeiro

Universidade Federal de Uberlândia – FEMEC – Faculdade de Engenharia Mecânica
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
jrribeiro@mecanica.ufu.br

Resumo. Este trabalho discute o emprego de controladores fuzzy, implementados em microcontroladores, no controle ativo de vibrações de sistemas dinâmicos. A metodologia de projeto proposta é avaliada no controle de um sistema vibratório de um grau de liberdade, acionado por atuadores magnéticos. A não linearidade dos atuadores magnéticos e as incertezas dos parâmetros do sistema motivam o uso dos controladores fuzzy que, implementados em microcontroladores, conferem ao sistema de controle elevada flexibilidade a um custo muito baixo. O controlador fuzzy proposto é do tipo Sugeno com 9 regras e três funções de pertinência sintonizadas de forma heurística. A entrada do controlador é o deslocamento do sistema vibratório e a saída é uma tensão elétrica modulada por largura de pulso (PWM). O sinal de saída aciona um driver de corrente, que alimenta o atuador e gera a força de controle.

Palavras chave: Controle, Fuzzy, Microcontroladores, Vibrações.

1. Introdução

Um dos problemas mais importantes no estudo da dinâmica dos sistemas físicos é o problema das vibrações indesejáveis. Sistemas mecânicos, quando sujeitos à excitações perturbadoras podem gerar ruídos indesejados, ocasionar desconforto aos usuários e ter o tempo de vida severamente diminuído.

A evolução vertiginosa da eletrônica nos últimos anos tem possibilitado o aparecimento de novas tecnologias na área de controle de vibrações. O controle de sistemas físicos utilizando a lógica fuzzy foi introduzida por Zadeh (1965), e pode ser aplicada em áreas onde as informações sobre o sistemas são vagas e imprecisas. As principais razões do emprego crescente da tecnologia fuzzy em controle são:

- o controlador é formulado a partir de um conhecimento intuitivo do sistema e utiliza descritores lingüísticos no seu projeto;
- o projeto do controlador dispensa o conhecimento do modelo matemático do sistema a ser controlado;
- aplica-se à sistemas lineares como a sistemas não lineares e
- é de implementação fácil e de baixo custo.

Talvez a maior vantagem do controlador fuzzy seja o seu ciclo de desenvolvimento e sua facilidade de implementação. A Figura (1) ilustra uma seqüência de projeto para o desenvolvimento de um controlador, usando controle convencional e lógica fuzzy, que evidencia esta vantagem, (Aptronix Inc., 1993).

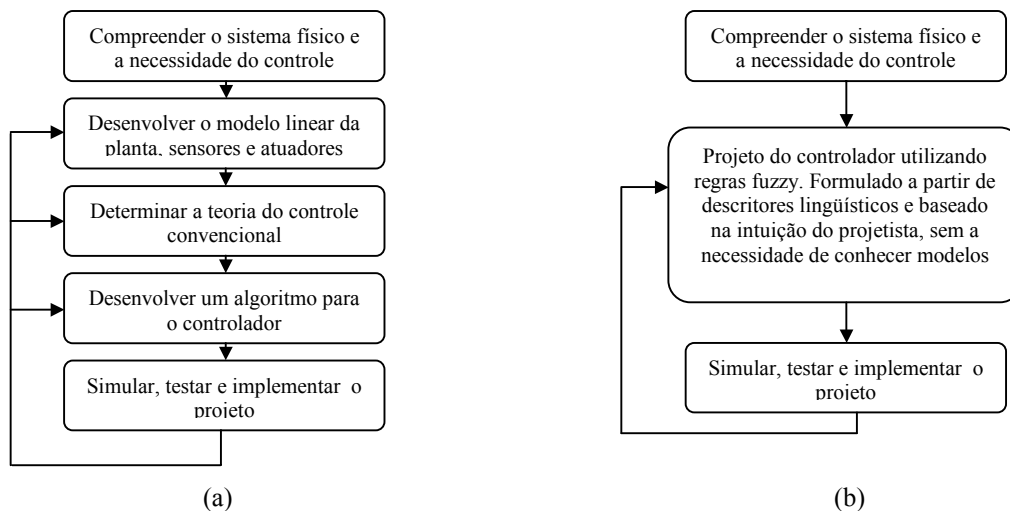


Figura 1. Etapas de um projeto usando controle convencional (a) e o controle fuzzy (b).

Dado, as características dos controladores fuzzy e a disponibilidade nos dias atuais de componentes eletrônicos sofisticados e de baixo custo, o objetivo deste trabalho é propor, implementar e testar uma unidade eletrônica para o controle de sistemas mecânicos de um grau de liberdade, baseados na lógica fuzzy e implementados em microcontroladores. O microcontrolador é um componente eletrônico que possui muitos dos periféricos dos microprocessadores comuns embutidos em uma só pastilha, o que facilita o desenvolvimento de sistemas pequenos e baratos, razoavelmente complexos e sofisticados. No item dois e três deste trabalho faz-se uma apresentação do controlador fuzzy e dos microcontroladores. Nos itens seguintes descreve-se o experimento realizado e os resultados auferidos.

2. A Lógica Fuzzy

A metodologia de projeto de um controlador fuzzy envolve a construção de uma base de regras, que utiliza operadores lógicos e funções de pertinência. O controlador fuzzy mapeia, através de um processo de inferência variáveis de entrada em variáveis de saída.

O problema de controle utilizando a lógica fuzzy obedece normalmente as seguintes etapas:

- Etapa 1: caracterização do intervalo de valores que as variáveis de entrada e as variáveis de saída podem assumir;
- Etapa 2: definição de um conjunto de funções, denominadas funções de pertinência, que mapeiam as variáveis de entrada e de saída do universo do discurso;
- Etapa 3: definição de um conjunto de regras, usando operadores lógicos, que buscam estabelecer uma relação entre valores de entrada e valores de saída;
- Etapa 4: uma vez obtidas as regras, passa-se à fase de tradução matemática da linguagem simbólica construída. Isto é realizado através da utilização de operadores lógicos definidos pela teoria dos conjuntos fuzzy. Esta etapa se divide em três sub-etapas, (Driankov et al, 1996):

- *Fuzzificação* - transforma os valores reais das variáveis de entrada em graus de pertinência de um determinado conjunto fuzzy;
- *Inferência difusa* - opera com as regras e com o resultado da fase de *fuzzificação* e gera um conjunto de variáveis fuzzy através de um mecanismo de inferência e
- *Defuzzificação* - transforma o resultado da inferência difusa em uma saída numérica. A Figura (2) ilustra estas sub-etapas.

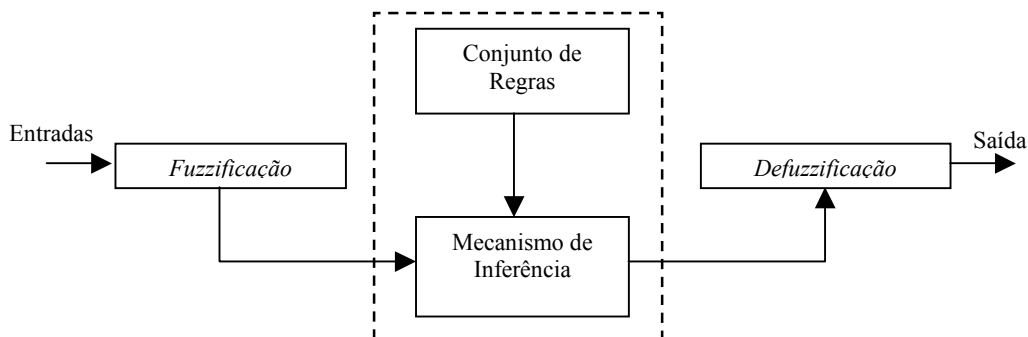


Figura 2. Sistema de inferência difuso.

3. Microcontroladores

O microcontrolador é um descendente direto do microprocessador. É utilizado em tarefas de controle e tem-se convertido na unidade fundamental de artefatos de uso cotidiano como: eletrodomésticos, máquinas ferramentas, equipamentos médicos, periféricos de computadores e uma infinidade de outras aplicações.

O microcontrolador é um sistema de um só circuito integrado, que incorpora numa única pastilha várias facilidades da eletrônica, como por exemplo memória de programa, memória de dados, portas I/O, conversores de sinais etc.

Geralmente os microcontroladores tem os seguintes componentes básicos, (Garcia, 1998):

- unidade central de processamento (CPU);
- memória de acesso aleatório para dados (RAM);
- memória apenas de leitura (ROM, EPROM);
- linhas de entradas/saídas (I/O);
- timers;
- interrupções;
- conversores A/D e
- saídas PWM.

A existência de ferramentas de desenvolvimento, baseados em linguagem de alto nível, tem facilitado, nos dias atuais, a tarefa de programação e o desenvolvimento de sistemas baseado em microcontroladores. Neste projeto foi utilizado o compilador C da PCW[®] (1), e como interface entre o compilador e o gravador de microcontroladores PICSTAR-PLUS[®] (2) utilizou-se o software MPLAB[®] (2).

Para a realização deste projeto foi utilizado o microcontrolador da Microchip[®] “PIC16C74A”, que possui 4K de memória, 192 bytes de RAM, 3 timers, 8 canais A/D de 8 bytes, 2 canais de saída PWM, frequência máxima de operação de 20 MHz e voltagem de alimentação entre 2,4 a 6,0 V.

4. Formulação do Problema de Controle

Para validar o objetivo deste trabalho, foi proposto, o controle de um mesa vibratório de um grau de liberdade, em que pretende-se controlar os deslocamentos laterais do sistema vibratório da Fig. (3), utilizando um controlador fuzzy implementado num microcontrolador.

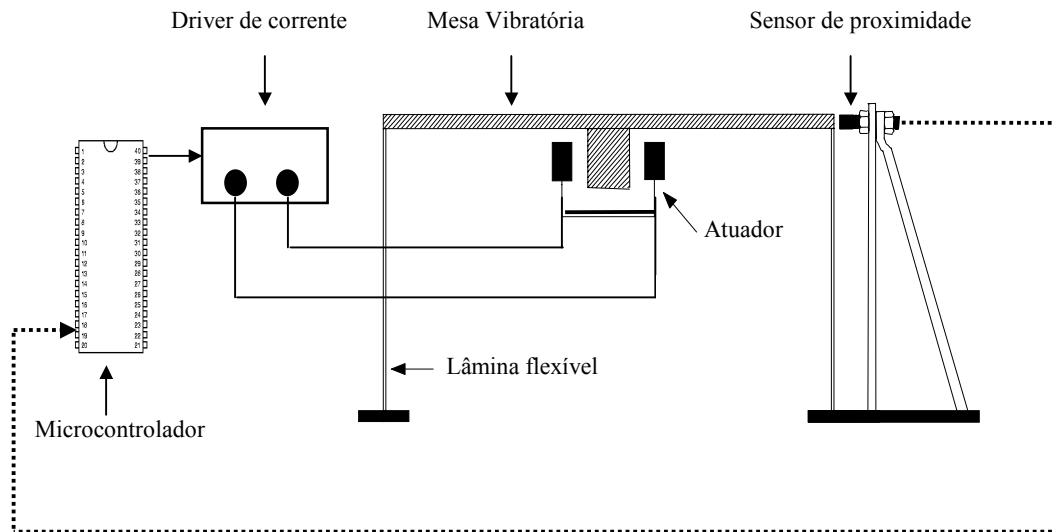


Figura 3. Sistema vibratório.

A entrada do controlador é o deslocamento do sistema vibratório, sendo este, detectado por um sensor indutivo de proximidade. A saída, é uma tensão elétrica modulada por largura de pulso (PWM). O sinal de saída aciona um driver de corrente, que alimenta o atuador, e gera a força de controle. A Figura (4) mostra os componentes do sistema completo.

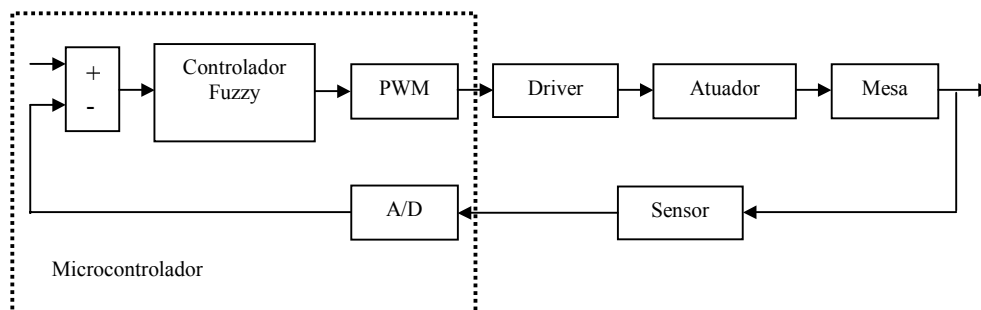


Figura 4. Componentes do sistema e do controlador.

(1) PCW[®] - Compilador C comercializado pela Custom Computer Services, Inc., www.ccsinfo.com

(2) PICSTAR-PLUS[®] e MPLAB[®] - Ferramentas de desenvolvimento comercializados pela Microchip[®], www.microchip.com

5. Desenvolvimento do software de controle

O controlador fuzzy teve como entrada o sinal do erro e a sua derivada temporal. Foram definidas para cada uma destas entrada três funções de pertinência, como mostra a Fig. (5). A saída, como já mencionado, admitiu valores "crisp". A base de regras é também mostrada na Fig. (5) e o mecanismo de inferência utilizado foi máx-min.

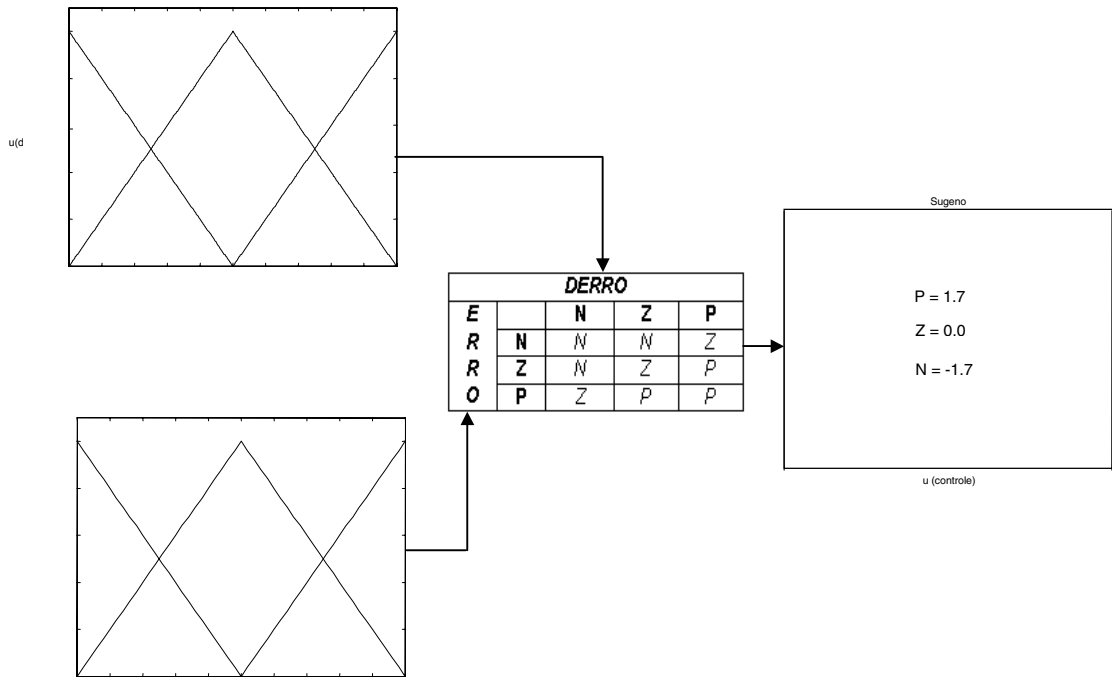


Figura 5. Funções de pertinência e regras utilizadas.

Cada célula da tabela mostrada na Fig. (5) representa uma regra construída da seguinte forma:

$$R1: \text{Se o Erro é } N \text{ e Derro é } N \text{ então a saída do controlador vale } N = -1.7$$

O programa que adquire o sinal do A/D, calcula o erro e sua derivada, implementa o controlador fuzzy e gera a saída PWM, foi desenvolvido em um PC, na linguagem C, compilado e transferido para o microcontrolador. Foi também construído alguns circuitos periféricos que condicionaram e filtraram os sinais eletrônicos envolvidos. A taxa de amostragem do programa foi programada a partir de temporizadores internos do microcontrolador via interrupção.

6. Resultados experimentais

O microcontrolador com o programa de controle implementado, faz um interfaceamento com o sensor e o driver de corrente em um tempo de amostragem de 5 ms. Os principais parâmetros do sistema são (Ribeiro et all, 1996):

- Frequência natural: $W_n = 77,5 \text{ [rad/s]}$
- Amortecimento global: $\xi = 7,07E-3$
- Rigidez global: $k = 11377,882 \text{ Nm}^{-1}$

Para a verificação da eficiência do controlador são apresentados a seguir a simulação e o teste experimental realizado para o sistema sem controle e controlado. A simulação numérica do sistema foi implementada utilizando o Matlab®. A Figura (6) mostra o diagrama de blocos do sistema em malha fechada com o controlador fuzzy utilizado na simulação numérica.

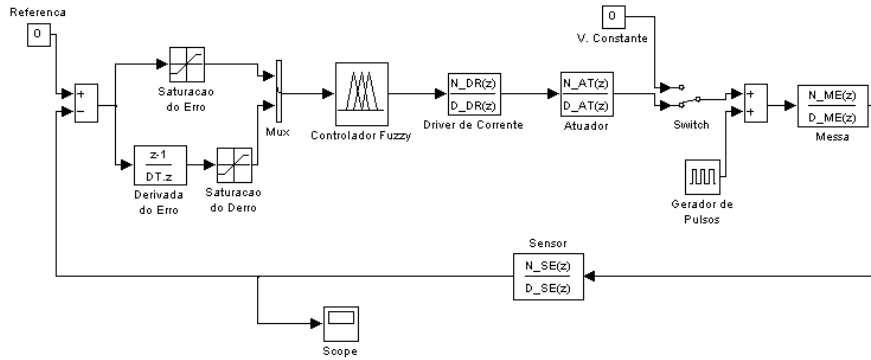


Figura 6. Sistema de simulação de controle utilizando o controlador fuzzy.

O sistema em malha aberta sem o controlador foi simulado numericamente com uma entrada tipo impacto resultando no gráfico apresentado na Fig. (7).

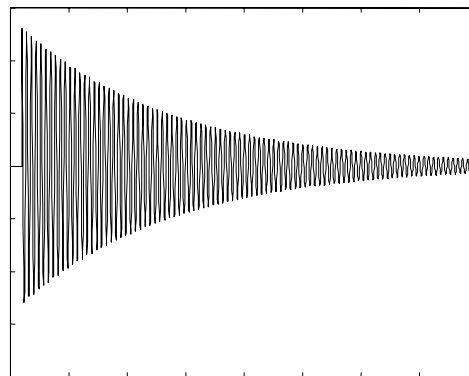


Figura 7. Resposta numérica do sistema sem controle a uma excitação impulsiva.

No ensaio experimental o sistema sem controle, excitado por uma força de impacto, apresentou o seguinte resultado Fig. (8).

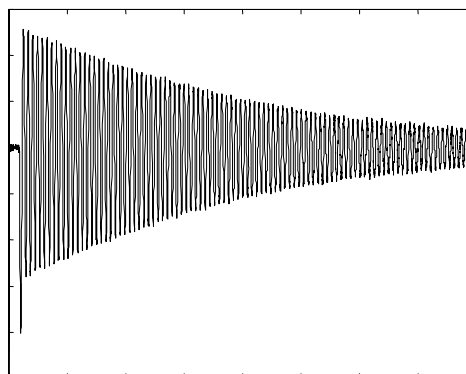


Figura 8. Resposta experimental do sistema sem controle a uma excitação impulsiva.

Como pode ser visto na Fig. (7) e Fig. (8), existe uma boa semelhança entre a resposta teórica e a experimental, com destaque para o elevado tempo de acomodação do sistema, decorrente do amortecimento muito baixo.

Foi determinado, através de simulação numérica, a resposta ao impacto para o sistema vibratório com o controlador fuzzy. A Figura (9) mostra o resultado de simulação obtido.

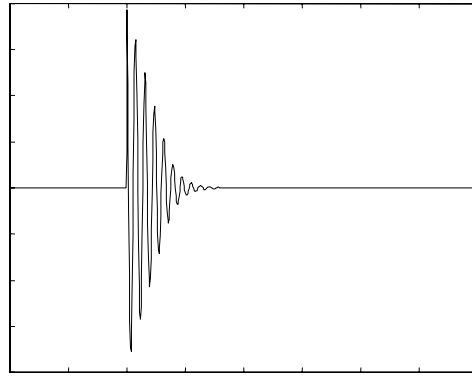


Figura 9. Resposta numérica do sistema com controle quando sujeito a uma excitação impulsiva.

No ensaio experimental, utilizando o microcontrolador fuzzy, observou-se, o seguinte resultado Fig. (10).

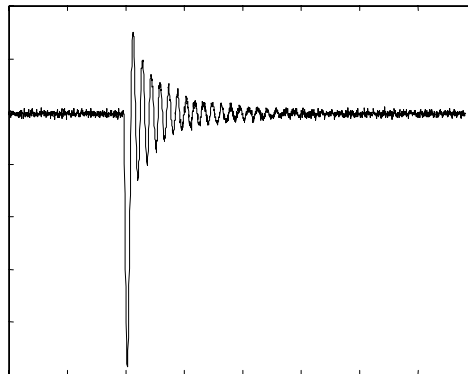


Figura 10. Resposta experimental do sistema com controle a uma excitação impulsiva.

Pode-se verificar na Fig. (10) a boa aproximação entre os resultados teóricos e os experimentais e o excelente comportamento do controle que conseguiu introduzir um amortecimento significativo no sistema diminuindo o tempo do regime transitório para aproximadamente 1.5 segundos. Observa-se no caso experimental um ruído em regime decorrente, provavelmente, das imperfeições do sensor e/ou de outros componentes eletrônicos envolvidos.

7. Conclusões

Neste trabalho apresentou-se um controlador fuzzy, implementado em um microcontrolador e voltado para a atenuação das vibrações num sistema mecânico de um grau de liberdade. Pode-se comprovar, com este trabalho, a facilidade no desenvolvimento do controlador, que não demandou formulações teóricas elaboradas, pois foi todo construído com base na intuição do projetista. A implementação do controle no microcontrolador foi também facilitada pelo uso de sistemas de desenvolvimento amigáveis fornecidos pelo fabricante do componente eletrônico.

Quanto ao controlador, este apresentou um desempenho excelente, atendendo plenamente aos objetivos propostos. No futuro pretende-se controlar sistemas vibratórios mais complexos e introduzir no controlador mecanismos de adaptação que possibilitem o uso do mesmo software em diferentes sistemas físicos.

8. Referências

- Apronix Inc., 1993, "Fuzzy Logic – From Concept to Implementation", www.aptronix.com.
- Driankov, D., Hellendoorn, H. and Reinfrank, M., 1996, "An Introduction to Fuzzy Control", Ed. Springer, United States of America, 316 p.
- Garcia, D., 1998, "Curso de Introducción a los Microcontroladores", Bogota, Colombia, 82p.
- Ribeiro, J.F., Lépore, F.P., 1996, "O uso de Atuadores Magnéticos no Controle Ativo de Vibrações de Sistemas Mecânicos", IVCEM-NNE, 439-444.
- Zadeh, L.A., 1965, "Fuzzy Sets", Information and Control, Vol.3, pp. 338-353.

VIBRATION CONTROL USING FUZZY LOGIC AND IMPLEMENTED IN MICROCONTROLLERS

Israel Jorge Cárdenas Nuñez

Universidade Federal de Uberlândia – FEMEC – Faculdade de Engenharia Mecânica
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
icardenas@mecanica.ufu.br

Prof. Dr. José Francisco Ribeiro

Universidade Federal de Uberlândia – FEMEC – Faculdade de Engenharia Mecânica
Av. João Naves de Ávila, 2160 – Bloco M – Sala 212, CEP 38400-902 – Uberlândia – MG
jribeiro@mecanica.ufu.br

Abstract. This work discusses the application of Fuzzy controllers in the active control of vibrations of dynamic systems. The project methodology proposed is evaluated in the control of a vibratory system of a degree of freedom, where the control are electromagnetic forces. The non linearity of actuators and the uncertainties of the parameters of the system justify the use of fuzzy control. The controller is implemented in microcontrollers, that offers a elevated flexibility at a very low cost. The Fuzzy controller proposed is of the type Sugeno with 9 rules, two inputs (error and change of error) and one output. The input of the microcontroller is the displacement of the vibratory system measured by a proximity sensor and the output is an electric tension pulse width modulated (PWM). The output feeds the electronic of actuator, and it generates the control force. The work concludes presenting some experimental results that prove the proposed controller's efficiency.

Keywords: Control, Fuzzy, Microcontrollers, Vibration.

Metodologia de Projeto de um Robô para Aplicações Cirúrgicas

Walter de Britto Vidal Filho

Departamento de Mecatrônica e Sistemas Mecânicos, Escola Politécnica - USP
Av. Prof. Mello morais 2231 – São Paulo – SP - 05508-900 sala: MC-1
wbritto@usp.br

Lucas Antônio Moscato

Departamento de Mecatrônica e Sistemas Mecânicos, Escola Politécnica - USP
Av. Prof. Mello morais 2231 – São Paulo – SP 05508-900
Lamoscat@usp.br

Raul Gonzalez Lima

Departamento de Mecatrônica e Sistemas Mecânicos, Escola Politécnica - USP
Av. Prof. Mello morais 2231- 05508-900

Resumo. O presente artigo aborda o projeto mecânico de um robô para aplicações cirúrgica, bem como propõe uma metodologia de projeto. A aplicação de robôs em atividades médicas não é algo totalmente novo. As primeiras iniciativas de usar um manipulador robótico em atividades médicas remonta ao final da década de 80. Foram usados para posicionamento preciso de sistemas de diagnóstico e para tanto foram adaptados robôs industriais. A aplicação de manipuladores em cirurgia, remonta aos meados da década de 90, quando apareceram robôs para auxílio a implantação de próteses, para posicionamento de sondas em neurocirurgia e telecirurgia. Observou-se que era necessário o projeto dedicado do manipulador para a tarefa a ser realizada, pois a adaptação de robôs industriais incorria em riscos elevados. Segundo as normas técnicas os manipuladores industriais não devem atuar nos mesmos ambientes que os seres humanos. Isto decorre da pouca confiabilidade dos sistemas de segurança e da grande estrutura dos manipuladores. Uma forma de aumentar a confiabilidade, consiste em projetar a estrutura mecânica com restrições de movimento, no desenvolvimento de estruturas mais compactas e em sistemas de segurança que considerem a presença humana próximo ao volume de trabalho do equipamento.

Palavras chave: Robô Cirúrgico, Projeto de manipuladores, robótica

1. Introdução

Apesar da grande quantidade de dispositivos eletrônicos para diagnóstico e monitoramento, os procedimentos cirúrgicos ainda são manuais e com auxílio de ferramentas desenvolvidas no começo do século passado. Contudo vem surgindo uma nova ferramenta para auxílio a medicina: o robô. Ele se impõem pela grande precisão de posicionamento que pode fornecer.

A utilização de robôs em aplicações cirúrgicas datam de meados da década de 80, quando robôs industriais foram adaptados para auxiliar cirurgias. Os primeiros manipuladores usados, eram robôs industriais adaptados, como o Puma 200 que foi empregado em 1985 no Memorial Medical Center para uma neurocirurgia de biopsia. No Reino Unido, um Puma 560, também foi adaptado para auxiliar em uma cirurgia, em 1987. Neste mesmo ano, na Universidade de Tóquio, o departamento de neurocirurgia relatou a aplicação de um robô para posicionamento preciso em cirurgia (Gray & Caldwell, 96).

O aparecimento de robôs especialmente projetados para aplicações cirúrgicas, só ocorreu na década de 90. Em 1992 tem-se os primeiros relatos de aplicações robóticas em cirurgias de implantes protéticos, culminando com o projeto do Robodoc. Um robô especialmente projetado para auxiliar a usinagem dos ossos de membros humanos para receber um encaixe preciso de uma prótese. (Pransky, 97). Deste ponto em diante os robôs empregados na medicina, passaram a ser especialmente projetados para este fim. Isto decorre, da necessidade de um aumento na segurança do dispositivo, bem como em uma melhora no desempenho do equipamento com relação a tarefa a ser executada. Contudo falta uma sistemática de projeto de robôs voltada a área cirúrgica, cada projeto segue critérios isolados e isto é decorrente da complexidade de cada campo cirúrgico a ser abordado.

Basicamente, diversos campos da cirurgia vem sendo alvo de pesquisa para aplicações de manipuladores. As cirurgias podem ser classificadas de forma simplificada, em dois campos: cirurgias invasivas e as minimamente invasivas. As invasivas são as convencionais nas quais os cirurgiões abrem passagem no tecido externo para as mãos atingirem os pontos internos de interesse. As minimamente invasivas são as resultantes de técnicas mais recentes, nas quais o cirurgião, através de pequenas incisões nos tecidos externos (furos), atinge os pontos internos de interesse através de hastes contendo um dispositivo mecânico de pinças. A visualização da região afetada é indireta, sendo feita através de uma câmera de vídeo. As principais vantagens das cirurgias minimamente invasivas são: mínimo trauma ao organismo, baixos índices de infecções hospitalares e baixos custos do pós-operatório (Bass et al., 93). Contudo tais cirurgias têm a sua difusão restringida pelas limitações mecânicas dos instrumentos cirúrgicos disponíveis e pela dificuldade de manuseio de tais dispositivos (Cohn et al., 95). Os instrumentos disponíveis (forceps) são hastes rígidas contendo uma garra (pinça) acionadas mecanicamente por alavancas na extremidade como visto na Fig.(1).

Desta forma, a aplicação de manipuladores robóticos para substituir tais mecanismos mostrou-se bastante atraente. Contudo, inicialmente pensou-se em empregar robôs em cirurgias invasivas convencionais, explorando a possibilidade da teleoperação a distância.

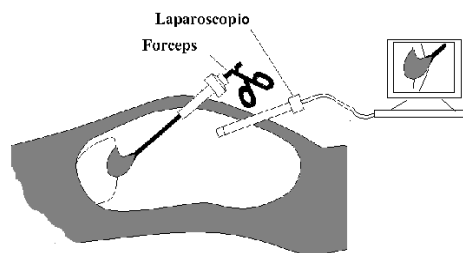


Figura1. Cirurgia minimamente invasiva

Neste caso o efetuator final do robô substitua a mão do médico no local da cirurgia, ficando o médico a distância enviando comando que seriam reproduzidos pelo robô. Esta abordagem esbarrou no atraso de tempo provocado pela linhas de transmissão para enviar comandos a distância. No caso das cirurgias minimamente invasivas o médico ficaria próximo do paciente, mas teleoperaria um efetuator final dentro do paciente, o qual penetraria através das pequenas incisões laterais. A vantagem desta abordagem reside nos benefícios das cirurgias minimamente invasivas: mínimo trauma ao organismo, baixos índices de infecções hospitalares e baixos custos do pós-operatório.

2. Estado da arte

A tecnologia robótica aplicada a cirurgia pode ser, atualmente, agrupada e classificada em três categorias: quanto a aplicação cirúrgica, quanto ao controle e quanto a participação no procedimento cirúrgico.

A classificação dos manipuladores quanto a aplicação pode ser vista na Tab.(1) abaixo:

Tabela 1. Classificação dos robôs quanto a aplicação cirúrgica

Quanto a aplicação cirúrgica	
Sistemas para cirurgias invasivas (convencionais)	São robôs que substituem as mão do médico no local da cirurgia, geralmente são teleoperados a distância.
Sistemas para cirurgias minimamente invasivas.	São robôs que substituem os dispositivos mecânicos como forceps e hastes empregados nas cirurgias minimamente invasivas.

O manipulador conhecido como Robodoc (Pransky,97) é um exemplo de um sistema para cirurgias invasivas, contudo não é teleoperado a distância. É baseado em uma configuração SCARA e destina-se a usinagem de ossos para colocação de próteses. O manipulador AESOP da empresa Computer Motion Inc. é um exemplo de sistema para cirurgias minimamente invasivas. Este manipulador posiciona um endoscópio através de comando vocal. Utiliza também uma configuração SCARA.

Uma outra forma de classificar os manipuladores é quanto ao controle, ou seja como o cirurgião controla o manipulador durante o procedimento cirúrgico.

Tabela 2. Classificação dos robôs quanto ao controle

Quanto ao controle	
Sistemas Teleoperados	São robôs que estão sob o controle direto do cirurgião, reproduzindo os movimentos da mão do médico durante a cirurgia.
Sistemas Programados	São robôs que são programados antes da cirurgia e realizam procedimentos automáticos durante a cirurgia.
Sistemas Passivos.	São robôs que servem como posicionadores espaciais de ferramentas. Muitas vezes movidos diretamente pela mão do médico, restringindo o movimento fora da região programada.

Um exemplo de sistema teleoperado é o manipulador desenvolvido no MIT (Madhani,98). Este dispositivo é composto de um mecanismo de cinco barras que posiciona um endopunho dentro do corpo do paciente. Um outro sistema teleoperado é o ‘Da Vinci’ desenvolvido pela empresa IntuitiveSurgical Inc. Este sistema também move um endopunho dentro do corpo do paciente (Salisbury,98). Um sistema programado, já comentado, é o Robodoc. Um exemplo de sistema passivo é o PADyC (Passive Arm with Dynamic Constraints) desenvolvido na Faculdade de Medicina de Grenoble (Schneider et al,99). Este sistema é baseado em uma configuração SCARA e tem como objetivo

localizar uma posição espacial no corpo do paciente. É movido pela mão do médico que movimentava o efetuador final. Os atuadores só restringem os movimentos da mão do médico na vizinhança da região alvo.

Pode-se também classificar os manipuladores quanto a forma de participação na cirurgia, Tab.(3), isto é, se eles participam de forma atuante realizando incisões ou como um assistente segurando ferramentas. A maioria dos manipuladores autorizados para venda, são sistemas assistentes, tais como o AESOP(automatic endoscopic system for optional positioning) e o EndoSista da empresa Armstrong (Finlay & Ornstein,95). Quanto aos manipuladores atuantes, a maioria está em fase de pesquisa e desenvolvimento, tais como o sistema ZEUS da ComputerMotion e o sistema Da Vinci da IntuitiveSurgical, que ainda não foram liberados para venda nos EUA.

Tabela 3. Classificação dos robôs quanto a participação na cirurgia

Quanto a participação na cirurgia	
Sistemas atuantes	São robôs que realizam os procedimentos principais de uma cirurgia, tais como incisões, suturas, etc .
Sistemas assistentes	São robôs que realizam procedimentos auxiliares, tais como posicionar um endoscópio, localizar as coordenadas espaciais de instrumentos, etc

3. Metodologia de projeto

No presente projeto do robô para endocirurgias, o método de projeto adotado apesar de seguir a sistemática de projeto de robôs industriais possui modificações significativas, principalmente quanto aos critérios de projeto e especificações técnicas.

Apesar da grande difusão das aplicações de robôs industriais, a sua metodologia de projeto ,ainda é pouco difundida, devido a segredos industriais para garantir reservas de mercado. Observando o que existe nas normas e em metodologias divulgadas, extrai-se como critérios básicos de projeto: Máxima produtividade e Mínimo custo. No tocante a robótica cirúrgica estes critérios não são aceitáveis. Os critérios básicos que devem servir de alicerce para um projeto voltado a medicina são: Máxima segurança, mínimo custo e máximo desempenho.

Um ponto que distingue a aplicação industrial da médica é que na aplicação industrial, homens e robôs são proibidos de compartilhar o mesmo volume de trabalho, enquanto na aplicação médica o robô vai dividir o mesmo ambiente com os homens.

A abordagem de projeto de manipuladores é normalmente sistêmica, ‘top-down’, ou seja, o projeto global é dividido em subprojetos agrupado por sistemas afins. E estes sistemas são projetados separadamente. Pode-se dividir um manipulador robótico nos seguintes sistemas:

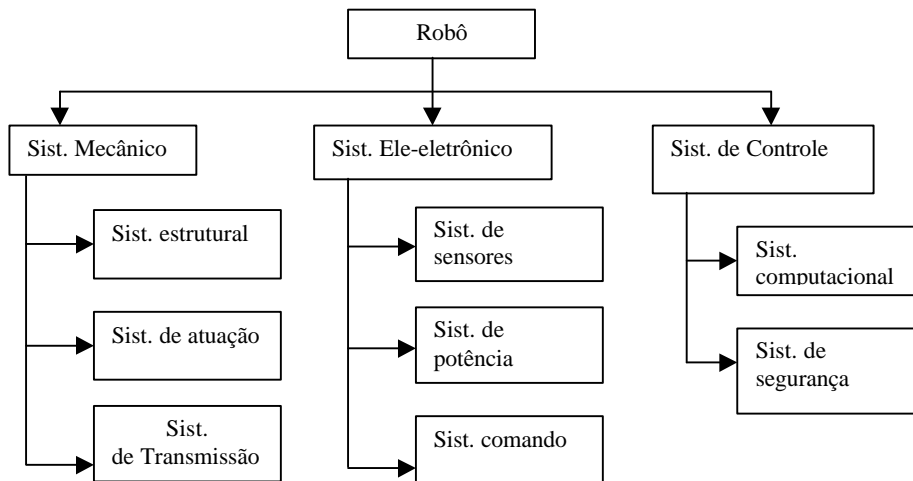


Figura 2. Diagrama dos sistemas que compõem um robô

Observa-se que muito foi feito no desenvolvimento do sistema eletro-eletrônico, contudo hoje em dia, o projeto deste sistema é a especificação de componentes comerciais. O projeto criativo é voltado mais para o sistema mecânico e o sistema de controle. No projeto do sistema mecânico deve-se levar em conta variáveis de projeto que influenciam no projeto e desempenho do sistema de controle. No projeto do sistema mecânico, até pouco tempo, não se preocupava em otimizar variáveis que melhorassem o sistema de controle, cada sistema era tratado isoladamente (Martins,86) Em metodologias de projeto mais recente (Urbano,97), considera-se no projeto do sistema mecânico variáveis que simplifiquem e melhorem o desempenho do sistema de controle. O projeto dos diversos sistemas passam a ser integrados por alguns critérios de projeto.

Segundo (Shimon,86) os pontos básicos para o projeto mecânico de manipuladores são

- Especificação do sistema
- Configuração do sistema
- Performance do sistema

Na especificação do sistema determina-se : o alcance do manipulador, conjunto de atividade que é ele capaz de realizar, o volume de trabalho e a capacidade de carga. Este último é um ponto importante para obter a máxima produtividade, pois a carga esta ligada a aceleração e a velocidades do manipulador. Contudo a importância de um ou outro parâmetro está na aplicação do manipulador, por exemplo para o projeto de robôs de montagem a aceleração e rigidez estrutural são mais importantes que capacidade de carga e velocidade.

Na configuração do sistema, determina-se a configuração de junta, o número de graus de liberdade e a configuração do acionamento. A configuração de junta é determinada pelo movimento, controle, obstáculos e requerimentos estruturais. Um fato que deve ser observado é que quanto maior o número de graus de liberdade, tanto de melhor o sistema de controle deve ser, para garantir a mesma acurácia do sistema.

Na performance do sistema são determinados os critérios de desempenho e avaliado se a configuração adotada satisfaz os critérios de desempenho.

O projeto do sistema de controle pode ser dividido no projeto do sistema computacional, onde são implementadas as estratégias e algoritmos de controle, e no sistema de segurança, onde são implementados algoritmos e dispositivos que garantam a segurança do usuário do sistema. No projeto do sistema de controle se considerava o controle isolado dos atuadores de cada junta, não levando em conta o acoplamento dinâmico dos vários elos. O efeito do acoplamento era considerado pelo controle como perturbações no sistema.

A sistemática adotada inicia-se com um projeto conceitual do sistema como um todo, em seguida tem-se a subdivisão do sistema, onde cada subsistema passa a ser tratado como um sistema, com entradas e saídas. Estas entradas e saídas correspondem a variáveis de projeto usadas para integrar os subsistemas afins. Pela figura(3) abaixo pode-se ver o fluxo de informações entre o projeto dos diversos sistemas. Cada fluxo corresponde a uma ou mais variáveis de projeto que integram os respectivos projetos de sistemas.

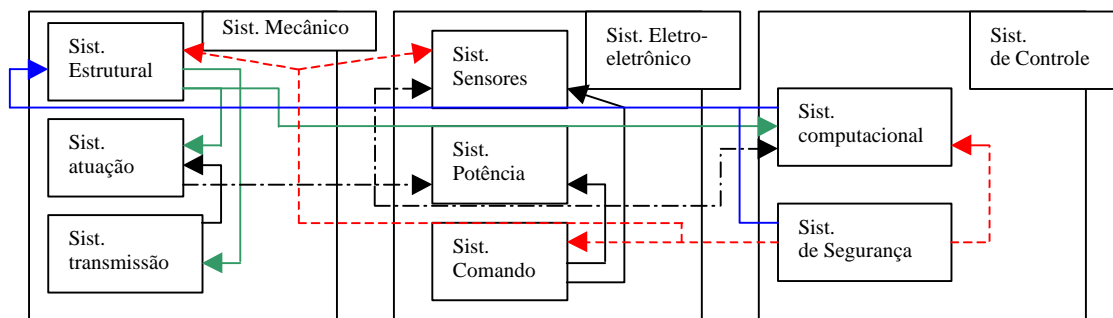


Figura 3. Fluxo de informação no projeto dos sistemas

Geralmente o projeto inicia-se pelo sistema mecânico, projetando-se a estrutura do manipulador segundo os critérios e especificações desejados. Observa-se, Fig.(3), que o sistema estrutural influencia o projeto dos sistemas de atuação e transmissão. As variáveis que são passadas ao projeto do sistema de atuação são: potência, torque e localização dos atuadores. O projeto do sistema de transmissão também recebe variáveis do sistema estrutural, tais como, localização dos atuadores e juntas atuadas, espaço para transmissão e rigidez desejadas. O sistema estrutural recebe variáveis do sistema de controle, algumas provenientes do sistema computacional e outras do sistema de segurança. As variáveis provenientes do sistema computacional referem-se as estratégias de controle, elas são: Desacoplamento das juntas motoras, inércia, balanceamento dinâmico e do peso próprio. Estes fatores quando tratados como variáveis de projeto da estrutura do manipulador simplificam e otimizam o sistema de controle e são as variáveis retornadas ao sistema de controle. O sistema computacional recebe também variáveis do sistema de atuação, pois dependendo do tipo dos atuadores, tem-se técnicas de controle diferentes.

Os sistemas de segurança até pouco tempo eram sistemas externos ao robô, como cercas protetoras e barreiras de feixes de luz que quando atravessados faziam o sistema de segurança desligar automaticamente o manipulador. Hoje em dia existe sensores incorporados ao manipulador que checam o ambiente de trabalho e os sistemas do manipulador, em busca de anormalidades. Contudo para aumentar a confiabilidade dos manipuladores para atuar em ambientes cirúrgicos, a segurança deve começar no projeto do manipulador, pelo estabelecimento de variáveis de projeto que sejam influenciadas pela segurança. O sistema estrutural recebe variáveis de projeto do sistema de segurança, tais como velocidade, inércia, movimentos restringidos e volume de trabalho seguro. O sistema de controle deve evitar o choque do manipulador com as pessoas na sala de operação, através de um sistema de segurança que considere a possibilidade de pessoas na área de risco. Contudo o projeto deve minimizar as conseqüências de um possível contato, ou seja, caso ocorra o contato não deve ser grave como ocorreria com robôs industriais. Para tanto as inercias desses manipuladores devem ser pequenas, como também a velocidade de trabalho. A velocidade de trabalho sendo igual a velocidade de um braço humano em um procedimento cirúrgico, propicia um tempo para a reação humana, suficiente para evitar o choque. Por outro lado, as velocidades dos manipuladores devem ser compatíveis com a das mãos do médico, pois na

maioria das vezes os manipuladores, em aplicações cirúrgicas, são teleoperados e seguem movimentos da mão do cirurgião.

Observando o projeto de cada subsistemas separadamente, utiliza-se a sistemática vista na Fig.(4)

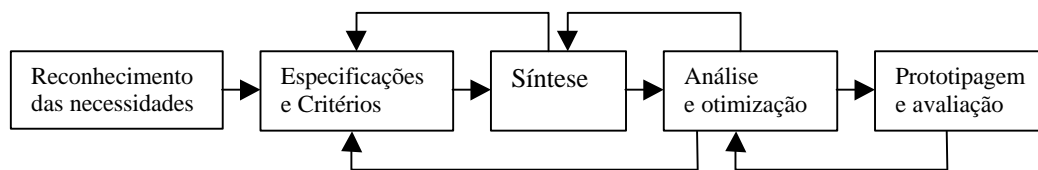


Figura 4. Sistemática de projeto de cada sistema

A sistemática para o projeto de cada sistema isoladamente inicia-se pelo reconhecimento das necessidades do usuário, ou seja, da equipe médica. Esta é uma etapa importante e de difícil realização, devido a dificuldade de comunicação entre engenheiros e médicos. Determinadas as necessidades, elaboram-se as especificações técnicas e os critérios de projeto que atendem a essas necessidades. Em seguida sintetiza o sistema de acordo com as especificações e critérios. Muitas vezes tem-se que retornar e rever a especificações devido a inviabilidade técnica ou financeira. Diante da configuração desejada é feita uma análise e otimização. Por fim deve-se gerar um protótipo e avaliar o desempenho. Desde a etapa de síntese deve-se considerar a viabilidade de construção de um protótipo. Hoje em dia o emprego de ferramentas CAD, reduziu o número de protótipos necessários para finalizar um projeto.

4. Projeto conceitual

O objetivo deste projeto é a criação de um manipulador robótico para aplicações em endoscópias, classificado quanto ao controle em um sistema teleoperado supervisionado e classificado quanto a participação na cirurgia em um sistema atuante. O objetivo do sistema é aumentar a habilidade do cirurgião em procedimentos minimamente invasivos, permitindo que o mesmo possa realizar movimentos que são difíceis ou impossíveis de realizar com os instrumentos cirúrgicos convencionais.

Apesar do sistema ser teleoperado, o cirurgião se encontra próximo ao paciente. O objetivo da teleoperação não é a cirurgia a distância e sim permitir que o médico possa operar como se estivesse em uma cirurgia aberta. Deseja-se aqui criar uma telepresença, ou seja, gerar a impressão no médico que suas mãos estão dentro do corpo do paciente Fig.(5).

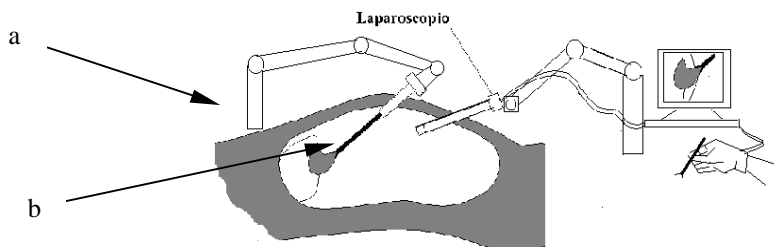


Figura 5. Plataforma robótica, composta de dois manipuladores

O projeto do manipulador que efetua a intervenção, pode ser dividido em duas partes, uma externa ao corpo que será chamada de exo-robô, Fig.(5a), e uma que penetra no organismo, chamado endo-robô, Fig.(5b). O exo-robô é um manipulador com a finalidade posicionar a base do endo-robô o qual irá substituir os forceps penetrando no organismo por uma pequena incisão. Os dois manipuladores serão teleoperados em série pelo cirurgião que executa os procedimentos cirúrgicos.

Com a teleoperação surge a necessidade de uma interface de comando, onde as coordenadas da mão do cirurgião são capturadas e as forças de contato com os tecidos são retornadas a mão do médico. Este tipo de interface é conhecida como interface háptica e são empregadas, atualmente, em realidade virtual (Salisbury & Srinivasan,97). A idéia que está por trás do funcionamento desta interface surgiu na década de 70 com os manipuladores teleoperados empregados nos laboratórios nucleares.(Sheridan,89). A interface háptica que será inicialmente usada para teleoperação do robô cirúrgico é a descrita em (Vidal et al,2000).

O projeto do endopunho não será descrito neste trabalho, restringindo-se aqui ao projeto do exorobô, ou seja, ao projeto da estrutura do manipulador que está fora do corpo.

5. Determinação das necessidades e especificações dos sistemas

Na cirurgia endoscópica o cirurgião usa uma haste contendo uma pinça. Esta haste se move ao redor do ponto de incisão no corpo do paciente Fig.(6). Como se pode observar tal instrumento mecânico possui 4 graus de liberdade sem contar com o de pinçar. São dois de rotação em torno do ponto de incisão, um de translação passando pelo ponto de incisão e um de rotação sobre si próprio Fig.(6). Estas restrições limitam a maneabilidade do cirurgião, impedindo-o de realizar todos os procedimentos que é capaz de realizar em uma cirurgia aberta. Além das restrições mecânicas, uma

outra restrição que dificulta este tipo de operação é a restrição visual, pois a visualização da região afetada é indireta através de uma câmera de vídeo o que causa a perda da noção de profundidade, como também uma visão reversa dos movimentos na tela com relação aos movimentos da base da haste. Isto é, quando o cirurgião quer levantar a pinça na tela do vídeo, deve baixar a parte da haste que está fora do corpo.

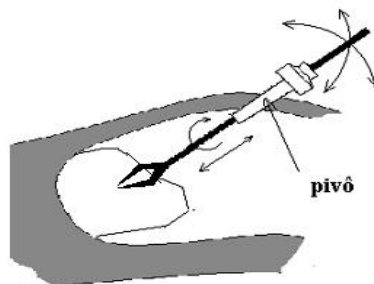


Figura 6. Graus de liberdade do procedimento

Diante dessas necessidades se estabelece os seguintes critérios de projeto:

- O mecanismo do manipulador deve garantir movimentos centrados no ponto de incisão
- O mecanismo do manipulador deve aumentar a maneabilidade pelo aumento dos graus de liberdade.
- O mecanismo do manipulador deve garantir movimentos semelhantes entre a mão do cirurgião e a pinça vista na tela.
- As restrições de segurança devem ser inerentes ao projeto do mecanismo do manipulador.
- O sistema de controle deve garantir um acoplamento dinâmico entre a mão do cirurgião e o efetuator final.

Analisando-se os movimentos e dispositivos convencionais para este tipo de cirurgia chega-se as seguintes especificações técnicas para o sistema mecânico:

- Amplitude angular de $\pm 60^\circ$ em relação a vertical.
- Máxima força realizada pela pinça (efetuator final) : 1kgf
- Amplitude máxima de penetração: 200 mm
- Velocidade de trabalho entre 200 e 300mm/s nas três direções (x,y,z) no efetuator final.

6. Projeto do Sistema Mecânico

O desenho do mecanismo deve satisfazer o primeiro critério de projeto: movimentos centrados no ponto de incisão. Como sabemos, qualquer manipulador articulado pode mover o efetuator final de forma a ter um movimento centrado em um pivô Fig.(7). Contudo isto requer um considerável trabalho do sistema de comando e do sistema computacional, para coordenar vários atuadores e interpolar uma trajetória circular. Qualquer falha de controle dos atuadores ou de cálculos de interpolação resulta em um acidente que irá romper a parede externa do corpo do paciente. O que não satisfaz o quarto critério: restrições de segurança devem ser inerentes ao projeto do mecanismo. Desta forma pode-se criar uma estrutura para o manipulador que tenha um centro remoto, ou seja, possua restrições mecânicas que permitem só movimentos centrado no ponto de incisão Fig.(8). Na Fig.(8a) tem-se uma estrutura de seguimentos curvos com articulações cujas linhas de eixo passam pelo ponto de incisão o que criam um centro remoto para os movimentos esféricos que são gerados quando os elos se movem entorno das articulações. Segundo (Faraz & Payandeh, 97) este tipo de mecanismo tem a tendência a desalinhar as linhas de eixo, quando sob carga, o que o torna difícil de manipular.

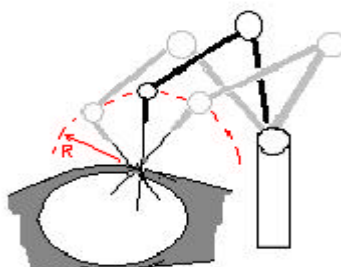


Figura 7. Manipulador articulado

O dispositivo mais simples que pode gerar movimentos com centro remoto é a guia linear curva Fig.(8b). Observe que o dispositivo da Fig.(8b) pode ter dois giros perpendiculares centrados no ponto de incisão: um através da guia e outro ao redor do eixo da base. Este tipo de configuração é de difícil construção, quando se deseja baixo coeficiente de atrito e ausência folgas. A aquisição comercial de guias lineares curvas com esferas re-circulantes também é difícil.

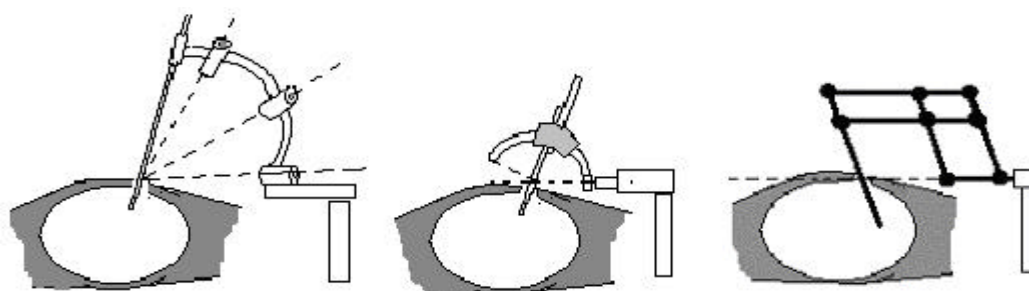


Figura 8. a) Elos curvos b) guias lineares curvas c) mecanismo de 5 barras

Na Fig.(8c) tem-se um mecanismo de 5 barras que é capaz de gerar movimentos centrados no ponto de incidência. Este tipo de mecanismo já foi empregado por (Taylor et al,95) e (Kobayashi et al, 99) para posicionamento de endoscópio, com sucesso. Algumas das vantagens desse mecanismo são a fácil construção e a boa rigidez estrutural.

6.1. Sistema estrutural

6.1.2. Análise da cinemática

Analisando as estruturas Fig.(8) quanto ao volume de trabalho, observa-se que a estrutura da Fig.(8b) e Fig. (8c) possui volumes de trabalho parecidos a um manipulador esférico Fig. (9a).

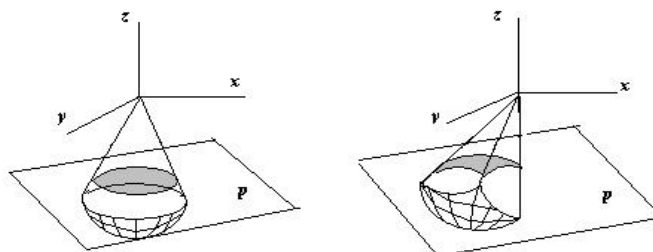


Figura 9. Volumes de trabalho a) b)

A estrutura da Fig.(8a), apesar de possuir movimentos pivotados, possui um volume de trabalho complexo. Se for Interceptado por um plano obtém-se uma área convexa. Isso dificulta a geração de trajetórias retas que não saiam do volume de trabalho. As estruturas da Fig. (8b) e (8c) possuem uma área de trabalho circular. Dado dois pontos, pode-se sempre obter uma trajetória reta dentro do volume de trabalho.

O mecanismo de 5 barras possui uma vantagem em relação a guia curva que reside no paralelismo entre as barras verticais. Desta forma os ângulos das juntas da base podem ser transferido ao ponto de incidência, o que facilita a modelagem cinemática. Observando do ponto de vista do local de incidência, que deve permanecer estático, a parte do robô vista de dentro da cavidade pode ser modelada pela Fig.(10). Os ângulos *f* e *q* são os mesmos da base *f*1 e *f*2 Fig.(12).

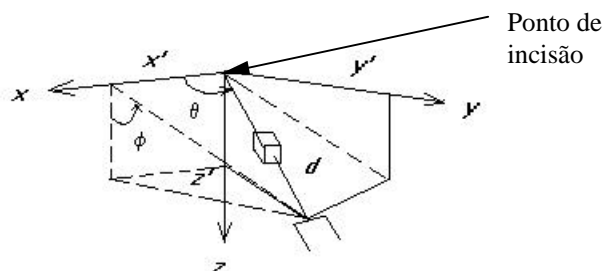


Figura 10. Diagrama de junta visto de dentro do corpo

A cinemática direta de posição para este caso é dada pelas equações (1), (2) e (3) obtida através do modelo de manipulador esférico Fig.(10).

$$z' = (d \cdot \text{sen } q) \cdot \cos f \tag{1}$$

$$y' = (d \cdot \text{sen } q) \cdot \text{sen } f \tag{2}$$

$$x' = d \cdot \cos q \tag{3}$$

Estas equações fornecem a cinemática da estrutura sem considerar os graus de liberdade do punho. As equações estão em função dos ângulos θ e ϕ , bem como do deslocamento axial d , pois são correspondentes as juntas controladas do manipulador Fig.(12).

Observando a Fig.(12) tem-se que um atuador paralelo a y faz o estrutura descrever um ângulo θ em relação ao ponto de incidência. Um atuador axial ao eixo x faz a estrutura descrever um ângulo ϕ em relação ao ponto de incidência.

A cinemática inversa de posição pode ser também obtida pela Fig.(10) e pode ser expressa pelas equações (4), (5) e (6).

$$q = \text{atg}\left(\frac{\sqrt{z'^2 + y'^2}}{x'^2}\right) \tag{4}$$

$$f = \text{atg}\left(\frac{y'}{z'}\right) \tag{5}$$

$$d = \sqrt{x'^2 + y'^2 + z'^2} \tag{6}$$

Devido às vantagens cinemáticas, a boa rigidez do mecanismo de 5 barras e a facilidade de construção frente a guia curva, foi escolhido o mecanismo da Fig.(8c).

6.1.1. Configuração

A estrutura adotada foi um mecanismo de 5 barras com uma guia linear na extremidade, barra 5, Fig.(11). O desenho do mecanismo foi feito de forma a embutir cada barra dentro da anterior, isto é, as barras 4 e 2 estão embutidas nas barras 3 e 1.

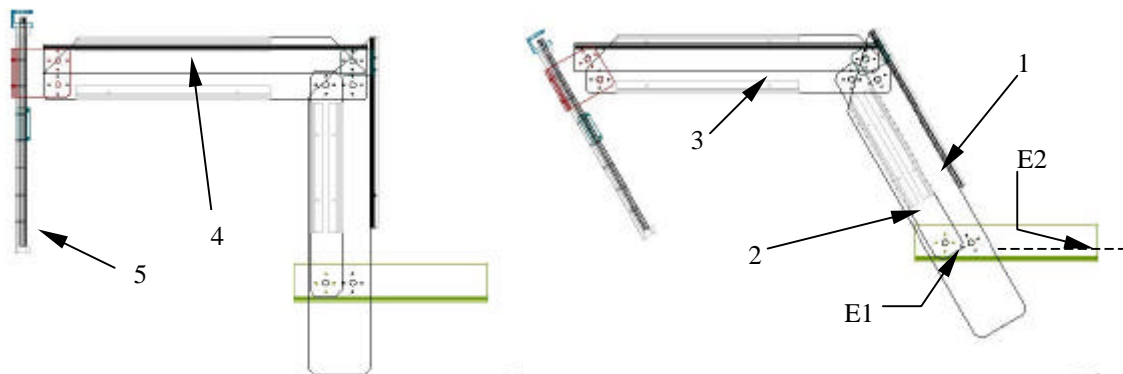


Figura 11. Desenho da configuração adotada

Como se pode observar na Fig.(11) o mecanismo move a barra 5 paralela a barra 1, desta forma controlando a posição angular da barra 1 controla-se a posição angular da barra 5. Portanto foram colocados atuadores nas base do manipulador, eixos E1 e E2, diminuindo a inércia da estrutura.

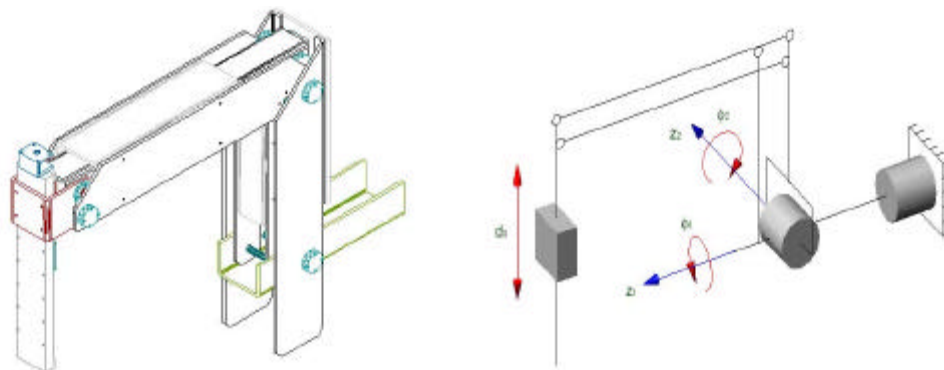


Figura 12. Perspectiva do manipulador e diagrama de juntas

6.2 Sistemas de atuação e transmissão.

Os atuadores mais empregados em manipuladores são elétricos, mais precisamente os servo-motores. Contudo tais dispositivos tornam o projeto caro e complexo. Uma forma de reduzir os custos de projeto está no emprego de motores de passo, uma vez que as potências envolvidas e as precisões requeridas sejam compatíveis com as características do projeto. Os motores de passo possuem um custo bem menor que os servos-motores e são facilmente encontrados no mercado nacional. Seu emprego é recomendado para baixas potências e requisitos de precisão baixos. A precisão destes dispositivos podem ser aumentada pelo emprego de redutores e 'drives' de micropasso. Portanto serão empregados inicialmente neste manipulador, para estudar sua viabilidade. Será analisado o emprego de servo-motores para atuação do endo-punho, cuja necessidade de precisão de posicionamento é maior.

Nem sempre os atuadores estão conectados diretamente nos eixos das juntas, o que implica na necessidade da transmissão dos movimentos destes à junta. Uma vantagem de não colocar os atuadores no braço manipulador está na redução da inércia do braço.

Os principais tipos de transmissão são: eixos, engrenagens, parafusos sem-fim, correias, correntes e cabos. Cada um destes possui um leque de aplicações, sendo que para aplicações em manipuladores geralmente usa-se transmissões com elevada rigidez. Contudo para algumas aplicações em que as restrições geométricas ou de custos imperem, tem se feito uso de transmissões flexíveis como a transmissão por cabo (Madhani et al, 1998). Como pode ser observado na (Fig.9), o mecanismo de barras permite a colocação de dois atuadores na base. O atuador que move a guia linear, barra 5 (Fig.8), é o único que necessita de uma transmissão, se tiver que ser colocado na base. Inicialmente optou-se em colocá-lo na barra 5 que resulta em usar um contra-peso maior na base.

7. Conclusão

No presente trabalho verificou-se a necessidade de uma metodologia no projeto de um robô para aplicações cirúrgicas, bem como forneceu esta metodologia. O conjunto de critérios de projeto e especificações foram extraídas das necessidades da equipe cirúrgica e de artigos da área.

O grande limitador da aplicação de manipuladores robóticos em cirurgias é a confiabilidade do sistemas atuais, que são projetados baseados em manipuladores industriais. Aumentando-se a segurança do sistema quanto a falhas possíveis de operação, tais sistemas podem ser implantados na prática. Para alcançar a confiabilidade desejada, o projeto do sistema deve seguir uma metodologia, na qual os requisitos de segurança sejam inerentes ao projeto dos diversos subsistemas que compõem o manipulador. E estes subsistemas devem ser projetados de forma integrada, onde variáveis de projeto de um sistema que influenciam outro sistema devem ser consideradas no projeto de ambos. Servindo assim como meio de integração dos projetos de cada subsistema.

O projeto mecânico, descrito aqui, aplica a metodologia sugerida no projeto de um manipulador destinado a aumentar a destreza do cirurgião em a endocirurgia. O conceito deste projeto não é uma teleoperação à distância e sim uma teleoperação que fornece a destreza que as ferramentas cirúrgicas disponíveis para endocirurgias manuais não fornece.

8. Agradecimentos

Agradecemos o apoio financeiro da FAPESP e do CNPq, bem como aos alunos de iniciação científica que colaboraram na construção de componentes e na realização de desenhos técnico em CAD.

9. Referências

- Bass, E. B., Pitt, H., and Lilemoe, K, 1993, "Cost-effectiveness of laparoscopic cholecystectomy versus open cholecystectomy". The American Journal of surgery, 165:466-471
- Cohn, M. B. et al, 1995, "Surgical Applications of mini-robots", journal of robotic systems 12(6),401-416
- Finlay, P. A. and Ornstein, H. M, 1995, "Controlling the Movement of a surgical laparoscope", IEEE Engineering in medicine and biology, May/June 95, pp289-291
- Gray, J.O. & Caldwell, D. G., 1996., "Advanced robotics & Intelligent machine", IEE control engineering series, v51, published by Institution of Electrical Engineers, London.
- Kobayashi, E.; Masamune, K; Sakuma I.; Dohi, T.; Hashimoto, D., 1999 "A New Safe Laparoscopic manipulator System With a Five-Bar Linkage mechanism and an Optical Zoom" Computer Aided Surgery 4:182-192
- Madhani, A.; Niemeyer, G. and Salisbury, J.K., 1998 "The Black Falcon: A Teleoperated Surgical Instrument for Minimally Invasive Surgery", IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), Victoria B.C., Canada, October 13-17.

- Martins, F. P. R., 1986, “*Aspectos Fundamentais do Projeto Mecânico de um Robô 6R*”, Dissertação de mestrado, Escola Politécnica –USP
- O. Schneider, J. Troccaz, O. Chavanon, D. Blin, 1999, “*Synergistic Robotic Assistance in Cardiac Procedure*”, Computer Assisted Radiology and Surgery, pp23-26, Juin 99, Paris.
- Pransky, J., 1997, “*ROBODOC- Surgical Robot Success Story*” - Industrial robot vol 24 número 3 pp231-233
- Salisbury J. K., 1998, “*The Heart of Microsurgery*” - Mechanical Engineering, December 98, pp46 –51
- Salisbury J. K. and Srinivasan, M. A., 1997, “*Phanton-Based Haptic interaction with virtual objects*” IEEE Computer Graphics and Applications, Vol.17, N° 5, pp6-10, September.
- Sheridan, T. B., 1989, “*Telerobotics*”, Automatica, Vol.25, N° 4, pp 487-507.
- Shimon Y., 1986, “*Handbook of industrial robotics*”, J. Wiley, New York
- Taylor, H. R.; Funda, J.; Eldridge, B.; Gomory, S.; Gruben, K.; Larose, D.; Talomini, M.; Kavoussi, L.; Anderson, J. 1995 “*Telerobotic Assistant for laparoscopic Surgery*” IEEE Engineering in Medicine and Biology, pp279-288, May/june 1995
- Urbano, M. S., 1997, “*Projeto mecânico de um manipulador de seis graus de liberdade, acionado por motores pneumáticos rotativos*”, Dissertação de mestrado, Escola Politécnica –USP.
- Vidal, W. B; Moscato, L. A. and Lima, R. G, 2000, “*Development of ‘spatial’ interface for surgery simulator*” IV industry Applications conference, Porto Alegre, Brasil.

Design Methodology of a Robot for Surgical Applications

Walter de Britto Vidal Filho

Departamento de Mecatrônica e Sistemas Mecânicos, Escola Politécnica – USP
Av. Prof. Mello morais 2231 – São Paulo – SP - 05508-900 sala: MC-1
wbritto@usp.br

Lucas Antônio Moscato

Departamento de Mecatrônica e Sistemas Mecânicos, Escola Politécnica – USP
Av. Prof. Mello morais 2231 – São Paulo – SP - 05508-900
Lamoscat@usp.br

Raul Gonzalez Lima

Departamento de Mecatrônica e Sistemas Mecânicos, Escola Politécnica – USP
Av. Prof. Mello morais 2231 – São Paulo – SP - 05508-900

Abstract. *This paper describes a methodology and a partial project of a robotic arm for surgical applications. The robots application in medical activities is not something new. The first initiatives of using a robotics manipulator in this field occurred in late of the 80s. At that time, industrial robots were adapted to be used for precise positioning of systems for diagnostics. The actual beginning of application of manipulators in surgery occurred in the middle of the 90s, when robots were used for aiding the placement of prosthesis and for positioning of probes in neurosurgery and telesurgery. It was observed that it was necessary a dedicated design for the task to be accomplished, as the adaptation of industrial robots incurred in high risks. According to technical standards, the industrial manipulators should not act in the same environment as human beings. This results from the little reliability of the system. A form of increasing the reliability is based in the kinematics design with movement restrictions, through the development of more compact structures and in safety systems that foresee human presence inside the work volume.*

Keywords. *Surgical robotics, robot design, robotics*

VISUAL SERVO CONTROL OF NONHOLONOMIC MOBILE ROBOTS

Geraldo F. Silveira

Robotics and Computer Vision Laboratory/ITI PO Box 6162, 13081-970, Campinas-SP Brazil
Geraldo.Silveira@iti.gov.br

J. Reginaldo H. Carvalho

Robotics and Computer Vision Laboratory/ITI PO Box 6162, 13081-970, Campinas-SP Brazil
Reginaldo.Carvalho@iti.gov.br

Pedro Mitsuo Shiroma

Robotics and Computer Vision Laboratory/ITI PO Box 6162, 13081-970, Campinas-SP Brazil
Pedro.Shiroma@iti.gov.br

Patrick Rives

INRIA Sophia-Antipolis, 2004 Route des Lucioles, 06565 Valbonne Cedex-France
Patrick.Rives@sophia.inria.fr

Samuel Siqueira Bueno

Robotics and Computer Vision Laboratory/ITI PO Box 6162, 13081-970, Campinas-SP Brazil
Samuel.Bueno@iti.gov.br

Abstract. This work describes the application of the visual servoing of a mobile robot having as input the image from a camera mounted on a pan-tilt. The solution is based on the so called Interaction Matrix (IM). The velocities of the image features parameters are utilized to compute control signals for the robot actuators without any state reconstruction or camera calibration schemes. The paper also presents a study about the influence of the DOFs added by the pan-tilt on the maneuverability of the robotics ensemble, in two case studies: *i*) the pan and robot steering motions shares the same rotation axis. *ii*) the rotation axis for pan motion and robot are different. Experiments were done with a Nomad 200 mobile robot with a B&W CCD camera and a pan-tilt. This methodology finds application in several real scenarios, like trajectory tracking, parking and docking of AGVs. We are also investigating the application of the methodology for the automatic landing of an aerial unmanned vehicle.

Keywords. Mobile robots, visual servoing, robotic vision.

1. Introduction

How to use vision has always been a major research area in robotics. However, the computational demand necessary for image processing and parameters extraction makes the utilization of computer vision solutions a very difficult task in real problems. This happens mainly due to the combination of high data density in images, the complete lack of depth information and the lack of a rigid linkage between the mobile robot and the environment. Most applications work in the robot's configuration space and, thus, its state reconstruction demands a huge computational effort, making real-time implementation prohibitive. Moreover, usually camera parameters are very sensitive to lighting conditions and automatic calibration procedures are still an open field of research. For the case of visual-based control of mobile robots, one has that most used mobile robots are subject to nonholonomic constraints, bringing us to an area of research with open problems, specially related to dynamics modeling and controller design.

In this paper, the methodology proposed in (Espiau *et al.*, 1992) is applied to the visual servoing of a class of wheeled nonholonomic mobile robot. The solution is based on the so called Interaction Matrix (IM), whose purpose is to extend the robotics Jacobian to the image plane, utilizing certain geometric primitives as reference images. The velocities of the image features parameters are then utilized to compute control signals for driving the robot actuators without any state reconstruction or camera calibration schemes.

The paper also presents a study about the influence of the DOFs¹ added by the pan-tilt on the maneuverability of the robotic ensemble in two case studies: *i*) the pan and robot steering motions share the same rotation axis. *ii*) the rotation axis for pan motion and robot are different. It is shown that, although in (*i*) the Jacobian is singular, the positioning is still possible by subdividing the main problems into two. However, as both subproblems are not decoupled, the solution is not ensured for all initial conditions. On the other hand, for (*ii*) the non-singularity of the Jacobian ensures the solution of the problem when utilizing the IM methodology.

The setup used in the experiments is based on a Nomad 200 mobile robot with a monocular non-calibrated black and white CCD camera mounted on a pan-tilt and a Data Translation DT-3155 frame grabber. In terms of software, all

¹Degrees Of Freedom

algorithms were implemented using Tcl/Tk and run in a PC Intel Pentium 200 MHz embedded in the robot with Linux.

This methodology finds application in several real scenarios, like trajectory tracking, parking and docking of AGVs. We are also investigating the application of the methodology for the automatic landing of an aerial unmanned vehicle as well as road following, see (Silveira *et al.*, 2001b) and (Silveira *et al.*, 2001a).

2. Problem Formulation

The main objective of this paper is to develop a full vision-based control application which is both useful and feasible using off-the-shelf components.

The problem considered here consists of positioning a camera mounted on a pan-tilt carried by a Nomad 200 mobile robot (see Fig.(1)) with respect to a reference image formed by four points as in the vertex of a square and located on the wall (see Fig.(2)). This image is related to the methodology used to control the robot and it is explained in the next sections.



Figure 1. *Nomad 200* mobile robot at LRV/ITI.

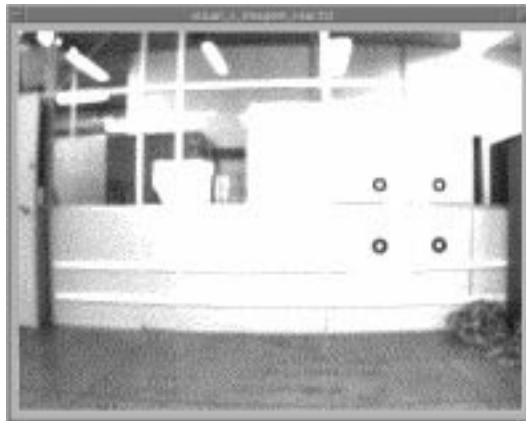


Figure 2. The reference image at the initial condition.

Real-world applications of visual servoing techniques can be divided in vision aspects and control aspects. Concerning vision, the tasks include image acquisition and segmentation, feature tracking and computation of feature parameters. Concerning control, one has the configuration and speed transformations among all frames assigned to the system, kinematic and/or dynamic modeling, Jacobian computation, and controller design.

This work will focus on the control aspects. Refer to (Carvalho *et al.*, 2000) and (Donnoui *et al.*, 2001) for a detailed description of the vision aspects. The vision problem consists of extracting from the image, during robot motion, the pixel coordinates of each circle centroid using a sequence of morphological filters. Therefore, our vector of visual signals is given by $s = [x_1, x_2, x_3, x_4, y_1, y_2, y_3, y_4]^T$.

3. Implementation of the control tasks

The approach proposed by Espiau *et al.* (Espiau *et al.*, 1992) basically consists of deriving a suitable mathematical relationship (the *interaction matrix* L^T) between the motion in the image plane of the parameters vector s

of certain geometric primitives (like points, lines, or circles) and the motion of the 3D frame attached to the camera $T_c = [{}^c\dot{x}_c, {}^c\dot{y}_c, {}^c\dot{z}_c, {}^c\dot{w}_x, {}^c\dot{w}_y, {}^c\dot{w}_z]^T$.

$$\dot{s} = L^T \cdot T_c \quad (1)$$

In this paper, we are interested in an interaction matrix that permits no motion for an unchanged image, which can be implemented by utilizing four points as the reference image.

Given the velocities of the 3D camera frame, one has to derive a non-singular Jacobian to express these velocities in terms of robot actuator speeds, and then, design a suitable controller. Both Jacobian and controller must consider spatial and/or mechanical constraints of the robotic platform.

Consider the system of Fig.(3) as a representation for the Nomad 200, where \mathcal{F}_b is the frame attached to the center of rotation of the robot base; \mathcal{F}_t is the frame attached to the center of rotation of the robot turret; \mathcal{F}_p is the frame attached to the pan-tilt rotation axes; \mathcal{F}_c is the frame attached to the camera, whose X-Y projection is coincident with the image plane. Both \mathcal{F}_b and \mathcal{F}_t share the same axes of rotation, which is assumed to be vertical. Their origins are located at the center of the robot circular section, where \mathcal{F}_t is displaced by h in the y-axes of \mathcal{F}_b , and \mathcal{F}_p by d_p away from \mathcal{F}_t . The usual assumption of movement without slipping in a horizontal plane holds.

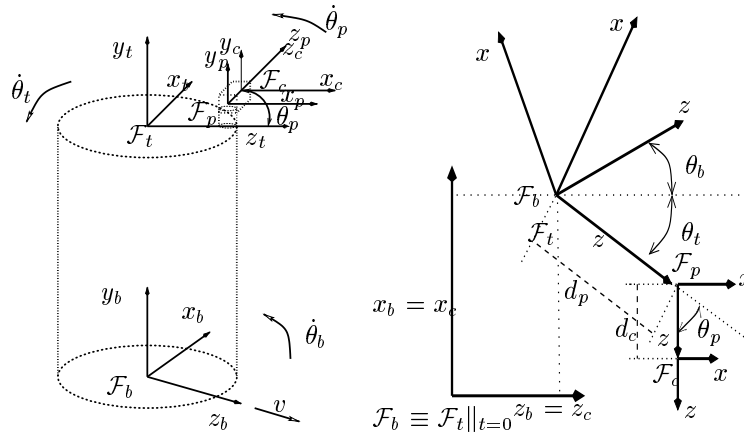


Figure 3. 3D and 2D frames attached to the Nomad 200 mobile robot with camera and pan-tilt.

The Nomad 200 mobile robot has some particular characteristics that are interesting to point out:

- A fixed reference frame is set when the robot receives its reset command. This command sets all encoders output to zero and brings the robot actuators to a pre-defined configuration. In this case the turret and base frames are aligned (their 2D Z-X projections are coincident);
- The robot configuration given by odometry $(x_b, y_b, \theta_b, \theta_t)$ is set according to this fixed reference frame (see Fig.(3)). The orientation of the turret frame \mathcal{F}_t with respect to the base frame \mathcal{F}_b is in fact the difference between the turret angle θ_t and the base angle θ_b given by Nomad 200 odometry.

The task function formalism, introduced in (Samson *et al.*, 1990), used is:

$$e(r, t) = L^{T+} \Big|_{s=s^*} (s(r, t) - s^*) \quad (2)$$

where $s(\cdot)$ is the current visual features parameters related with the robot configuration r , $s^*(\cdot)$ are the desired values of $s(\cdot)$ and $L^{T+} \Big|_{s^*}$ is the pseudo-inverse of the interaction matrix, whose value computed at s^* is given by (Espiau *et al.*, 1992)

$$L^T \Big|_{s=s^*} = \begin{bmatrix} \frac{-1}{z_d} & 0 & \frac{-a}{z_d} & -a^2 & -1 - a^2 & a \\ \frac{-1}{z_d} & 0 & \frac{a}{z_d} & a^2 & -1 - a^2 & a \\ \frac{-1}{z_d} & 0 & \frac{a}{z_d} & -a^2 & -1 - a^2 & -a \\ \frac{-1}{z_d} & 0 & \frac{-a}{z_d} & a^2 & -1 - a^2 & -a \\ 0 & \frac{-1}{z_d} & \frac{a}{z_d} & 1 + a^2 & a^2 & a \\ 0 & \frac{-1}{z_d} & \frac{a}{z_d} & 1 + a^2 & -a^2 & -a \\ 0 & \frac{-1}{z_d} & \frac{-a}{z_d} & 1 + a^2 & a^2 & -a \\ 0 & \frac{-1}{z_d} & \frac{a}{z_d} & 1 + a^2 & -a^2 & a \end{bmatrix} \quad (3)$$

where $a = \frac{1}{2}z_d$, l is the vertex length and z_d the desired depth. The control law, computed in terms of \mathcal{F}_c speed, which will bring $e(r, t)$ to zero is selected as (Espiau *et al.*, 1992)

$$T_c = -\lambda e \quad \text{with} \quad \lambda > 0. \quad (4)$$

Given the velocities of the frame \mathcal{F}_c attached to the camera (T_c), it is now necessary to derive a Jacobian from T_c to the robot control inputs. In this paper we consider two cases: *i*) using the forward speed v , steering speed $\dot{\theta}_b$, and turret rotation speed $\dot{\theta}_t$ (next subsection), and *ii*) replacing $\dot{\theta}_b$ by the pan speed $\dot{\theta}_p$ (in Subsection 3.2.). In practical terms, the difference between them is that for the first case, camera and robot base have the same axes of rotation, while in the second case this is not true. In any case, it is still possible to solve the problem of positioning the camera in front of the reference image dividing it in two subtasks according to the control inputs.

3.1. Visual servo control without pan motion

In this case, the visual servoing technique is applied to the Nomad 200 mobile robot using as control input $u = (v, \dot{\theta}_b, \dot{\theta}_t)^T$. The angle θ_p and the pan speed $\dot{\theta}_p$ are fixed to zero and the orientation of \mathcal{F}_t with respect to \mathcal{F}_b is $\Delta\theta = \theta_t - \theta_b$.

As the turret speed $\dot{\theta}_t$ (used to rotate the camera) and base speed $\dot{\theta}_b$ (used to steer the robot) share the same rotation axes, the Jacobian from the velocities of \mathcal{F}_c in the plane to the control vector is singular. We solve this singularity problem by separating the variables which compose the task (${}^c\dot{x}_c, {}^c\dot{z}_c, {}^c w_y$) in two sub-tasks and solve them separately in the frames where the Nomad 200 control inputs is defined.

The first sub-task is composed by ${}^c w_y$, which will be solved using the $\dot{\theta}_t$ control input. This part is trivial, as \mathcal{F}_c is rigidly attached to \mathcal{F}_t and due to the property of the Nomad 200 of not transferring to upper frames the rotation of the base, then one has

$${}^t w_y = {}^p w_y = {}^c w_y \quad (5)$$

The remaining sub-task, composed by (${}^c\dot{x}_c, {}^c\dot{z}_c$) is solved in the base frame \mathcal{F}_b . In this case, one has

$${}^b v_c = {}^b Rot_c * {}^c v_c \quad (6)$$

where ${}^c v_c = [{}^c\dot{x}_c, {}^c\dot{z}_c]^T$ and

$${}^b Rot_c = \begin{bmatrix} \cos \Delta\theta & -\sin \Delta\theta \\ \sin \Delta\theta & \cos \Delta\theta \end{bmatrix} \quad (7)$$

Using a previous result from Samson and Ait-Abderrahim (Samson and Ait-Abderrahim, 1990) and after some computations, one obtains the relationship between robot control signal at \mathcal{F}_b ($U = [v, \dot{\theta}_b]^T$) and ${}^c v_c$:

$${}^b v_c = J * U \Rightarrow {}^c v_c = M * U \quad (8)$$

where

$$M = {}^c Rot_b * J, \quad \text{with} \quad J = \begin{bmatrix} -1 & {}^b x_\tau \\ 0 & -{}^b z_\tau \end{bmatrix} \quad (9)$$

and ${}^b z_\tau, {}^b x_\tau$ are the coordinates at \mathcal{F}_b of the distance from the origin of \mathcal{F}_c and the desired position. In order to avoid 3D reconstruction, and considering that the task is specified in terms of a final desired configuration of \mathcal{F}_τ with respect to \mathcal{F}_c (${}^c z_\tau(t_f) = z_d, {}^c x_\tau(t_f) = 0, \theta_\tau^c(t_f) = 0$) the strategy employed is to use ${}^c\dot{z}_\tau(t) = z_d$, for all t and a very simple approximation for ${}^c x_\tau(t)$ based directly on the sensor information:

$${}^c\hat{x}_\tau(t) = \frac{z_d}{4\lambda} * \sum_{i=1}^4 X_i \quad (10)$$

where λ is the focal length, which if not available may be set equal to one, and $X_i, i = \{1, 2, 3, 4\}$ are the X-coordinate at image plane of the four points.

Then, one has

$$\begin{bmatrix} {}^b\hat{z}_\tau \\ {}^b\hat{x}_\tau \end{bmatrix} = {}^b Rot_c \begin{bmatrix} d_c^t + {}^c\hat{z}_\tau \\ {}^c\hat{x}_\tau \end{bmatrix} \quad (11)$$

And then,

$$J = \begin{bmatrix} -1 & (d_c^t + z_d) \sin \Delta\theta + {}^c\hat{x}_\tau \cos \Delta\theta \\ 0 & -(d_c^t + z_d) \cos \Delta\theta + {}^c\hat{x}_\tau \sin \Delta\theta \end{bmatrix} \quad (12)$$

where $d'_c = d_p + d_c$ as in Fig.(3).

To find the relation M from U to ${}^c v_c$ one has to compute

$$M^{-1} = \frac{1}{D} \begin{bmatrix} (d'_c + z_d) & {}^c \hat{x}_\tau \\ \sin \Delta\theta & \cos \Delta\theta \end{bmatrix} \quad (13)$$

$$D = (d'_c + z_d) \cos \Delta\theta - {}^c \hat{x}_\tau \sin \Delta\theta \quad (14)$$

which is used to control the second sub-task.

These subtasks are not decoupled and to guarantee their convergence from any initial condition a control scheme for the whole base configuration must be used. An alternative is to use different rotation axes for the base and camera, which is done considering the pan speed instead of robot turret speed, as shown in the sequence.

3..2. Visual servo control using pan motion

In this section we make the turret frame \mathcal{F}_t always aligned with the base frame \mathcal{F}_b , and assume that the orientation of the turret with respect to the base is zero, which means that $\theta_t - \theta_b = 0$. In other words, both frames are rigidly connected.

The relationship between the velocity of the 3D frame attached to the camera \mathcal{F}_c and the 3D frame attached to the base \mathcal{F}_b is now non-singular and can be easily computed using the classical velocity transfer relationship along a sequence of revolute joints:

$${}^{i+1}w_{i+1} = \sum_{j=i}^{i+1} w_j^{i+1} \quad (15)$$

$${}^{i+1}v_{i+1} = Rot_i^{i+1} \left(v_i^i + w_i^i \wedge \overrightarrow{0_{i+1}^i} \right) \quad (16)$$

where, ${}^j v_i$ is the translational speed of frame i given in frame j , ${}^j w_i$ is the rotation of frame i given in frame j , $\overrightarrow{0_i^j}$ is the coordinates of the origin of frame i in the frame j

$${}^j Rot_i = \begin{bmatrix} \cos \theta_{ji} & \sin \theta_{ji} & 0 \\ -\sin \theta_{ji} & \cos \theta_{ji} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

is the rotation matrix from frame i to frame $i + 1$, where θ_{ji} is the angle formed by the x-axis of frame i and the x-axis of frame j given in the frame j .

For the frames showed in Fig.(3), one has

$${}^b w_b = [0 \quad 0 \quad \dot{\theta}_b]^T \quad (18)$$

$${}^b v_b = [v \quad 0 \quad 0]^T \quad (19)$$

Using the characteristics of Nomad 200, if base and turret rotate together, one has at the turret frame

$${}^t w_t = [0 \quad 0 \quad \dot{\theta}_t = \dot{\theta}_b]^T \quad (20)$$

$${}^t v_t = {}^t Rot_b ({}^b v_b + 0) = [v \quad 0 \quad 0]^T \quad (21)$$

Now, computing the speeds at the pan-tilt frame, one has

$${}^p w_p = [0 \quad 0 \quad \dot{\theta}_b + \dot{\theta}_p]^T \quad (22)$$

$${}^p v_p = {}^p Rot_t ({}^t v_t + w_t^t \wedge \overrightarrow{O_p^t}) \quad (23)$$

$${}^p v_p = \begin{bmatrix} \cos \theta_p & \sin \theta_p & 0 \\ -\sin \theta_p & \cos \theta_p & 0 \\ 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} v \\ d_p \dot{\theta}_b \\ 0 \end{bmatrix} \quad (24)$$

$${}^p v_p = \begin{bmatrix} v \cos \theta_p + d_p \dot{\theta}_b \sin \theta_p \\ -v \sin \theta_p + d_p \dot{\theta}_b \cos \theta_p \\ 0 \end{bmatrix} \quad (25)$$

Finally, in the camera frame, one has

$${}^c w_c = [0 \quad 0 \quad \dot{\theta}_b + \dot{\theta}_p]^T \quad (26)$$

$${}^c v_c = {}^c Rot_p ({}^c v_c + {}^c w_c \wedge \overrightarrow{cO_p}) \quad (27)$$

$${}^c v_c = {}^c Rot_p \left(\begin{bmatrix} v \cos \theta_p + d_p \dot{\theta}_b \sin \theta_p \\ -v \sin \theta_p + d_p \dot{\theta}_b \cos \theta_p \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \dot{\theta}_b + \dot{\theta}_p \end{bmatrix} \wedge \begin{bmatrix} d_c \\ 0 \\ 0 \end{bmatrix} \right) \quad (28)$$

$${}^c v_c = \begin{bmatrix} v \cos \theta_p + d_p \dot{\theta}_b \sin \theta_p \\ -v \sin \theta_p + d_p \dot{\theta}_b \cos \theta_p + d_c (\dot{\theta}_b + \dot{\theta}_p) \\ 0 \end{bmatrix} \quad (29)$$

where d_p is the displacement between the center of rotation and the pan frame \mathcal{F}_p ; and d_c corresponds to the distance between the pan frame and the camera frame.

Re-writing the Eq.(29) in the form

$$V = J * U \quad (30)$$

where $V = [{}^c \dot{x}_c, {}^c \dot{z}_c, {}^c w_y]^T$ is the vector composed with the translational and rotational speeds of the camera frame in the horizontal plane and $U = [v, \dot{\theta}_b, \dot{\theta}_p]^T$ is the vector of control signals for driving the Nomad 200, one has

$$J = \begin{bmatrix} \cos \theta_p & +d_p \sin \theta_p & 0 \\ -\sin \theta_p & d_p \cos \theta_p + d_c & d_c \\ 0 & 1 & 1 \end{bmatrix} \quad (31)$$

whose inverse, with the substitution of $\cos \theta$ and $\sin \theta$ to $c\theta$ and $s\theta$, is given by:

$$J^{-1} = \begin{bmatrix} c\theta_p & -c\theta_p & d_c c\theta_p \\ c\theta_p/d_p & c\theta_p/d_p & -d_c c\theta_p/d_p \\ -s\theta_p/d_p & -c\theta_p/d_p & (d_p + d_c c\theta_p)/d_p \end{bmatrix} \quad (32)$$

This is the relation used to compute the control signals based on the output of the Interaction Matrix L^{T+} .

4. Experimental Results

In this section the results of the two cases are presented. The hardware setup used in the experiment were basically a Nomad 200 mobile robot and a PC Intel 300 MHz, 64 MB RAM, used as robot terminal, to edit files and to remotely control the robot. All algorithms were implemented using Tcl/Tk in order to facilitate the design and debugging process.

The goal consists in positioning the camera frame parallel to the vertical plane of the reference frame with a distance of 75cm between them, as seen in Fig.(2). The experiment was performed first using only robot actuators and then using pan motion and disabling turret rotation from the same initial condition according to sections 3.1 and 3.2, respectively. Figure (4) shows the evolution at each iteration of the four dots.

The reference image was located in a half-wood-half-glass wall, where the two upper dots are located in the glass part. Although this setup is strongly non-recommended for a vision task, it is useful to validate the robustness of the vision algorithm, fully described in (Carvalho *et al.*, 2000) and refined in (Donnoui *et al.*, 2001).

Figures (5)-a to -c shows the evolution of the control signal for the visual servo control without pan motion while in Fig. 5-d the evolution of the Euclidean norm of the error of the image plane coordinates for each dot is shown. The same set of results for the second case study is depicted through Fig.(6)-a to -d.

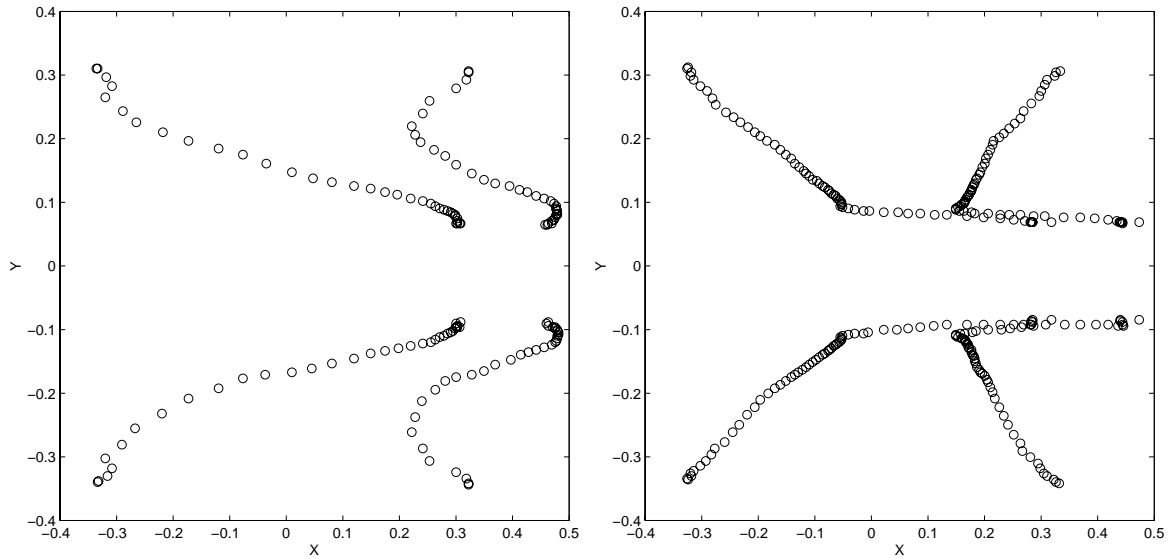


Figure 4. The evolution of the four dots during robot motion: without pan motion (left) and using pan motion (right).

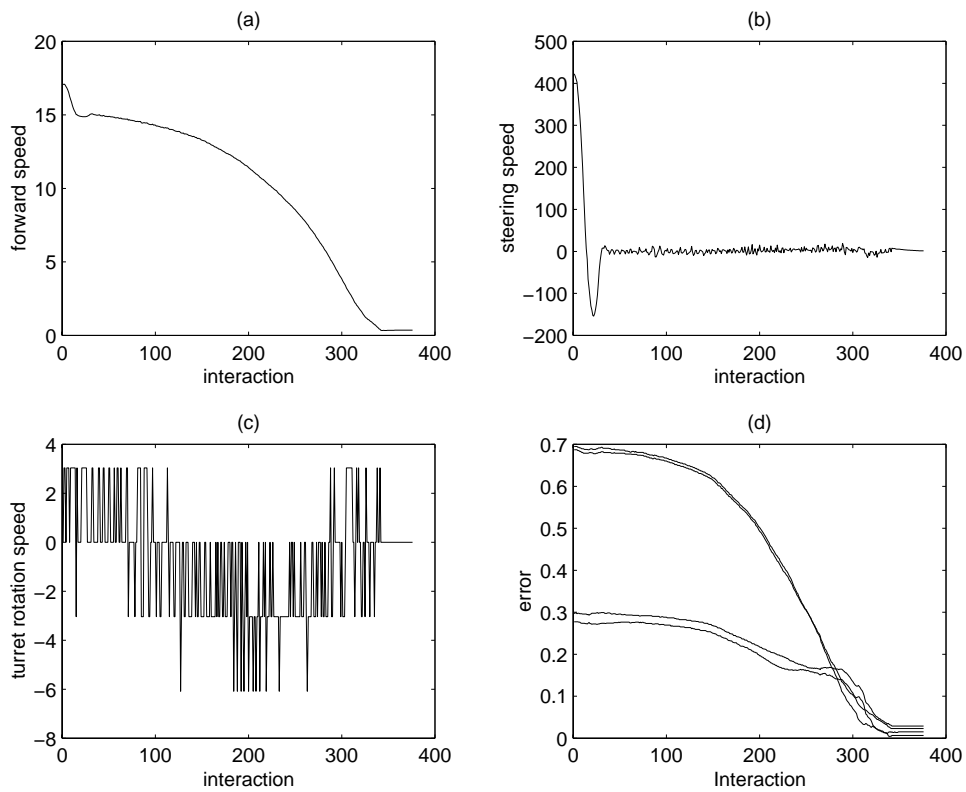


Figure 5. (a)-(c) Control signals. (d) Evolution of the norm of the errors (without pan motion).

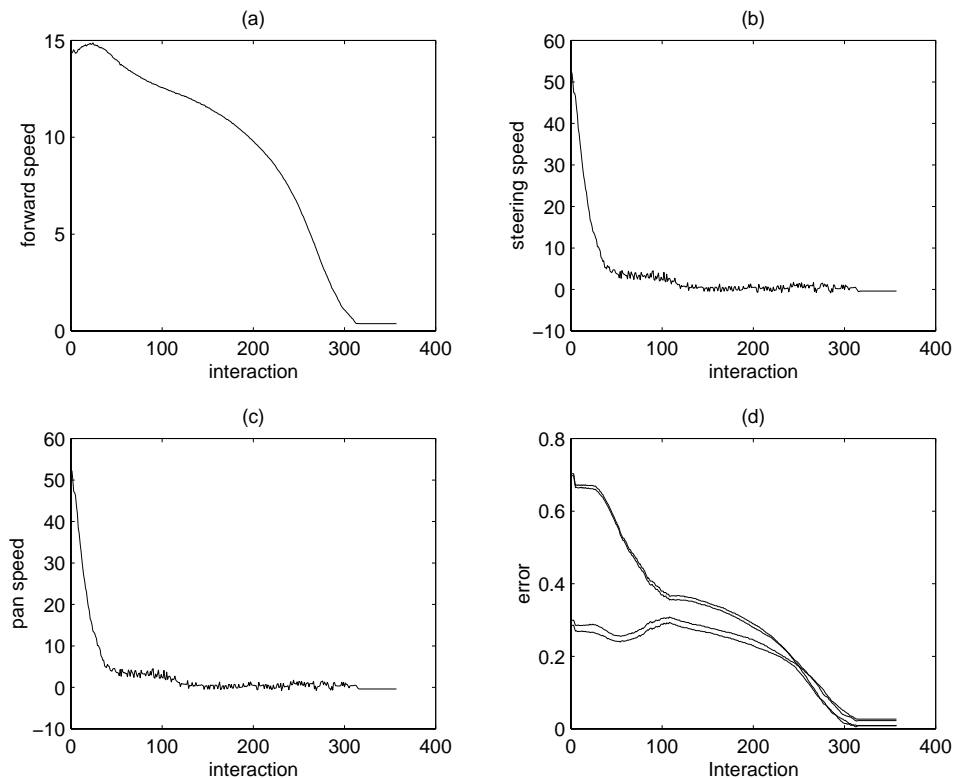


Figure 6. (a)-(c) Control signals. (d) Evolution of the norm of the errors (using pan motion).

5. Discussions and Conclusion

In this section we present an analysis of the results obtained in this work. First and foremost, the reader will recall that the interaction matrix was kept constant and computed for the desired final pose of the camera. A case study about the influence of adding more information to the computation of the interaction matrix can be found in (Silveira *et al.*, 2001b). However, this does not present a severe limitation of the method, since the positioning tasks are usually performed close to the goal position. Additionally, experiments were performed from a starting point about 10 times larger than the final desired depth of the image and the performance obtained was satisfactory.

Both robot and pan-tilt actuators present a saturation for large input signals. To avoid nonlinearities due to saturation, an upper bound for the control was used, selected as half of the manufacturer's stated saturation value. Also, robot and pan-tilt actuators operate with finite resolution. The resolution of robot actuators did not present any problem, but the resolution on pan motion degenerated the results. To overcome this problem we adopted the following strategy: whenever the pan input signal was calculated as a value between 30 and 100% of the smallest non-zero input signal, we set it equal to the smallest input signal.

The best execution time achieved was about 200ms per interaction. Since this can not be considered real-time, the forward speed had to be limited to 20% of the maximum speed and all rotations speed to 50% of the maximum. Execution time can be reduced significantly by using a compiled programming language and hardware based on a state-of-the-art microprocessor. Tcl/Tk was considered here because of its capabilities of data logging and plotting, reduced implementation time, platform independence, most software and hardware compatibility and for debugging purposes.

Most of industrial mobile robots are submitted to nonholonomic constraints. In the present case, despite the controllability of the system, the stabilization of the robot base around a given configuration by pure state feedback is not possible (Samson and Ait-Abderrahim, 1990). In this paper only the camera pose is controlled, and the final robot base configuration depends on the initial condition. For the first case study (without pan motion) and due to the characteristics of the Nomad 200 mobile robot, the execution of the task implies that the robot will always stop in the same position (although with different orientation), but this may not be sufficient in many real problems, for instance the problem of parking a robot. One approach to solve this problem is to utilize the pan motion along with a time-varying control strategy to stabilize the base configuration or to add more DOFs to the camera to control the whole mobile-manipulator platform. A complete discussion on this subject can be found in (Tsakiris *et al.*, 1998) and references therein.

In the view of the above discussion, the obtained results constitute a first step towards the conception of an accessible real-world implementation of visual servoing of mobile robots.

6. Acknowledgments

This project earns partial grant from FAPESP under the process 97/13384-7 and 98/13562-5 and ProTeM-CC CNPq / INRIA under the process 68.0140/98-0.

7. References

- Carvalho, José R. H., Patrick Rives, Ailton Santa-Bárbara and Samuel S. Bueno (2000). Visual servo control of a class of mobile robot. In: *2000 IEEE International Conference on Control Applications*. Alaska, USA.
- Donnouti, Leonardo S., Geraldo F. Silveira, Pedro M. Shiroma, J. Reginaldo H. Carvalho, Patrick Rives and Samuel S. Bueno (2001). Mobile robot positioning via visual servoing techniques. In: *Proceedings of the IV Brazilian Symposium on Intelligent Automation*. Gramado/RS, Brazil. submitted.
- Espiau, Bernard, François Chaumette and Patrick Rives (1992). A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation* **8**, 313–326.
- Samson, Claude and Karim Ait-Abderrahim (1990). Mobile robot control, part 1: Feedback control of a nonholonomic wheeled cart in cartesian space. Technical report. INRIA, Centre de Sophia-Antipolis.
- Samson, Claude, Bernard Espiau and M. le Borges (1990). *Robot Control: the Task Function Approach*. Oxford University Press. USA.
- Silveira, Geraldo F., J. Reginaldo H. Carvalho, Marconi K. Madrid, Patrick Rives and Samuel S. Bueno (2001a). A fast vision-based road following strategy applied to the control of aerial robots. In: *XIV Brazilian Symposium on Computer Graphics and Image Processing*. Florianópolis, Brazil. accepted.
- Silveira, Geraldo F., J. Reginaldo H. Carvalho, Marconi K. Madrid, Samuel S. Bueno and Patrick Rives (2001b). Towards vision guided navigation of autonomous aerial robots. In: *Proceedings of the IV Brazilian Symposium on Intelligent Automation*. Gramado/RS, Brazil. submitted.
- Tsakiris, D., P. Rives and C. Samson (1998). Extending visual servoing techniques to nonholonomic mobile robots. In: *The Confluence of Vision and Control*. Lecture Notes in Control and Information System. Springer-Verlag. D. Kriegman (Ed.).

PROJETO E IMPLEMENTAÇÃO DE UM SENSOR DE TATO PARA UMA GARRA ROBÓTICA

Fernando Silveira Madani

ITA, Div. Eng. Mecânica Aeronáutica, Pça. Mal. Eduardo Gomes, 50, CTA, S. J. dos Campos – SP, CEP: 12228-901
e-mail: madani@mec.ita.br

Alberto Adade Filho

ITA, Div. Eng. Mecânica Aeronáutica, Pça. Mal. Eduardo Gomes, 50, CTA, S. J. dos Campos – SP, CEP: 12228-901
e-mail: adade@mec.ita.br

Resumo. Sensores de tato são dispositivos que têm como objetivo indicar o contato entre o sensor e algum objeto, acusando se há contato ou não (sensores de toque) ou indicando com que força acontece tal contato (sensores táteis). Um sensor tátil, aplicado a robôs, é o objeto de estudo deste trabalho. A capacidade do robô medir a força que sua garra exerce sobre os objetos manipulados possibilita uma maior gama de aplicações. Este trabalho descreve o projeto e alguns dos principais aspectos da implementação de um sensor tátil para a garra de um manipulador robótico. Este sensor utiliza vários elementos de feltro de carbono como elemento sensível, formando uma malha, e as características de toque são decodificadas pela detecção de variação de voltagem em decorrência da variação de resistência elétrica no elemento. O sensor é avaliado computacionalmente, através de interface via porta serial, e são apresentados resultados da calibração e teste do sensor. A motivação deste desenvolvimento está centrada: (1) na realização do controle refinado de agarramento, possibilitando a manipulação de elementos frágeis; e (2) na capacitação para o desenvolvimento de sensores para aplicações na robótica.

Palavras chave: sensores táteis, sensor de feltro de carbono, sensores robóticos.

1. Introdução

Diversos métodos e dispositivos aplicados à robótica têm sido desenvolvidos com o intuito de tornar os robôs mais “inteligentes” e lhes possibilitar uma maior gama de aplicações. Os sensores são dispositivos que fornecem aos robôs uma capacidade perceptiva podendo então haver uma maior e melhor interação dos robôs com o meio ambiente, o espaço de trabalho, tornando os robôs capazes de realizar tarefas diversas utilizando-se de recursos semelhantes aos sentidos básicos dos seres humanos.

Os sensores de tato podem ser classificados segundo a característica de seu elemento sensor, que pode ser capacitivo, indutivo, óptico, piezoeletrico, piezoresistivo, entre outros (Andrew, 1990). Todos os materiais utilizados para este fim possuem vantagens e limitações, devendo a escolha ser feita levando-se em conta a aplicação.

As características do elemento de feltro de carbono para aplicação como sensor de tato foram apresentadas em (Larcombe, 1981; Webster, 1988 ; e Andrew, 1990) por exemplo, no entanto nenhuma destas referências mostra uma visão global e completa de um protótipo de sensor, possibilitando assim a verificação e requisitos do hardware necessário à montagem desse protótipo e também seu custo de montagem.

Este trabalho propõe um protótipo de sensor piezoresistivo à base de feltro de carbono que visa fornecer a uma garra robótica com fechamento proporcional o sentido do tato, possibilitando ao robô o controle da força com que este agarra os objetos que são manipulados. São verificadas e demonstradas as características básicas deste sensor.

É de interesse projetar um sensor simples e de custo relativamente baixo. Uma interface com conversor A/D e conexão via porta serial foram implementadas de modo a permitir a realização de diversos experimentos com o protótipo, para o desenvolvimento deste e buscando a capacitação para o desenvolvimento de sensores mais refinados e complexos e também para aplicações mais específicas.

O trabalho compreende o desenvolvimento completo do protótipo proposto. A escolha do feltro de carbono se justifica dadas as facilidades de obtenção do material, de montagem do protótipo, simplicidade do hardware, e ainda seu custo reduzido, características estas comparadas a de outros sensores descritos em (Andrew, 1990), como os sensores piezoresistivos utilizando elastomero condutivo, e os sensores piezoeletricos, ópticos, magnéticos, ultrasonicos, capacitivos e eletroquímicos.

O trabalho traz como principal contribuição um sensor de tecnologia simples e robusta e de baixo custo. É descrito integralmente a montagem deste sensor, tornando possível um maior conhecimento e visualização de seu funcionamento e ainda a realização de alterações tanto de hardware como de software para aprimoramento do protótipo ou adequá-lo a outras aplicações.

1.1. Organização do trabalho

O trabalho foi dividido em 9 seções que trazem a descrição de cada item relacionado à implementação do protótipo de sensor tátil.

A seção 2 apresenta uma breve descrição da garra na qual será montado o sensor proposto, na seção 3 é feita a descrição do elemento sensor, compreendendo a descrição do material e a maneira de montagem dos elementos. Os testes, os dispositivos e os circuitos para a obtenção das curvas que representam as variações de resistência e de tensão que o elemento gera devido à aplicação de carga sobre ele são apresentados na seção 4. Na seção 5 é proposta a malha de elementos sensores que deverá ser montada na garra que iremos utilizar, a seção 6 descreve o circuito utilizado para interfacear o sensor e um microcomputador PC através da porta serial, a seção 7 apresenta o software responsável pela “leitura” do sensor, a seção 8 mostra os testes e resultados obtidos com este sensor, e finalmente na seção 9 são apresentadas as conclusões obtidas com a pesquisa e implementação do sensor objeto deste trabalho.

2. A garra

Existem diversos modelos de garras robóticas, a maioria diretamente relacionada à aplicação. Alguns modelos são apresentados em (Andeen, 1988).

A garra utilizada é do tipo *standard*, mostrada na Fig.(1), e tem como mecanismo de acionamento um servo-motor, que possui como característica básica a relação de sua rotação com a tensão aplicada. A Fig.(2) mostra o circuito do servo e um gráfico que relaciona o fechamento com a tensão aplicada.

Uma garra com fechamento proporcional tem como característica principal a sua capacidade de controle gradual do fechamento, que é determinado de acordo com o sinal de comando de entrada, ao atuador da garra.

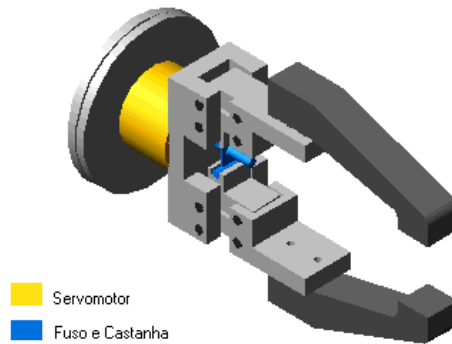


Figura 1. Modelo da garra utilizada

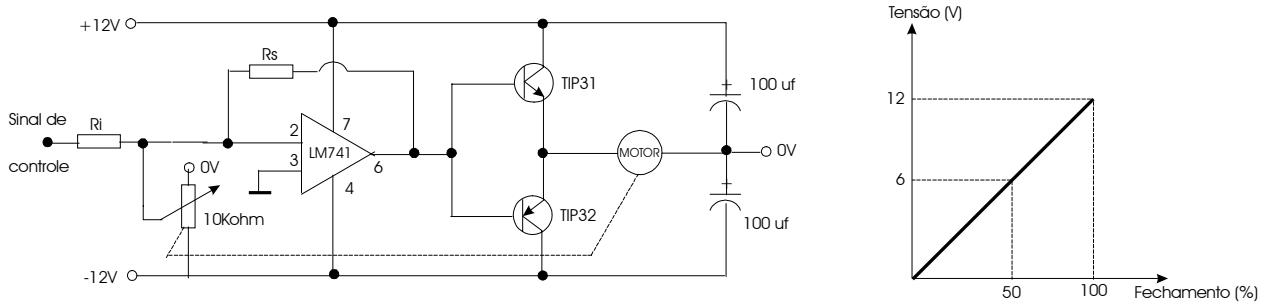


Figura 2. Circuito de acionamento da garra

Para a realização dos testes vai-se gerar o sinal de controle a partir da porta serial de um microcomputador PC-compatível. Para tanto é necessário um circuito conversor D/A (8 bits), sendo que a saída deste segue a Eq.(1):

$$V_{s\grave{a}ida} = -V_{ref} \cdot \sum_{n=1}^8 Bit_n \cdot 2^{-n} \tag{1}$$

onde:

V_{ref} é a tensão de referência para a conversão; e

Bit_n indica os bits de 1 a 8 do valor digital a ser convertido.

A região da garra onde o sensor é fixado está mostrada na Fig.(3).

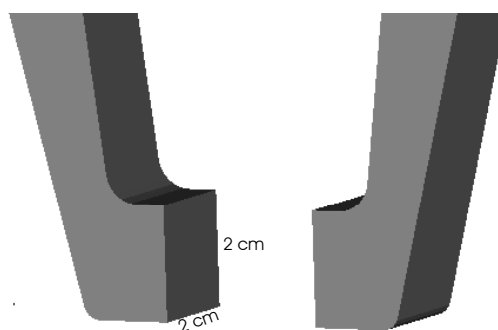


Figura 3 – Região de contato da garra.

3. O elemento sensor

Será utilizado o feltro de carbono como elemento sensor como proposto em (Larcombe, 1981). Este material possui características que podem limitar sua utilização, como a impossibilidade de trabalhar com elementos muito pequenos, sendo difícil o manuseio de elementos menores que 3mm, em contra partida este material apresenta outras características bastante interessantes que são:

- capacidade de suportar cargas elevadas sem dano ao material; em (Larcombe, 1981), após a aplicação de uma carga de 2GPa o erro para uma medida de carga igual a 1g foi de 1%;
- faixa de variação bastante larga não havendo a necessidade do uso de amplificadores;
- o sensor não apresenta variação perceptível com a mudança de temperatura, para utilização prática;
- massa específica (0,05 – 1,00 g/cm³), não comprometendo a capacidade de carga do robô;
- facilidade de obtenção do material, como descrito em (Nohara, 1998);
- o elemento pode ser montado de diversas formas pois o material é bastante maleável.

Com estas características acredita-se que este tipo de sensor seja bastante interessante para aplicações em ambiente industrial, observando que pode ser um ambiente bastante agressivo a outros tipos de sensores.

É importante comentar outras características do feltro de carbono que são relacionadas ao seu uso para o fim descrito neste trabalho:

Histerese – em (Davies, 1984) foi estimada em 6% e em (Pruski e Mutel, 1983) é relatado que a histerese do feltro de carbono é desprezível.

Ruído – é muito significativo quando o sensor está em contato com metais ou submetido a cargas menores que 10g, devendo-se evitar tais situações com o uso de isolamento e aplicação de pré-carga.

Vale ainda lembrar que os valores característicos do material podem variar de acordo com a forma de obtenção do mesmo, sendo então necessária uma atenção especial a estas características, principalmente no momento da calibração (aferição) inicial.

O elemento proposto neste trabalho visa sua instalação em uma garra já em uso, observando as dimensões dadas na Fig.(3). O elemento é composto por duas chapas metálicas e um pedaço de feltro de carbono montados na forma de sanduíche como se vê na Fig.(4). As dimensões do elemento sensor são 10x5x3 mm (larg.,compr.,espessura); essa espessura se deve ao feltro que foi disponibilizado para esse experimento, sendo que espessuras menores podem ser utilizadas.

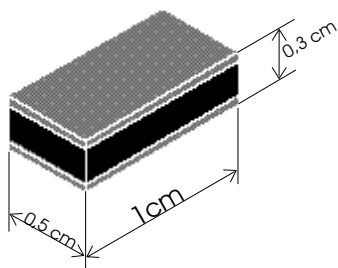


Figura 4 – Elemento sensor montado.

Com o elemento montado conforme ilustra a Fig.(4), é possível, através de testes, obtermos as curvas que relacionam as variações de resistência elétrica e tensão ao longo do elemento sensor, que tem como princípio de funcionamento a variação da resistência elétrica do feltro de carbono devido à aplicação de uma força comprimindo o elemento, aproximando assim as fibras, ilustradas na Fig.(5), e então diminuindo o valor de resistência elétrica em função da carga aplicada.

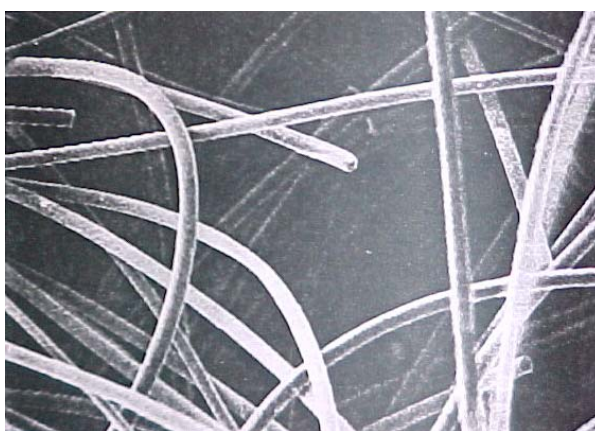


Figura 5 – Estrutura interna (extraída de Katahira, 1979, Fig.3).

4. A obtenção da curva de resposta

A característica de fundamental importância desses elementos sensores é a variação de sua resistência elétrica devido à aplicação de uma carga sobre ele. Através da medida desta variação podemos então determinar a que força o sensor está sendo submetido. Para “levantar” a curva que representa a resistência elétrica em função da carga, foi realizado o seguinte ensaio: utilizando-se uma sonda “haste” com suporte para a colocação de pesos calibrados, aplica-se uma carga conhecida ao sensor enquanto através de um aparelho capaz de medir a resistência elétrica podemos ler o valor da resistência correspondente. Assim, a curva (resistência x carga) obtida está mostrada no gráfico da Fig.(6).

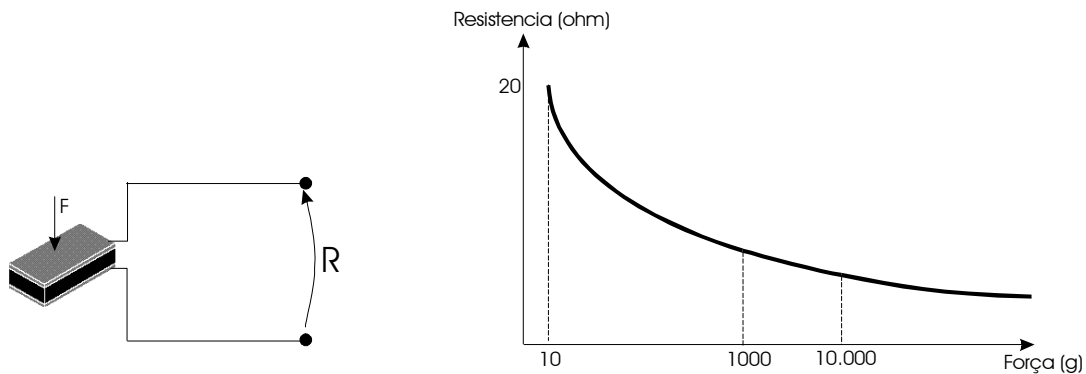


Figura 6 – Curva de Resistência x Carga.

O procedimento também pode ser aplicado para a obtenção da curva (tensão elétrica x carga). É necessário um circuito para a alimentação do sensor, e então teremos, como mostrado na Fig.(7), o gráfico da tensão em função da carga aplicada ao sensor.

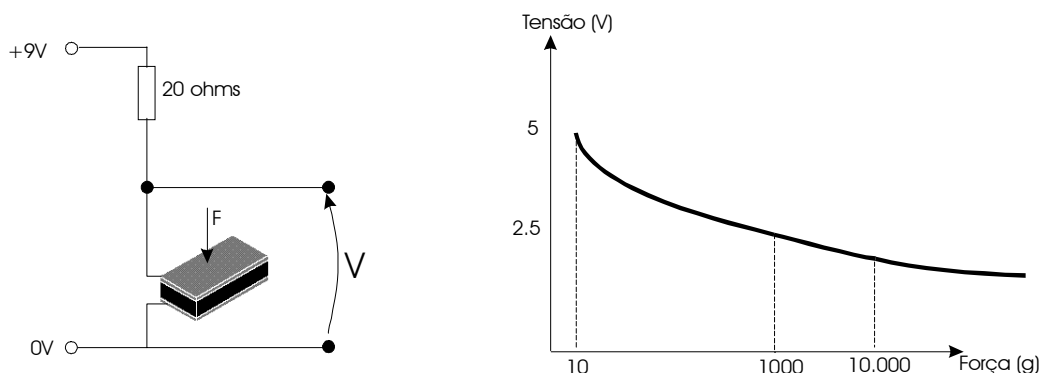


Figura 7 – Curva de tensão x carga.

5. A malha proposta para o sensor

Visando a montagem do sensor na garra podemos formar uma malha de elementos sensores (descritos na seção 3), malha esta que será capaz de fornecer a variação da força aplicada pela garra sobre os objetos manipulados em diversos pontos da garra, tantos pontos quanto for o número de elementos sensores da malha. A quantidade de pontos de medida pode ser prevista de acordo com a aplicação do sensor, observando-se a área da garra que entra em contato com os objetos agarrados Fig.(3).

Para este trabalho a malha de elementos sensores é montada conforme ilustrado na Fig.(8). O número de elementos foi determinado visando a simplificação do hardware, considerando que é um protótipo.

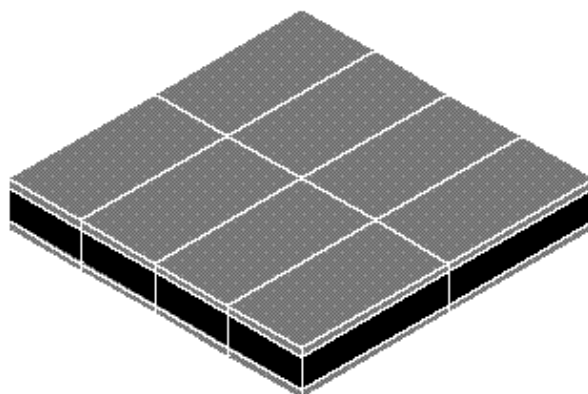


Figura 8 – Malha implementada

6. A interface (sensor x microcomputador)

Trata-se de um circuito eletrônico que irá transformar o sinal analógico do elemento sensor em sinal digital e transmitir os dados de forma serial padrão RS232 a um microcomputador PC-compatível.

A utilização da conexão serial foi escolhida pois a transmissão dos dados é mais confiável por esta do que através da transmissão paralela.

A conversão do sinal analógico para digital é a primeira etapa sendo o circuito integrado ADC0808 responsável por esta etapa. Este CI é capaz de transformar até 8 sinais analógicos de tensão em sinais digitais individualmente.

Os dados digitais na saída do ADC0808 são lidos pelo micro-controlador AT89C51 da Atmel Corp., que é responsável por ler os diversos valores na forma digital fazendo a varredura dos elementos sensores que compõe a malha sensorial (seção 5). Através de seu canal serial, o micro-controlador envia os dados já na forma desejada, faltando apenas a adequação da voltagem de operação entre o micro-controlador e o PC, tarefa esta que é de responsabilidade do CI dedicado MAX232 da Maxim.

A Figura(9) mostra o circuito da interface, através da qual os dados serão disponibilizados para o PC na forma serial, podendo então ser lidos através de software, como veremos na seção 7.

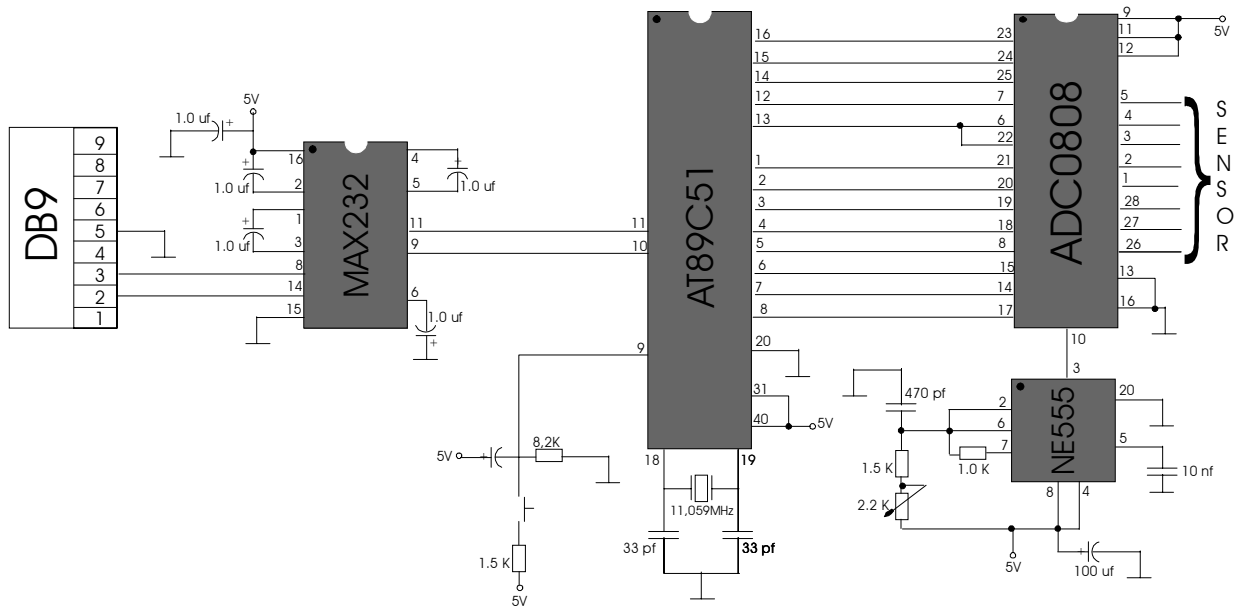


Figura 9 – Circuito de Interface sensor-microcomputador.

7. Software

7.1. O software de alto nível

Desenvolvido em Delphi, o software tem por objetivo realizar o protocolo de interfaceamento serial com o PC possibilitando a leitura do sinal que é fornecido pela interface, e mostrar na tela a força aplicada a cada elemento sensor.

O software oferece ainda a possibilidade de comandar a abertura da garra e o estabelecimento de controles simples da garra, comandada através da porta paralela do microcomputador.

A Fig. (10) mostra a tela do programa, que é sub-dividido em duas partes sendo:

1ª) *Janela do Sensor*, onde podemos visualizar as forças que estão sendo aplicadas a cada elemento sensor.

2ª) *Janela da Garra*, controla o estado de abertura da garra fornecendo a possibilidade de controle manual do fechamento ou determinar que a garra seja fechada até que se atinja uma determinada força de aperto.

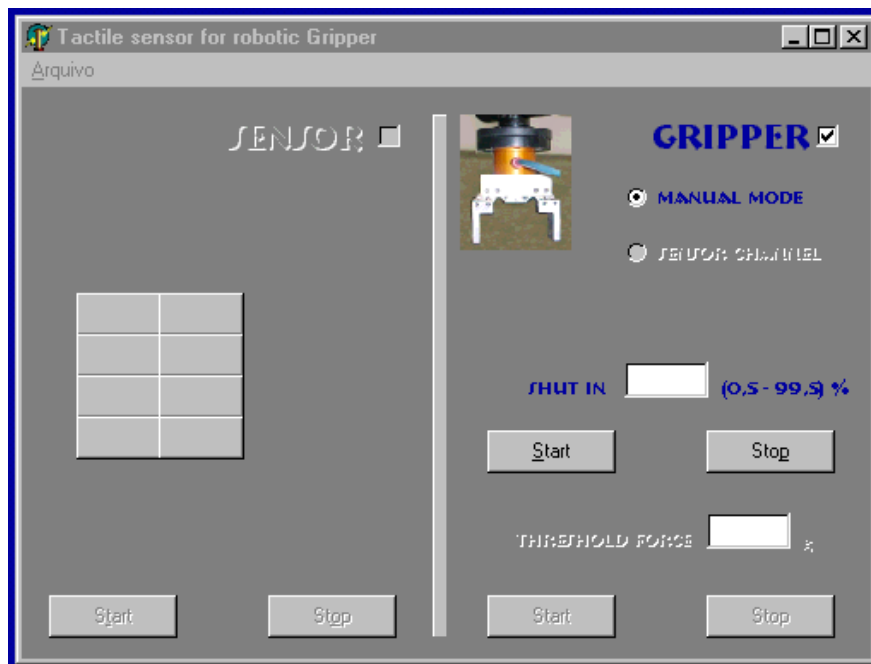


Figura 10 – Tela do software.

7.2. O software de baixo nível

Trata-se de um programa em linguagem assembler, responsável pelo funcionamento do microcontrolador, que irá receber um Byte enviado pelo PC através da porta serial na forma (0 1 0 1 0 C B A), sendo que os três bits menos significantes correspondem aos valores de C, B e A, que determinam o endereço do conversor analógico/digital que deverá ser lido, conforme Tab.(1). Após a leitura do endereço que foi pedido pelo PC o microcontrolador lê o dado digital convertido pelo ADC e então o envia na forma serial ao PC, com taxa de transferencia de 19,2 Kbps.

A listagem do software de baixo nível pode ser vista na Fig.(11).

Tabela 1 – Endereços do ADC 0808.

Canal analógico selecionado	Endereço			Byte enviado pelo PC
	C	B	A	
In0	0	0	0	50H
In1	0	0	1	51H
In2	0	1	0	52H
In3	0	1	1	53H
In4	1	0	0	54H
In5	1	0	1	55H
In6	1	1	0	56H
In7	1	1	1	57H

```

SP DATA 81H
P1 DATA 90H
P3 DATA 0B0H
SCON DATA 98H
TMOD DATA 89H
TCON DATA 88H
TH1 DATA 8DH
PCON DATA 87H
IE DATA 0A8H
SBUF DATA 99H
B DATA 0F0H
PSW DATA 0D0H

TRANS: JMP LOOP

RECEP: RRC A
MOV P3.4, C
RRC A
MOV P3.5, C
RRC A
MOV P3.6, C
SETB P3.3 ; START
CLR P3.3

MOV R5,#0FFH
ESPO: DJNZ R5,ESPO
JMP FORA
ESPOO: JB P3.2,ESPO

ESP1: JNB P3.2,ESP1

MOV A,P1
MOV SBUF, A
JMP LOOP

FORA: MOV SBUF,#0FFH
JMP LOOP

ORG 200H
JB SCON.0, SUB1
JB SCON.1, SUB2
JMP FIMROT

SUB1: CLR SCON.4
CLR SCON.0
MOV A, SBUF
ANL A, #00000111B
MOV R7, #0FFH
JMP FIMROT

SUB2: CLR SCON.1
MOV R7,#0H
SETB SCON.4

FIMROT: MOV R6,#1
RETI

END

ORG 0000H
JMP 100H

ORG 0023H
JMP 200H

ORG 100H
SETB P3.0
SETB P3.1
SETB P3.2

MOV P1,#0FFH
MOV R6,#0

MOV SP,#50H
CLR SCON.7 ; INICIA SERIAL
SETB SCON.6
CLR SCON.5
MOV IE,#10010000B
MOV TMOD,#20H
MOV TH1,#0FDH ; CARREGA

TH1 MOV PCON,#80H ; SET SMOD
SETB TCON.6 ; DISPARAR

TIMER SETB SCON.4 ; HAB RECEP

LOOP: CJNE R6,#1,LOOP
MOV R6,#0

CJNE R7,#0,RECEP
    
```

Figura 11 – Listagem do software de baixo nível.

8. Testes

Como citado na introdução, o sensor proposto é um protótipo para futuros testes e desenvolvimento de aplicações, portanto os testes realizados neste trabalho visaram somente a obtenção de características básicas e verificação do funcionamento correto do hardware. Assim, foi assegurado pelos testes: a leitura correta dos valores da carga aplicada, a repetitividade dos valores medidos, a não interferência de cargas em células (diferentes) não carregadas. A influência de variações de temperatura no funcionamento do sensor também foi avaliada sendo praticamente desprezível.

8.1. Resultados Obtidos.

A calibração do sensor foi feita utilizando-se o mesmo procedimento usado para o levantamento da curva de resposta (seção 4). Foram lidos no microcomputador os valores medidos pelo sensor repetidas vezes, com diversos valores de carga sendo que o erro verificado esteve sempre na faixa definida no DataSheet do ADC0808, o que garante a leitura correta do sensor.

A repetitividade dos valores medidos foi verificada da mesma forma e quando utilizados pesos de mesmo valor as medições verificadas foram praticamente as mesmas.

A não interferência de cargas em células (diferentes) não carregadas, foi verificada através do seguinte experimento: a aplicação de carga em uma única célula da malha ou de cargas diferentes nas células da malha, e através do microcomputador novamente foram lidos os valores; cada célula da malha apresentou valores medidos relativos somente a carga que estava sendo aplicada a ela, revelando assim a não ocorrência de interferência entre as células da malha.

A não influência de variações de temperatura no funcionamento do sensor foi identificada ainda na obtenção da curva de resposta do elemento sensor (seção 4): com o elemento “carregado” com um peso fixo, a leitura do valor de resistência elétrica medido no elemento sensor quando submetido a alterações de temperatura ficou praticamente constante, observando-se que neste teste a temperatura do elemento sofreu variações dentro da faixa de aproximadamente 10° a 50°.

9. Conclusões

O sensor proposto tem como pontos importantes a repetitividade nas medidas mesmo com variações de temperatura e o baixo custo para a montagem. A vida útil varia de acordo com a aplicação, mas o sensor é bastante resistente, suportando cargas elevadas sem que haja prejuízo ao seu funcionamento. A recuperação, isto é, a volta ao estado inicial após a liberação da carga pode ser vista ocorrer em dois estágios: no primeiro a forma retorna em direção a inicial rapidamente, garantindo assim que na maioria das aplicações as próximas medidas podem ser realizadas sem prejuízo no tempo de amostragem, um segundo estágio de recuperação mais lento, já próximo ao estado inicial poderia trazer alguma limitação em aplicações especiais.

No que diz respeito a calibração, o sensor só necessita ser “aferido” para o início de operação ou quando ocorrer alguma alteração física no mesmo, sendo importante a verificação periódica como manutenção preventiva.

Deve-se dar atenção especial à calibração inicial, pois, especialmente se forem utilizados feltros obtidos através de diferentes processos, podem ocorrer variações nas curvas da seção 4.

Com as características descritas e estudadas neste trabalho acredita-se que o sensor tátil baseado nas propriedades piezoresistivas do feltro de carbono é uma opção bastante interessante para uso em garras de robôs industriais. Para aplicações especiais como a utilização em robótica médica, acredita-se que seria melhor utilizar o elemento de PVDF (polyvinylidene fluoride, elastômero piezelétrico).

10. Agradecimentos

Ao Prof. Edgard José de Faria Guimarães do Departamento de Eletrônica Aplicada - IEEA do ITA, pelo apoio e orientação dadas na montagem e programação do microcontrolador.

11. Referências

Andeen G.B.,(1988) “Robot Design Handbook”, SRI international.

Andrew R.R., (1990) "Robot Tactile Sensing", Prentice Hall of Australia Pty Ltd.

Davies J. B. C.,(1984) "Carbon fiber sensor", Proc. 4th Int. Conf. Robot Vision Sensory Control, pp295-300.

Katahira K., (1979) "A new type of force sensitive device using carbon fiber and its biomedical application", Fukushima J. Med. Sci. vol.26, pp.121-132.

Larcombe M.H.E., (1981) "Carbon Fibre Tactile Sensors", Proc. Of the First Int. Conf. On Robot Vision and Sensory Controls, IFS Ltd, Bedford, UK, pp273-6

Nohara E.L., (1988) "Estabelecimento de parâmetros de processamento de compósitos CRFC aplicados na área de elementos de fricção.", Tese de mestrado, ITA, S.J. dos Campos.

Pruski A, e Mutel B.,(1983) "Capteur sensoriel pour robot industriel", Congres ACFET Productique et Robotique Intelligente, Bescanon, France, pp287-295.

Webster J.G., (1988) "Tactile Sensors for Robotics and Medicine", John Wiley, New York.

DESIGN AND IMPLEMENTATION OF A TACTILE SENSOR FOR A ROBOTIC GRIPPER.

Fernando Silveira Madani

ITA, Div. Eng. Mecânica Aeronáutica, Pça. Mal. Eduardo Gomes, 50, CTA, S. J. dos Campos – SP, CEP: 12228-901
e-mail: madani@mec.ita.br

Alberto Adade Filho

ITA, Div. Eng. Mecânica Aeronáutica, Pça. Mal. Eduardo Gomes, 50, CTA, S. J. dos Campos – SP, CEP: 12228-901
e-mail: adade@mec.ita.br

Abstract. Tactile sensors are devices used to indicate the contact between a sensor and some object. Their outputs indicate if there is contact (touch sensors) or the force when a contact is established (tactile sensors). The last one applied to robots, is the object of study in this paper. Several applications can be envisioned for robots with tactile sensing capability, since they have the ability to measure the force that its gripper does upon the objects it manipulates. This paper describes the design and some main implementation aspects of a tactile sensor for a robotic gripper. This sensor utilizes carbon felt cells as the sensitive element, and the touch characteristics are decodified through voltage variation detection due to electric resistance variation of the sensitive elements. This sensor is connected to a computer through a serial port, and results of the calibration and test of the sensor are presented in detail. The motivations for this work are: (1) the implementation of refined grasping control; (2) acquiring some expertise in developing sensors for robotic applications.

Key words: Tactile sensors, Carbon felt sensor, Robotic sensors.

Proposta de Elaboração de Movimento de Uma Prótese a partir da Modelagem Cinemática Direta e Inversa

Edna Rodrigues Cassemiro

UNICAMP-Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica, Departamento de Projeto Mecânico, CP 6051, CEP 13083-970, Campinas-SP.

e-mails: ednarc@fem.unicamp.br

Helder Anibal Hermini

UNICAMP-Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica, Departamento de Projeto Mecânico, CP 6051, CEP 13083-970, Campinas-SP.

e-mail hermini@uol.com.br

João Maurício Rosário

UNICAMP-Universidade Estadual de Campinas, Faculdade de Engenharia Mecânica, Departamento de Projeto Mecânico, CP 6051, CEP 13083-970, Campinas-SP.

e-mails: rosario@fem.unicamp.br

Resumo: A elaboração de próteses inteligentes exige o desenvolvimento de um modelo cinemático que expresse o movimento desses modelos em termos de seus graus de liberdade dependentes. Com a evolução tecnológica muitos trabalhos de pesquisa têm estabelecido leis no controle aplicada a processos de reabilitação, visando substituir parte do corpo perdida. Este trabalho tem o objetivo de desenvolver metodologia para modelagem, simulação e desenvolvimento de dispositivos biomecânicos para aplicação em próteses de membros inferiores ou superiores. A partir da modelagem matemática de membros e atuadores lineares fisiologicamente análogo aos músculos, para tanto, utilizando a mesma metodologia usada em modelagem de manipuladores robóticos e estudo de atuadores e sistema de transmissão mecânica, foi implementado um programa computacional (linguagem ADA) de visualização e geração automática de movimentos. Uma bancada de laboratório, permite a realização de testes da interface de comando (geração de sinais de referência para a interface de acionamento, calculados em função do modelamento dos diferentes graus de liberdade das próteses a serem implementadas) e sistemas de acionamento das próteses e sensores. Nesse estudo foi dado ênfase ao membro superior.

Palavras chave: biomecânica, prótese, robótica, controle.

1. Introdução:

Muitos trabalhos relacionados a essa linha de pesquisa têm sido desenvolvidos (robótica, controle, reabilitação), porém ainda não possuem a eficiência clínica desejada. Com o objetivo de estudar e validar a possibilidade de utilização de dispositivos biomecânicos para aplicações em prótese de membros humanos e assim dar condições para uma melhor qualidade de vida ao usuário de prótese. Para reproduzir os esforços e deslocamentos equivalentes ao sistema natural, muitos aspectos devem ser considerados, pois o alto grau de complexidade da estrutura do corpo humano, como o número de graus de liberdade pode resultar em redundâncias. Outro fator importante nesse aspecto, diz respeito ao a origem da amputação, tais como diabetes, câncer, traumas decorrentes de acidentes, bem como anomalias congênitas que implicarão no tipo de prótese a ser desenvolvida influenciando no projeto geral da prótese.

Para tanto, existe a necessidade da modelagem do sistema articular humano. O modelo geométrico mostra o sistema no espaço com seus graus de liberdade. No modelo cinemático direto, todos os deslocamentos da prótese são obtidos a partir dos valores dos ângulos das juntas. No entanto para se obter os ângulos das juntas (informação necessária para enviar ao sensor de posição) é feita então a modelagem cinemática inversa. Nela usa-se conceito de matriz jacobiana.

2.1 Modelagem Cinemática Direta

Para a modelagem de membros e atuadores lineares fisiologicamente análogo aos músculos, o estudo foi baseado na metodologia utilizada para modelagem de manipuladores robóticos, incluindo de atuadores e sistemas de transmissão mecânica.

A modelagem cinemática presente neste estudo visa a construção gráfica no programa computacional. A partir da definição das articulações da estrutura esquelética, é definido um sistema de referência para cada ponto Oi. As fig. 1 e 2 mostram o sistema articular superior e inferior, respectivamente.

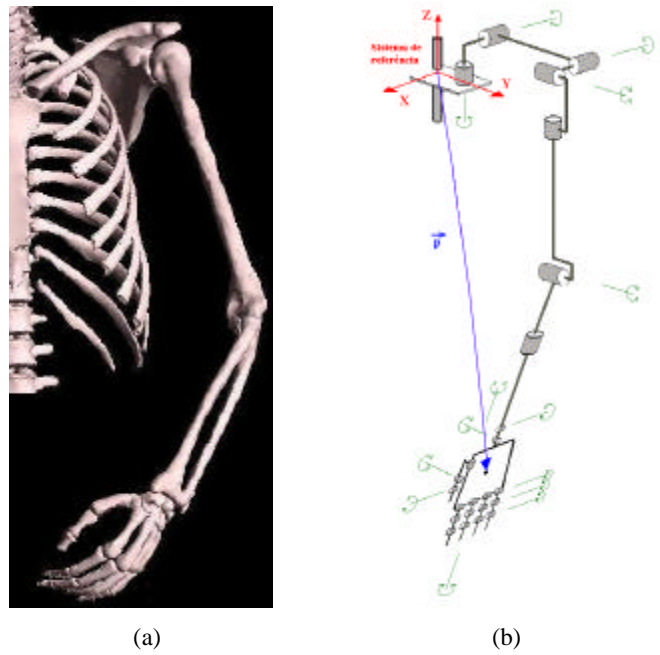


Fig.1- Sistema Articular do Membro Superior

(a)Esqueleto humano referente ao membro superior estudado

(b)Representação das articulações do membro superior, com as respectivas rotações locais

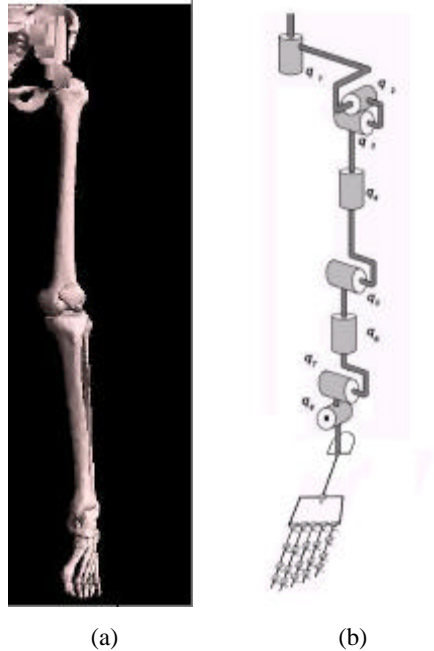


Fig 2. Sistema Articular do Membro Inferior

(a)Esqueleto humano referente ao membro inferior estudado

(b)Representação das articulações do membro inferior, com as respectivas rotações locais

Para uma melhor visualização do problema, vejamos a fig. 3. Ela representa as articulações localizadas no membro superior (mostradas apenas as rotações da escápula até o punho).

A partir de um sistema de referência, chamada de sistema de coordenadas de origem da base, foi relacionada com as demais rotações da cadeia articular (n graus de liberdade). Assim, com a eq. (2.1), obteve-se cada posição da junta, referentes a cada ângulo de rotação da mesma ou sua translação utilizando vetores locais.

$$\underline{O}_{i+1} = \underline{O}_i + MG_i^{i+1} * \underline{L}_i \tag{2.1}$$

Onde:

i = número de pontos necessários para a definição do modelo do corpo

\underline{O}_i = matriz 3x1- origem em relação a um sistema de referência local

\underline{O}_{i+1} = matriz 3x1- próxima origem em relação a um sistema de referência local

MG_i^{i+1} = matriz 3x3- matriz de passagem

$\underline{L}_i = (\underline{O}_{i+1} - \underline{O}_i)$ - Referenciais escolhidos convenientemente de modo a simplificar o sistema

$$MG_i^{i+1} = MG_1^2 * MG_2^3 * \dots * MG_i^{i+1} \tag{2.2}$$

Resultando:

$$MG_i^{i+1} = \begin{bmatrix} n_x & s_x & a_x \\ n_y & s_y & a_y \\ n_z & s_z & a_z \end{bmatrix} \tag{2.3}$$

Rx, Ry, Rz = rotações das juntas nos eixos x, y e z, respectivamente.

$$R_x(qn) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & C_i & -S_i \\ 0 & S_i & C_i \end{bmatrix} \quad R_y(qn) = \begin{bmatrix} C_i & 0 & S_i \\ 0 & 1 & 0 \\ -S_i & 0 & C_i \end{bmatrix} \quad \dots \quad R_z(qn) = \begin{bmatrix} C_i & -S_i & 0 \\ S_i & C_i & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

qn = número de graus de liberdade

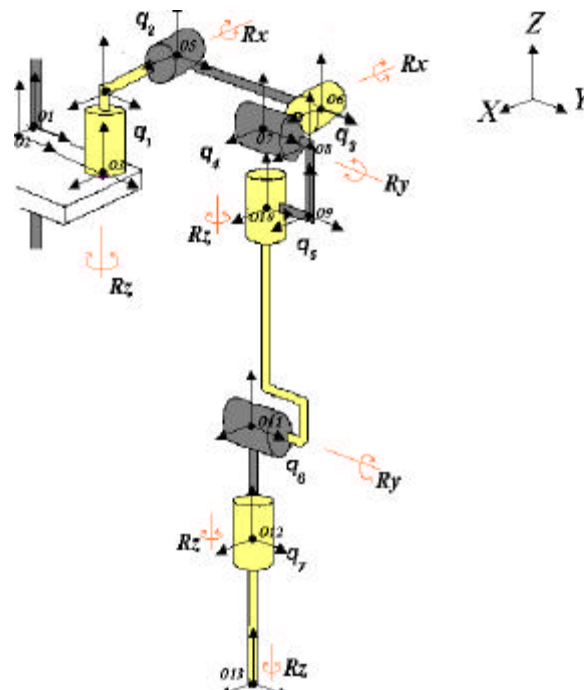


Fig.3 Sistema Articular detalhado do Membro Superior(da escápula ao punho)

No anexo I são apresentadas as matrizes de orientação e vetores de posição de cada um dos segmentos da fig. 3.

2.2 Modelagem Cinemática Inversa

A necessidade da obtenção de referências em coordenadas angulares, correspondentes a tarefas definidas no espaço cartesiano é expressa matematicamente pela inversão do modelo geométrico [SÁ & ROSÁRIO, 1996], isto é:

$$\mathbf{q} = \mathbf{f}^{-1}(\mathbf{x}) \quad (2.4)$$

A função f é não linear e composta de soma de produtos de senos e co-senos das coordenadas generalizadas (translações ou rotações elementares). Por isso, a sua inversão é em geral não trivial. Como f é não linear não se pode garantir a existência e/ou a unicidade de uma função inversa f^{-1} . No caso geral, só se pode determinar o número máximo de prováveis soluções. A mudança de coordenadas consistirá de um funcional que descreverá a correspondência existente entre a cadeia cinemática para um conjunto de variáveis articulares \mathbf{p} e sua posição \mathbf{X} correspondente.

$$\mathbf{X} - \mathbf{X}_0 = F(\mathbf{q} - \mathbf{q}_0) \quad (2.5)$$

onde

$\mathbf{q}_{[n \times 1]} \Rightarrow$ Vetor que representa os graus de liberdade da junta, onde n representa o número de graus de liberdade da prótese.

$\mathbf{X}_{[m \times 1]} \Rightarrow$ Vetor ($m \leq 6$) que representa a posição e a orientação do elemento terminal da prótese no espaço de trabalho em coordenadas cartesianas. No caso de uma representação completa de posição e orientação $m = 6$

$F \Rightarrow$ É uma função não linear obtida da modelagem geométrica da prótese que descreve a posição e a orientação do elemento terminal da mesma.

No caso da transformação inversa de coordenadas, uma determinada posição \mathbf{X} do volume de trabalho da prótese será atingida a partir de uma posição de repouso \mathbf{X}_0 (obtenção dos ângulos Roll, Pitch, Yaw a partir da matriz de orientação espacial).

$$(\mathbf{q} - \mathbf{q}_0) = F^{-1}(\mathbf{X} - \mathbf{X}_0) \quad (2.6)$$

A transformação direta de coordenadas não apresenta dificuldades na sua resolução, o mesmo não acontecendo com a transformação inversa que é muito complexa, não apresentando uma solução única. Para eliminarmos as indeterminações que aparecem no problema inverso, utiliza-se geralmente a matriz jacobiana, onde a mesma poderá ser utilizada para o controle cinemático de mecanismos.

Dada uma configuração inicial \mathbf{q}_0 e \mathbf{X}_0 de um robô, as coordenadas \mathbf{X} do elemento terminal são descritas pela eq. (2.6). Como uma trajetória é definida por pequenos incrementos angulares associados a cada grau de liberdade. Para pequenos deslocamentos $\delta \mathbf{X}$ associados aos deslocamentos das variáveis articulares $\delta \mathbf{q}$, podemos escrever:

$$\delta \mathbf{X}_{m \times 1} = \mathbf{J}_{m \times n} \delta \mathbf{q}_{n \times 1} \quad (2.7)$$

A matriz Jacobiana $\mathbf{J}(\theta)$ será definida como:

$$[J(\mathbf{q})]_{i,j} = [\partial F_i / \partial q_j] \quad (2.8)$$

onde:

i é a i -ésima coordenada generalizada da junta antropomórfica e

j é o j -ésimo grau de liberdade da junta antropomórfica.

que poderá ser construída a partir das relações cinemáticas que descrevem a arquitetura da prótese:

$$\left. \begin{aligned} X_1 &= F_1(q_1, q_2 \dots q_n) \\ X_2 &= F_2(q_1, q_2 \dots q_n) \\ \dots & \dots \dots \dots \\ X_m &= F_m(q_1, q_2 \dots q_n) \end{aligned} \right\} \underline{X} = F_n(q) \tag{2.9}$$

Onde;

$$\Delta x = J(q)\Delta t \tag{2.10}$$

Através de derivadas parciais, a matriz Jacobiana J(q) é definida como:

$$J(\underline{q}) = \begin{bmatrix} \frac{\partial F_1}{\partial q_1} & \frac{\partial F_1}{\partial q_2} & \dots & \dots & \frac{\partial F_1}{\partial q_n} \\ \frac{\partial F_2}{\partial q_1} & \frac{\partial F_2}{\partial q_2} & \dots & \dots & \frac{\partial F_2}{\partial q_n} \\ \dots & \dots & \dots & \dots & \dots \\ \frac{\partial F_m}{\partial q_1} & \frac{\partial F_m}{\partial q_2} & \dots & \dots & \frac{\partial F_m}{\partial q_n} \end{bmatrix} \tag{2.11}$$

Para uma prótese de um membro superior ou inferior, as coordenadas de seu elemento terminal serão descritas através de um vetor posição \underline{X} (x, y, z) e sua orientação definida a partir de três ângulos (ψ, θ, ϕ).

Levando-se em consideração que o sistema de decisão humana normalmente executa o controle de articulações em relação ao espaço de tarefas, o controle de uma prótese antropomórfica no espaço de juntas necessita de uma transformação inversa de coordenadas (F^{-1}). Esta transformação poderá ser realizada a partir da inversão da matriz Jacobiana:

$$\delta \underline{q}_{n \times 1} = J^\#(q)_{m \times n} \delta \underline{X}_{m \times 1} \tag{2.12}$$

onde:

$J^\#(q)$ representa a pseudoinversa da matriz Jacobiana, que deve atender as propriedades definidas a partir da matriz de inversão generalizada de Moore e Penrose.

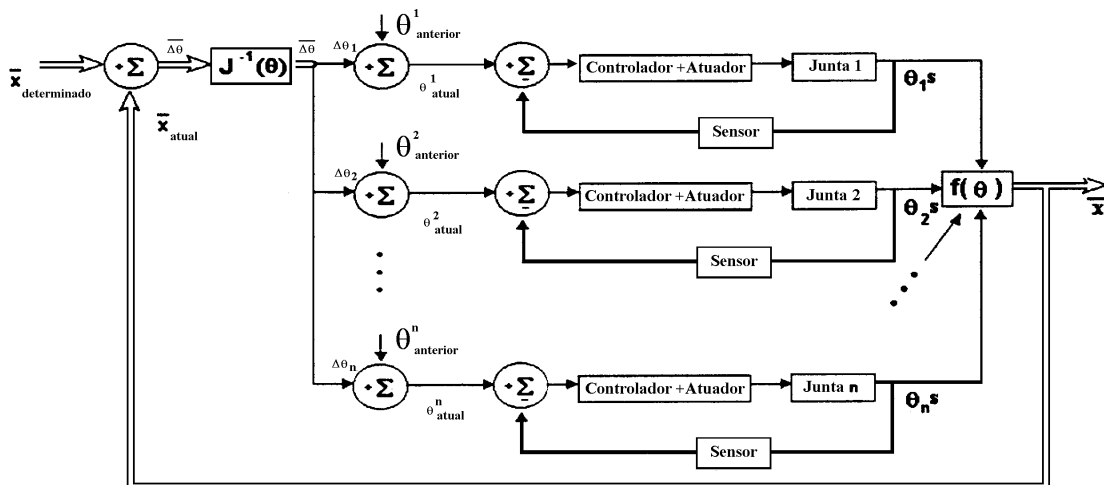


Figura 4- Malha de controle cinemático direto e inverso implementada em controlador

Matematicamente, a relação 2.10 indica a variação do incremento $\delta \underline{q}$ das variáveis articulares para um dado deslocamento $\delta \underline{X}$ do elemento terminal da prótese. Como a posição atual de cada articulação $q_{i \text{ atual}}$ é perfeitamente conhecida (através do sistema sensorial humano), a partir da utilização da eq. (2.10) de modo iterativo e recalculando J(q) a cada passo de iteração, uma trajetória $\underline{X}(t)$ poderá ser realizada num determinado tempo, em função da variação dos ângulos das juntas $q_{i \text{ atual}} + \delta q_i$.

A partir da comparação da posição atual da prótese \underline{X} (valor calculado a partir da posição atual \underline{X}_{atual} obtidas das informações de posições dos sensores de juntas ou da sinapse neural) e sua posição de referência \underline{X}_d , um sinal de erro é amplificado e transformado em termos de coordenadas articulares δq a partir do cálculo de $J^{\#}(q)$. O sinal de erro é integrado e depois utilizado como sinal de entrada para controle das variáveis articulares da prótese.

Finalmente, a obtenção da matriz Jacobiana, utilizada no método recursivo para o cálculo do modelo cinemático inverso, é uma forma multidimensional da derivada e relaciona a velocidade no espaço de juntas à velocidade no espaço cartesiano. A sua solução deverá ser encontrada em tempo real através da utilização de algoritmos numéricos, onde será aproximada por $\Delta \underline{x} = J \cdot \Delta q$. Em muitos casos, a solução de um sistema de equações lineares existe, mesmo quando a inversa da matriz não existe. Este problema e muitos outros podem ser resolvidos através do conceito da pseudoinversa, ou matriz inversa generalizada [PIRES & TENREIRO, 1999], [DUARTE & TENREIRO, 1999].

3. Implementação Computacional

Foi elaborado um programa computacional para simulação de membros e geração de movimentos off-line; (módulos de visualização gráfica e geração de trajetórias dos movimentos) em linguagem ADA, a partir da modelagem cinemática. Este programa computacional é constituído de diversas bibliotecas, e foi estruturado de modo a criar uma plataforma de desenvolvimento em robótica. Assim foi necessário o desenvolvimento de bibliotecas de visualização e geração de movimentos off-line dos diferentes elementos de uma prótese (graus de liberdade); o desenvolvimento de interface amigável com o usuário e a interface de entrada de sensores e saída de sinais de comandos para os atuadores.

Assim sendo, pacotes computacionais desenvolvidos anteriormente para um robô industrial foram adaptados de modo a permitir a visualização e geração automática de trajetórias. Outros módulos complementares, contendo o modelo geométrico correspondente as próteses, como: visualização gráfica e simulação dos movimentos; rotinas para correção de parâmetros de posicionamento na geração de trajetórias off-line e um protocolo de comunicação do micro-computador com a prótese.

Esses programas apresentam como característica principal a fácil utilização pelo usuário e a possibilidade de ser acrescido de novos módulos (bibliotecas).

Finalmente, este programa computacional apresenta um alto grau de estruturação, sendo desenvolvido a partir de módulos correspondentes a bibliotecas básicas (modelagem, simulação e visualização, controle, etc.)

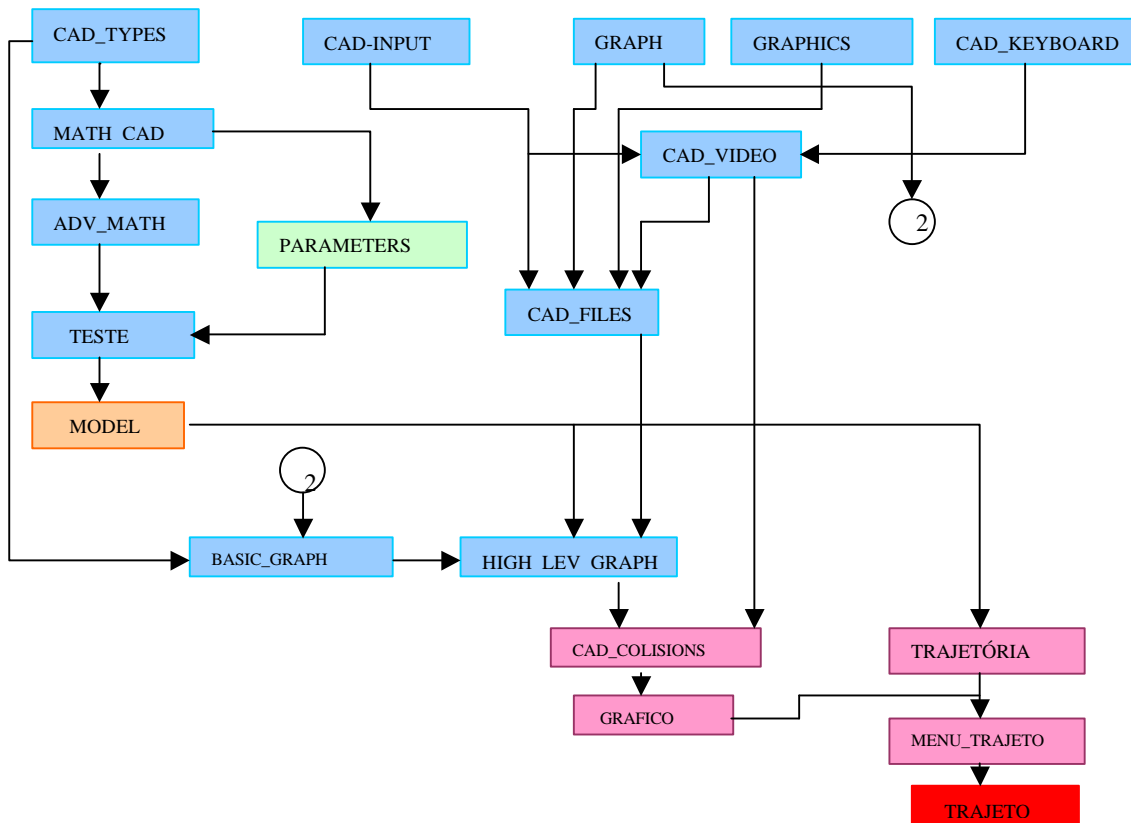


Figura 5: Fluxograma dos pacotes para a geração do módulo Trajetória.

4. Conclusões e Perspectivas Futuras

A partir dos resultados obtidos nesse projeto, os seguintes objetivos foram atingidos: elaboração de programa de simulação, visualização e geração de movimentos de prótese antropomórficas. Este programa permite a validação do modelo correspondente aos elementos constituintes de uma prótese antropomórfica, considerando-se geometria e esforços envolvidos; solidificação dos conhecimentos em biomecânica e modelagem de estruturas robóticas obtidos a partir de desenvolvimento de trabalho aplicação em problemas reais. Com a elaboração de ensaios em bancada, experimentos utilizando atuadores lineares e sensores de posicionamento (encoders incrementais e potenciômetros) e força (extensômetros) para aplicações na prótese membros superiores permitirão a extensão do problema para um protótipo. Ele representará os diferentes graus de liberdade de uma prótese correspondente a um sistema de preensão infantil e um dispositivo para a realização do movimento de adução e abdução dos dedos de uma mão (abertura e fechamento da mão).

Uma bancada de laboratório permitirá a realização de testes (quantitativos) da interface de comando (geração de sinais de referência para a interface de acionamento, calculados em função do modelamento dos diferentes graus de liberdade das próteses implementadas) e sistema de acionamento das próteses e sensores utilizados.

Para o desenvolvimento de um manipulador análogo ao braço/perna humana, torna-se necessário o estudo de acionadores lineares e sistemas de transmissão eficientes. Estes sistemas a transmissão mecânica limita-se a redução ou amplificação da velocidade de rotação solicitada pelos acionadores das articulações. A escolha destes sistemas de acionamento devem ser baseados nos seguintes aspectos: interface com micro-processador (rapidez e ruídos); resposta dinâmica e capacidade de forças elevadas; linearidade na resposta.

5. Agradecimentos

Agradecimentos à Faculdade de Engenharia Mecânica da UNICAMP e à FAPESP pelo apoio financeiro.

6. Referências

- Duarte, F. B. M.; Tenreiro Machado, J. A.. Controlo de Trajectórias de Robots Redundantes Através de Matrizes Pseudoinversas Revista Robótica e Automatização, Portugal, n. 36, p. 12-15, 3º Trimestre, 1999.
- Pires, E. J. Solteiro, Tenreiro Machado, J. A. Geração de Trajectórias para Manipuladores Redundantes Utilizando Algoritmos Genéticos. Revista Robótica e Automatização, Portugal, n. 35, p. 12-16, 2º Trimestre, 1999.
- SÁ, C.E.; Rosário, J.M. Implementation of Numerical Algorithms for the Resolution of the kinematic Inverse Problem of Robots Manipulators. In: ICONE'96 Second International Conference on Non-Linear Dynamics, Chaos, Control and their Applications in Engineering Sciences, 1996, São Pedro-S.P. Proceedings of Second International Conference on Non-Linear Dynamics, Chaos, Control and their Applications in Engineering Sciences, 1996. p.246-252.

Proposal of Prosthesis Movement from the Modeling Forward and Inverse Kinematics

Edna Rodrigues Casemiro

UNICAMP –State University of Campinas - College of Mechanical Engineering PO box 6051, zip code 13083-970, Campinas SP

e-mails: ednarc@fem.unicamp.br

Helder Anibal Hermini

UNICAMP –State University of Campinas - College of Mechanical Engineering PO Box 6051, zip code 13083-970, Campinas SP

e-mail hermini@uol.com.br

João Maurício Rosário

UNICAMP –State University of Campinas - College of Mechanical Engineering PO Box 6051, zip code 13083-970, Campinas SP

e-mails: rosario@fem.unicamp.br

Abstract: For intelligent prosthesis elaboration is necessary the development of cinematic model that shows these models in terms of its dependents degrees of freedom(DOF's). With the technological evolution, many works have established control laws applied to rehabilitation process, aiming to replace part of lost body. This work has the aim of development methodology for modeling, simulation and biomechanical design for application in prosthesis of upper and lower limbs. It was made a computational program (in ADA language) of visualization and automatic movement, from of the mathematical model of limbs and of the physiological linear actuators similar to muscles, for that, using the same methodology for robotic manipulators and for mechanical transmission system. A workbench allows tests of command interface (signals generation to drive interface, it will calculate in function of the different DOF's of the prosthesis that will be implemented) and drive systems and sensors. In this work was given emphasis to upper limb.

Key words: Biomechanics, Prosthesis, Robotic, Control.

Anexo I:

Tabela 1: Matrizes de rotação ou translação referentes aos pontos O_i e seus respectivos deslocamentos.

O_i	Mg	L_{i-1}	O_i	Mg	L_{i-1}
$O_0 = O_1$	-	$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$	O_8	$\begin{bmatrix} C_4 & 0 & S_4 \\ 0 & 1 & 0 \\ -S_4 & 0 & C_4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ Ly \\ 0 \end{bmatrix}$
O_2	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} Lx \\ 0 \\ 0 \end{bmatrix}$	O_9	$\begin{bmatrix} C_4 & 0 & S_4 \\ 0 & 1 & 0 \\ -S_4 & 0 & C_4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ Lz \end{bmatrix}$
O_3	$\begin{bmatrix} C_1 & -S_1 & 0 \\ S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ Ly \\ 0 \end{bmatrix}$	O_{10}	$\begin{bmatrix} C_5 & -S_5 & 0 \\ S_5 & C_5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ Ly \\ 0 \end{bmatrix}$
O_4	$\begin{bmatrix} C_1 & -S_1 & 0 \\ S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ Lz \end{bmatrix}$	O_{11}	$\begin{bmatrix} C_4 & 0 & S_4 \\ 0 & 1 & 0 \\ -S_4 & 0 & C_4 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ Lz \end{bmatrix}$
O_5	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & C_2 & -S_2 \\ 0 & S_2 & C_2 \end{bmatrix}$	$\begin{bmatrix} Lx \\ 0 \\ 0 \end{bmatrix}$	O_{12}	$\begin{bmatrix} C_5 & -S_5 & 0 \\ S_5 & C_5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ Lz \end{bmatrix}$
O_6	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & C_2 & -S_2 \\ 0 & S_2 & C_2 \end{bmatrix}$	$\begin{bmatrix} 0 \\ Ly \\ 0 \end{bmatrix}$	O_{13}	$\begin{bmatrix} C_5 & -S_5 & 0 \\ S_5 & C_5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ Lz \end{bmatrix}$
O_7	$\begin{bmatrix} C_1 & -S_1 & 0 \\ S_1 & C_1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$	$\begin{bmatrix} Lx \\ 0 \\ 0 \end{bmatrix}$			

Laser Interferometry Technique Applied to the Measurement of Flextensional Transducers Displacements

Gilder Nader

Department of Mechatronics and Mechanical Systems Engineering
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231, 05590-900 - São Paulo - SP, Brazil
e-mail: gnader@usp.br

Alessandro Barbosa Shirahige

Department of Mechatronics and Mechanical Systems Engineering
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231, 05590-900 - São Paulo - SP, Brazil
e-mail: alessandro.shirahige@poli.usp.br

Emílio Carlos Nelli Silva

Department of Mechatronics and Mechanical Systems Engineering
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231, 05590-900 - São Paulo - SP, Brazil
e-mail: ecnsilva@usp.br

Julio Cezar Adamowski

Department of Mechatronics and Mechanical Systems Engineering
Escola Politécnica da Universidade de São Paulo
Av. Prof. Mello Moraes, 2231, 05590-900 - São Paulo - SP, Brazil
e-mail: jcadamow@usp.br

Abstract

The application of laser interferometry techniques to measure vibration is growing in industry field. In this work, it is implemented a Michelson-type interferometer to measure displacements of piezoelectric transducers. It consists of phase measuring of two optical waves interference, deriving from a same light source. This technique allows the measurement of displacements in nanometric scale. The main purpose of these measurements is to verify the displacement amplification introduced by new designs of flextensional piezoelectric transducers, obtained by optimization techniques and simulated by finite element using the software ANSYS. A flextensional transducer consists of a piezoceramic connected to a flexible structure, which amplifies and changes the direction of generated piezoceramic displacement. To verify the amplification, both displacements of the single piezoceramic and the complete transducer are measured through laser interferometry and compared. The measurements are made considering harmonic and transient excitations. All experimental results are compared with finite element simulations and the predicted amplification rates provided by these transducers are verified.

Keywords: laser interferometry, flextensional transducers, topology optimization, piezoelectric transducers, finite element.

1. Introduction

Several interferometric techniques are applied to measure displacements (Monchalin, 1986, Scruby, 1990). The laser interferometry can measure mechanical displacements over length scales of nanometers to meters (Royer, 1986, Royer, 1994) and is able to reach an accuracy of picometers (Lawall, 2000). The principle of laser interferometry for length metrology is to measure displacements from a phase shift (Creath, 1988) of an optical wave due to the movement of a sample. The displacement of the sample is obtained in terms of the known laser wavelength (λ). A phase difference of π corresponds to a displacement of $\lambda/4$. This is a sensitive and efficient method to measure small displacements and deformations.

The main purpose of this work is to apply laser interferometry to measure the displacement amplification of flextensional transducers. A flextensional transducer consists of a piezoceramic connected to a flexible structure, which amplifies and changes the direction of generated piezoceramic displacement. New models of optimized flextensional piezoelectric transducers have been designed by using the topology optimization technique. Topology optimization is a method that combines finite element method (FEM) with optimization algorithm, and it was implemented to design the flexible structure of these transducers to improve their performance (Silva, 1999, Silva, 2000).

Some prototypes of these flextensional piezoelectric transducers were manufactured. Prototypes designed by topology optimization (Silva, 2000) are illustrated in Fig. (1). These designs have mechanical parts with complex forms that can provide high stiffness (and thus blocking force) or high flexibility (and thus output displacement). These prototypes consist essentially of two elements, a PZT5A piezoceramic bonded with epoxy to an aluminum flexible structure manufactured by using a wire EDM (Electrical Discharge Machining) machine. To characterize the resonance

frequencies, the electrical impedance curves were obtained by using an Impedance Gain/Phase Analyzer HP4194A. The piezoceramic displacement in y-direction, see Fig. (1) and (2), is compared with flextensional transducer displacements to verify the amplification rates.

The experimental displacement measurements were done through laser interferometry, performed by exciting the piezoelectric transducers with harmonic and transient excitations (Royer, 1994). These results were compared with the finite element simulation results (Gindre, 1988) obtained by using ANSYS software, a commercial FEM package.

2. Transducers assembly

Piezoceramics applied to build transducer prototypes were rectangular PZT5A with dimensions shown in Tab. (1). These ceramics were polarized in the y-direction and the electrode surfaces are normal to this direction. The designed actuators composed by an aluminum flexible structure and PZT5A are shown in Fig. (1). They are called f1a1025 and f2b0830 and their dimensions are illustrated in the same figure.

The PZT5A piezoceramic was bonded to the aluminum flexible structure by using a thin film of epoxy resin, for f1b0830 and rosin for f1a1025. Special care was taken to electrically insulate the electrodes of piezoceramic from the flexible structure in these transducers, to avoid a short circuit, because of the closed shape of their endcap, which connects top and bottom electrodes.

Table 1. Piezoceramic dimensions.

Length (L)	30.0 mm	x-direction
Thickness (T)	5.0 mm	y-direction
Width(W)	13.0 mm	z-direction

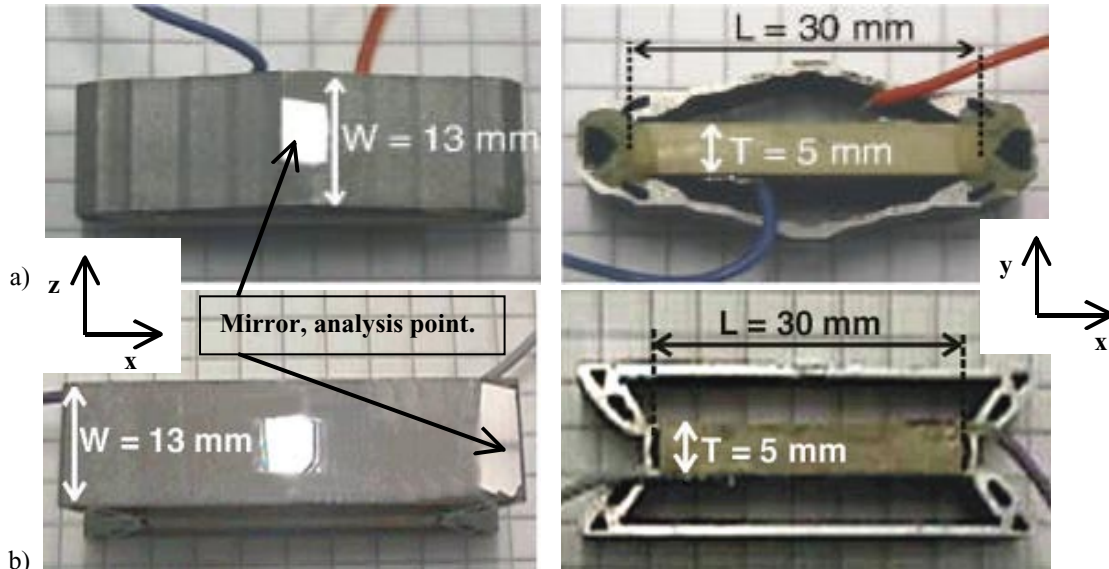


Figure 1. Flextensional piezoelectric transducers: (a) f1a1025 and (b) f2b0830.

3. Finite element method

The flextensional transducers were modeled by finite element using the software ANSYS. Two-dimensional finite element models were built based on plane strain assumption, due to the rectangular shape of the piezoceramic and the extrusion-like geometry of these transducer prototypes. These models are shown in Fig. (2). The model was built considering the symmetry of the transducer with electrodes in both surfaces normal to the y-direction.

The material properties of piezoelectric ceramic PZT5A and aluminum are shown in Tab. (2). The damping is not given in handbooks, because it depends on several intrinsic manufacturing parameters of each material. In Table (2) are shown the damping assumed on presented simulations.

4. Laser interferometry

4.1. Michelson interferometer – phase measurement principle

Flextensional transducers displacements are measured by using laser interferometry. The interferometric technique applied is based on a Michelson-type interferometer (Djelouah, 1992), which measures a one-point displacement of the

transducer surface. The points that are measured in each actuator are described in Fig. (1). These actuators were designed to generate large displacement amplification in these points.

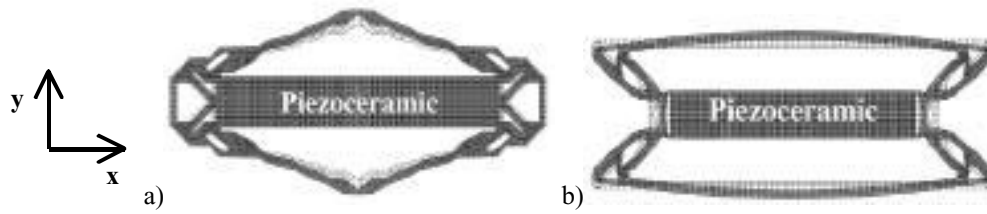


Figure 2. Displacement motion at 10 kHz of flextensional transducers obtained by FEM (a) f1a1025 and (b) f2b0830.

Table 2. Material Properties of PZT5A and aluminum used in FEM.

PZT5A	
elastic constants	(10 ¹⁰ N.m ⁻²)
C ₁₁ ^E	12.00
C ₁₂ ^E	7.51
C ₁₃ ^E	7.52
C ₃₃ ^E	11.10
C ₄₄ ^E	2.10
C ₆₆ ^E	2.24
piezoelectric constants	(C.m ⁻²)
e ₃₁	-5.35
e ₃₃	15.8
e ₅₁	12.3
ε: dielectric constants	(10 ⁻⁹ C ² .N ⁻¹ .m ⁻²)
ε ₁₁	15.31
ε ₃₃	13.38
Density	7800 kg.m ⁻³
Damp	5 x 10 ⁻⁸
Aluminum	
Young's modulus	71 x 10 ⁹ N.m ⁻²
Density	2800 kg.m ⁻³
Damp	4 x 10 ⁻⁷
Poisson's ratio	0.33

The interferometric system shown in Fig. (3) uses a He-Ne laser (λ = 632.8 nm). A beam-splitter (BS) is a semitransparent mirror, which splits the light laser into two beams (see Fig. (3)). These two beams are directed along orthogonal paths, usually called arms of the interferometer, where they strike two mirrors, the reference mirror (R) and the sample mirror (S). These beams return to the beam splitter where they interfere. The interference pattern is illustrated in Fig. (4). A convergent lens (L) is applied to expand the laser beam and enlarge the interference pattern on amplified photo-diode (PD-A). Due to the PD-A saturation, an intensity attenuate (A) is applied before it.

The phase measurement technique used consists of measuring the intensity variation of an interference pattern (Nader, 2001), as illustrated in Fig. (4). The intensity resulting from the interference is (Born, 1980):

$$I_L = I_R + I_S + 2\sqrt{I_R I_S} \cos\phi \tag{1}$$

where: I_L = the light intensity on the photo-diode (PD-A); I_R = reference beam intensity; I_S = sample beam intensity; and

$$\phi = \frac{4\pi\Delta d}{\lambda} \tag{2}$$

where λ is the laser wavelength and the phase variation (φ) is proportional to the difference of optical path Δd:

$$\Delta d = L_R - L_S \tag{3}$$

where L_R is the length between BS and R, and L_S is the length between BS and S, (see Fig. 4).

The sample displacement is equal to the relation $L_R - L_S$, if the reference length path (L_R) does not change. The displacement of the sample surface is determined by measuring the phase difference (ϕ), which is related to the intensity variation of interference pattern as described by Eq. (1). The maximum intensity detected by the photo-diode (PD-A) will be equal to:

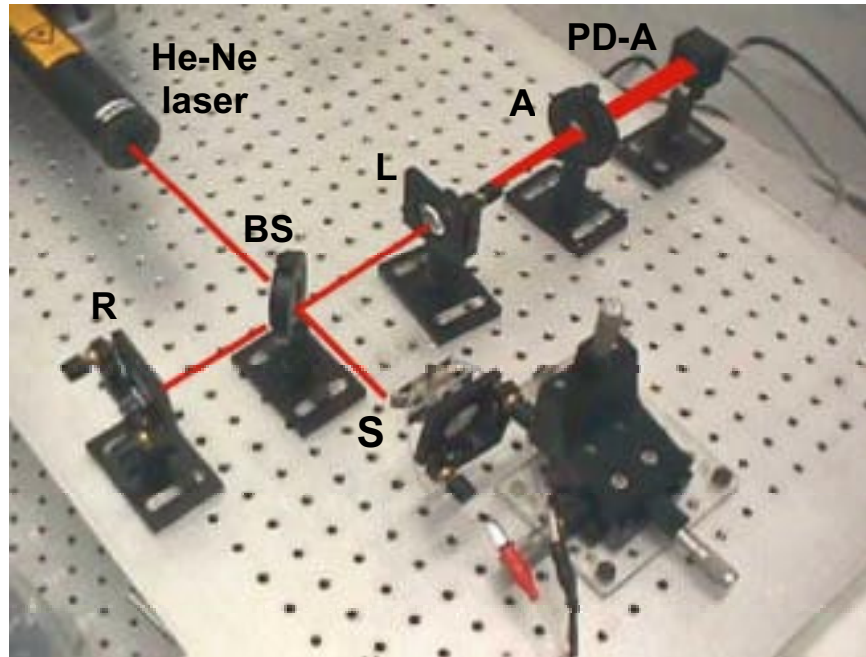


Figure 3. Michelson interferometer.

$$\left. \begin{aligned} |\phi| &= 0, 2\pi, 4\pi \\ \Delta d = L_R - L_S &= n\lambda \end{aligned} \right\} |n| = 0, \frac{1}{2}, 1, \dots, \tag{4}$$

and the minimum intensity detected will be equal to:

$$\left. \begin{aligned} |\phi| &= \pi, 3\pi, 5\pi \\ \Delta d = L_R - L_S &= n\lambda \end{aligned} \right\} |n| = \frac{1}{4}, \frac{3}{4}, \dots, \tag{5}$$

as shown in Fig. (4).

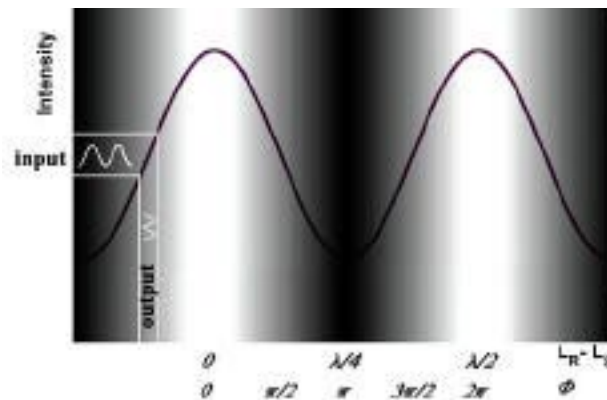


Figure 4. Light interference pattern in a Michelson interferometer.

Notice from Fig. (4) that the difference between a crest and a trough is equal to a phase difference of π . From Eq. (2), this phase difference occurs when the optical path difference change by $\lambda/4$. However, in this method, it is necessary to stay in the linear part of the light interference pattern, as illustrated in Fig. (4), therefore, considering the He-Ne laser, the maximum displacement that can be measured is equal to 158.2 nm (including the nonlinear part of the curve).

4.2. Displacement measurement technique

During interferometric tests, flextensional transducers are held by a device fixed by three points perpendicular to the displacement to be measured. This device gives all freedom to flextensional transducers to oscillate in the positive and the negative y-direction, see Fig. (5). To minimize the transducer energy losses, measurements were made in air.

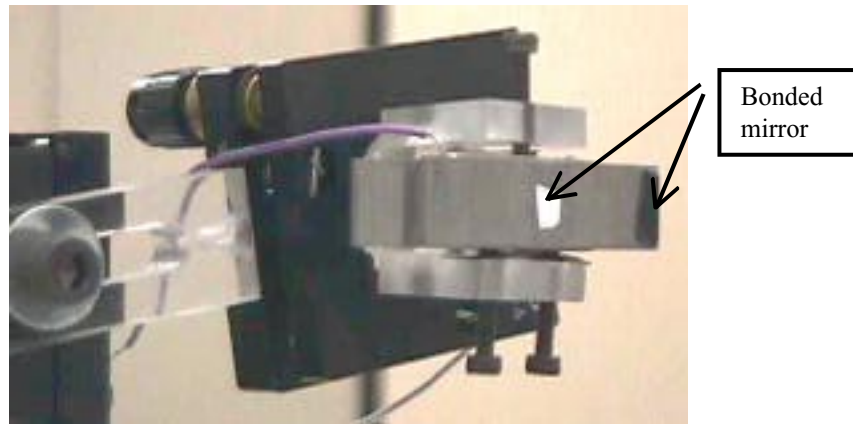


Figure 5. Flextensional transducer holder.

The flextensional transducer surface could not become reflexive through an optical polish due to their complex shape. The solution was to manufacture 200 μm thickness mirrors by aluminum vaporization in glass plates and bond them to the flextensional transducer surface with epoxy (see Figs. (1) and (5)).

The experimental setup is shown in Fig. (6). This interferometric system was built to measure displacements in the range of nanometers. Therefore, transducers were excited with a quite low voltage, in a range between 3 and 30 V. A laser probe pointed to the mirror on the flextensional piezoelectric transducers (see Figs. (1)) performs the displacement measurements from interferometry.

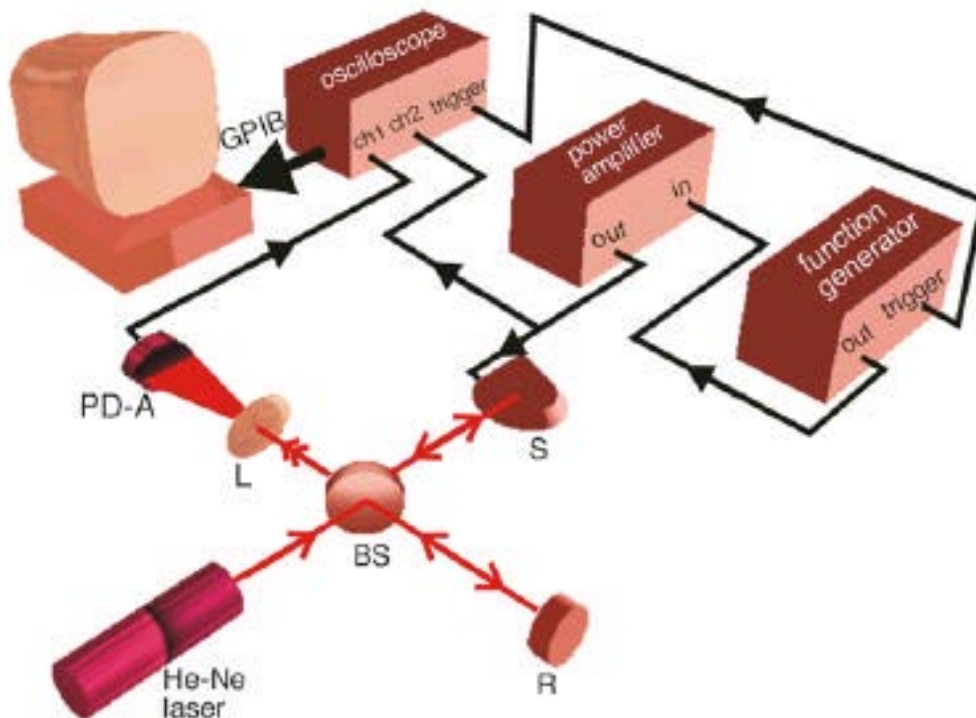


Figure 6. Experimental setup of the Michelson interferometer.

The He-Ne laser source ($\lambda = 632.8 \text{ nm}$) is vertically polarized, with 10 mW output nominal power. The electronic apparatus is composed by a function generator, which sends an harmonic sinusoidal signal or one sine cycle (burst) to a power amplifier, exciting the piezoelectric transducer (S). A digital oscilloscope monitors the output of the power amplifier and acquires the response signal from an amplified photo-diode (PD-A) with a variable gain, 35 ns of rise/fall time and 0.16 mW of saturation.

The transducer displacement is obtained by measuring the light intensity variation (due to a phase shift) detected by photo-diode, which exports the data to a microcomputer through GPIB communication device.

The calibration is obtained in this system by making the piezoelectric sample to vibrate with an amplitude larger than $\lambda/4$. The fringe pattern is not very stable, therefore the calibration is performed frequently. Care must be taken to keep minimum vibrations on the assembling.

5. Experimental and simulated results

5.1. Electrical impedance measurements

The PZT5A piezoceramics used to build the prototypes were initially analyzed by the Impedance Gain/Phase Analyzer (HP4194A) to determine their resonance frequencies and electrical impedance curves. These curves were compared with the curves obtained after bonding the flexible aluminum structure to the piezoceramic (to assemble the flextensional transducer), to verify if the bonding was successful. After bonding, the transducer set should present different resonance frequencies. Therefore, it can be guaranteed that the flexible structure is well coupled to the piezoceramic.

These electrical impedance curves acquired by HP4194A, between 10 - 40 kHz, were compared with simulated data calculated by ANSYS in the same range, considering damping. Experimental and simulated electrical impedance curves for flextensional transducers, f1a1025 and f2b0830, are illustrated in Fig. (7). Table (3) shows experimental and simulated resonance frequencies calculated by modal analysis, not considering damping on materials.

From Fig. (7) and Tab. (3), it can be observed that new resonance frequencies appear due to the coupling between piezoceramic and the flexible structure. The first resonance frequency of PZT5A shown in Tab. (3) is larger than the resonance frequencies of these flextensional transducers, illustrated in Fig. (7).

Experimental and simulated resonance frequencies are shown in Tab. (3) and Fig. (7), showing a good agreement among experimental and simulated results. However, electrical impedance curves show a difference larger than 20 %. This occurs due to inaccurate values of dielectric constants used in FEM.

Experimental and simulated resonance frequencies show a good agreement, and simulated resonance frequencies are usually slightly larger than experimental resonance frequencies due to the inaccuracy values of piezoceramic stiffness properties used in FEM.

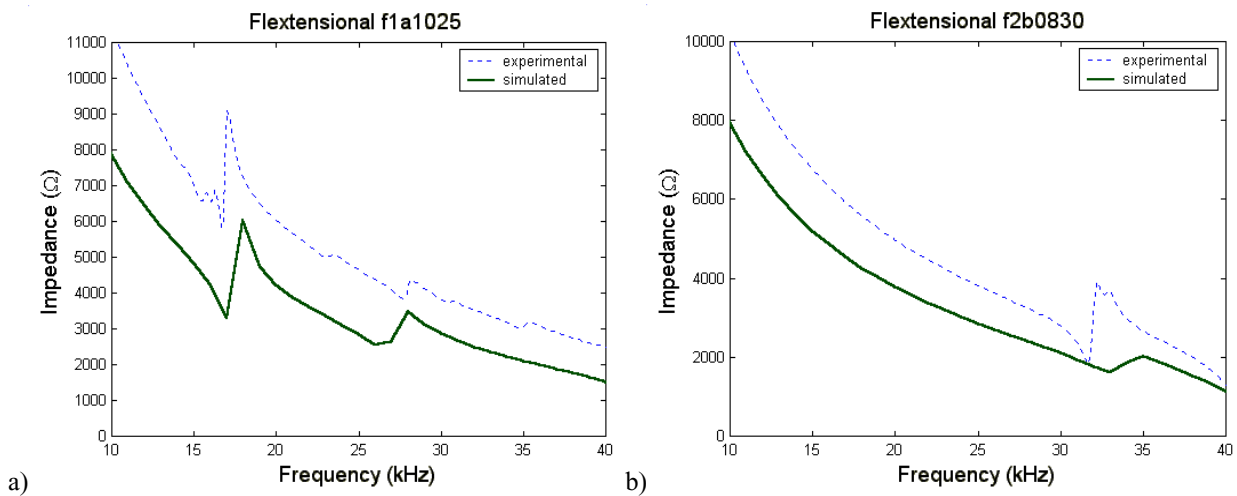


Figure 7. Impedance curve of a) f1a1025 and b) f2b0830.

Table 3. Resonance frequencies of transducers.

Transducer	Experimental (kHz)	Simulated (kHz)
PZT5A	46.60	49.42
f1a1025	16.50	17.17
	27.70	26.92
f2b0830	31.65	33.77

5.2. Displacement measurements

Displacement measurements were done considering harmonic sinusoidal excitation (continuous mode) and a burst signal excitation (transient mode). The objective is to compare simulated and experimental displacement results in a

frequency range (displacement frequency response) to determine the displacement amplification, using harmonic excitation, and response of the flextensional transducers to the transient excitation, comparing the fast fourier transforms (FFT).

5.2.1. Harmonic excitation

Flextensional actuators were designed for quasi-static applications. Therefore, the displacement frequency responses of flextensional transducers were obtained by exciting each transducer, with continuous sinusoidal signals, in a low kHz range of frequency until its first resonance frequency, as illustrated in Fig. (8).

Simulated and experimental normalized displacement frequency response curves of flextensional actuators are illustrated in Fig. (8). Measured and simulated displacement amplification rates are shown in Fig. (9).

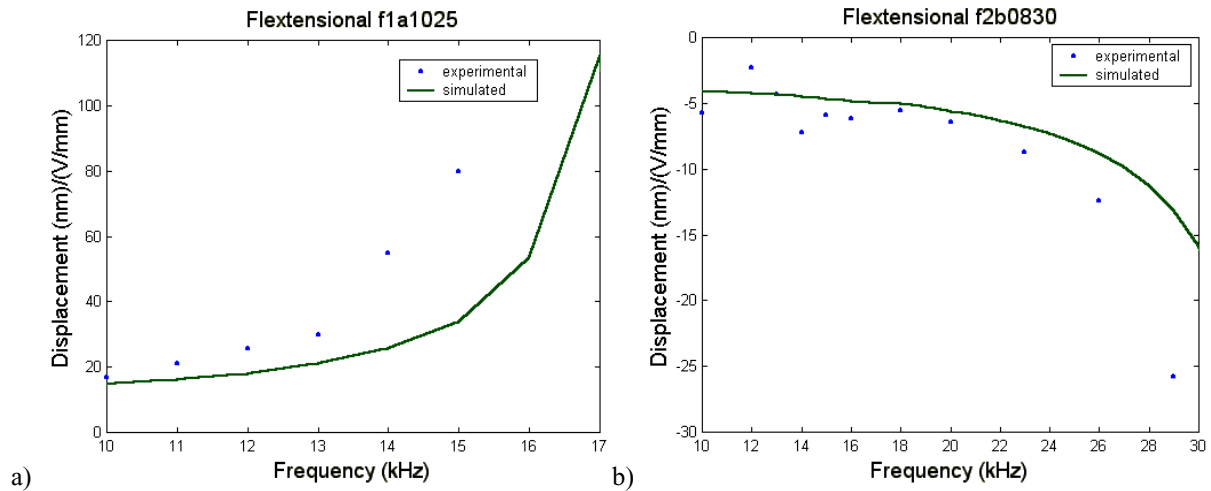


Figure 8. Displacement frequency response of a) f1a1025 and b) f2b0830.

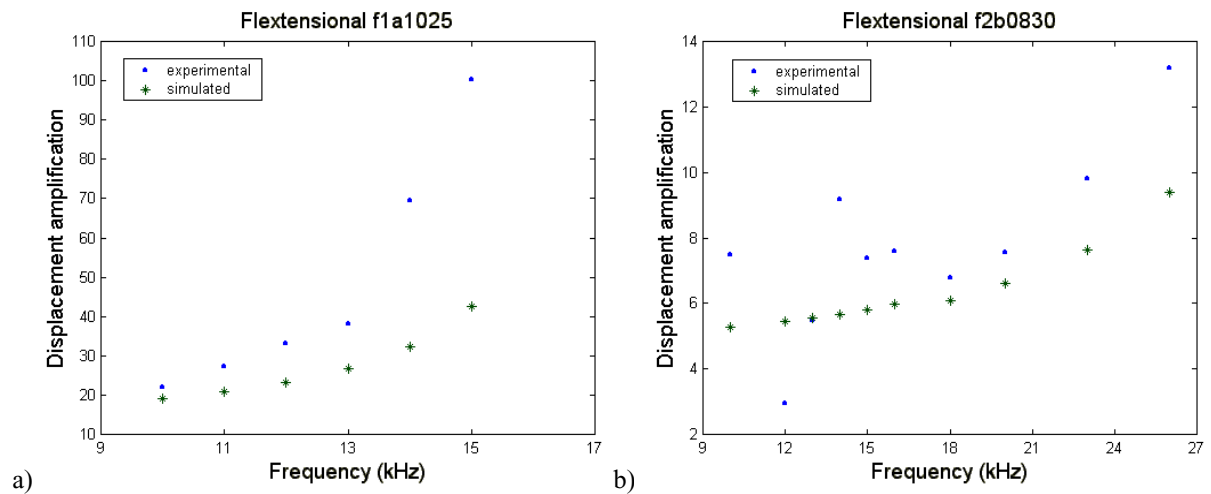


Figure 9. Displacement amplification rate of a) f1a1025 and b) f2b0830.

These results are very time consuming to obtain because the interferometer has to be calibrated for each frequency measurement. The amplification is obtained for each frequency by dividing the displacement generated by the actuator (in a specific point – see Fig.(1)) and the displacement generated by the piezoceramic considering the same applied voltage. The amplification rate obtained for design f1a1025 is comparable to the amplification rates obtained for the latest flextensional actuator designs (Dogan, 1997). The displacement presented are normalized, therefore they can be compared with other actuators with different piezoceramic thickness and applied voltage.

From Figs. (8) and (9), it is verified that the simulated and experimental results match very well for first points, however, as it gets close to the resonance the simulated and experimental values start diverging, however the measured displacements are always larger than the simulated values. This may occur due to the epoxy bond. After transducers were bonded, a decrease of the mounted transducer overall stiffness was realized due to the epoxy or the rosin. It is very hard to model the epoxy bond in finite element method, due to possible micro-air bubbles inside the bond, inaccurate measurement of the bond thickness, and its non-linear behavior. To verify the epoxy influence on flextensional transducer behavior, several simulations were performed considering different thickness of epoxy or rosin layer. From

these simulations, it was observed a large variation on results, therefore, proving epoxy influence. In addition, it is very important to know accurately the value of the physical properties of transducer elements.

5.2.2. Transient excitation

Laser interferometry was also applied to measure the transient response of the flextensional transducers. Results obtained are compared with simulated transient response by using FEM.

Figure (10) shows simulated and experimental transient displacement responses of these actuators considering the excitation pulse illustrated in the lower left corner of these figures. Notice from these figures that experimental and simulated transient responses are very similar, demonstrating the potential application of the laser interferometry to measure transient behavior of these devices.

FFTs of displacement frequency response are shown in Fig. (11). From these figures it is noticed that results are also in perfect agreement in the frequency domain. The fact that experimental and simulated results match very well suggests that measurements and simulations done are reliable.

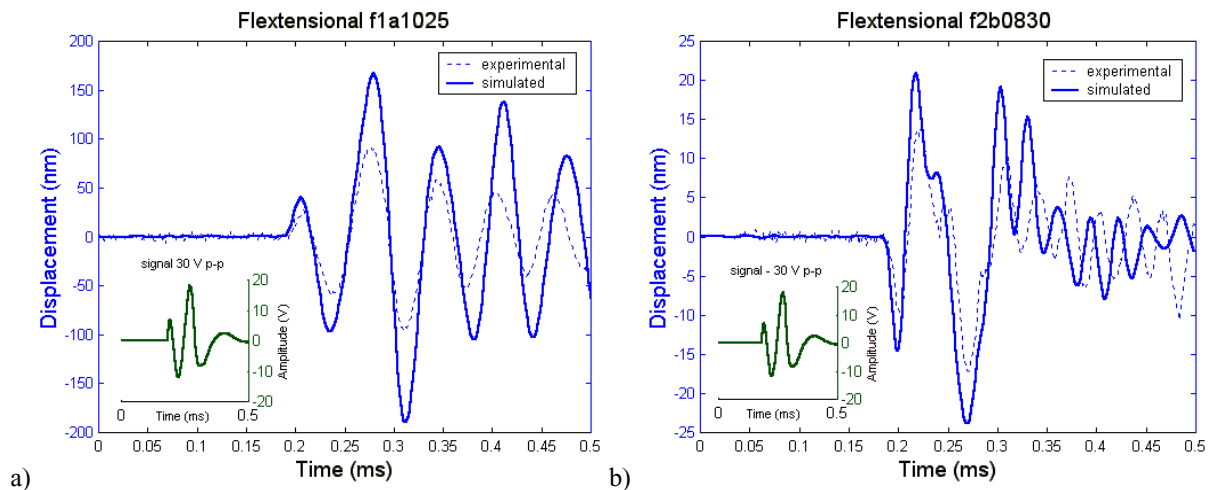


Figure 10. – Displacement transient response of a) f1a1025 and b) f2b0830.

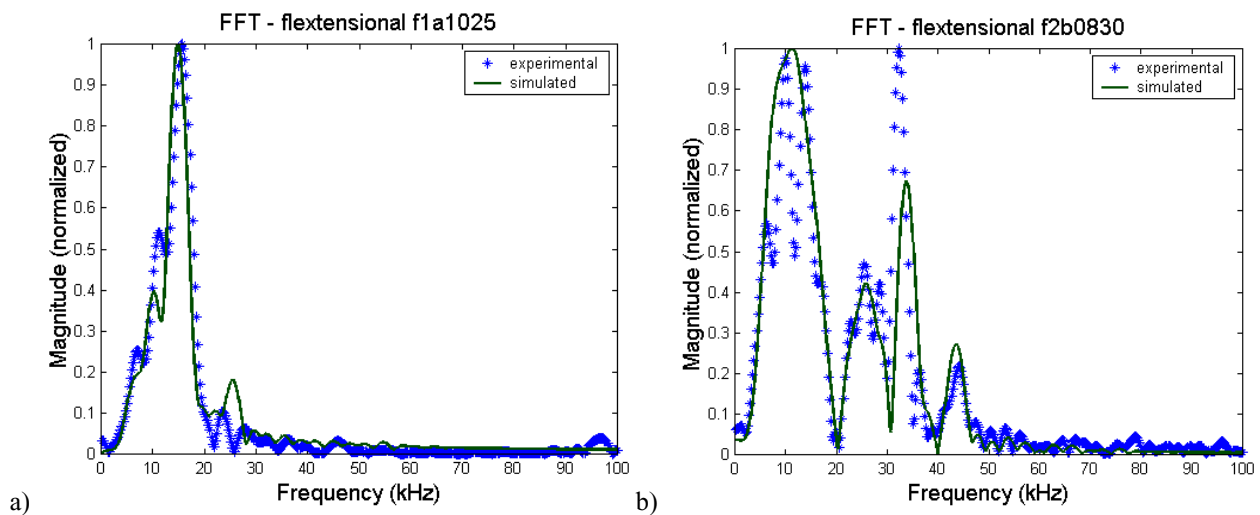


Figure 11. – FFTs of displacement transient response of a) f1a1025 and b) f2b0830.

6. Conclusion

A Michelson interferometer was successfully implemented to conduct the experimental verification of amplification rate of novel types of flextensional actuators. Harmonic and transient measurements were conducted to demonstrate the potential application of the laser interferometer to measure piezoelectric actuators displacements. The interferometric measurement technique proved to be very sensitive and accurate to measure displacements in the nanometer range. The main difficulty in the displacement measurements was to obtain a good reflection target on the transducer surface. This was solved with a very small mirror bonded to the transducer surface. Thanks for the laser

interferometer, the novel actuator designs could be analyzed and from the results obtained it is proved that the actuators designed by topology optimization perform as predicted and desired in the initial design specifications.

It was verified that a perfect match between simulated and experimental results depends on accurate values of material properties, which usually are not available specifically for piezoceramics.

As a future work, experimental techniques must be developed to measure piezoceramic properties, and the interferometer will be adapted to measure large static displacements. Regarding the epoxy influence, new FEM simulations will be conducted to try to model it, however, this influence could be eliminated by finding other technique to assemble the actuator.

7. Acknowledgments

The authors thank Dr. O. Horikawa, for helpful valuable comments and suggestions and following sponsor agencies: FAPESP (Fundação de Amparo à Pesquisa do Estado de São Paulo) and CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

8. References

- Born, M and Wolf, E., 1980, "Principles of Optics", sixth edition, Cambridge University Press, Cambridge, UK.
- Creath, K., 1988, "Phase-Measurement Interferometry Techniques", Progress in Optics XXVI, Elsevier Science Publishers.
- Djelouah, H. and Baboux, J.C., 1992, "Transient ultrasonic field radiated by a circular transducer in a solid medium", Journal of the Acoustic Society of America, Vol. 92, No. 5, pp. 2932-2941.
- Dogan, A., Uchino K. and Newnham, R.E., 1997, "Composite piezoelectric transducer with truncated conical endcaps "Cymbal"", IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, Vol. 44, pp. 597-605.
- Gindre, M., Urbach, W., Coursant, R.H. and Fink, M., 1988, "Mechanical displacement induced in a piezoelectric structure: Experimental measurement by laser interferometry and simulation by a finite element method", Journal of the Acoustic Society of America, Vol. 84, No. 1, pp. 11-19.
- Lawall, J. and Kessler, E., 2000, "Michelson interferometry with 10 pm accuracy", Review of Scientific Instruments, Vol. 71, No. 7, pp. 2669-2676.
- Monchalain, J.P., 1986, "Optical detection of ultrasound", IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, Vol. 33, No. 5, pp.485-499.
- Nader, G., Silva, E.C.N. and Adamowski, J.C., 2001, "Experimental verification of flexensional transducers designed by using topology optimization method", Proceeding of SPIE: Smart Structures & Materials – NDE for Health Monitoring. accept.
- Royer, D. and Dieulesaint, E., 1986, "Optical detection of sub-angstrom transient mechanical displacements", IEEE Ultrasonic Symposium Proceeding, pp. 527-530.
- Royer, D. and Casula, O., 1994, "Quantitative imaging of transient acoustic fields by optical heterodyne interferometry", IEEE Ultrasonic Symposium Proceeding, pp. 1153 – 1162.
- Royer, D., 1997, "Génération et détection optiques d'ondes élastiques", Techniques de L'Ingénieur", Traité Électronique, E4 415, France, pp. 1 – 16.
- Scruby, C.B. and Drain, L.E., 1990, "Laser Ultrasonics, Techniques and Applications", Ed. Adam Hilger – EUA.
- Silva, E.C.N. and Nikuchi, N., 1999, "Design of piezoelectric transducers using topology optimization", Journal of Smart Materials and Structures, Vol. 8, pp. 350-364.
- Silva, E.C.N., Nishiwaki, S. and Kichichi, N., 2000, "Topology optimization design of flexensional actuators", IEEE Transactions on Ultrasonics, Ferroelectrics and Frequency Control, Vol. 47, No. 3, pp. 657-671

A DESIGN METHODOLOGY FOR THE ESTIMATION OF POSITIONING DEVIATION FOR GANTRY MANIPULATORS

Claudio Violante Ferreira (D.Sc. Student)

Vitor Ferreira Romano

Universidade Federal do Rio de Janeiro E.E. / COPPE
Departamento de Engenharia Mecânica – Laboratório de Robótica
Centro de Tecnologia, bloco G, sala 204
P. Box - 68503 - CEP 21945-970 Rio de Janeiro – RJ, Brasil
Tel.: 562-7764 / 562 – 8404
email: violante@serv.com.ufrj.br, violante@labrob.coppe.ufrj.br, romano@serv.com.ufrj.br

Abstract: This work presents recent results concerning a design methodology used to estimate the positioning deviation for a gantry (Cartesian) manipulator, related mainly to structural elastic deformation of components during operational conditions. The case-study manipulator is classified as gantry type and its basic dimensions are 1,53m x 0,97m x 1,38m. The dimensions used for the calculation of effective workspace due to end-effector path displacement are: 1m x 0,5m x 0,5m. The manipulator is composed by four basic modules defined as module X, module Y, module Z and terminal arm, where is connected the end-effector. Each module controlled axis performs a linear-parabolic positioning movement. The planning path algorithm has the maximum velocity and the total distance as input parameters for a given task. The acceleration and deceleration times are the same. Denavit-Hartenberg parameterization method is used in the manipulator kinematics model. The gantry manipulator can be modeled as four rigid bodies with three degrees-of-freedom in translational movements, connected as an open kinematics chain. Dynamic analysis were performed considering inertial parameters specification such as component mass, inertia and center of gravity position of each module. These parameters are essential for a correct manipulator dynamic modelling, due to multiple possibilities of motion and manipulation of objects with different masses. The dynamic analysis consists of a mathematical modelling of the static and dynamic interactions among the modules. The computation of the structural deformations uses the finite element method (FEM).

Keywords: Robotics, design of manipulators, manipulator modelling.

1. Introduction

The main catalogue specifications of any industrial robot, provided and guaranteed by its manufacturer for normal operational conditions, are related to its workspace and other performance parameters such as payload, maximum velocity and acceleration, positioning accuracy and repeatability (Rivin, 1988; ABB, 1996; DEA, 1995 and Warneck et al, 1985). The usual control techniques applied to these manipulators are mainly based on kinematics variables.

In the case that the value of payload, velocity or acceleration is above the specification, dynamic effects will cause positioning deviations on accuracy and repeatability. These deviations can be mainly associated to structural deformations. To avoid these problems, the most adopted solution by robot manufacturers is to increase the rigidity of the manipulator, resulting in a high ratio between manipulator mass and working load.

The design of less rigid manipulators in relation to payload requires real time monitoring of positioning deviations for control system compensation, in order to obtain the operational conditions previously specified (Yong et al, 1985). Another limiting aspect is related to its natural frequency behavior, whose numerical value is reduced when the structural rigidity is decreased.

2. Manipulator Characteristics

The design parameters used in this article are based on a manipulator prototype designed at LabRob COPPE-PEM/UFRJ. This case-study manipulator Fig.(1) is classified as a gantry type, with three degrees of freedom (not considering the end-effector) and four structural modules: Module X, Module Y, Module Z and Arm, besides the End-effector that can be coupled to the Arm.

The manipulator estimated mass is 42.5 Kg, not taking in account the end-effector and payload masses. The basic dimensions are 1530mm x 972mm x 1379mm and the dimensions used for the calculation of effective workspace due to end-effector path displacement are: 1000mm x 500mm x 500mm. The actuation system is composed by three actuators, in this case three step motors, which are controlled by a microstep control card. The transmission elements consists of ball screws, linear guides and bearings.

3. Manipulator Kinematics Analysis

The kinematics analysis of the manipulator, which has a fixed base (Module X) and 4 links (Module Y, Module Z, Arm and End-effector) with three degrees-of-freedom in translational movements, connected as an open kinematics chain Fig.(2), is referred in the Joints Space Coordinates.

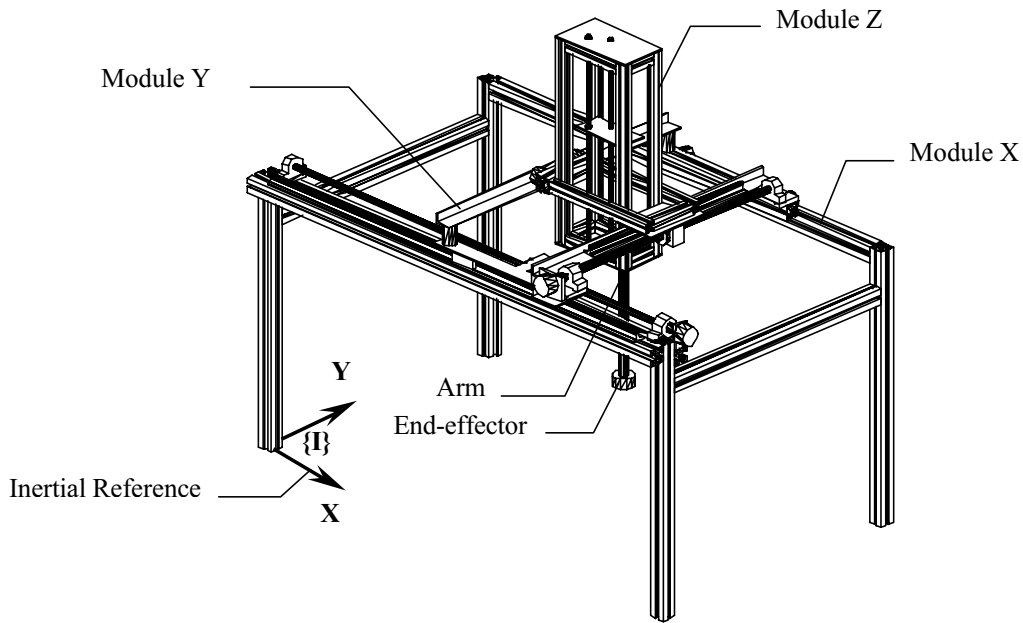


Figure 1. Case-study Manipulator.

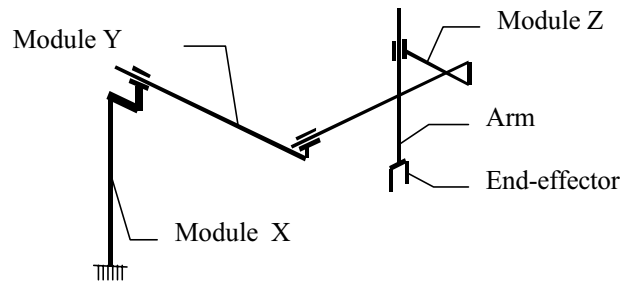


Figure 2. Manipulator simplified model.

Denavit-Hartenberg (D-H) parameterization method (Sciavicco and Siciliano, 1995) is used in the manipulator kinematics model, Fig.(3). In order to optimize D-H parameters the Module Z was subdivided in two links, Link 2 and Link 3. Besides this, since the End-effector is fixed to the Arm, both of them can be represented as one link (Link 4).

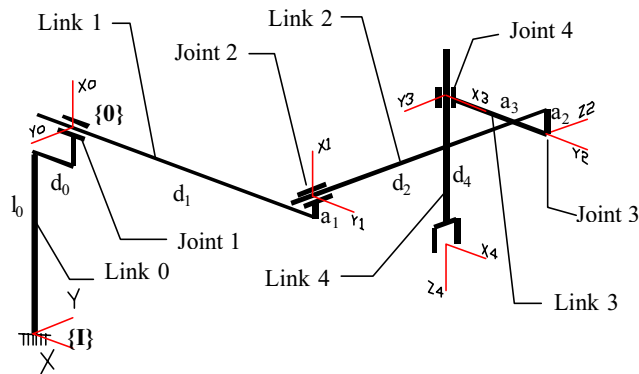


Figure 3. Simplified model with inertial and local reference frames and D-H parameters.

Local homogeneous transformation matrices 0T_1 , 0T_2 , 2T_3 and 3T_4 are calculated based on D-H parameters. From the Fig. (3) is obtained the homogeneous transformation matrix 1T_0 , relating the local {0} to the global {I} inertial reference frames. The global homogeneous transformation matrix 1T_4 , which describes the end-effector position and orientation in the global inertial reference {I} is given by:

$${}^1T_4 = {}^1T_0 \cdot {}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot {}^3T_4 = \begin{bmatrix} 1 & 0 & 0 & -a_3 + d_1 + d_0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & -1 & l_0 + a_1 - a_2 - d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where:

- a_i = D-H geometrical parameter (body i length), mm;
- d_i = D-H geometrical parameter (joint i variable), mm;
- ${}^i T_{i-1}$ =homogeneous transformation matrix, relating the reference frames $\{i-1\}$ and $\{i\}$.

The linear velocity extremity can be obtained as a function of the homogeneous transformation matrices. The equation of the linear velocity of the end-effector for the gantry manipulator studied here is (Fu et al, 1987):

$${}^1V_E = \left[\left(Q_0 \cdot {}^1T_0 \cdot \dot{d}_0 \cdot {}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot {}^3T_4 \right) + \dots + \left({}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot Q_4 \cdot {}^3T_4 \cdot \dot{d}_4 \right) \right] \cdot {}^4P_E \quad (2)$$

where:

- Q_i = link matrix for prismatic joints, (dimensionless);
- 4P_E = end-effector position in the reference frame $\{4\}$, (m).

The solution of equation (2) yields:

$${}^1V_E = (\dot{d}_1 \quad \dot{d}_2 \quad -\dot{d}_4)^T \quad (3)$$

where the Jacobian obtained from

$$[\dot{d}] = J^{-1} \cdot [V] \quad (4)$$

has the following elements:

$$J = \begin{bmatrix} 0 & 0 & -1 \\ 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (5)$$

This Jacobian matrix is invertible. Therefore it has no singular points and the manipulator can move into any point of its workspace.

4. Dynamic Analysis

The manipulator dynamics relates the kinematics of the links composing the kinematics chain, with the loads applied at the joints, through the actuators. The generic dynamic equation for a manipulator with n degrees of freedom and joint vector $q(t)$ can be expressed as (Sciavicco and Siciliano, 1995):

$$\tau(t) - \Pi(t) = M(q(t)) \cdot \ddot{q}(t) + C(q(t), \dot{q}(t)) \cdot \dot{q}(t) + \tau_V \cdot \dot{q}(t) + \tau_C(q(t), \dot{q}(t)) + G(q(t)) \quad (6)$$

where:

- τ = torque/force vector applied in the joints, (N m) or (N);
- Π = external torque/force applied in the ith link, (N m) or (N);
- M = manipulator inertia matrix, (Kg m²) or (Kg m);
- C = Coriolis and centrifuge term, (Kg m²/s) or (Kg m/s);
- τ_V = viscosity friction term, (Kg m²/s) or (Kg m/s);
- τ_C = Coulomb friction term, (N m) or (N);
- G = gravity term, (N m) or (N).

The Newton-Euler equations are used for the calculation of static and dynamic load interactions among the modules of the case-study manipulator. These interactions occur at the joints and transmissions elements (ball screws, linear guides and bearings). The analysis is referred in the Actuators Space Coordinates and friction is taken into consideration (Ferreira, 2000).

4.1. Motor Z

The forces considered in the calculation of the motor Z total torque are represented in Fig. (4). This torque is calculated by Eq. (7) through Eq.(10), where the signal (-) is used for movements in Z positive direction and (+) otherwise.

$$T_{Tz} = M_Z \cdot \ddot{\theta}_Z + \tau_{CZ} + G_Z \quad (7)$$

$$M_Z = \frac{2 \cdot \pi \cdot \eta_Z \cdot d_Z}{(2 \cdot \pi \cdot \eta_Z \cdot d_Z \mp \mu_Z \cdot p_Z)} \cdot (I_{\text{Rotorz}} + I_{\text{Eqz}}) \quad (8)$$

$$\tau_{CZ} = \frac{2 \cdot \pi \cdot \eta_Z \cdot d_Z \cdot M_{\text{RLZ}} + p_Z \cdot d_Z \cdot F_{\text{aZ}}}{(2 \cdot \pi \cdot \eta_Z \cdot d_Z \mp \mu_Z \cdot p_Z)} \quad (9)$$

$$G_Z = \frac{p_Z \cdot d_Z}{(2 \cdot \pi \cdot \eta_Z \cdot d_Z \mp \mu_Z \cdot p_Z)} \cdot m_{\text{A/EL}} \cdot g \quad (10)$$

where:

η_Z = mechanical efficiency coefficient, (dimensionless);

d_Z = distance between the guide Z and screw Z, (mm);

μ_Z = frictional coefficient, (dimensionless);

p_Z = screw Z lead, (mm);

I_{Rotorz} = mass moment of inertia of rotor z, (Kg m^2);

I_{Eqz} = mass moment of inertia of moving components in respect to screw Z, (Kg m^2);

M_{RLZ} = bearing friction torque, (N m);

$m_{\text{A/EL}}$ = mass of End-effector, Arm and Load, (Kg);

g = gravity acceleration, (m/s^2).

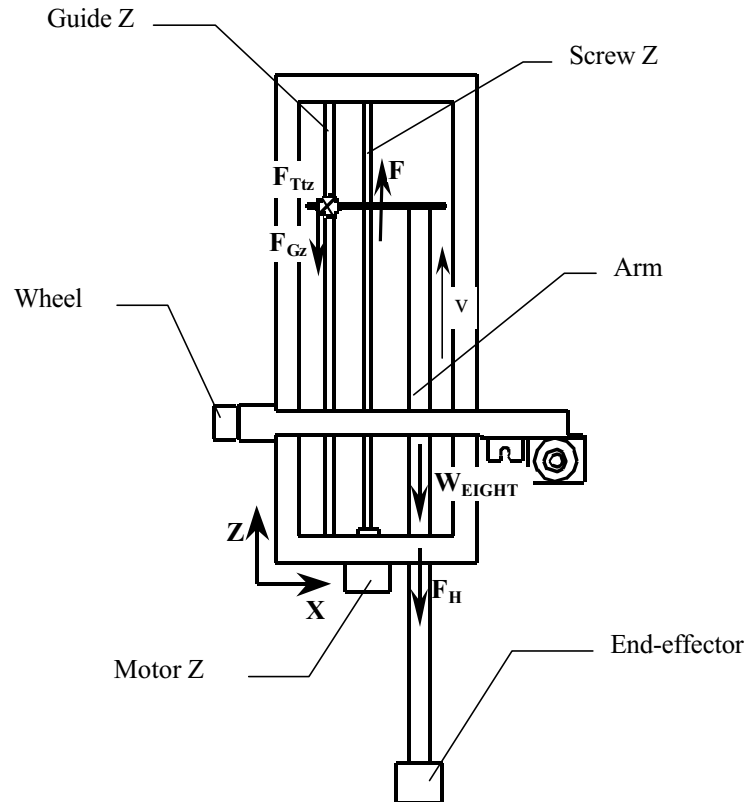


Figure 4. Representation of the Module Z with the loads considered in the calculation of the motor Z total torque.

4.2. Motor Y

In Fig. (5) are represented all the forces considered for the calculation of the motor Y total torque given by Eq. (11) through Eq. (14), where the signal (+) is used when the movement is in Y positive direction and (-) otherwise.

$$T_{\text{Ty}} = M_Y \cdot \ddot{\theta}_Y + \tau_{\text{CY}} + G_Y \quad (11)$$

$$M_Y = \frac{d_y}{(2 \cdot \pi \cdot \eta_y \pm \mu_{gy} \cdot p_y)} \cdot \left[\frac{p_y^2}{\pi \cdot r_y} \cdot I_{Roly} + 2 \cdot \pi \cdot \eta_y \cdot (I_{Rotory} + I_{Eqy}) \right] \quad (12)$$

$$\tau_{CY} = \frac{p_y \cdot d_y \cdot F_{ay} + 2 \cdot \pi \cdot \eta_y \cdot d_y \cdot M_{RLy}}{(2 \cdot \pi \cdot \eta_y \pm \mu_{gy} \cdot p_y)} \quad (13)$$

$$G_Y = \left[\frac{(\mu_{ry} + \mu_{ey} \cdot \frac{r_{ey}}{r_y}) \cdot x_1 + \mu_{gy} \cdot x_2}{(x_1 + x_2)} \right] \cdot \frac{p_y \cdot d_y}{(2 \cdot \pi \cdot \eta_y \pm \mu_{gy} \cdot p_y)} \cdot m_Z \cdot g \quad (14)$$

where:

d_y = distance between the guide Y and screw Y, (mm);

η_y = mechanical efficiency coefficient, (dimensionless);

p_y = screw Z lead, (mm);

I_{Roly} = mass moment of inertia of the wheel, (Kg m²);

I_{Rotory} = mass moment of inertia of rotor Y, (Kg m²);

I_{Eqy} = mass moment of inertia of moving components in respect to screw Y, (Kg m²);

M_{RLy} = bearing friction torque, (N m);

μ_y = frictional coefficient, (dimensionless);

m_Z = module Z mass, (Kg).

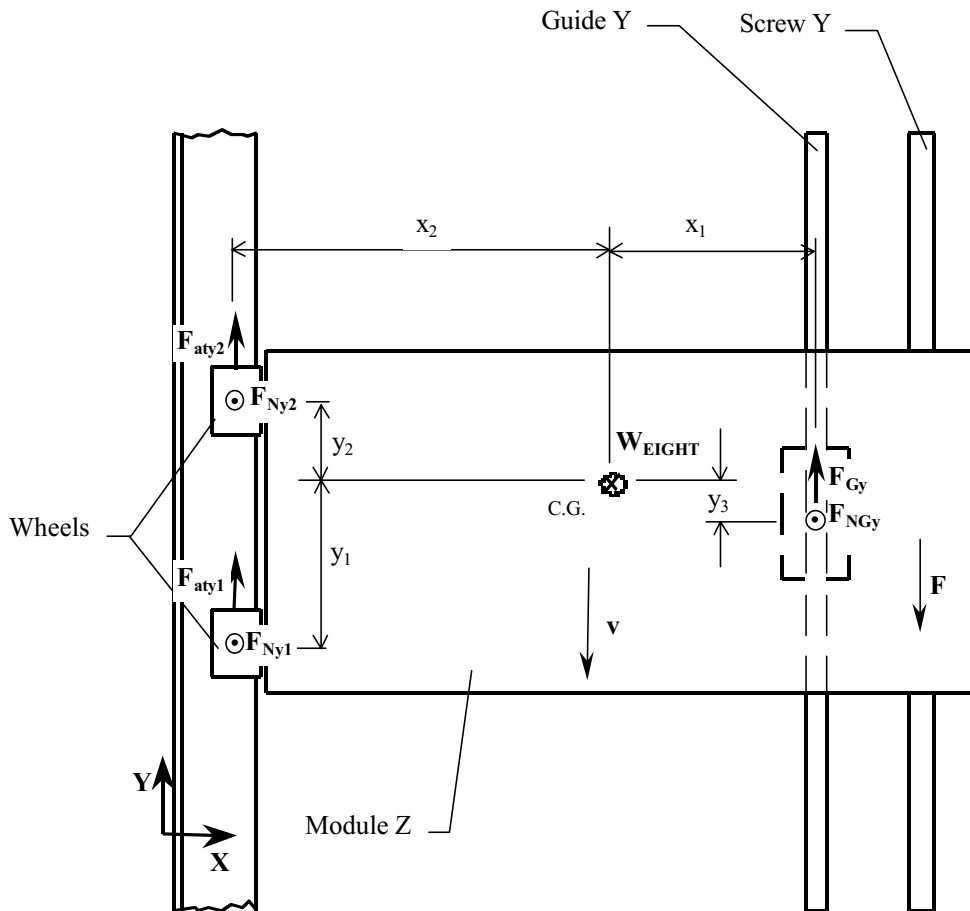


Figure 5. Representation of the Module Y with the loads considered in the calculation of the motor Y total torque.

4.3. Motor X

The forces taken into consideration to calculate the motor X total torque are represented in Fig. (6). This torque is calculated by Eq. (16) through Eq.(18), where one has to use the signal (-) for movement in X positive direction and (+) otherwise.

$$T_{Tx} = M_X \cdot \ddot{\theta}_X + \tau_{CX} + G_X \tag{15}$$

$$M_X = \left(\frac{d_X}{2 \cdot \pi \cdot \eta_X \mp \mu_{gx} \cdot p_X} \right) \cdot \left[\frac{p_X^2}{\pi \cdot r_X} \cdot I_{Rox} + 2 \cdot \pi \cdot \eta_X \cdot (I_{Rotorx} + I_{Eqx}) \right] \tag{16}$$

$$\tau_{CX} = \frac{2 \cdot p_X \cdot d_X \cdot F_{ax} + 2 \cdot \pi \cdot \eta_X \cdot d_X \cdot M_{RLx}}{(2 \cdot \pi \cdot \eta_X \mp \mu_{gx} \cdot p_X)} \tag{17}$$

$$G_X = \frac{\left(\mu_{rx} + \mu_{ex} \cdot \frac{r_{ex}}{r_X} \right) \cdot y'_1 + \mu_{gx} \cdot y'_2}{(y'_1 + y'_2)} \cdot \left(\frac{p_X \cdot d_X}{2 \cdot \pi \cdot \eta_X \mp \mu_{gx} \cdot p_X} \right) \cdot m_{ZY} \cdot g \tag{18}$$

where:

- d_Y = distance between the guide Y and screw Y, (mm);
- η_Y = mechanical efficiency coefficient, (dimensionless);
- p_Y = screw Y lead, (mm);
- I_{Roly} = mass moment of inertia of the wheel, (Kg m²);
- I_{Rotory} = mass moment of inertia of rotor Y, (Kg m²);
- I_{Eqy} = mass moment of inertia of moving components in respect to screw Y, (Kg m²);
- M_{RLy} = bearing friction torque, (N m);
- μ_Y = frictional coefficient, (dimensionless);
- m_{ZY} = mass of modules Z and Y, (Kg).

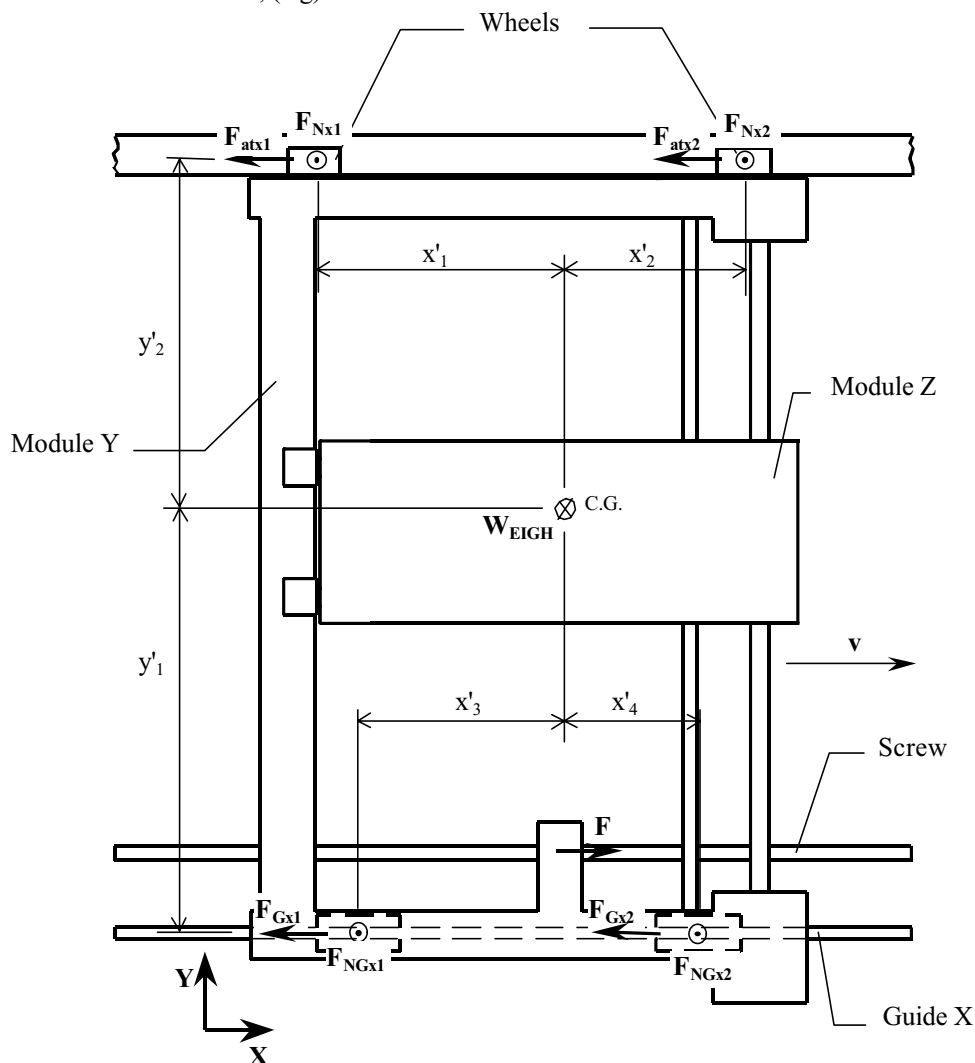


Figure 6. Representation of the Module X with the loads utilized to calculate the motor X total torque.

Finite element model

A commercial FEM program, ANSYS[®] 5.4, is used for the calculation of the structural deformations. The element “3-D Elastic Beam” (ANSYS-PC, 1991) is used to model the manipulator structural components, since it can support tension, compression, torsion and flexion. This element also has six degrees-of-freedom in each node (three translations and three rotations referred to X, Y and Z axis).

Figure (7) presents the model of the manipulator used to calculate the structural deformations, with all the loads applied at the contact surfaces, the acceleration gravity vector (node 1) and the restrictions to the six degrees of freedom at the manipulator base (nodes 1, 10, 11 and 20). Figure (8) represents the deformed condition of the manipulator calculated by the FEM program ANSYS[®].

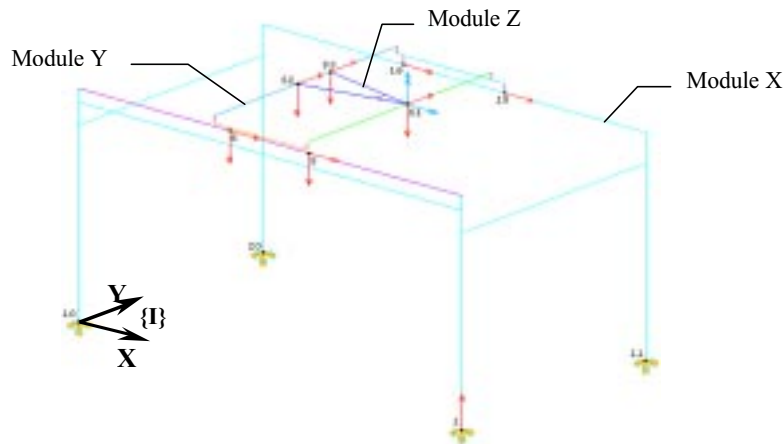


Figure 7. FEM model of the manipulator used for the calculation of the structural deformations in the ANSYS[®] 5.4 .

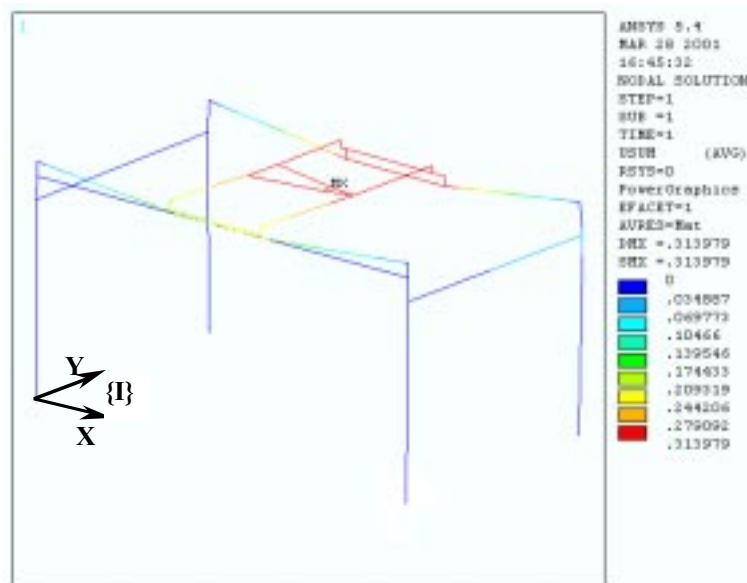


Figure 8. Deformed condition of the manipulator calculated by the FEM program ANSYS[®], for the position X=0,5m , Y=0,25m and Z=0,25m.

End-effector Positioning Deviation

The values of the positioning deviations of the end-effector extremity are essentially dependent on the elastic deformations caused in the manipulator during the realization of a given task. In the FEM modeling the module Z, Fig.(9), was defined by the elements connected by the nodes 51, 52 and 53; and nodes 62 and 63, which are defined to locate point 60. This node (Node 60) is the point where the arm containing the end-effector is assumed to orthogonally intercept the plane π , see Fig.(10). The planes π and π_D represent, respectively, the undeformed and deformed conditions of the manipulator. In order to obtain the end-effector extremity coordinates is used the D-H parameter d4 (defined in the kinematics analysis) which represents the arm extended length. The End-effector position deviation D is given by Eq. (19), where x, y and z are the desired End-effector extremity coordinates in the reference systems{I}.

$$D = \begin{cases} x - \frac{x_{62D} + x_{63D}}{2} + d_4 \cdot n_{Dx} \\ y - \frac{y_{62D} + y_{63D}}{2} + d_4 \cdot n_{Dy} \\ z - \frac{z_{62D} + z_{63D}}{2} + d_4 \cdot n_{Dz} \end{cases} \quad (19)$$

where:

- x_{ijD} = position of Node ijD in the X direction, (mm); see Fig. (10).
- y_{ijD} = position of Node ijD in the Y direction, (mm);
- z_{ijD} = position of Node ijD in the Z direction, (mm);
- n_{Dk} = component k of the plane π_D normal vector.

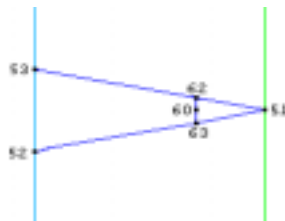


Figure 9. XY plane view of the module Z modeling in FEM.

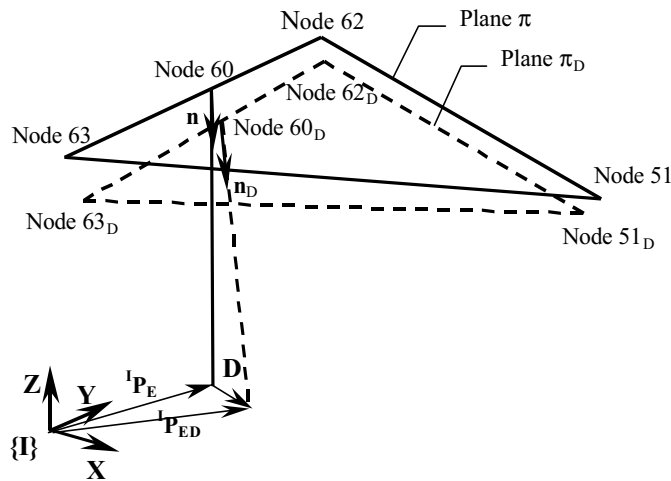


Figure 10. End-effector representation for the deformed (Plane π_D) and undeformed (Plane π) conditions of the manipulator.

Results

The manipulator performance is related to the specification of its tasks. A task can be interpreted as a temporal relation between kinematics and inertial parameters. After the definition of the manipulator movement, the kinematics parameters are obtained for the specified path and the values of the mechanical loads applied at the elements are calculated. Then these data are used in the FEM modeling to calculate the deformation of the manipulator, which results are used to determine the end-effector extremity position deviation.

In this article, the task is defined as the simultaneous movement of the manipulator through the directions X (0 to 1m), Y(0 to 0,5m) and Z(0,5 to 0m), supporting loads (payload and end-effector masses) of 0, 1, 5 and 10 Kg. Each step motor is assumed to perform a linear-parabolic movement, due to the fact that this kind of motion is the most employed for tasks such as manipulation of objects. The planning path algorithm has the maximum velocity and the total distance as input parameters for a given task. The acceleration and deceleration times are the same. For the case study manipulator the maximum velocities are: 0,41m/s at direction X and 0,20 m/s at directions Y and Z.

To study the end-effector extremity position deviation of the manipulator when submitted to a specified task, the analysis is done according to the following criteria: five points at the accelerated interval, seven points at the constant velocity interval and five points at the decelerated interval, as presented in Fig. (11).

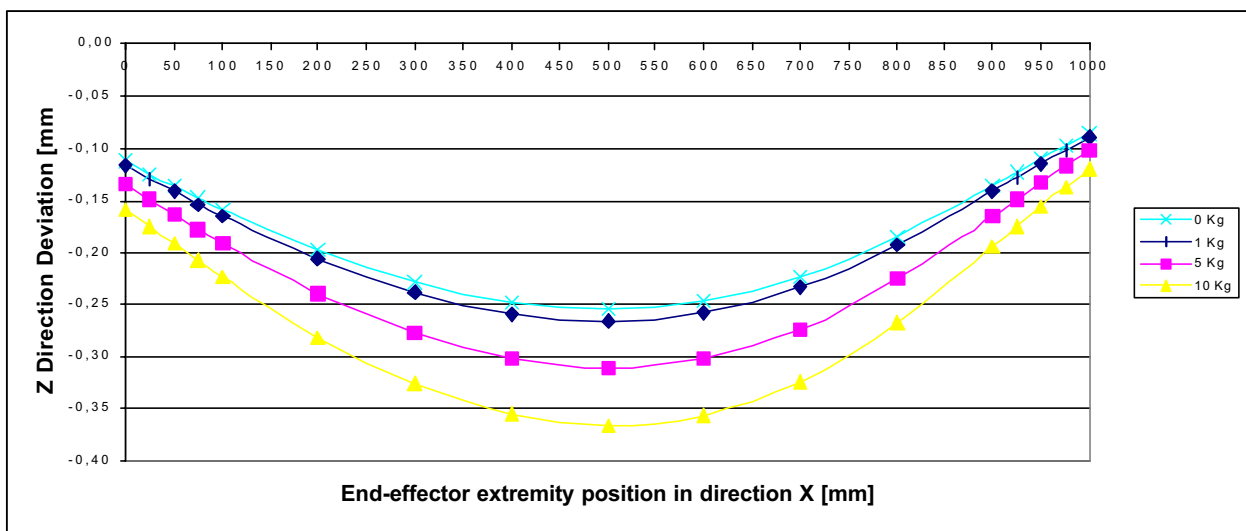
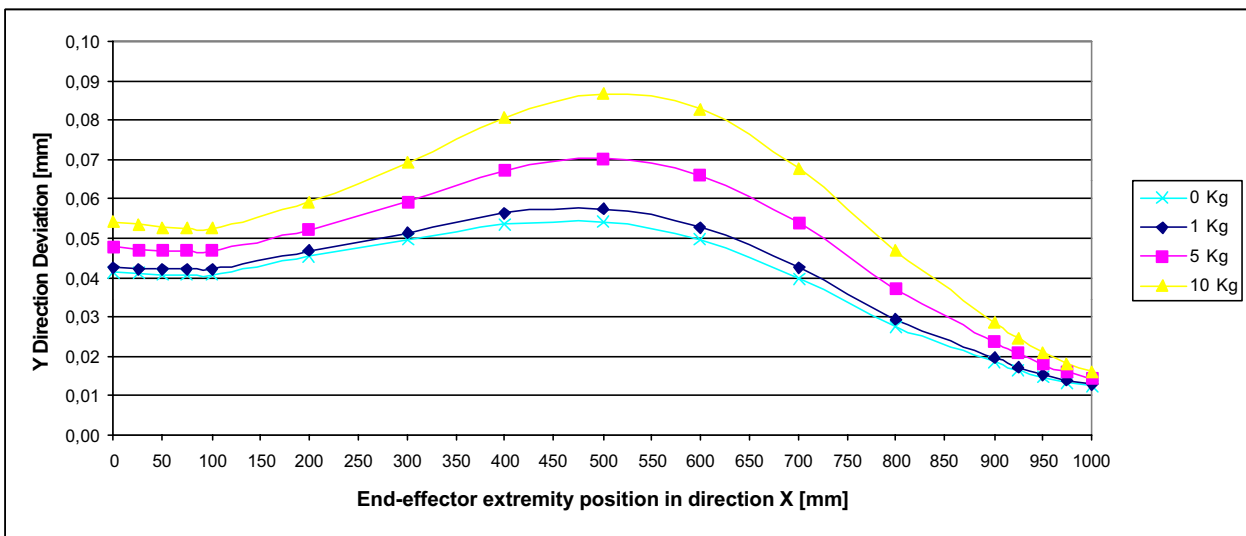
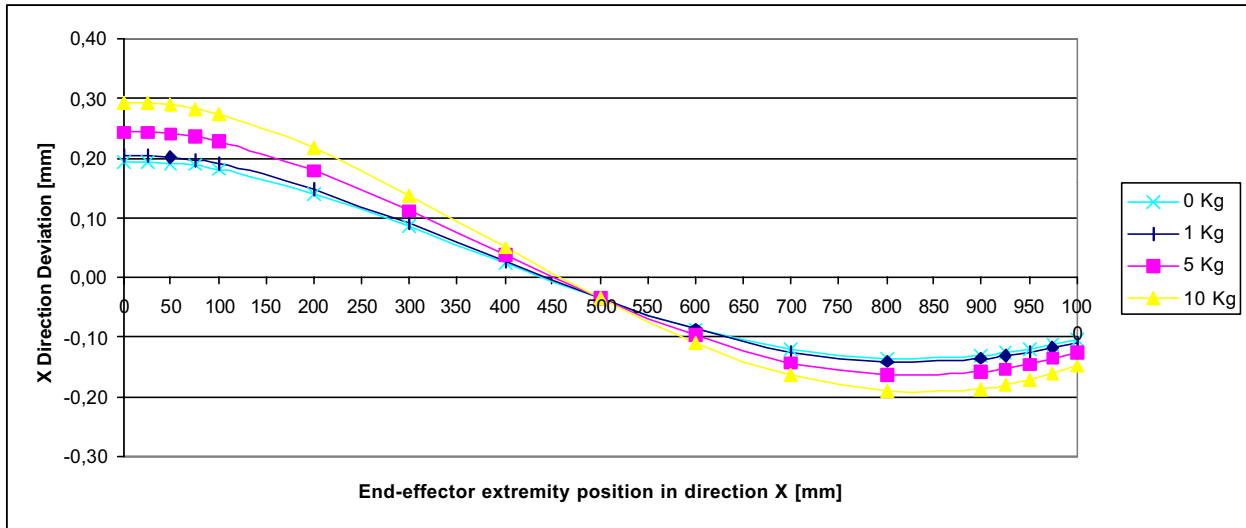


Figure 11. End-effector extremity position deviation history in the directions X, Y and Z.

Conclusions

It can be observed in Fig. (11) the proportionality between the end-effector extremity position deviation the payload/end-effector masses, due to the elastic nature of the element used in the FEM modelling, beside this, the hypothesis of the friction coefficients to be constant implies in the elimination of the non-linearities due to friction in the torque/loads equations.

Experimental analysis of the manipulator is of fundamental importance to provide the necessary modelling adjustments and data validation, due to non-linearities and non-precise components assembly.

The design methodology developed by this research can be used as an optimization tool for a conventional application of gantry manipulators, since it is able to identify structural limits during the design process. The most significant phases are the following:

- a) determination of inertial and geometrical properties of each component;
- b) kinematics modelling based on homogeneous transformation matrices;
- c) dynamic modelling based on Newton-Euler equations;
- d) definition of the planned workspace where tasks will be performed (i.e., CIM environment);
- e) determination of the desired paths and kinematics parameters;
- f) calculation of the manipulator static and dynamic loads (based on phases b to e);
- g) structural deformations mapping (based on phase f);
- h) calculation of the deviation-function at the manipulator end-effector.

At this point, the modal analysis and the interpolation functions for positioning deviation are being implemented for different boundary conditions. The manipulator is actually under construction and further results will be presented in the near future.

Acknowledgements

The authors would like to express their gratitude to FAPERJ for the financial support provided during the course of this present research through grant 170.818/95. Additional partial financial support was provided by CNPq through grant 520864/97.

References

- ABB, 1996, "ABB Products Catalog", ABB Flexible Automation AG, Switzerland.
- ANSYS-PC, 1991, "Reference Manual for Revision 4.4 Vol. I", Swanson Analysis Inc., USA.
- DEA, 1995, "Products Catalog", DEA spa, Italy.
- Ferreira, C.V., 2000, "Compensation of Positioning Deviations for a Gantry Manipulator" (In Portuguese), M.Sc. Thesis, Federal University of Rio de Janeiro, Rio de Janeiro, R.J., Brazil, 119 p.
- Fu, K. S., Gonzalez, R. C., Lee, 1987, C. S. G., "Robotics Control, Sensing, Vision, and Intelligence", McGraw-Hill, Singapore, 580 p.
- Rivin, E. I., 1988, "Mechanical Design of Robots", McGraw-Hill, USA.
- Romano, V. F., Ferreira, C. V., 1999, "Positioning Error Compensation for Gantry Telerobot Multi-task Performance" (in portuguese), Proceedings of the 15th Brazilian Congress of Mechanical Engineering, Águas de Lindóia, SP, Brazil, registered in CD.
- Sciavicco, L., Siciliano, B., 1995, "Robotica Industriale – Modellistica e Controllo di Manipulatori", McGraw-Hill, Italy, 374 p.
- Warneck, H. J., Schraft, R. D., Wanner, M. C., 1985, "Mechanical Design of the Robot System", in Handbook of Industrial Robotics, ed S. Y. Norf, John Wiley & Sons, USA, pp. 44-79.
- Yong, Y. F., Gleave, J. A., Green, J. C., Bonney, M. C., 1985, "Off-line Programming of Robots", in Handbook of Industrial Robotics, ed. S. Y. Norf, John Wiley & Sons, USA, , pp. 366-380.

Modelagem Cinemática de um Robô Hexápode

Salvador Martins da Costa Netto

COPPE/UFRJ - Programa de Engenharia Mecânica
salnetto@yahoo.com.br

Max Suell Dutra

COPPE/UFRJ – Programa de Engenharia Mecânica
maxdutra@ufrj.br

Resumo. O objetivo deste trabalho é o estudo da cinemática de um robô hexápode. Com a finalidade de controlar seu corpo de forma a realizar qualquer movimento no espaço, faz-se necessário obter a evolução dos ângulos de junta de suas pernas (com 3 graus de liberdade cada perna) ao longo do tempo. Para isto, é necessário analisar a cinemática do robô como um todo, o que consiste numa tarefa árdua devido à complexidade envolvida quando as cadeias cinemáticas fechadas (com os pés tocando o solo) são tratadas por métodos tradicionais. Neste trabalho, essas dificuldades são expostas e é proposto um método alternativo, que será chamado cinemática pseudo-aberta. Este método consiste no tratamento das cadeias fechadas como se abertas fossem; para isto, o corpo do robô é cortado em pedaços imaginários estabelecendo-se “cadeias cinemáticas abertas fictícias”, onde movimentos virtuais são impostos. Com isto, pode-se aproveitar os resultados da cinemática de cadeia aberta, que são mais facilmente obtidos e trabalhá-los convenientemente para a solução da cinemática em cadeia fechada, resultando numa formulação simples e geral, ou seja, aplicável igualmente a todas as pernas do robô.

Palavras chave: robótica, cinemática, hexápode, cinemática pseudo-aberta, sistema multicorpos

1. Introdução

Com o intuito de desenvolver um sistema capaz de controlar um robô (Costa Netto, 2001) de maneira que seu corpo possa realizar um movimento qualquer no espaço (giro, curva, inclinação, abaixar, levantar, etc), torna-se necessário obter a evolução dos ângulos das juntas de cada perna ao longo do tempo que resulte no movimento requerido. Para isso precisa-se analisar a cinemática do robô como um todo. Isso significa que se deve estudar o comportamento das variáveis de cada perna para realizar um objetivo comum (o movimento do robô), o que consiste numa tarefa árdua; principalmente por se tratar de cadeias cinemáticas fechadas, devido à alta complexidade envolvida e às diversas possibilidades de movimentos. Esta análise apresenta alguns fatores que a tornam demasiadamente complicada se feita de modo tradicional (pela análise da cinemática de cadeias fechadas). Neste capítulo, procura-se expor estas dificuldades e propor um método alternativo que será doravante chamado *Cinemática Pseudo-aberta*.

O que se chama método tradicional neste texto consiste em qualquer técnica de análise que se baseie no estudo da cinemática de cadeia fechada aplicada ao mecanismo como um todo (seu corpo + suas pernas em contato com o solo). A análise feita desta forma apresenta alguns inconvenientes no caso do robô hexápode especificamente. Entre eles, pode-se citar Dutra et al (1999):

- *Geometria variável:* Durante o movimento o número de pernas e quais as pernas que estão em contato com o solo variam com o tempo;
- *Necessidade de muitas análises:* Neste caso, em que todas as pernas são idênticas, necessita-se de quatro análises diferentes - para 3, 4, 5 e 6 pernas em contato com o solo. Este número de análises pode aumentar muito na medida em que haja um número crescente de pernas não idênticas;
- *Necessidade de diferentes análises:* Cada perna é considerada como um mecanismo distinto apesar de suas semelhanças, o que torna necessário um tratamento diferenciado para cada uma delas;

2. Cinemática de Cadeia Aberta (pernas sem contato com o solo)

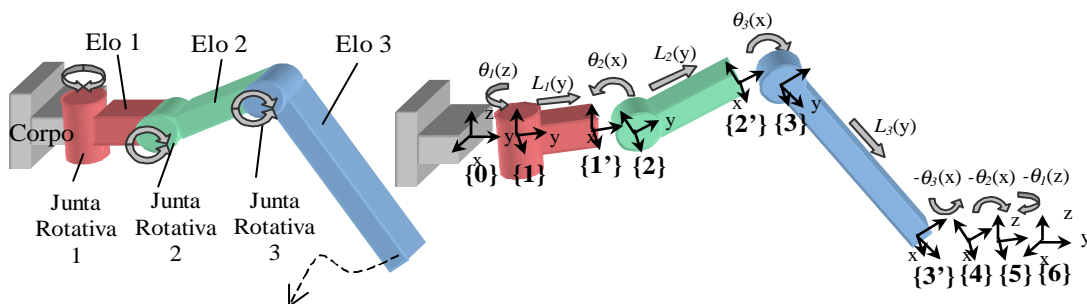


Figura 1. O modelo de perna e seus referenciais.

2. 1. Cinemática Direta da Perna

O problema cinemático direto consiste em determinar as coordenadas \mathbf{x} operacionais, de dimensão n , da ponta da perna em função das coordenadas $\boldsymbol{\theta}$ generalizadas, de dimensão m , das juntas $[\mathbf{x} = f(\boldsymbol{\theta})]$. Sua solução pode ser facilmente obtida através de uma matriz de transformação homogênea (Eq. 2) obtida pela multiplicação sucessiva das matrizes de rotação e translação (Eq. 1) que relacionam entre si os referenciais da perna (Figura (1)).

$${}^0\mathbf{M}_6 = \text{Rot}(\theta_1, z) \cdot \text{Trans}(L_1, y) \cdot \text{Rot}(\theta_2, x) \cdot \text{Trans}(L_2, y) \cdot \text{Rot}(\theta_3, x) \cdot \text{Trans}(L_3, y) \cdot \text{Rot}(-\theta_3, x) \cdot \text{Rot}(-\theta_2, x) \cdot \text{Rot}(-\theta_1, z) \quad (1)$$

Da equação 1 resulta a matriz de transformação homogênea do referencial {6} em {0}:

$${}^0\mathbf{M}_6 = \begin{bmatrix} 1 & 0 & 0 & -(L_1 + L_2 C_2 + L_3 C_{23}) S_1 \\ 0 & 1 & 0 & (L_1 + L_2 C_2 + L_3 C_{23}) C_1 \\ 0 & 0 & 1 & L_2 S_2 + L_3 S_{23} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

onde: $S_i = \text{sen } \theta_i$, $C_i = \text{cos } \theta_i$, $S_{ij} = \text{sen}(\theta_i + \theta_j)$ e $C_{ij} = \text{cos}(\theta_i + \theta_j)$.

Assim, de ${}^0\mathbf{M}_6$ (Eq. 2) aplicado à origem do referencial {6} (no pé) resultam nas equações que definem o vetor de translação em relação a {0} desejado; ou seja:

$$[{}^0x_6, {}^0y_6, {}^0z_6, 1]^T = {}^0\mathbf{M}_6 \cdot [0, 0, 0, 1]^T \quad (3)$$

de (Eq. 3):

$${}^0x_6 = -[L_1 + L_2 \text{cos } \theta_2 + L_3 \text{cos}(\theta_2 + \theta_3)] \text{sen } \theta_1 \quad (4)$$

$${}^0y_6 = [L_1 + L_2 \text{cos } \theta_2 + L_3 \text{cos}(\theta_2 + \theta_3)] \text{cos } \theta_1 \quad (5)$$

$${}^0z_6 = L_2 \text{sen } \theta_2 + L_3 \text{sen}(\theta_2 + \theta_3) \quad (6)$$

2. 2. Cinemática Inversa da Perna

Ao contrário do problema anterior, o problema cinemático inverso consiste em determinar as coordenadas $\boldsymbol{\theta}$ generalizadas, de dimensão m , das juntas em função das coordenadas \mathbf{x} operacionais, de dimensão n , da ponta da perna $[\boldsymbol{\theta} = f^{-1}(\mathbf{x})]$.

Denominando por (x_p, y_p, z_p) as coordenadas do pé (ref. {6}) em relação ao ref. {0}, tem-se: $(x_p = {}^0x_6, y_p = {}^0y_6, z_p = {}^0z_6)$. Com o objetivo de analisar geometricamente o mecanismo da perna para obter a cinemática inversa deste, é preciso definir adequadamente alguns parâmetros auxiliares, tais como os representados pelo desenho esquemático da Figura (2). Para facilitar a análise, esta representação é feita com a ajuda de um plano Ψ , que define o plano que contém a perna do robô a cada instante do movimento.

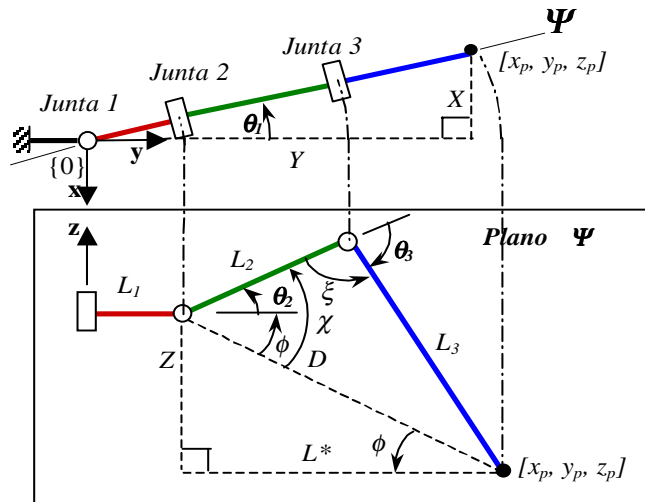


Figura 2. Desenho esquemático da perna do robô

Note que, facilmente pode-se obter θ_1 , fazendo:

$$\theta_1 = \arctan\left(\frac{X}{Y}\right) = \arctan\left(-\frac{x_p}{y_p}\right) \quad (7)$$

Para calcular θ_2 e θ_3 recorre-se aos ângulos auxiliares ϕ , χ e ξ , atentando para suas orientações. Faz-se:

$$\theta_2 = \chi - \phi \quad (8)$$

$$\xi - \theta_3 = \pi \therefore \theta_3 = \xi - \pi \quad (9)$$

onde: $-\pi/2 < \phi < \pi/2$; $0 < \chi < \pi$ e $0 < \xi < \pi$.

Para explicitar estes resultados em termos das coordenadas operacionais x_p , y_p e z_p , é necessário fazer a manipulação trigonométrica como segue:

$$\phi = \arcsen\left(\frac{Z}{D}\right) = \arcsen\left(-\frac{z_p}{D}\right), \quad (-\pi/2 < \phi < \pi/2) \quad (10)$$

Da lei dos cosenos aplicada ao triângulo formado pelos elos L_2 , L_3 da perna e pelo parâmetro auxiliar D , tem-se:

$$L_3^2 = L_2^2 + D^2 - 2 \cdot L_2 \cdot D \cdot \cos \chi \quad (11)$$

$$\therefore \chi = \arccos\left(\frac{L_2^2 + D^2 - L_3^2}{2 \cdot L_2 \cdot D}\right) \quad (0 < \chi < \pi) \quad (12)$$

$$D^2 = L_2^2 + L_3^2 - 2 \cdot L_2 \cdot L_3 \cdot \cos \xi \quad (13)$$

$$\therefore \xi = \arccos\left(\frac{L_2^2 + L_3^2 - D^2}{2 \cdot L_2 \cdot L_3}\right) \quad (0 < \xi < \pi) \quad (14)$$

onde, formando os parâmetros Z , D e L^* um triângulo retângulo, tem-se:

$$D = |\sqrt{L^{*2} + Z^2}| = |\sqrt{L^{*2} + z_p^2}| \quad (15)$$

sendo, da mesma forma com o triângulo formado por X , Y e (L^*+L_1) :

$$L^* = |\sqrt{X^2 + Y^2}| - L_1 = |\sqrt{x_p^2 + y_p^2}| - L_1 \quad (16)$$

Finalmente, substituindo a Eq.(16) em Eq.(15) e aplicando esta nas Eqs. (10, 12 e 14), obtém-se:

$$\phi = \arcsen\left[-\frac{z_p}{|\sqrt{(\sqrt{x_p^2 + y_p^2} - L_1)^2 + z_p^2}|}\right] \quad (-\pi/2 < \phi < \pi/2) \quad (17)$$

$$\chi = \arccos\left[\frac{L_2^2 + \left(\sqrt{x_p^2 + y_p^2} - L_1\right)^2 + z_p^2 - L_3^2}{2 \cdot L_2 \cdot \left|\sqrt{(\sqrt{x_p^2 + y_p^2} - L_1)^2 + z_p^2}\right|}\right] \quad (0 < \chi < \pi) \quad (18)$$

$$\xi = \arccos\left[\frac{L_2^2 + L_3^2 - \left(\sqrt{x_p^2 + y_p^2} - L_1\right)^2 + z_p^2}{2 \cdot L_2 \cdot L_3}\right] \quad (0 < \xi < \pi) \quad (19)$$

Desta forma, da Eq. (7) e substituindo Eq. (17), Eq. (18) e Eq. (19) em Eq. (8) e Eq. (9), a cinemática inversa fica solucionada pelas seguintes equações:

$$\theta_1 = \arctan\left(-\frac{x_p}{y_p}\right) \quad (-\pi/2 < \theta_1 < \pi/2) \quad (20)$$

$$\theta_2 = \arccos\left[\frac{L_2^2 + \left(\sqrt{x_p^2 + y_p^2} - L_1\right)^2 + z_p^2 - L_3^2}{2 \cdot L_2 \cdot \left|\sqrt{\left(\sqrt{x_p^2 + y_p^2} - L_1\right)^2 + z_p^2}\right|}\right] - \arcsen\left[-\frac{z_p}{\left|\sqrt{\left(\sqrt{x_p^2 + y_p^2} - L_1\right)^2 + z_p^2}\right|}\right] \quad (-\pi/2 < \theta_2 < \pi/2) \quad (21)$$

$$\theta_3 = \arccos\left[\frac{L_2^2 + L_3^2 - \left(\sqrt{x_p^2 + y_p^2} - L_1\right)^2 + z_p^2}{2 \cdot L_2 \cdot L_3}\right] - \pi \quad (-\pi < \theta_3 < 0) \quad (22)$$

Os resultados obtidos para a cinemática inversa podem ser bem pesados computacionalmente e, nem sempre são facilmente determinados. Por isso, métodos alternativos podem ser usados vantajosamente, tais como *Lógica Fuzzy* e *Redes Neurais*; para este caso, estes métodos podem ser vistos em Costa Netto et al (2000).

3. Cinemática de Cadeia Fechada

A cinemática de cadeia aberta para o modelo de uma perna já foi desenvolvida no tópico 2 sem considerar a posição desta em relação ao centro geométrico do corpo do robô Eqs. (4, 5, 6, 20, 21 e 22). Esta consideração pode facilmente ser feita ao arbitrar-se uma distribuição adequada de referenciais para as bases das pernas no corpo. O modelo de robô estudado possui pernas idênticas, dispostas simetricamente ao longo de seu corpo, sendo três de cada lado. A denominação que se utiliza para diferenciar uma perna da outra, baseia-se na sua localização em relação ao corpo do robô, que pode ser direita (D) ou esquerda (E) e ainda frontal (F), intermediária (I) ou traseira (T); cada qual com seu referencial {0} relativo à sua base fixa ao corpo.

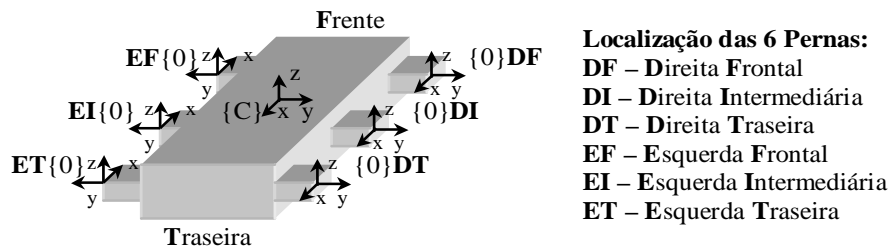


Figura 3. Distribuição dos referenciais do corpo do robô {C} e os referenciais das bases das pernas {0} fixos a este.

Note pela Figura (3) que dependendo do lado em que a perna se situa (direita/esquerda), o referencial {0} tem uma determinada orientação em relação ao corpo. Isso acontece para que se possa aproveitar a análise generalizada já feita para o modelo de uma perna.

Para relacionar o referencial {6} da ponta de cada perna com o referencial {C} do centro geométrico do corpo do robô (${}^C M_6$), sendo conhecido ${}^0 M_6$ (Eq. 2), basta fornecer as coordenadas (${}^C x_0, {}^C y_0, {}^C z_0$) da base de cada perna em relação a {C} e, no caso das pernas situadas à esquerda do corpo, a mudança de orientação do referencial {0} deve ser considerada; definindo a matriz de transformação homogênea ${}^C M_0$.

Assim, fica:

$${}^C M_6 = {}^C M_0 \cdot {}^0 M_6 \quad (23)$$

Ou seja:

$${}^C M_6 = \begin{bmatrix} 1 & 0 & 0 & {}^C x_0 \\ 0 & 1 & 0 & {}^C y_0 \\ 0 & 0 & 1 & {}^C z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & {}^0 x_6 \\ 0 & 1 & 0 & {}^0 y_6 \\ 0 & 0 & 1 & {}^0 z_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & {}^C x_0 + {}^0 x_6 \\ 0 & 1 & 0 & {}^C y_0 + {}^0 y_6 \\ 0 & 0 & 1 & {}^C z_0 + {}^0 z_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (24)$$

e

$${}^c\mathbf{M}_6 = \begin{bmatrix} -1 & 0 & 0 & {}^c x_0 \\ 0 & -1 & 0 & {}^c y_0 \\ 0 & 0 & 1 & {}^c z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & {}^0 x_6 \\ 0 & 1 & 0 & {}^0 y_6 \\ 0 & 0 & 1 & {}^0 z_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & {}^c x_0 - {}^0 x_6 \\ 0 & -1 & 0 & {}^c y_0 - {}^0 y_6 \\ 0 & 0 & 1 & {}^c z_0 + {}^0 z_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (25)$$

para as pernas direitas e esquerdas respectivamente.

Sendo definido:

$$[{}^c x_6, {}^c y_6, {}^c z_6, 1]^T = {}^c\mathbf{M}_6 \cdot [0, 0, 0, 1]^T \quad (26)$$

e,

$$(x_p, y_p, z_p) = ({}^0 x_6, {}^0 y_6, {}^0 z_6) \quad (27)$$

resulta das Eqs. (24, 25, 26 e 27):

$$\text{Pernas direitas: } {}^c x_6 = {}^c x_0 + x_p; \quad {}^c y_6 = {}^c y_0 + y_p; \quad {}^c z_6 = {}^c z_0 + z_p \quad (28)$$

$$\text{Pernas esquerdas: } {}^c x_6 = {}^c x_0 - x_p; \quad {}^c y_6 = {}^c y_0 - y_p; \quad {}^c z_6 = {}^c z_0 + z_p \quad (29)$$

As Eqs. (28) e (29) definem a posição da ponta da perna em relação ao centro geométrico do corpo do robô. Esse resultado pode ser estendido à situação de cadeia cinemática fechada, pois as relações entre os referenciais da perna não mudam. Porém, com o intuito de estabelecer a movimentação do robô (com os pés no solo), o vetor $(x_p, y_p, z_p) = ({}^0 x_6, {}^0 y_6, {}^0 z_6)$ usado em Eq. (28) e Eq. (29) deve ser obtido através da abordagem da cinemática fechada, e não mais das Eqs. (4, 5 e 6); isto feito, pode-se aplicar seus resultados em Eqs (20, 21 e 22) para obter a cinemática inversa. Esta abordagem é tema para o tópico a seguir.

3.1. Cinemática Pseudo-aberta: Uma Alternativa para Cadeias Fechadas

No caso das pernas do robô em contato com o solo, estabelece-se uma cadeia cinemática fechada. Visando evitar as já comentadas dificuldades envolvidas na sua análise, propõe-se um método alternativo que será chamado *Cinemática Pseudo-aberta*. Ao invés de obter o movimento do corpo em função do movimento das juntas da perna, este método consiste em obter o movimento passivo das juntas das pernas para um dado movimento do corpo do robô. Para isso, é preciso primeiramente impor um *movimento virtual* ao corpo e obter a evolução dos ângulos de junta necessários para acompanhar este movimento.

Sabe-se que independente do número de pernas em contato com o solo, o número de graus de liberdade (GL_f) conferido ao corpo do robô é igual a seis (3 rotações + 3 translações) (Costa Netto, 2001). Sabendo disso, pode-se imaginar como um artifício que o corpo do robô esteja dividido em pedaços onde, cada um desses está ligado a uma perna em contato com o solo, ou seja, cada perna em contato com o solo pode ser considerada como um mecanismo independente em *cinemática aberta*, tendo como extremidade livre um *pedaço de corpo*. Isto pode ser feito desde que se considere o seu ponto de contato com o solo como uma *junta esférica fictícia*.

Obviamente, cada uma destas pernas deve trabalhar com um objetivo em comum, ou seja, seus movimentos se restringem a estabelecer a movimentação de um corpo único (constituído da junção rígida dos *pedaços imaginários*). Para essa análise, é necessário se estabelecer um sistema de referenciais conveniente.

3.1.1. Os Referenciais

Na Seção 2.1 são descritos os referenciais de uma perna do robô (ver Figura (1)). Além do referencial {0} relativos à base da perna (fixos ao corpo) e {6} relativo à ponta da perna (lembrando que este é paralelo a {0}), é necessário usar, para a completa descrição da cinemática pseudo-aberta do robô, o referencial {C} relativo ao corpo. O referencial {C} permite descrever o movimento do centro geométrico corpo do robô em relação a um referencial {I} fixo ao solo (considerado um referencial inercial), através de rotações e translações definidas por uma matriz ${}^I\mathbf{M}_C$ (ver Figura (4)).

- Referencial {C}: Sua origem localiza-se no centro geométrico do corpo do robô. Este referencial também se posiciona paralelo ao referencial {0};
- Referencial {I}: Um referencial inercial qualquer a ser escolhido, neste caso um ponto qualquer fixo ao solo (considerando a Terra um referencial inercial)

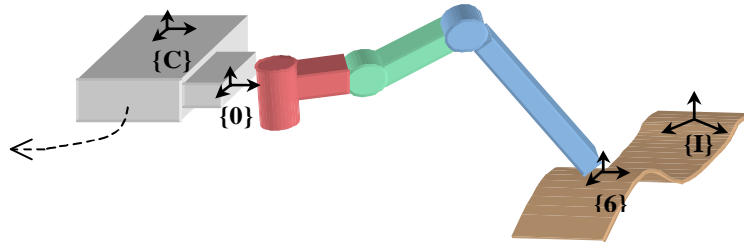


Figura 4. Disposição dos referenciais de cada perna e do corpo do robô e o referencial inercial.

3.1.2. Mudanças de Referenciais

Conforme já exposto anteriormente, a mudança de um referencial $\{i\}$ para um referencial $\{j\}$ pode ser feita considerando-se uma matriz de transformação homogênea ${}^j\mathbf{M}_i$ que define rotações e/ou translações entre um referencial e o outro. No caso da análise da perna em cinemática aberta, o referencial absoluto era o próprio corpo do robô $\{C\}$. Neste caso, diferentemente, o referencial absoluto de movimentação é o referencial inercial $\{I\}$. Assim, com o objetivo de determinar a evolução dos ângulos de junta da perna através da já conhecida cinemática inversa da perna, baseada no ref. $\{C\}$, que neste caso varia com o tempo (bem como o ref. $\{6\}$), necessita-se de um índice t que represente o instante de tempo a que se refere um determinado referencial (Ex.: Ref. $\{C(t)\}$, Ref. $\{6(t)\}$). Isso ficará mais claro à medida que se detalhe cada etapa de movimentação do robô.

O objetivo final da análise consiste em obter a mudança de referenciais ${}^{C(t)}\mathbf{M}_{6(t)}$, da ponta de uma perna em contato com o chão, num instante de tempo $t=\mathbf{T}$, de um determinado movimento que o robô tenha que executar.

A mudança de referencial da ponta da perna em relação ao corpo ${}^C\mathbf{M}_6$ está definida por Eq. (24) e Eq. (25) e já é conhecida no instante $t = \mathbf{0}$ (Eqs. (4, 5 e 6), conhecendo sua postura inicial).

Conhecido:

$${}^{C(0)}\mathbf{M}_{6(0)} = \begin{bmatrix} \pm 1 & 0 & 0 & {}^{C(0)}x_{6(0)} \\ 0 & \pm 1 & 0 & {}^{C(0)}y_{6(0)} \\ 0 & 0 & 1 & {}^{C(0)}z_{6(0)} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (30)$$

A transformação ${}^I\mathbf{M}_C$, representa a mudança do referencial do corpo do robô em relação ao referencial inercial, portanto, no instante inicial ($t = \mathbf{0}$) este é conhecido e composto de rotações e/ou translações em relação a $\{I\}$, definindo o posicionamento inicial do robô.

A descrição das translações pode ser obtida através uma matriz que contenha o vetor de translação (${}^Ix_{C(0)}$, ${}^Iy_{C(0)}$, ${}^Iz_{C(0)}$) em relação ao referencial inercial $\{I\}$. Para descrever as rotações, precisa-se definir os ângulos de posicionamento (γ_0 , β_0 , α_0) em torno dos eixos coordenados deste referencial $\{I\}$, fixo; estas rotações (em torno de eixos fixos) são denominadas na literatura por *roll* (rolamento), *pitch* (inclinação) e *yaw* (guinada) (Craig, 1986). Desta forma segue:

Conhecido:

$$\begin{aligned} {}^I\mathbf{M}_{C(0)} &= \mathbf{Rot}(\alpha_0, z) \cdot \mathbf{Rot}(\beta_0, y) \cdot \mathbf{Rot}(\gamma_0, x) \cdot \mathbf{Trans}({}^Ix_{C(0)}, {}^Iy_{C(0)}, {}^Iz_{C(0)}) = \\ &= \begin{bmatrix} c\alpha_0 & -s\alpha_0 & 0 & 0 \\ s\alpha_0 & c\alpha_0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\beta_0 & 0 & s\beta_0 & 0 \\ 0 & 1 & 0 & 0 \\ -s\beta_0 & 0 & c\beta_0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\gamma_0 & -s\gamma_0 & 0 \\ 0 & s\gamma_0 & c\gamma_0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & {}^Ix_{C(0)} \\ 0 & 1 & 0 & {}^Iy_{C(0)} \\ 0 & 0 & 1 & {}^Iz_{C(0)} \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ \therefore {}^I\mathbf{M}_{C(0)} &= \begin{bmatrix} \mathbf{R}_{C(0)} & \mathbf{T}_{C(0)} \\ \hline 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} c\alpha_0 c\beta_0 & c\alpha_0 s\beta_0 s\gamma_0 - s\alpha_0 c\gamma_0 & c\alpha_0 s\beta_0 c\gamma_0 + s\alpha_0 s\gamma_0 & {}^Ix_{C(0)} \\ s\alpha_0 c\beta_0 & s\alpha_0 s\beta_0 s\gamma_0 + c\alpha_0 c\gamma_0 & s\alpha_0 s\beta_0 c\gamma_0 - c\alpha_0 s\gamma_0 & {}^Iy_{C(0)} \\ -s\beta_0 & c\beta_0 s\gamma_0 & c\beta_0 c\gamma_0 & {}^Iz_{C(0)} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (31) \end{aligned}$$

onde, $ci = \cos(i)$ e $si = \sin(i)$.

▪ Pode-se determinar agora, para o instante inicial, a matriz de transformação homogênea que leva o ref. $\{6(0)\}$ da ponta do pé, no ref. $\{I\}$ inercial (fixo).

Calcula-se de Eq. (30) e Eq. (31):

$${}^I\mathbf{M}_{6(0)} = {}^I\mathbf{M}_{C(0)} \cdot {}^{C(0)}\mathbf{M}_{6(0)} \quad (32)$$

O *deslocamento virtual* conferido ao corpo pode ser interpretado com uma dada movimentação (com seis graus de liberdade) do referencial {C} do centro geométrico do corpo ao longo do tempo - {C(t)}.

Assim, define-se: ${}^{C(0)}\mathbf{M}_{C(T)}$ como um deslocamento num instantâneo de tempo $t = \mathbf{T}$, a partir de {C(0)} inicial resultando no novo referencial {C(T)}, subsequente. Este deslocamento é dado pelo *deslocamento virtual* imposto ao centro de gravidade do corpo.

A descrição das translações pode ser obtida através uma matriz que contenha o vetor de translação (${}^{C(0)}x_{C(T)}$, ${}^{C(0)}y_{C(T)}$, ${}^{C(0)}z_{C(T)}$) do referencial corpo {C(T)} com relação ao seu referencial anterior {C(0)}. Para descrever as rotações, definem-se os ângulos de movimentação (γ , β , α) em torno dos eixos coordenados do referencial {C(0)}; como o referencial {C} é dependente do tempo (diferente do referencial {I}) estes ângulos não são absolutos, e sim relativos a cada nova orientação assumida por cada eixo coordenado deste referencial, as rotações podem ser convenientemente descritas pelos denominados *ângulos de Euler Z-Y-X* (Craig, 1986). Desta forma:

Dado:

$${}^{C(0)}\mathbf{M}_{C(T)} = \mathbf{Rot}(\alpha, z) \cdot \mathbf{Rot}(\beta, y) \cdot \mathbf{Rot}(\gamma, x) \cdot \mathbf{Trans}({}^{C(0)}x_{C(T)}, {}^{C(0)}y_{C(T)}, {}^{C(0)}z_{C(T)}) =$$

$$= \begin{bmatrix} c\alpha & -s\alpha & 0 & 0 \\ s\alpha & c\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\beta & 0 & s\beta & 0 \\ 0 & 1 & 0 & 0 \\ -s\beta & 0 & c\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\gamma & -s\gamma & 0 \\ 0 & s\gamma & c\gamma & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & {}^{C(0)}x_{C(T)} \\ 0 & 1 & 0 & {}^{C(0)}y_{C(T)} \\ 0 & 0 & 1 & {}^{C(0)}z_{C(T)} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\therefore {}^{C(0)}\mathbf{M}_{C(T)} = \begin{bmatrix} & & & \\ & {}^{C(0)}\mathbf{R}_{C(T)} & & \\ & & & \\ 0 & & & 1 \end{bmatrix} = \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma & {}^{C(0)}x_{C(T)} \\ s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma & {}^{C(0)}y_{C(T)} \\ -s\beta & c\beta s\gamma & c\beta c\gamma & {}^{C(0)}z_{C(T)} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (33)$$

onde, $ci = \cos(i)$ e $si = \sin(i)$.

Com isso, no instante $t = \mathbf{T}$ também se pode definir outras mudanças de referenciais úteis, tais como:

${}^I\mathbf{M}_{C(T)}$, que define a mudança entre o novo referencial do corpo em relação ao referencial inercial. Calcula-se da Eq. (31) e Eq. (33):

$${}^I\mathbf{M}_{C(T)} = {}^I\mathbf{M}_{C(0)} \cdot {}^{C(0)}\mathbf{M}_{C(T)} \quad (34)$$

${}^I\mathbf{M}_{6(T)}$ define a mudança entre o novo referencial da ponta da perna em relação ao referencial inercial. Por {6} ser sempre paralelo a {C}, suas componentes de rotação em relação a {I} são idênticas (${}^I\mathbf{R}_{6(T)} = {}^I\mathbf{R}_{C(T)}$ (35)). E como o referencial {6} não translada quando em cinemática fechada (pé no chão sem deslizamento), tem-se que a componente de translação do ref. {6} em relação ao ref. {I} é invariável (${}^I\mathbf{T}_{6(T)} = {}^I\mathbf{T}_{6(0)}$ (36)).

Assim, da Eq. (35) e Eq. (36):
monta-se:

$${}^I\mathbf{M}_{6(T)} = \begin{bmatrix} & & & \\ & {}^I\mathbf{R}_{6(T)} & & \\ & & & \\ 0 & & & 1 \end{bmatrix} = \begin{bmatrix} & & & \\ & {}^I\mathbf{R}_{C(T)} & & \\ & & & \\ 0 & & & 1 \end{bmatrix} \quad (37)$$

Agora, finalmente já se pode obter ${}^{C(T)}\mathbf{M}_{6(T)}$, fazendo da Eq. (34) e Eq. (37):

$${}^{C(T)}\mathbf{M}_{6(T)} = {}^{C(T)}\mathbf{M}_I \cdot {}^I\mathbf{M}_{6(T)} \quad (38)$$

A Eq. (38) define a matriz de transformação homogênea que relaciona o referencial da ponta da perna em relação ao centro do corpo do robô a partir do movimento deste. A utilidade desta matriz consiste na sua importância como ferramenta para a obtenção da cinemática direta e, na facilidade com que nos permite obter a cinemática inversa, como será visto mais adiante.

Note que, apesar da Eq. (38) relacionar os referenciais somente no instante $t=\mathbf{T}$ (do *deslocamento virtual*), para a construção dessas matrizes precisou-se relacionar referenciais no instante de tempo $t=0$. Da Eq. (34) em Eq. (38):

$${}^{C(T)}\mathbf{M}_{6(T)} = {}^{C(0)}\mathbf{M}_{C(T)}^{-1} \cdot {}^I\mathbf{M}_{C(0)}^{-1} \cdot {}^I\mathbf{M}_{6(T)} \quad (39)$$

Esta formulação pode ser generalizada para qualquer instante de tempo $t=(t+\Delta t)$ de um *deslocamento virtual*, desde que se tenha em mãos os dados relativos ao instante de tempo anterior $t=(t)$. Assim, um movimento completo pode ser obtido de uma seqüência de definida ao longo de uma sucessão de instantes de tempo, aplicada às diversas pernas em contato com o solo. Da Eq. (39):

$${}^{C(t+\Delta t)}\mathbf{M}_{6(t+\Delta t)} = {}^{C(t)}\mathbf{M}_{C(t+\Delta t)}^{-1} \cdot {}^I\mathbf{M}_{C(t)}^{-1} \cdot {}^I\mathbf{M}_{6(t+\Delta t)} \quad (40)$$

Para monitorar a cada novo instante a posição do robô em relação ao referencial inercial {I}, basta fazer a mudança de referenciais de {C} em {I} dada por Eq. (34):

$${}^I\mathbf{M}_{C(t+\Delta t)} = {}^I\mathbf{M}_{C(t)} \cdot {}^{C(t)}\mathbf{M}_{C(t+\Delta t)} \quad (41)$$

3.1.3 - Cinemática Direta

Lembrando da definição da cinemática direta já exposta neste trabalho, sabendo que seu objetivo é a obtenção das coordenadas operacionais em função das coordenadas de junta ($\mathbf{x} = f(\boldsymbol{\theta})$), e sabendo que a matriz de transformação homogênea consiste numa ferramenta capaz de fazer este mapeamento; pode-se obtê-la facilmente fazendo:

$$[{}^{C(t+\Delta t)}x_{6(t+\Delta t)}, {}^{C(t+\Delta t)}y_{6(t+\Delta t)}, {}^{C(t+\Delta t)}z_{6(t+\Delta t)}, 1]^T = {}^{C(t+\Delta t)}\mathbf{M}_{6(t+\Delta t)} \cdot [0,0,0,1]^T \quad (42)$$

Onde os componentes (${}^{C(t+\Delta t)}x_{6(t+\Delta t)}, {}^{C(t+\Delta t)}y_{6(t+\Delta t)}, {}^{C(t+\Delta t)}z_{6(t+\Delta t)}$) do vetor obtido (em Eq.42) definem as coordenadas operacionais que descrevem a posição da ponta da perna (ref.{6}) em relação ao centro geométrico do corpo (ref.{C}), que satisfaz um dado posicionamento do corpo do robô.

Das Eqs. (31, 33 e 35) em Eq. (40), e Eq. (40) em Eq. (42), resulta (sendo $ci = \cos(i)$, $si = \sin(i)$):

$${}^{C(t+\Delta t)}x_{6(t+\Delta t)} = ({}^{C(t)}x_{6(t)} - {}^{C(t)}x_{C(t+\Delta t)}) ca c\beta + ({}^{C(t)}y_{6(t)} - {}^{C(t)}y_{C(t+\Delta t)}) sa c\beta - ({}^{C(t)}z_{6(t)} - {}^{C(t)}z_{C(t+\Delta t)}) s\beta \quad (43)$$

$${}^{C(t+\Delta t)}y_{6(t+\Delta t)} = ({}^{C(t)}x_{6(t)} - {}^{C(t)}x_{C(t+\Delta t)})(-sa c\gamma + ca c\beta s\gamma) + ({}^{C(t)}y_{6(t)} - {}^{C(t)}y_{C(t+\Delta t)})(ca c\gamma + sa s\beta s\gamma) + ({}^{C(t)}z_{6(t)} - {}^{C(t)}z_{C(t+\Delta t)}) c\beta s\gamma \quad (44)$$

$${}^{C(t+\Delta t)}z_{6(t+\Delta t)} = ({}^{C(t)}x_{6(t)} - {}^{C(t)}x_{C(t+\Delta t)})(sa s\gamma + ca s\beta c\gamma) + ({}^{C(t)}y_{6(t)} - {}^{C(t)}y_{C(t+\Delta t)})(-ca s\gamma + sa s\beta c\gamma) + ({}^{C(t)}z_{6(t)} - {}^{C(t)}z_{C(t+\Delta t)}) c\beta c\gamma \quad (45)$$

Note que as Eqs. (43, 44 e 47) estão em função apenas de (${}^{C(t)}x_{6(t)}, {}^{C(t)}y_{6(t)}, {}^{C(t)}z_{6(t)}$), que representa a conhecida posição da ponta da perna em relação ao corpo, e (${}^{C(t)}x_{C(t+\Delta t)}, {}^{C(t)}y_{C(t+\Delta t)}, {}^{C(t)}z_{C(t+\Delta t)}$) e (γ, β, α), que representam o novo posicionamento que o corpo deve assumir em relação ao seu posicionamento anterior, ou seja, a movimentação relativa do robô.

Para saber qual o resultado desta movimentação em relação ao referencial inercial (movimentação absoluta), usa-se o resultado de ${}^I\mathbf{M}_{C(t+\Delta t)}$ Eq. (41), cujos componentes de translação definem a nova posição e os componentes de rotação definem a orientação do corpo do robô em relação ao referencial inercial {I}, ou seja (de CRAIG, 1986):

Posição:

$$\begin{aligned} {}^I x_{C(t+\Delta t)} &= M_{14} \\ {}^I y_{C(t+\Delta t)} &= M_{24} \\ {}^I z_{C(t+\Delta t)} &= M_{34} \end{aligned} \quad (46)$$

Orientação:

$$\begin{aligned} \gamma_o(t+\Delta t) &= \text{Arctg}(M_{21} / M_{11}) \\ \beta_o(t+\Delta t) &= \text{Arctg}[-M_{31} / (M_{11}^2 + M_{21}^2)^{1/2}] \\ \alpha_o(t+\Delta t) &= \text{Arctg}(M_{32} / M_{33}) \end{aligned} \quad (47)$$

onde M_{ij} representa a componente (i, j) da matriz ${}^I\mathbf{M}_{C(t+\Delta t)}$.

3.1.4. Cinemática Inversa

O objetivo principal desta abordagem da *cinemática pseudo-aberta* é o controle do robô, ou seja, saber a evolução das variáveis de junta necessárias para obter determinada movimentação no centro geométrico do robô. Para a realização deste objetivo torna-se necessário o cálculo da cinemática inversa do sistema ($\boldsymbol{\theta} = f^{-1}(x)$).

Lembrando da análise obtida para a cinemática inversa para a perna em cadeia cinemática aberta, pode-se aproveitar este resultado na presente abordagem. Isso se dá devido à consideração feita, que define a cinemática pseudo-aberta como uma *cadeia aberta imaginária* (corpo dividido em pedaços). Assim, dada a posição de cada perna no corpo (${}^C x_0, {}^C y_0, {}^C z_0$), invariável com o tempo, segue de Eq. (28) e Eq. (29):

Se perna direita:

$$\begin{aligned} x_p(t+\Delta t) &= {}^{C(t+\Delta t)}x_{6(t+\Delta t)} - {}^C x_0 \\ y_p(t+\Delta t) &= {}^{C(t+\Delta t)}y_{6(t+\Delta t)} - {}^C y_0 \\ z_p(t+\Delta t) &= {}^{C(t+\Delta t)}z_{6(t+\Delta t)} - {}^C z_0 \end{aligned} \quad (48)$$

Se perna esquerda:

$$\begin{aligned} x_p(t+\Delta t) &= -{}^{C(t+\Delta t)}x_{6(t+\Delta t)} + {}^C x_0 \\ y_p(t+\Delta t) &= -{}^{C(t+\Delta t)}y_{6(t+\Delta t)} + {}^C y_0 \\ z_p(t+\Delta t) &= {}^{C(t+\Delta t)}z_{6(t+\Delta t)} - {}^C z_0 \end{aligned} \quad (49)$$

Pode-se agora aplicar as Eqs. (43, 44 e 45) em Eq. (48) e Eq. (49).

Da solução da cinemática inversa para uma perna (Eq. (20), Eq. (21) e Eq. (22)), pode-se obter as variáveis de juntas ($\theta_1(t+\Delta t)$, $\theta_2(t+\Delta t)$, $\theta_3(t+\Delta t)$) da perna relacionadas com o deslocamento virtual imposto.

$$\theta_1(t+\Delta t) = f_1^{-1}(x_p(t+\Delta t), y_p(t+\Delta t)) \quad (50)$$

$$\theta_2(t+\Delta t) = f_2^{-1}(x_p(t+\Delta t), y_p(t+\Delta t), z_p(t+\Delta t)) \quad (51)$$

$$\theta_3(t+\Delta t) = f_3^{-1}(x_p(t+\Delta t), y_p(t+\Delta t), z_p(t+\Delta t)) \quad (52)$$

Obs: Note que o resultado obtido para a cinemática direta, bem como para a inversa, é **dependente do tempo** e deve portanto ser recalculada a cada instante de tempo; Este procedimento deve ser feito **para todas as pernas em contato com o chão**, de forma a obter a evolução dos ângulos de junta destas para seguir uma determinada trajetória desejada; As velocidades e acelerações angulares das juntas estarão **automaticamente atreladas** às velocidades e acelerações impostas ao movimento virtual do corpo, devido à prévia escolha de Δt .

4. Conclusões

Neste trabalho desenvolveu-se um método eficiente e simples de solução da cinemática de um robô hexápode, tarefa esta essencial para o controle de seus movimentos no espaço, porém de difícil abordagem através de métodos convencionais (principalmente no caso inverso). Com isso, contribui-se para o estudo da cinemática, neste caso aplicada a um robô hexápode com três graus de liberdade em cada perna, podendo-se entretanto generalizar seu desenvolvimento para qualquer outro tipo de robô andante, desde que este possua total liberdade de posicionamento no espaço tridimensional.

5. Agradecimentos

Este projeto recebeu recursos da Fundação de apoio à pesquisa do estado do Rio de Janeiro (FAPERJ).

6. Referências

Costa Netto, S. M. da, Evsukoff, A., Dutra, M. S., 2000, "Fuzzy Systems to Solve Inverse Kinematics Problem in Robots Control: Application to an Hexapod Robot's Leg". In: *Proceedings of the 8th SBRN*, v.1, pp. 150-155, Rio de Janeiro, Brazil.

Costa Netto, S. M. da, 2001, "Estudo e Projeto de um Robô Hexápode com Atuadores utilizando Materiais com Memória de Forma". Tese de Mestrado em Engenharia Mecânica, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

Craig, J. J., 1986, "Introduction to Robotics – Mechanics and Control". 1 ed. Stanford, Addison-Wesley Publishing Company, Inc., USA.

Dutra, M. S., Oliveira, A. M., 1999, "Utilização do Método das Cadeias Cinemáticas Virtuais para o estudo de um Robô Hexápode". *CIDIM'99 – IV Congresso Iberoamericano de Ingenieria Mecânica*, Santiago, Chile.

Sciavicco, L., Siciliano, B., 2000, "Modelling and Control of Robot Manipulators", 2nd ed., Springer Verlag, London, Great Britain

7. Direitos autorais

KINEMATICS MODELLING OF AN HEXAPOD ROBOT

Salvador Martins da Costa Netto

Programa de Engenharia Mecânica – PEM – COPPE/UFRJ
salnetto@yahoo.com.br

Max Suell Dutra

Programa de Engenharia Mecânica – PEM – COPPE/UFRJ
P.O. Box 68503, 21945-970 – Rio de Janeiro, Brazil
maxdutra@ufrj.br

Abstract. *The objective of this work is the study of an hexapod robot's kinematics. To establish the control of your body intending to accomplish any movement in the space, it's necessary to obtain the joint angles evolution of its legs (with 3 degrees of freedom each) along the time. So, it's necessary to analyze the robot's kinematics as a whole, that consists of an arduous task, due to the complexity involved if the closed kinematical chains are treated using traditional methods. In this work, those difficulties are exposed and it's proposed an alternative method, that will be called pseudo-open kinematics. This method consists on treating the closed kinematical chains as it was an open one; so, the robot's body is divided in imaginary pieces, establishing "fictitious open kinematical chains", where virtual movements are imposed. Thus, the open kinematical chain results (more easily obtained) may be adapted and used to solve the closed chain problem, resulting in a simple and general formulation, equally applicable for any robot's leg.*

Keywords. *Robotics, Kinematics, Hexapod, Multibody Systems*

PROJETO DE UM CONTROLADOR LQG PARA UM SISTEMA PNEUMÁTICO

Benedito Santana de Oliveira

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB,
Fax: 2167186, E – mail: bs@openline.com.br

Clivaldo Silva Araújo

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB,
Fax: 2167186, E – mail: clivaldo@ct.ufpb.br

Simplicio Arnaud da Silva

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB,
Fax: 2167186, E – mail: lacerdaarnaud@uol.com.br

José Antônio Riul

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB,
Fax: 2167186, E – mail: riul@ct.ufpb.com.br

Resumo. Este trabalho tem como objetivo utilizar técnicas de identificação de sistemas para determinar a função de transferência de um servoposicionador eletro - pneumático e projetar um controlador baseado na técnica Linear Quadrática Gaussiana (LQG). A identificação é realizada usando-se os modelos paramétricos de Box Jenkins e ARMAX, a partir de dados experimentais (entrada e saída) obtidos em uma bancada de testes. São apresentados resultados da resposta do servoposicionador utilizando-se a função de transferência, e estas são comparadas com resultados experimentais. O projeto do controlador LQG é realizado usando-se a função de transferência identificada. Finalmente, realiza-se simulação com o controlador LQG, e a resposta para o posicionamento do cilindro pneumático é comparada com respostas obtidas utilizando-se controladores clássicos.

Palavras - chave: Pneumático, válvula, identificação, controle, ótimo.

1. Introdução

O modelo matemático de um sistema pode ser obtido através da física do sistema. Este tipo de modelagem é conhecido como caixa branca. Uma outra maneira de se obter um modelo matemático é através da identificação, que é uma área de estudo de técnicas alternativas de modelagem matemática. Este tipo de modelagem é conhecido como caixa preta e não requer conhecimento prévio do sistema. Com a identificação pode-se obter diversos tipos de modelos para um mesmo sistema. O modelo escolhido deve possuir características do sistema real (Aguirre, 2000). O modelo identificado deve conter características dinâmicas do sistema. O objetivo deste trabalho é determinar um modelo, de um sistema servoposicionador, representado por um cilindro pneumático de dupla ação com haste simples e uma válvula eletro-pneumática proporcional, através de uma modelagem tipo caixa preta. Obtendo-se este modelo, é feito um projeto de um controlador baseado na técnica de controle ótimo LQG, “Linear Quadratic Gaussian”, de forma que a ação do mesmo em malha fechada, satisfaça a critérios de projeto posteriormente estabelecidos. São realizadas simulações usando o modelo sob a ação de controladores Proporcional (P), proporcional-derivativo (PD) e LQG. Estes resultados são comparados com os obtidos experimentalmente usando-se os controladores P e PD, projetados pelo método do ganho crítico (Hermerly, 1996).

2. Descrição do sistema a ser identificado e bancada de testes

O sistema a ser identificado é constituído de uma válvula eletro-pneumática proporcional de cinco vias, marca FESTO, tipo MPYE – 5 - 1/8HF – 010B, com pressão de suprimento de 0 a 10bar, sinal de excitação de 0 a 10V, e um cilindro de dupla ação com 25mm de diâmetro, haste simples de 200mm de curso, marca PARKER, tipo : P1A – R025DS –0200. Para obter-se os dados a serem utilizados na identificação, montou-se uma bancada de testes, composta por um computador PC – K6II – 500, com 32Mb de memória RAM; um compressor SCHULZ, modelo : MS –2,6/60; um filtro processador de ar com duas saídas reguláveis de pressão, marca Ar Air Fil 1 e uma régua potenciométrica da FESTO, modelo NLO – POT – 225 – TFL, para captar o sinal do deslocamento do pistão. A Figura (1) mostra a montagem do experimento. A interface entre o computador e o sistema é realizada através de uma placa de entrada e saída de dados, marca Quatech, modelo daq.801 e do programa computacional LabView.

3. Identificação do sistema

Na obtenção dos dados para identificação do sistema, a válvula eletro-pneumática é excitada em malha aberta com um sinal do tipo:

$$u = \left(\frac{A}{0.25t} \right) \text{sen}(0.25t) + 5 \tag{1}$$

onde A é uma constante, t é o tempo e u é o sinal de excitação em Volt.

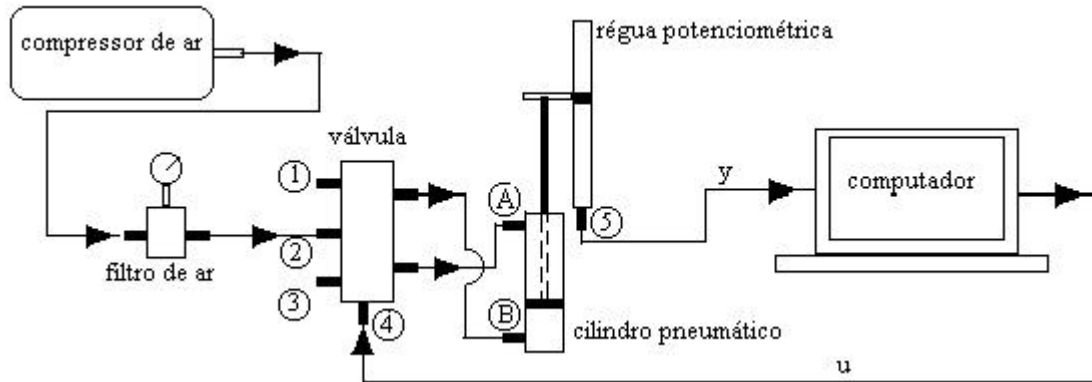


Figura 1. Esquema da bancada de testes para o controle do servoposicionador.

O centro da válvula corresponde a uma tensão de excitação de 5V. Tensão abaixo de 5V implica que a pressão na câmara A é maior que na câmara B, provocando um deslocamento do pistão no sentido de A para B, conforme mostrado na Fig. (1). Caso contrário, o deslocamento do pistão é no sentido oposto. As curvas de resposta do sistema e de excitação da válvula estão mostradas na Fig. (2) e na Fig. (3) respectivamente. Estas curvas foram obtidas com uma pressão de suprimento de 400 KPa, para diversas faixas de amplitude A. Na identificação do sistema, utilizou-se o pacote de identificação do MATLAB, usando os modelos BJ (Box Jenkins model), e o ARMAX, que são técnicas de identificação paramétricas (Ljung, 1987). Ambas as técnicas convergiram para um sistema de terceira ordem. Verificou-se que os modelos seguiam o modelo linear de Virvalo (Santos, 1996 e Vieira, 1998), definido na Eq. (2), onde ω_n é a frequência natural, ξ é o fator de amortecimento e K é o ganho.

$$G(s) = \frac{K \omega_n^2}{s(s^2 + 2\xi\omega_n s + \omega_n^2)} \tag{2}$$

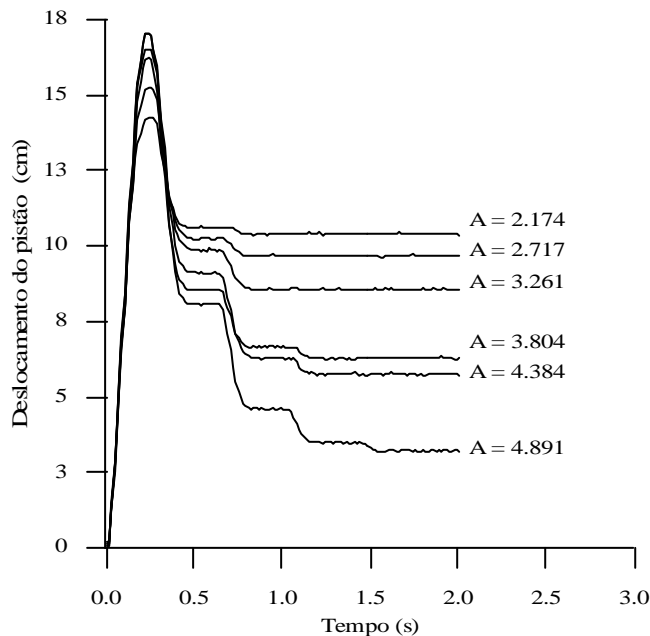


Figura 2. Resposta do sistema em malha aberta em função da variação da amplitude de excitação.

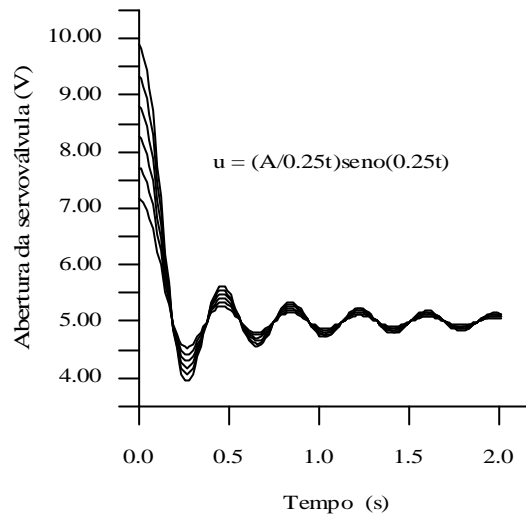


Figura 3. Sinal de excitação da válvula para diversas faixas de amplitude A.

Após a seleção do modelo para representar o sistema, deve-se fazer uma sintonia do mesmo, que são ajustes nos seus parâmetros para obter-se uma melhor aproximação entre a resposta obtida através da planta identificada e a obtida experimentalmente (Aguirre, 2000). Posteriormente, deve-se fazer a validação, que é a verificação se modelo ajustado tem uma resposta similar a do sistema real quando submetido a outro sinal de excitação. Realizada a sintonia, obteve-se a seguinte função de transferência para o sistema:

$$G(s) = \frac{31237,5}{s(s^2 + 32,13s + 1274,49)} \tag{3}$$

De acordo com o modelo de Virvalo, tem-se : $\omega_n = 35.7\text{rad/s}$; $\xi = 0.45$ e $K = 24.5$.

A Figura (4) mostra a resposta em malha aberta do sistema identificado e a experimental, usando-se como excitação a função representada pela Eq. (1), com $A = 2,174$, conforme mostrado na Fig. (5).

Para validar o sistema, comparou-se a resposta do servoposicionador obtida experimentalmente, num circuito de malha fechada, sob a ação de um controlador proporcional projetado pelo método do ganho crítico, com a resposta do modelo atuando em malha fechada, sob ação do mesmo controlador. A entrada de referência utilizada foi do tipo degrau, com amplitude de 15cm. O ganho do controlador utilizado foi $K_p = 0.6$. A Figura (6) mostra a resposta do modelo e a do sistema sob a ação deste controlador. Estes resultados demonstram que o modelo apresenta características dinâmicas similares as do sistema real, podendo-se desta forma adotá-lo como modelo matemático para representar o servoposicionador.

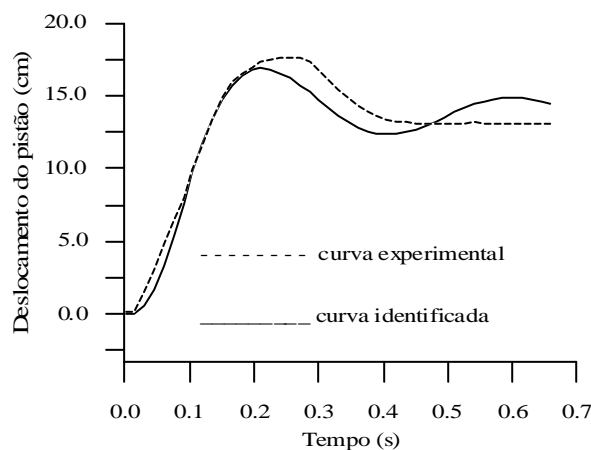


Figura 4. Resposta do modelo e do sistema em malha aberta, após a sintonia.

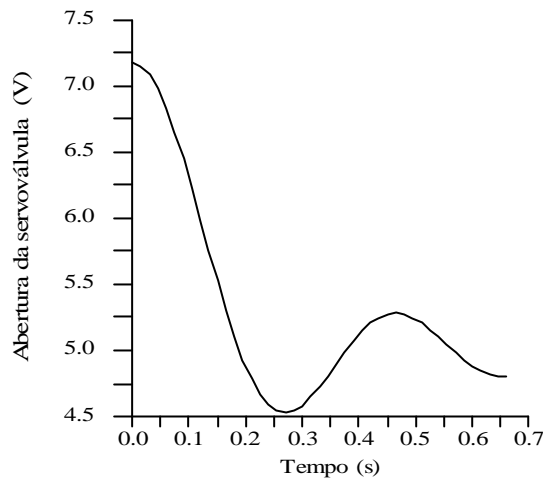


Figura 5. Sinal de excitação na válvula utilizado na sintonia do modelo.

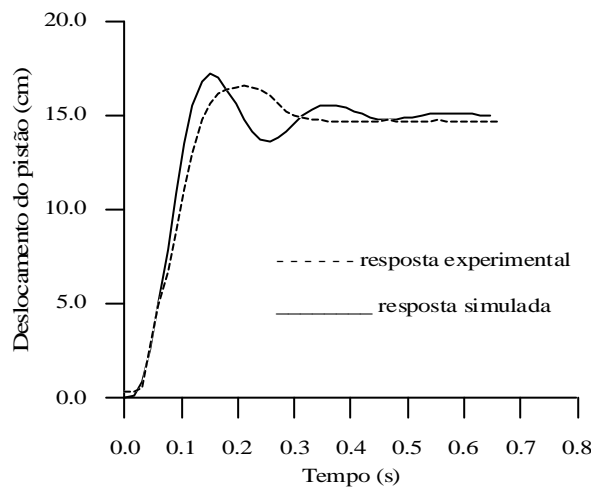


Figura 6. Resposta do modelo e do sistema real, a uma entrada degrau, sob a ação de controlador proporcional.

4. Projeto do controlador ótimo LQG

O projeto de um controlador LQG é um problema de controle ótimo estocástico. Esta técnica foi introduzida na década de sessenta considerando-se que a planta é um sistema linear, na forma de espaço de estado, cujo objetivo é determinar uma lei de controle que minimize o valor esperado de um índice de desempenho quadrático. Considera-se que tanto os estados quanto a saída são afetados por ruídos brancos gaussianos, de média nula e não correlacionados entre si, (Cruz, 1996 e Bryson et all, 1987) . A Figura (7) mostra a estrutura deste controlador, onde G é o ganho de realimentação de estados e H é o ganho do observador.

As equações que definem o problema são:

$$\dot{x} = Ax + Bu + \Gamma_{wx} \tag{4}$$

$$\dot{\hat{x}} = A \hat{x} + Bu + Hv \tag{5}$$

$$y = Cx \tag{6}$$

$$e_y = y + \Phi_{wy} \tag{7}$$

$$v = -e_y - Cx \tag{8}$$

$$u = -Gx \tag{9}$$

onde: Γ_{wx} e Φ_{wy} são o ruído de entrada e de saída na planta respectivamente.

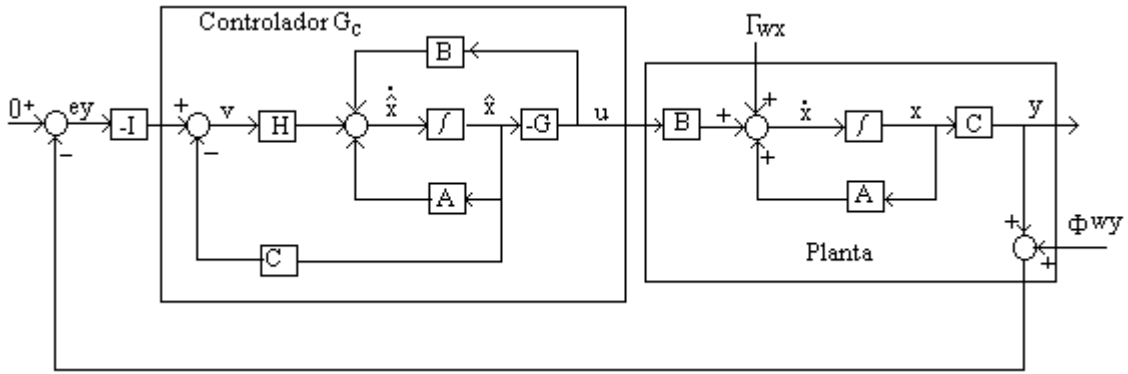


Figura 7. Estrutura do controlador G_c , projetado pela teoria de controle ótimo LQG, em série com a planta.

As variáveis \hat{x} são as variáveis de estado estimadas da planta. A matriz de ganho G é obtida a partir da solução da equação de Riccati da forma:

$$A^t P + PA + Q - PBR^{-1}B^tP = 0 \tag{10}$$

onde: a matriz de penalização de estados Q é positiva semi-definida e a de penalização de controle R é uma matriz positiva definida, escolhidas de forma a atenderem as especificações de projeto.

A matriz Q pode ser selecionada a princípio como definida pela Eq. (11) abaixo:

$$Q = C^t C \tag{11}$$

A matriz de ganho G é determinada a partir da seguinte equação:

$$G = R^{-1}B^tP \tag{12}$$

onde P é a solução da Eq. (10).

Com a determinação de G , obtém-se um regulador linear quadrático, com realimentação de estados \hat{x} no controlador. Esta solução requer que o par (A, B) seja controlável.

A matriz H é determinada de forma similar, porém deve-se considerar que os ruídos w_x e w_y na entrada e saída da planta sejam brancos gaussianos, de média nula e não correlacionados entre si, ou seja:

$$E\{w_x(t)\} = 0 \text{ e } E\{w_y(t)\} = 0 \tag{13}$$

$$E\{w_x(t)w_x(t + \tau)\} = Q_0 \delta(t - \tau) \tag{14}$$

$$E\{w_y(t)w_y(t + \tau)\} = R_0 \delta(t - \tau) \tag{15}$$

$$E\{w_x(t)w_y(t + \tau)\} = 0 \text{ para } t \text{ e } \tau \tag{16}$$

A matriz Q_0 é positiva semi-definida e representa a densidade espectral do ruído nos estados. A matriz R_0 é positiva definida e representa a densidade espectral do ruído de medida. Este problema consiste na determinação de um estimador ótimo de estados, de forma que o erro de estimativa $e_x = x - \hat{x}$ seja minimizado. Isto requer que o par (A, C) seja observável. A princípio pode-se considerar $Q_0 = I$, matriz identidade e $\Gamma = B$, e variar R_0 até que o erro e_x seja minimizado de forma satisfatória. A matriz de ganho H é determinada pela Eq. (17) abaixo:

$$H = \Psi C^t R^{-1} \tag{17}$$

onde Ψ é a solução da equação algébrica de Riccati, conforme Eq. (18):

$$A\Psi + \Psi A^t + \Gamma Q_0 \Gamma^t - \Psi C^t R_0 C\Psi = 0 \tag{18}$$

A função de transferência do controlador após a determinação das matrizes de ganho G e H é definida da seguinte forma:

$$G_c(s) = G(sI - A + BG + HC)H \tag{19}$$

Para uma entrada de referência $y_r \neq 0$, sugere-se que na Fig. (7), o vetor de estados estimados \hat{x} seja substituído por um outro vetor, como por z , que possui a mesma dimensão de x do modelo nominal. Neste caso, fazendo-se uma transformação de similaridade da forma $\Omega = x - z$, as equações de números (4) a (9), podem ser escritas nas seguintes formas:

$$\begin{Bmatrix} \dot{\hat{x}} \\ \hat{U} \end{Bmatrix} = \begin{bmatrix} (A - BG) & BG \\ 0 & (A - HC) \end{bmatrix} \begin{Bmatrix} x \\ \hat{U} \end{Bmatrix} + \begin{Bmatrix} \tilde{A} \\ \tilde{A} \end{Bmatrix} w_x + \begin{Bmatrix} 0 \\ -H \end{Bmatrix} w_y + \begin{Bmatrix} 0 \\ H \end{Bmatrix} y_r \quad (20)$$

$$\begin{Bmatrix} y \\ u \end{Bmatrix} = \begin{bmatrix} C & 0 \\ -G & G \end{bmatrix} \begin{Bmatrix} x \\ \Omega \end{Bmatrix} \quad (21)$$

A estabilidade do sistema nominal consiste em garantir que : $\text{Re}[\lambda_i(A - BG)] < 0$ e $\text{Re}[\lambda_i(A - HC)] < 0$.

Devido as limitações no sinal de excitação da servoválvula , onde $0 \leq u \leq 10V$, o projeto do controlador no domínio do tempo foi elaborado de forma que o sistema em malha fechada, sob a ação do controlador LQG, e para uma entrada degrau de amplitude de 15cm, tivesse um tempo de acomodação no máximo igual 0.5s e não apresentasse sobre – elevação.

O modelo do sistema, representado na forma de função de transferência é mostrado na forma de espaço de estados através da Eq. (22) e da Eq. (23).

$$\begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{Bmatrix} = \begin{bmatrix} -35.7 & -127.5 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} + \begin{Bmatrix} 1 \\ 0 \\ 0 \end{Bmatrix} u \quad (22)$$

$$y = \begin{bmatrix} 0 & 0 & 31237.5 \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix} \quad (23)$$

No projeto do controlador LQG, as matrizes de ganho G e H podem ser determinadas separadamente devido ao princípio da separação, (Cruz, 1996 e Shahian et all, 1993). Neste projeto, A matriz H foi determinada considerando-se uma entrada de referência $y_r = 0$, e os ruídos w_x e w_y foram gerados no MATLAB de forma que fossem brancos gaussianos com densidades espectrais $Q_0 = 10$ e $R_0 = 1,5$ respectivamente. As matrizes de penalização de estados e de controle, foram consideradas iguais a: $Q = I_{3 \times 3}$ e $R = 1$. No projeto considerou-se : $\Gamma = B$ e $\Phi = 1$. Com esses valores, tem-se $H = [-0.3009 \ 0.0316 \ 0.0014]^t$.

A matriz de ganho G foi determinada considerando-se uma entrada de referência y_r igual a um degrau com amplitude de 15cm, mantendo-se as matrizes Q_0 e R_0 constantes e os ruídos w_x e w_y foram desprezados. O melhor resultado para a matriz de ganho G foi obtido considerando-se, $Q = C^t C$, onde $C = [0 \ 0 \ 31237.5]$ e $R = [1.5]$. Com estes valores obteve-se $G = [25.01 \ 1188.23 \ 36018.21]$. Com G e H determinados, a função de transferência do controlador LQG , definida na Eq.(19) é da seguinte forma:

$$G_c(s) = \frac{146.08s^2 + 4913.66s + 166971.46}{s^3 + 121.40s^2 + 8127.17s + 302939.64} \quad (24)$$

5. Simulações e resultados

Para fins de comparação de resultados, serão mostradas curvas utilizando-se controladores P e PD, (Ogata, 1982 e D’Azzo et all, 1978), cujos ganhos foram obtidos experimentalmente pelo método do ganho crítico e curvas simuladas considerando a função de transferência da Eq. (3). Nos resultados experimentais e nos simulados, utilizou-se um controlador proporcional com um ganho $K_P = 0.6$ e um controlador PD com os seguintes parâmetros: $K_P = 0.72$ e $K_D = 0.01584$.

A Figura (8) mostra a resposta do modelo e do sistema real sob a ação de um controlador proporcional, para uma entrada de referência do tipo degrau, com amplitude de 10cm. Verificou-se que para um critério de 5% do valor final, o tempo de acomodação do modelo e do sistema foram iguais a 0.3s e 0.4s, respectivamente. O modelo apresentou um sobre-sinal de 13.5% ,enquanto o sistema de 7%. O tempo de subida do modelo e do sistema foram iguais a 0.09s e 0.125s, respectivamente, ambos para um critério de 5% a 95% do valor final.

A resposta do modelo e do sistema real sob a ação do controlador PD está mostrada na Fig. (9). Verificou-se que o tempo de acomodação do modelo e do sistema foram iguais a 0.23s e 0.2s, respectivamente. Apenas o modelo apresentou um sobre-sinal de 4%. O tempo de subida do modelo e do sistema foram iguais a 0.08s e 0.16s, respectivamente.

Os resultados da simulação demonstraram que o sistema sob a ação do controlador PD tem um resposta melhor que sob a ação de um controlador P. Os resultados experimentais confirmaram esta observação como se pode ver na Fig.(10). O sistema sob a ação do controlador P apresentou um erro no regime permanente de 3%.

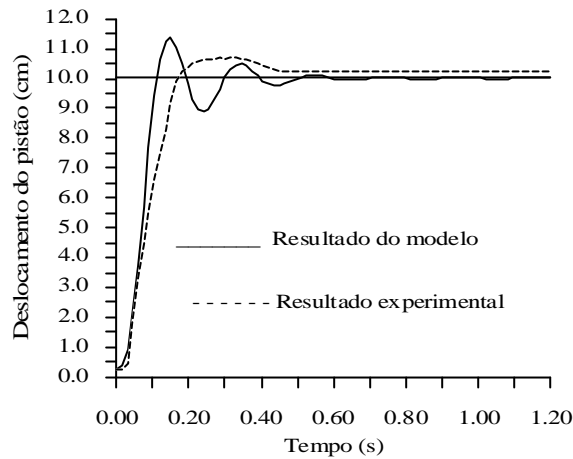


Figura 8. Resposta do modelo e do sistema real, a uma entrada de referência degrau, sob ação de um controlador P.

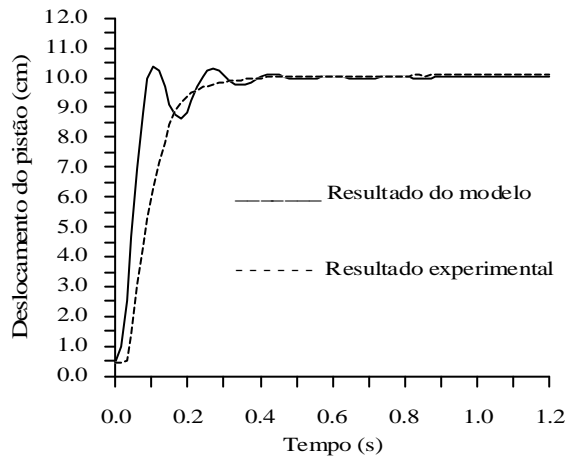


Figura 9. Resposta do modelo e do sistema real, a uma entrada de referência degrau, sob ação de um controlador PD.

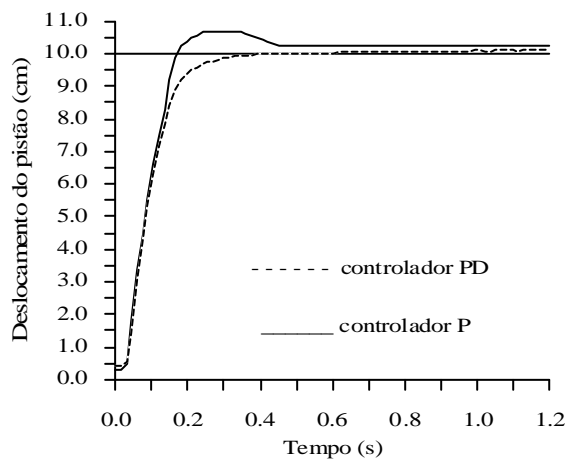


Figura 10. Resposta do sistema real, a uma entrada de referência degrau, sob ação dos controlador P e PD.

A Figura (11) mostra a resposta do modelo sob a ação dos controladores P, PD e LQG, para uma entrada de referência do tipo degrau com uma amplitude de 10cm. O sistema sob a ação do controlador LQG apresentou um tempo de subida de 0.175s e de acomodação igual a 0.19s, ausência de sobre-sinal. Os resultados das simulações demonstram que o sistema sob a ação do controlador LQG apresenta um bom desempenho em relação aos demais controladores, apesar de apresentar um tempo de subida maior que o sistema sob a ação dos controladores clássicos utilizados.

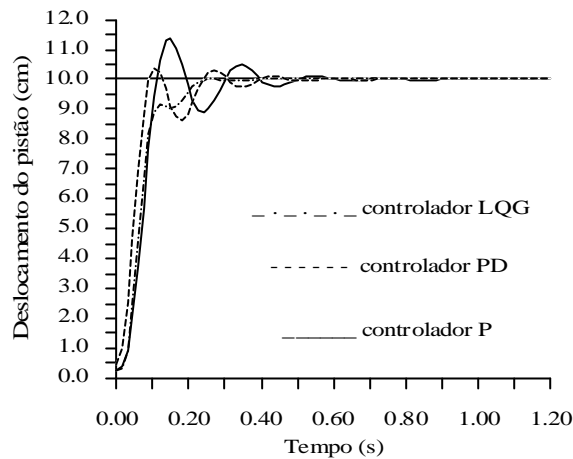


Figura 11. Resposta do modelo a uma entrada de referência degrau, sob a ação dos controladores P, PD e LQG.

6. Conclusões

De acordo com os resultados apresentados, conclui-se que o modelo identificado possui características dinâmicas similares a do sistema real, podendo o mesmo ser utilizado para projetos de controladores. Os resultados experimentais, comparados com os simulados, apresentaram diferenças em regime permanente e transitório, observadas com o uso dos controladores convencionais. Estas diferenças devem-se às incertezas tais como: atrito; amortecimento; zona morta da válvula e outros efeitos físicos não representados pelo modelo. Através das simulações realizadas, verificou-se ainda que o sistema apresentou um desempenho superior quando foi utilizado o controlador LQG, portanto, espera-se obter bom desempenho do sistema real quando este controlador for implementado.

7. Referências

- Aguirre, L. A., 2000, "Introdução à Identificação, Técnicas Lineares e Não – Lineares Aplicadas a Sistemas Reais", Ed. UFMG, Belo Horizonte, Brasil.
- Bryson, A.E., Ho, Y., 1987, "Applied Optimal Control, Optimization, Estimation and Control", Ed. Taylor & Francis, USA.
- Cruz, J. J., 1996, "Controle Robusto Multivariável", Ed. Universidade de São Paulo, São Paulo, Brasil.
- D'Azzo, J. J., Houpis, C. H., 1978, "Análise e Projeto de Sistemas de Controle Lineares", Ed. Guanabara Dois, Rio de Janeiro, Brasil.
- Hemerly, E. M., 1996, "Controle Por Computador de Sistemas Dinâmicos", Ed. Edgard Blucher LTDA, S. Paulo, Brasil.
- Ljung, L., 1987, "System Identification: Theory for the User", Ed. Prentice – Hall, Inc, Englewood Cliffs, New Jersey.
- Ogata, K., 1982, "Engenharia de Controle Moderno", Ed. Prentice Hall do Brasil, Rio de Janeiro, Brasil.
- Santos, E. A. P., 1987, "Análise Teórico Experimental de um Posicionador Pneumático", Dissertação de mestrado, UFSC, Santa Catarina, Brasil.
- Shahian, B., Hassul, M., 1993, "Control System Design Using Matlab", Ed. Prentice Hall, New Jersey.
- Vieira, A. D., 1998, "Análise Teórico – Experimental de Servoposicionadores Lineares Pneumáticos", Dissertação de Mestrado Submetida à Universidade Federal de Santa Catarina, Florianópolis, Brasil.

PROJECT OF A CONTROLLER LQG FOR A PNEUMATIC SYSTEM

Benedito Santana de Oliveira

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, E – mail: bsa@openline.com.br

Clivaldo Silva Araújo

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, E – mail: clivaldo@ct.ufpb.br

Simplicio Arnaud da Silva

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, E – mail: lacerdaarnaud@uol.com.br

José Antônio Riul

Departamento de Tecnologia Mecânica – CT – UFPB – Campus I - CEP: 58059 – 900 – João Pessoa – PB, Fax: 2167186, E – mail: riul@ct.ufpb.com.br

Abstract. *This work is mainly using system identification technique to find a transfer function of a servopositioner electro - pneumatic and to project a controller based in Linear Quadratic Gaussian (LQG) technique. The identification is realized using the parametric models Box Jenkins and ARMAX, from the experimental datas (input and output) found in a test bed. The output results of the servopositioner are presented making use of the transfer function and this are compared with the experimental results. The LQG controller design is made using the identify function transfer. Finally, the simulation with the controller LQG is realized, and the output to the pneumatic cylinder position is compared with obtained answers being used classical controllers.*

Keywords *Pneumatic, valve, identification, control, optimal*

O CONTROLE EM CASCATA DE UM SISTEMA PNEUMÁTICO DE POSICIONAMENTO

Raul Guenther

Robotics Laboratory
Mechanical Engineering Department
Federal University of Santa Catarina - UFSC
88040 – 900 - Florianópolis - SC – Brazil
Caixa Postal 486
guenther@emc.ufsc.br

Eduardo A. Perondi

Robotics Laboratory – UFSC
perondi@emc.ufsc.br

Resumo. *O propósito deste trabalho é contribuir no desenvolvimento de controladores para sistemas pneumáticos de posicionamento que possibilitem melhorar seu desempenho dinâmico e, conseqüentemente, viabilizem sua maior aplicação em tarefas onde as exigências de velocidade e precisão são severas. Com base em uma modelagem adequada, e a partir de uma discussão das dificuldades de controle dos sistemas pneumáticos de posicionamento, propõe-se uma estratégia de controle em cascata para superar diversas dificuldades dinâmicas estruturais destes sistemas de posicionamento. Apresentam-se as propriedades de estabilidade decorrentes dessa estratégia e a sua comprovação experimental.*

Palavras chave: *controle de posicionadores, controle em cascata, pneumática, robótica.*

1. Introdução

Os sistemas que possibilitam posicionar uma carga mecânica em uma dada localização definida por uma coordenada “y” são chamados de sistemas de posicionamento. Esta localização pode ser fixa ou variável no tempo, ou seja, $y(t)$, e nestes casos os sistemas de posicionamento são, algumas vezes, chamados de seguidores.

Os sistemas de posicionamento são utilizados em diversas aplicações dentre as quais pode-se citar laminadores, turbinas hidráulicas, máquinas agrícolas, lemes de aeronaves, antenas, telescópios, direcionadores de foguetes, máquinas de precisão e robôs manipuladores.

Um componente importante do sistema é o elemento que aplica a força sobre a carga mecânica para leva-la até a posição desejada. Este elemento é chamado de motor ou acionamento e pode ser elétrico, hidráulico ou pneumático. Uma discussão sobre as características e aplicações de cada um desses motores pode ser encontrada em Bollmann e Guenther (1997). Aqui, discutem-se as características dos sistemas com acionamento pneumático, doravante denominados sistemas pneumáticos de posicionamento, compostos basicamente por uma servoválvula que controla a vazão de ar na direção de um cilindro, a cujo êmbolo está ligada a massa que se deseja posicionar.

Os sistemas de posicionamento pneumático são relativamente baratos, leves e de fácil instalação, apresentando também uma boa relação entre a força aplicada e o seu peso. Isso os torna bastante atraentes para diversas aplicações.

A estrutura dinâmica do sistema com parcelas oscilatórias introduzidas pela compressibilidade do ar, as não linearidades associadas ao escoamento do ar na servoválvula e ao atrito entre o êmbolo e a camisa do cilindro em conjunção com os atrasos devidos à baixa velocidade de propagação das ondas de pressão no ar, tornam os sistemas pneumáticos de posicionamento difíceis de serem controlados de forma a obter-se o desempenho necessário para algumas tarefas de posicionamento.

O objetivo de desenvolver sistemas pneumáticos de posicionamento rápidos e precisos tem motivado diversos trabalhos em controle destes sistemas (Bobrow e McDonnell, 1998, Perondi e Guenther, 1999a, 1999b, 2000b). Busca-se superar os problemas inerentes aos sistemas pneumáticos através de técnicas de controle adequadas.

Este trabalho é desenvolvido dentro desta linha de pesquisa. Propõe-se aqui uma estratégia de controle que permite superar as não linearidades associadas ao escoamento do ar na servoválvula, mantém limitadas as oscilações devido à compressibilidade do ar e cria um caminho para enfrentar as dificuldades introduzidas pelo atrito entre o êmbolo e a camisa do cilindro.

A estratégia consiste em interpretar o sistema de posicionamento como dois subsistemas interconectados: um subsistema mecânico comandado por uma lei de controle capaz de garantir sua estabilidade na presença de perturbações, e um subsistema pneumático onde, após uma linearização por realimentação, é empregada uma lei de controle proporcional.

Após a descrição do sistema na seção 2, apresenta-se, na seção 3, sua modelagem dinâmica, utilizada no projeto de controlador e na discussão das características de controle dos sistemas de posicionamento pneumáticos apresentada na seção 4. Na seção 5 descreve-se o controlador em cascata e apresentam-se suas propriedades de estabilidade. Os resultados experimentais são apresentados e comentados na seção 6. Na seção 7 delineiam-se algumas conclusões.

2. Sistema pneumático de posicionamento

O sistema em estudo está ilustrado na Fig. (1).

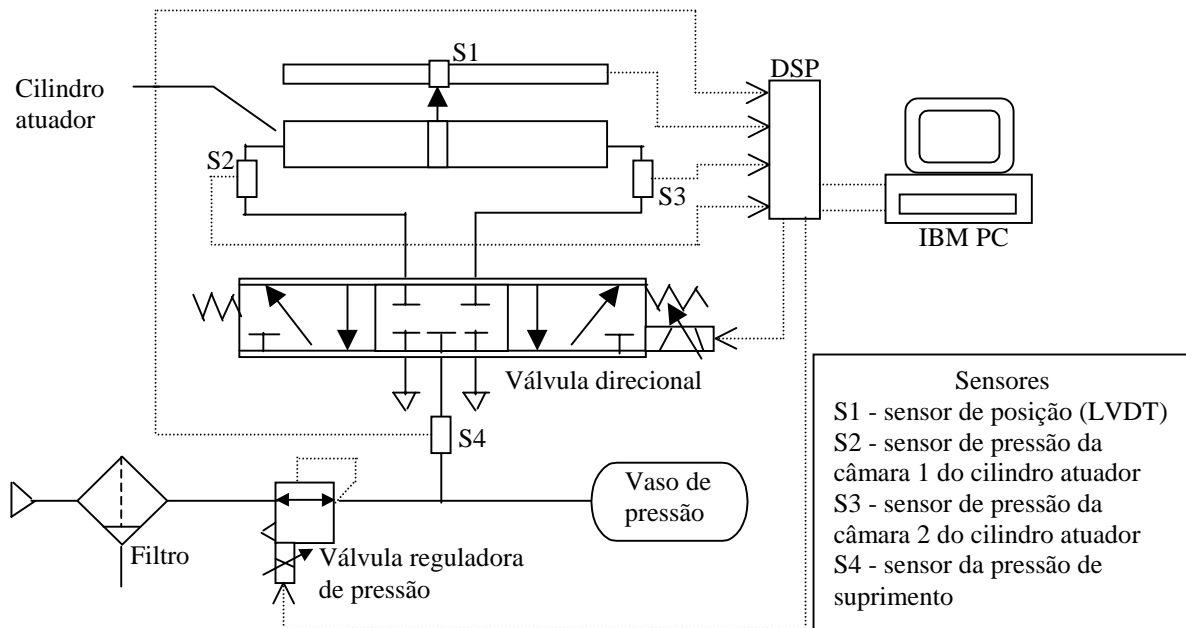


Figura 1. Esquema da bancada experimental.

Uma servoválvula proporcional 5/3 (MPYE-5-1/8 FESTO) opera um cilindro de dupla ação com transmissão de movimento por fita com diâmetro interno de 0,025 m e curso 1 m (DGPL-1000 FESTO).

A válvula possui vazão nominal (medida) $Q_N = 7.10^{-3} \text{ m}^3/\text{s}$ (420 l/min) independente da pressão de suprimento devido a uma realimentação interna. Com isso o comportamento da abertura da válvula em função da tensão aplicada é aproximadamente linear. Um pequeno sobrepassamento (*overlap*) ocasiona uma zona-morta de cerca de 4%. O êmbolo móvel do cilindro é conectado a um sensor linear (Festo POT-1000-TFL) de curso igual ao do cilindro atuador (1,0 m). Três sensores eletroresistivos de pressão medem a pressão absoluta em cada câmara do cilindro e a pressão manométrica de suprimento. O aparato de controle e aquisição de dados consiste em uma placa de aquisição de dados rápida (dSPACE GmbH – DS1102) acoplada a um microcomputador IBMPC DX266MHz com tempo de amostragem de 1 ms. Nas conversões AD e DA, a placa utilizado apresenta programas auxiliares para gerenciamento da aquisição de dados e módulos de acoplamento para o *MatLab/Simulink*. Assim, os algoritmos em tempo real de medições e controle estão programados via módulo *dSPACE* para o *MatLab/Simulink* (ver *dSPACE GmbH*, 1996).

3. O modelo dinâmico

O modelo para o posicionador pneumático é obtido a partir da descrição da vazão mássica em cada câmara do cilindro e do equilíbrio de forças no êmbolo. Neste trabalho, a descrição da vazão mássica de ar no cilindro é feita de acordo com Liu e Bobrow (1988). Seguindo este enfoque a relação entre a vazão mássica de ar e a variação da pressão na câmara 1 (ou câmara 2) do cilindro é estabelecida usando a conservação de energia. A energia interna da massa que entra na câmara 1 do cilindro é $C_p \dot{m}_1 T$, onde C_p é o calor específico do ar a pressão constante, T é a temperatura do ar de suprimento, $q_{m1} = (dm_1/dt)$ é a vazão mássica na câmara 1. A potência do movimento do pistão é $p_1 \dot{V}_1$, onde p_1 é a pressão na câmara 1 e $\dot{V}_1 = (dV_1/dt)$ é a vazão volumétrica. A variação na energia interna do volume de controle é $d(C_v \rho_1 V_1 T)/dt$, onde C_v é o calor específico do ar a volume constante e ρ_1 é a massa específica do ar. Definindo a relação de calores específicos $r = C_p/C_v$ e lembrando que para um gás ideal a equação da massa específica é $\rho_1 = C_v/(RT)$, sendo R a constante do gás, pode-se realizar o balanço energético, do qual resulta

$$q_{m1}T - \frac{p_1}{C_p} \frac{dV_1}{dt} = \frac{1}{rR} \frac{d}{dt} (p_1 V_1) \quad (1)$$

onde a potência calorífica dissipada através das paredes do cilindro (\dot{Q}) é desprezada. O volume total na câmara 1 é dado por $V_1 = Ay + V_{10}$, onde A é a área do êmbolo, y é o deslocamento do êmbolo e V_{10} é o volume morto na câmara 1 incluindo as tubulações. A taxa de variação deste volume é $\dot{V}_1 = A\dot{y}$, onde $\dot{y} = dy/dt$ é a velocidade do êmbolo. Com isso, resolvendo a Eq. (1) para \dot{p}_1 , após efetuar a derivada do termo da direita, chega-se a

$$\dot{p}_1 = \frac{Rr q_{m1} T - A p_1 \dot{y}}{A y + V_{10}} \quad (2)$$

onde foi utilizada a relação $C_p = (rR)/(r-1)$ (veja por exemplo Streeter, 1981). Sendo L o comprimento útil do cilindro atuador, obtém-se para a câmara 2,

$$\dot{p}_2 = \frac{Rr q_{m2} T - A p_2 \dot{y}}{A(L-y) + V_{20}} \quad (3)$$

Assumindo que as vazões mássicas são funções das pressões no cilindro e da tensão u aplicada à servoválvula, ou seja, $q_{m1} = q_{m1}(p_1, u)$ e $q_{m2} = q_{m2}(p_2, u)$, as expressões (2) e (3) resultam:

$$\dot{p}_1 = -\frac{A \dot{y}}{A y + V_{10}} p_1 + \frac{RrT}{A y + V_{10}} q_{m1}(p_1, u) \quad (4)$$

$$\dot{p}_2 = -\frac{A \dot{y}}{A(L-y) + V_{20}} p_2 + \frac{RrT}{A(L-y) + V_{20}} q_{m2}(p_2, u) \quad (5)$$

O equilíbrio de forças no êmbolo é obtido pela aplicação da 2ª lei de Newton, e resulta

$$M \ddot{y} + F_a + F_e = A(p_1 - p_2) \quad (6)$$

onde M é a massa deslocada, F_a é a força de atrito, F_e é a força externa e $A(p_1 - p_2)$ é a força resultante da diferença de pressão entre as câmaras (ver Fig. (2)).

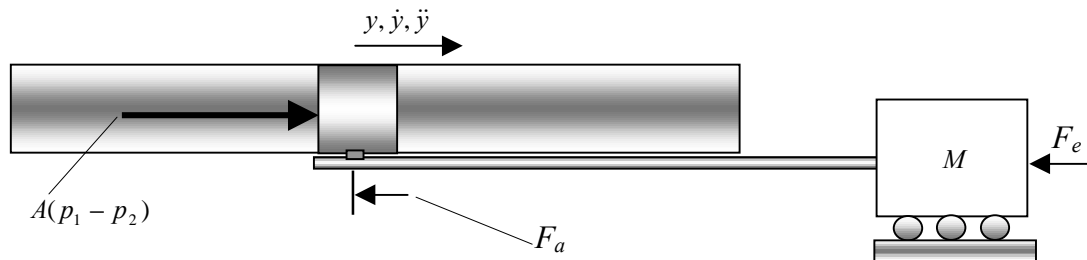


Figura 2. Equilíbrio de forças no êmbolo do cilindro atuador.

As equações (4), (5) e (6) constituem um modelo dinâmico de quarta ordem do sistema pneumático de posicionamento, cujas variáveis de estado podem ser $x = [y \ \dot{y} \ p_1 \ p_2]^T$. Este modelo tem não linearidades caracterizadas pelo produto e quociente de variáveis de estado presentes nas relações (4) e (5), pela não linearidade das funções que descrevem as vazões mássicas $q_{m1} = q_{m1}(p_1, u)$ e $q_{m2} = q_{m2}(p_2, u)$, e pela parcela de atrito estático incluída em F_a .

Observação 1 (modelo linear de quarta ordem): Expandindo as funções que descrevem as vazões mássicas em série de Taylor em torno de um ponto de funcionamento “i” obtém-se uma expressão linearizada para a vazão mássica que passa pelo orifício da servoválvula 1 dada por

$$\Delta q_{m1} = \left[\frac{\partial q_{m1}}{\partial u} \right]_i \Delta u + \left[\frac{\partial q_{m1}}{\partial p_1} \right]_i \Delta p_1 \quad (7)$$

onde o coeficiente $[\partial q_{m1}/\partial u]_i$ define o ganho de vazão na entrada da câmara 1 e o coeficiente $[\partial q_{m1}/\partial p_1]_i$ define o ganho de pressão nesta entrada. O índice “i” indica que a pressão p_1 e a tensão para o cálculo destes coeficientes são tomadas para o ponto de funcionamento “i”. Lembrando que este ponto de funcionamento inclui ainda uma posição $[y]_i$ e uma velocidade $[\dot{y}]_i$, pode-se linearizar a expressão (4) em torno deste ponto. O mesmo procedimento pode ser desenvolvido para a expressão (5).

As expressões assim obtidas a partir de (4) e (5) e a expressão (6), desconsiderando-se o atrito estático, constituem um modelo dinâmico linear de quarta ordem com um zero (Perondi e Guenther, 1999).

Observação 2 (modelo linear de terceira ordem): Se o ponto de funcionamento “i” é o ponto central do cilindro e os ganhos de pressão $[\partial q_{m1}/\partial p_1]_i$ e $[\partial q_{m2}/\partial p_2]_i$ são considerados iguais, o zero cancela um pólo, o sistema tem uma dinâmica não observável e a função de transferência resulta na forma (Boudart et al., 1992):

$$G(s) = \frac{b_0}{s(s^2 + a_2s + a_1)} \quad (8)$$

onde os pólos, para as características usuais, são reais (Perondi e Guenther, 2000b).

Observação 3 (modelo não linear de quarta ordem): Definindo $p_\Delta = p_1 - p_2$ como a diferença de pressão entre as câmaras 1 e 2, a expressão (6) resulta:

$$M\dot{y} + F_a + F_e = Ap_\Delta \quad (9)$$

Utilizando as expressões (4) e (5) pode-se escrever a derivada da diferença de pressão em relação ao tempo como

$$\dot{p}_\Delta = RrT \left[\frac{q_{m1}(p_1, u)}{Ay + V_{10}} - \frac{q_{m2}(p_2, u)}{A(L - y) + V_{20}} \right] - rA\dot{y} \left[\frac{p_1}{Ay + V_{10}} + \frac{p_2}{A(L - y) + V_{20}} \right] \quad (10)$$

Desta forma é possível agrupar os termos que dependem da tensão na servoválvula u em uma função $\hat{u} = \hat{u}(p_1, p_2, y, u)$, e os termos que independem da tensão em uma função $\hat{h} = \hat{h}(p_1, p_2, y, \dot{y})$:

$$\hat{u}(p_1, p_2, y, u) = RrT \left[\frac{q_{m1}(p_1, u)}{Ay + V_{10}} - \frac{q_{m2}(p_2, u)}{A(L - y) + V_{20}} \right] \quad (11)$$

$$\hat{h}(p_1, p_2, y, \dot{y}) = -rA\dot{y} \left[\frac{p_1}{Ay + V_{10}} + \frac{p_2}{A(L - y) + V_{20}} \right] \quad (12)$$

Com isso a expressão (10) resulta:

$$\dot{p}_\Delta = \hat{h}(p_1, p_2, y, \dot{y}) + \hat{u}(p_1, p_2, y, u) \quad (13)$$

As expressões (9) e (13) são uma outra forma de descrever o sistema não linear de quarta ordem.

4. O controle do sistema de posicionamento pneumático

Utilizando a técnica do lugar das raízes verifica-se que, com uma realimentação proporcional, dois pólos da malha fechada do modelo linear de terceira ordem do posicionador (8) tornam-se oscilatórios com um amortecimento que diminui a medida que o ganho proporcional aumenta.

Se o modelo considerado é o linear de quarta ordem (veja observação 1 acima), a presença do zero faz com que, com uma realimentação proporcional, dois pólos dirijam-se para o semiplano complexo direito à medida que o ganho de realimentação cresce. Ou seja, o sistema em malha fechada tende a ser instável com o crescimento do ganho.

As oscilações pouco amortecidas e o risco de instabilidade limitam o ganho proporcional a pequenos valores. Por isso, realimentado proporcionalmente, o sistema de posicionamento resulta lento em malha fechada. Os pequenos ganhos proporcionais também redundam em grandes erros provocados pelo atrito estático.

Analisando o lugar das raízes verifica-se também que a introdução de uma parcela derivativa na realimentação não altera significativamente as dificuldades estruturais apresentadas acima. Uma parcela integral pode auxiliar na redução do erro provocado pelo atrito, mas também não altera a posição dos pólos em malha fechada de forma a reduzir as oscilações pouco amortecidas e os riscos de instabilidade. Conseqüentemente, os controladores PID (Proporcional-Integral-Derivativo) são insuficientes quando as exigências da tarefa do sistema de posicionamento pneumático são severas.

Como o modelo linear de terceira ordem (Eq.(8)) é controlável, as dificuldades estruturais decorrentes do posicionamento dos pólos em malha aberta podem ser superadas utilizando uma realimentação de estados. Sendo controlável, os pólos do sistema em malha fechada podem ser alocados em qualquer posição do plano complexo, o que possibilita obter a resposta especificada através de algum critério de projeto. Na prática o desempenho é limitado pela

saturação da tensão de controle na servoválvula (u). Mesmo assim, o controlador com realimentação de estados possibilita velocidades de posicionamento mais elevadas que os controladores PID (Perondi e Guenther, 1999b).

Diversos autores desenvolvem controladores de estado para sistemas de posicionamento pneumático utilizando o modelo linear de terceira ordem (8) como base (Virvalo, 1995, Uebeling e Vaughan, 1997). Estes controladores são referenciados como controladores PVA por utilizarem medidas de posição (P), velocidade (V) e aceleração (A). A realimentação de aceleração compensa a força de atrito (Tafazoli et al, 1998, Perondi e Guenther, 1999a).

Uma dificuldade prática para a implementação do controlador PVA é a necessidade de dispor da aceleração. Sua medição é difícil, seu cálculo numérico é contaminado pelos ruídos do sinal de velocidade e sua obtenção através de filtros introduz erros. Por isso, utiliza-se muitas vezes um controlador de estados em que a aceleração é substituída pela diferença de pressão nas câmaras $p_{\Delta} = p_1 - p_2$, definida na observação 3 da seção anterior, e o controlador é chamado de PV Δ p (Surgenor et al, 1996, Perondi e Guenther, 1999b). Neste caso a alocação de pólos é apenas aproximada e um ajuste fino dos ganhos através de simulações torna-se necessário na prática (Perondi e Guenther, 1999a). Além disso, substituindo a realimentação da aceleração perde-se a compensação do atrito (Perondi e Guenther, 1999a)

Os controladores de estado (PVA e PV Δ p) são projetados no modelo linear de terceira ordem (8). Por isso, o projeto de seus ganhos depende do ponto de funcionamento "i". Utilizando os mesmos ganhos para pontos diferentes, os desempenhos são diferentes em cada ponto, podendo, inclusive, ocorrer instabilidades. Essa é uma forte limitação da aplicação dos controladores de estado nos sistemas de posicionamento pneumático. O controlador em cascata proposto na próxima seção tem por objetivo superar essa limitação.

5. O Controle em cascata

Nesta seção apresenta-se uma estratégia de controle em cascata desenvolvida a partir da metodologia de redução de ordem proposta por Utkin (1987), utilizada com sucesso no controle de robôs acionados eletricamente (Guenther e Hsu, 1993) e no controle de acionamentos hidráulicos (Guenther e De Pieri, 1997).

Segundo esta estratégia o posicionador pneumático é interpretado como dois subsistemas interconectados: um subsistema mecânico acionado por uma força gerada em um subsistema pneumático, como na Fig (3).

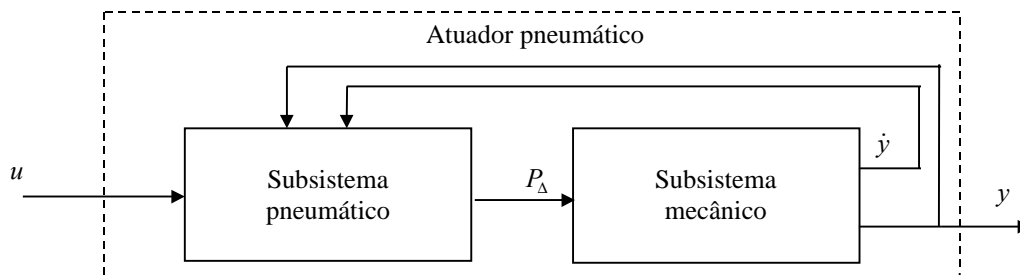


Figura 3. Interpretação do posicionador pneumático como dois subsistemas interconectados.

Neste trabalho essa interpretação é aplicada ao sistema não linear (9), (13) (veja observação 3 da seção 3), reescrevendo-o de forma conveniente. Para tanto, define-se o erro no seguimento da diferença de pressão como:

$$\tilde{p}_{\Delta} = p_{\Delta} - p_{\Delta d} \tag{14}$$

onde $p_{\Delta d}$ é a diferença de pressão desejada, a ser definida com base na força $g_d = Ap_{\Delta d}$ que deve ser aplicada à massa para obter um desempenho desejado no posicionamento.

Utilizando a definição (14) as expressões (9) e (13) podem ser reescritas como:

$$M\ddot{y} = Ap_{\Delta d} + A\tilde{p}_{\Delta} - F_a - F_e \tag{15}$$

$$\dot{p}_{\Delta} = \hat{h}(p_1, p_2, y, \dot{y}) + \hat{u}(p_1, p_2, y, u) \tag{16}$$

Este sistema está na forma em cascata. A expressão (15) pode ser interpretada como um subsistema mecânico de segunda ordem acionado por uma força desejada $g_d = Ap_{\Delta d}$ sujeito a uma perturbação na entrada $d = A\tilde{p}_{\Delta} - F_a - F_e$. A expressão (16) descreve o comportamento do subsistema pneumático de segunda ordem.

A estratégia em cascata consiste em:

- (i) - Projetar uma lei de controle (força desejada $g_d(t) = Ap_{\Delta d}(t)$) para o subsistema mecânico (15) de modo que a saída $y(t)$ siga uma trajetória desejada $y_d(t)$ na presença da perturbação $d = A\tilde{p}_{\Delta} - F_a - F_e$; e então

(ii) – Projetar uma lei de controle $u(t)$ para que o subsistema (16) gere uma força pneumática $g(t) = Ap_{\Delta}(t)$ que siga a força desejada $g_d(t) = Ap_{\Delta d}(t)$.

Neste trabalho a lei de controle para o subsistema mecânico $g_d(t)$ é baseada no controlador proposto em Slotine e Li (1988) e a lei de controle $u(t)$ é sintetizada de forma a obter um bom desempenho no seguimento da força no subsistema pneumático.

5.1 – Seguimento no subsistema mecânico

A lei de controle para obter o seguimento de trajetória no subsistema mecânico é dada por (Slotine e Li (1988):

$$g_d = M\dot{y}_r - K_D s \quad (17)$$

onde K_D é uma constante positiva, \dot{y}_r é uma velocidade de referência e “s” é uma medida do erro de seguimento da velocidade. De fato, \dot{y}_r pode ser calculada modificando a velocidade desejada \dot{y}_d de acordo com o que segue

$$\dot{y}_r = \dot{y}_d - \lambda \tilde{y} ; \tilde{y} = y - y_d ; s = \dot{y} - \dot{y}_r = \dot{\tilde{y}} + \lambda \tilde{y} \quad (18)$$

onde λ é uma constante positiva.

5.2 – Seguimento no subsistema pneumático

Para alcançar o seguimento da força no subsistema pneumático propõe-se a lei de controle:

$$\hat{u} = \bar{u} - K_p \tilde{p}_{\Delta} \quad (19)$$

onde \bar{u} é um sinal de controle auxiliar e K_p é uma constante positiva.

O projeto do sinal \bar{u} e da constante K_p é realizado com o objetivo de obter um sistema em malha fechada globalmente estável, e é realizado utilizando o método direto de Lyapunov (Guenther e Perondi, 2001). A constante K_p define velocidade como que o erro na diferença de pressão e, conseqüentemente, o erro na força aplicada, convergem para zero. O sinal auxiliar \bar{u} resulta

$$\bar{u} = \dot{p}_{\Delta d} - \hat{h}(p_1, p_2, y, \dot{y}) \quad (20)$$

5.3 – Controle do sistema de posicionamento

O controlador em cascata do sistema pneumático de posicionamento é obtido pela combinação da lei (17) para o seguimento no subsistema mecânico com a lei (19) para o subsistema pneumático.

Utilizando (17) calcula-se a diferença de pressão desejada para obter o seguimento da trajetória no subsistema mecânico, $p_{\Delta d} = g_d / A$.

Empregando (19) e (20) calcula-se \hat{u} . Para tanto necessita-se obter a derivada da diferença de pressão, calculada anteriormente, e a função $\hat{h} = \hat{h}(p_1, p_2, y, \dot{y})$ calculada através de (12).

Calculado o valor de \hat{u} a tensão de controle é obtida pela inversão da função \hat{u} definida em (11), ou seja,

$$u = u(p_1, p_2, y, \hat{u}) \quad (21)$$

Essa inversão é descrita no apêndice A.

A utilização da função $\hat{h} = \hat{h}(p_1, p_2, y, \dot{y})$ e da inversa definida em (21) na lei de controle, pode ser interpretada como a utilização de uma linearização por realimentação (veja Slotine e Li, 1991, por exemplo).

5.4 – Propriedades de estabilidade da malha fechada

As propriedades de estabilidade da malha fechada formada pelo sistema de posicionamento (9)(13) e pelo controlador em cascata (17)(19), podem ser caracterizadas através do vetor erro de seguimento deste sistema, aqui definido por $\rho = \begin{bmatrix} \tilde{y} & \dot{\tilde{y}} & \tilde{p}_{\Delta} \end{bmatrix}^T$. Empregando o método direto de Lyapunov, Guenther e Perondi (2001) demonstram as seguintes propriedades:

- (i) – Quando os parâmetros do sistema de posicionamento pneumático são conhecidos, desconsiderando as forças externa F_e e de atrito F_a , o vetor de erros ρ , definido acima, converge exponencialmente para zero.
- (ii) – Quando os parâmetros são conhecidos, na presença de forças externa F_e e de atrito F_a , o vetor de erros $\rho(t)$ converge globalmente para um conjunto residual que depende dos ganhos do controlador.
- (iii) – As incertezas nos parâmetros do sistema são mais uma fonte de erro e ampliam o conjunto residual para o qual o vetor $\rho(t)$ converge.

6. Resultados experimentais

O controlador em cascata dado pelas equações (17) e (19) foi implementado no sistema de posicionamento pneumático descrito na seção 2. Para o cálculo da força desejada g_d (17), foi utilizada uma massa $M = 2,9Kg$.

Os sinais de velocidade e aceleração foram obtidos derivando numericamente a posição. Além disso, por serem ruidosos, os sinais de posição, velocidade, aceleração e pressão, são filtrados antes de serem utilizados no controlador. As frequências de corte dos filtros passa baixa utilizados foram, respectivamente, 40, 30, 25 e 50 rad/s. Os ganhos do controlador (K_d, λ, K_p) foram ajustados até o limite em que a resposta apresentou oscilações de alta frequência, resultando em $K_d = 70$, $\lambda = 13$ e $K_p = 500$. Com isso, para uma trajetória senoidal $y_d(t) = 0,45 \text{sen}(t)$ (veja linha vermelha na Fig. (5.a)) obteve-se a trajetória $y(t)$ (indicada pela linha azul na mesma figura). A Fig. (5.a) mostra também o erro de seguimento $\tilde{y}(t)$ em verde, que pode ser visto em detalhe na Fig. (5.b).

Na Fig. (6) estão apresentadas força desejada (linha vermelha) e força aplicada (linha azul) e o erro de seguimento de força (linha verde). Os resultados comprovam o seguimento da força desejada no subsistema pneumático, validando a linearização por realimentação proposta assim como sua implementação experimental.

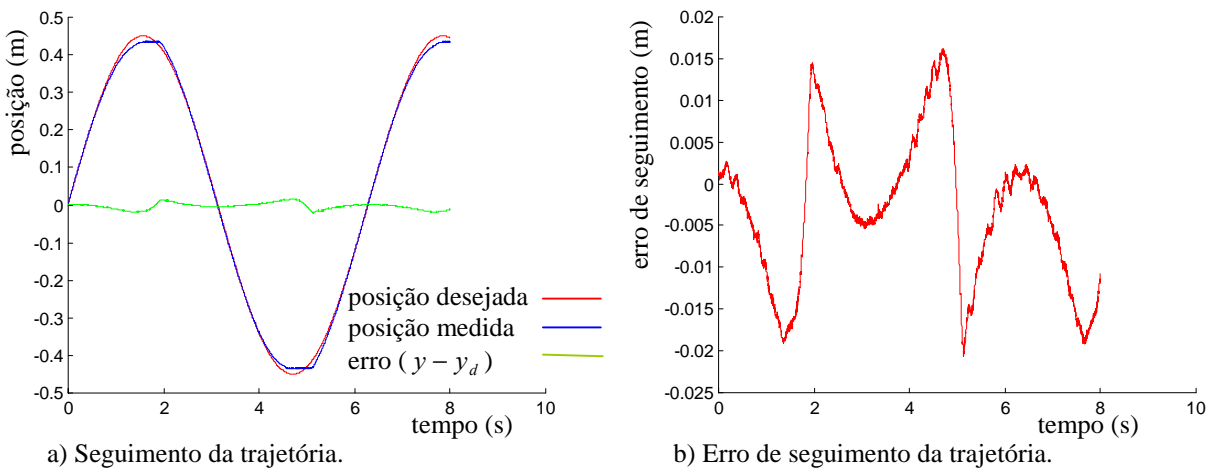


Figura 5. Seguimento da trajetória de posição.

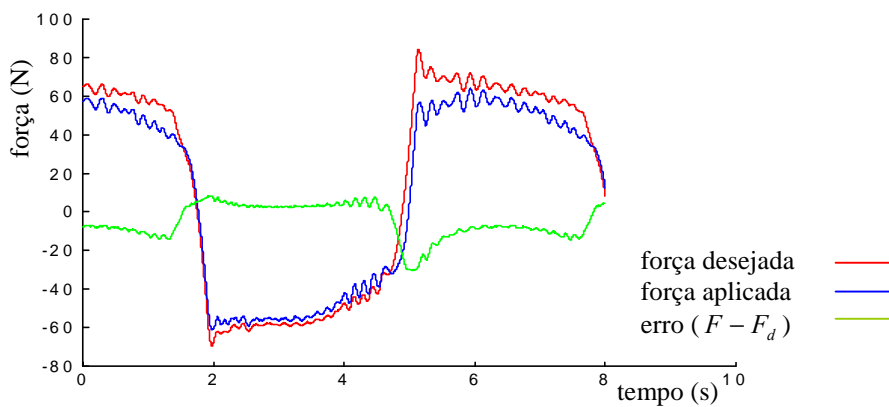


Figura 6. Seguimento da força no subsistema pneumático.

Os erros de seguimento de trajetória mostrados na Fig.(5) demonstram o bom desempenho da estratégia em cascata. Nos testes experimentais foi observado que, mesmo para altas velocidades, com os ganhos utilizados não ocorrem oscilações de alta frequência presentes quando da utilização de controladores PID.

Cabe assinalar ainda que, utilizando um controlador de estados $PV\Delta p$, e ajustando seus ganhos até o limite em que a resposta passa a apresentar oscilações de alta frequência, o desempenho de seguimento é semelhante. Isso ocorre porque o controlador em cascata é um controlador de estados que tem uma linearização por realimentação.

Realizando experimentos variando a massa M e a frequência da trajetória senoidal desejada (portanto, a velocidade e a aceleração desejadas), verificou-se que, para um mesmo ajuste de ganhos, o desempenho do controlador de estados $PV\Delta p$ é sensível a essa variação de parâmetros e que o controlador em cascata mantém o desempenho praticamente inalterado com essa variação. Isso comprova a superação das características introduzidas pelas não linearidades associadas ao escoamento do ar na servoválvula.

7. Conclusões

Apresenta-se uma proposta de controlador para sistemas pneumáticos de posicionamento, baseada numa estratégia em cascata. A estratégia em cascata é desenvolvida a partir de uma modelagem adequada e de uma interpretação do sistema pneumático de posicionamento como um subsistema mecânico interconectado com um subsistema pneumático. Para o subsistema pneumático propõe-se uma lei de controle que utiliza a vazão mássica medida experimentalmente. Apresenta-se uma metodologia experimental para a medição da vazão mássica. Com as leis de controle propostas para os dois subsistemas a estabilidade resulta global. Os resultados experimentais obtidos demonstram a eficácia da estratégia na superação das linearidades associadas ao escoamento de ar na servoválvula, e na limitação das oscilações devido à compressibilidade do ar.

Além disso, a estratégia proposta cria um caminho para a compensação do atrito utilizando uma metodologia que faça essa compensação diretamente no subsistema mecânico como em Canudas et al (1999).

8. Referências

- Bobrow, J.E., McDonell, W., 1998, "Modeling, Identification, and Control of a Pneumatically Actuated, Force Controllable Robot", IEEE Transactions on Robotics and Automation, vol.15, n° 5, pp. 732-742.
- Bollmann, A., Guenther, R., 1997, "Posicionadores Hidráulicos e Pneumáticos: Características e Técnicas de Controle". Anais do 5º Seminário Nacional de Hidráulica e Pneumática, SENAI/CTAI. Florianópolis, SC, BR.
- Boudart, S., Richard, E., Scavarda, S., 1991, "A Modified Linear Reduced Observer for a Pneumatic Servodrive". FluidPower System, Modeling and Control. Proceedings of Fourth Bath Int. Fluid Power Workshop, pp. 357-371.
- dSPACE Digital Signal Processing and Control Engineering, 1996, GmbH, DS1102 User's Guides, Germany.
- Fox, R.W., McDonald, A.T., 1981, "Introdução à Mecânica dos Fluidos". Ed. Guanabara Dois S.A. Rio de Janeiro-RJ.
- Guenther, R., Hsu, L., 1993, "Variable Structure Adaptive Cascade Control of Rigid-Link Electrically-Driven Robot Manipulators". Proc. IEEE 32nd CDC, San Antonio, Texas, December, pp.2137-2142.
- Guenther, R., De Pieri, E.R., 1997, "Cascade Control of the Hydraulic Actuators". Revista Brasileira de Ciências Mecânicas, vol. 19, n° 2, Junho, pp. 108-120.
- Guenther, R., Perondi, E.A., 2001, "Controle em Cascata de Servoposicionadores Pneumáticos", Relatório Técnico, Laboratório de Robótica, Centro de Tecnologia, Universidade Federal de Santa Catarina, Florianópolis, SC, Brasil
- Liu, S., Bobrow, J., 1988, "An Analysis of a Pneumatic Servo System and its Application to a Computer-controlled Robot". ASME Journal of Dynamic Systems, Measurement and Control, Vol. 110.
- Lischinsky, P., Canudas-de-Wit, C., e Morel, G., 1999, "Friction Compensation for na Industrial Hydraulic Robot", IEEE Control Systems, February, pp.25-32.
- Perondi, E.A., Guenther, R., 1999a, "Controle em Malha Fechada de um Servoposicionador Pneumático". Anais do XV Congresso Brasileiro de Engenharia Mecânica – COBEM99. Águas de Lindóia, SP, Brasil.
- Perondi, E.A. e Guenther, R., 1999b, "Controle com Compensação de Atrito de um Servoposicionador Pneumático", anais do IV Congreso Iberoamericano de Ingeniería Mecánica, CIDIM '99 Santiago, Chile.
- Perondi, E.A. e Guenther, R., 2000a, "Controle de um Servoposicionador Pneumático por Modos Deslizantes", Congresso Nacional de Engenharia Mecânica - CONEM/2000, Natal, Brasil.
- Perondi, E.A. e Guenther, R., 2000b, "Control of a Servopneumatic Drive with Friction Compensation" 1st FPNI – PhD Symposium, Fluid Power Net International, Hamburgo,.
- Slotine, J.J.E., Li, W., 1991, "Applied Nonlinear Control", Prentice-Hall Inc.
- Streeter, V. L., 1981, "Mecânica dos Fluidos", McGraw-Hill, São Paulo.
- Surgenor, B. W., Vaughan, N. D. and Uebing, M., 1995, "Continuous Sliding Mode Control of a Pneumatic Positioning System". Design and Performance, Proceedings of Eighth Bath Int. Fluid Power Workshop, pp.270-285.
- Tafazoli, S., Silva, C. W., Lawrence, P.D., 1998, "Tracking Control of an Eletrohydraulic Manipulator in the Presence of Friction". IEEE Transaction on Control Systems Technology, vol. 6, pp 401-411.
- Uebing, M., Vaughan, M.D., Surgenor, B.W., 1997, "On Linear Dynamic Modeling of a Pneumatic Servo System". The fifth Scandinavian International Conference on Fluid Power, SICFP'97, Linköping, Sweden, pp. 363-378.
- Utkin, V.I., 1987, "Discontinuous Control Systems: State of Art in Theory and Applications". IFAC World Congress on Automatic Control. Preprints, Munich, v.1, pp75-94.
- Virvalo, T., 1995, "Modeling and Design of a Pneumatic Position Servo System Realized with Commercial Components". PhD Thesis, Tampere, Finland.
- Wikander, J. 1988, "Adaptive Control of Pneumatic Cylinder". Dr. Thesis, Stockholm.

Apêndice A – Inversão da função \hat{u}

A inversão da função $\hat{u} = \hat{u}(p_1, p_2, y, u)$ é realizada a partir do levantamento experimental das funções que descrevem as vazões mássicas $q_{m1} = q_{m1}(p_1, u)$ e $q_{m2} = q_{m2}(p_2, u)$.

Diversos autores (Wikander, 1988, Virvalo, 1995, Maré et al, 2000) assumem que a vazão mássica através da válvula ocorre de forma semelhante à vazão mássica de ar no orifício de uma restrição situada no interior de um tubo, onde a área do orifício (A_o) varia de acordo com a tensão de controle u , ou seja, $A_o = A_o(u)$. Empregando essa hipótese a vazão mássica teórica (sem considerar perdas) é determinada através de uma expressão na forma:

$$q_m = A_o(u)p_e\Psi(p_s/p_e) \quad (A1)$$

onde p_e é a pressão a montante do orifício, p_s é a pressão a jusante, e a função $\Psi(p_s/p_e)$ tem um valor constante para $p_s/p_e \leq 0,528$, característico do fluxo saturado, e um valor dependente da relação de pressões quando o fluxo é subsônico ($p_s/p_e > 0,528$) (veja Fox e MacDonald, 1981, por exemplo).

Essa descrição não leva em conta a geometria efetiva dos orifícios, desconsiderando os fenômenos relacionados às perdas e irreversibilidades que ocorrem na passagem do gás pelas aberturas. Essas perdas fazem com que as relações de pressões críticas sejam menores na prática (Maré et al, 2000). Além disso, Bobrow e McDonnell (1998) mostram que, mesmo ajustando a abertura efetiva e a relação de pressões críticas, a expressão para a vazão subsônica pode não ser suficientemente precisa para descrever a vazão mássica na servoválvula.

Observando a expressão (A1) verifica-se que a vazão mássica é dada pelo produto de uma função que depende da tensão na servoválvula (u) por uma função das pressões a montante e a jusante. Com base nisso, utiliza-se neste trabalho, como em Bobrow e McDonnell (1998), a descrição da vazão mássica na servoválvula como o produto de funções da tensão $f_u(u)$ e das pressões $f_p(p_e, p_s)$, na forma:

$$q_m(p_e, p_s, u) = [q_m]_{\max} f_p(p_e, p_s) f_u(u) \quad (A2)$$

onde $[q_m]_{\max}$ é a vazão mássica máxima através do orifício, $0 \leq f_p(p_e, p_s) \leq 1$ é a função das pressões, e $0 \leq f_u(u) \leq 1$ é uma função que considera a abertura efetiva e depende da tensão na servoválvula u .

Nos sistemas pneumáticos considera-se geralmente que:

- 1 – A pressão de suprimento (p_{sup}) é constante; e
- 2 – A pressão de exaustão é igual à pressão atmosférica (p_{atm}).

Desta forma, no enchimento de um cilindro a pressão a montante é $p_e = p_{\text{sup}}$, a pressão a jusante é a pressão p_i na câmara “i” do cilindro, e a vazão mássica de enchimento da câmara “i” (q_{mi}^{ench}) escrita a partir de (A2) resulta

$$q_{mi}^{\text{ench}}(p_i, u) = [q_{mi}^{\text{ench}}]_{\max} f_{pi}^{\text{ench}}(p_i) f_{ui}^{\text{ench}}(u) \quad (A3)$$

onde $f_{pi}^{\text{ench}}(p_i)$ é a função de pressão no enchimento da câmara “i”, e $f_{ui}^{\text{ench}}(u)$ é a função de tensão correspondente ao enchimento da câmara “i”.

Na exaustão a pressão a montante é a pressão no interior da câmara “i”, ou seja, $p_e = p_i$, e a pressão a jusante é a pressão atmosférica ($p_s = p_{\text{atm}}$). Assim, a vazão mássica de exaustão da câmara “i” resulta

$$q_{mi}^{\text{exau}}(p_i, u) = [q_{mi}^{\text{exau}}]_{\max} f_{pi}^{\text{exau}}(p_i) f_{ui}^{\text{exau}}(u) \quad (A4)$$

onde $f_{pi}^{\text{exau}}(p_i)$ é a função de pressão na exaustão da câmara “i”, e $f_{ui}^{\text{exau}}(u)$ é a função de tensão na exaustão da câmara “i”.

Procedimento experimental

O levantamento experimental das funções de pressão ($f_{pi}^{\text{ench}}(p_i)$ e ($f_{pi}^{\text{exau}}(p_i)$), e das funções de tensão ($f_{ui}^{\text{ench}}(u)$ e $f_{ui}^{\text{exau}}(u)$) é feito observando que, quando o êmbolo do cilindro está parado, o volume nas câmaras é constante (V) e a velocidade do êmbolo é nula ($\dot{y} = 0$). Com isso, de (4) ou de (5) pode-se escrever

$$q_{mi}[p_i(t), u] = \frac{V}{RrT} \dot{p}_i(t) \tag{A5}$$

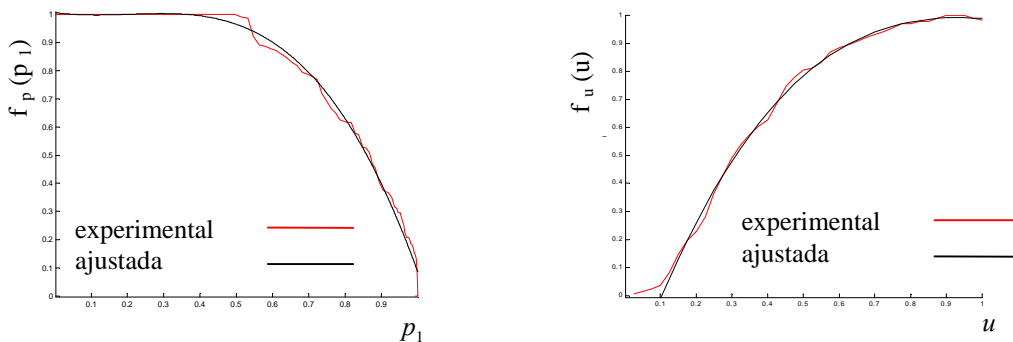
O procedimento experimental consiste em:

- 1 – Medir o comportamento da pressão na câmara ao longo do tempo ($p_i(t)$), no enchimento e na exaustão para uma seqüência de valores de tensão na válvula;
- 2 – Calcular a derivada da pressão em relação ao tempo, obtendo a vazão mássica em função do tempo através de (A5), para cada um dos valores de tensão na válvula;
- 3 – Utilizar o comportamento da pressão na câmara ao longo do tempo $p_i(t)$, medido no passo 1, na expressão (A5) para obter a vazão mássica $q_{mi}(p_i, u)$ para cada um dos valores de tensão na válvula.
- 4 – Os valores máximos de vazão em cada câmara são iguais a $[q_{mi}]_{\max}$ (de enchimento ou exaustão);
- 5 – A razão entre a vazão mássica calculada no passo 3 pelo respectivo valor máximo (passo 4) mantendo-se a tensão constante fornece a função de pressão correspondente;
- 6 – A razão entre a vazão mássica calculada no passo 3 pelo respectivo valor máximo, determinado no passo 4, mantendo a pressão constante fornece a função de tensão correspondente.

Ajuste de curvas

Utilizando o método dos mínimos quadrados ajustaram-se polinômios de terceira ordem tanto para as funções de pressão quanto para as funções de tensão (veja resultados na Fig. 7).

Para as funções de pressão esse ajuste facilita o cálculo e para as funções de tensão o ajuste possibilita a inversão através do cálculo das raízes do polinômio.



a) Curvas normalizadas vazão versus pressão com vazão entrando na câmara 1.

b) Curvas normalizadas vazão versus tensão com vazão entrando na câmara 1.

Figura 7. Curvas experimentais para o caso da vazão entrando na câmara 1.

CASCADE CONTROL OF A PNEUMATIC SERVODRIVE

Raul Guenther

Robotics Laboratory
 Mechanical Engineering Department
 Federal University of Santa Catarina - UFSC
 88040 – 900 - Florianópolis - SC – Brazil
 Caixa Postal 486
 guenther@emc.ufsc.br

Eduardo A. Perondi

Robotics Laboratory – UFSC
 perondi@emc.ufsc.br

Abstract. *In this paper the problem of pneumatic positioning system control is adressed. The pneumatic positioning systems controllers are developed to improve the dynamic performance of pneumatic drives in a way that they could be applied to precise and fast tasks. With the knowledge about pneumatic servodrives control difficulties and using an appropriate system model, we propose a cascade control strategy to overcome several structural dynamic problems of this kind of system. We present the cascade control stability properties and its experimental confirmation.*

Keywords. Positioning drives control, cascade control, pneumatics, robotics.

ANÁLISE E DESENVOLVIMENTO EXPERIMENTAL DE UMA TRAÇADORA PARA DESENHAR EM SUPERFÍCIE VERTICAL

Solly A. Segenreich.

Pontifícia Universidade Católica do Rio de Janeiro – PUC-RIO

Rua Marquês de São Vicente 225 – Gávea – RJ – CEP: 22453-900

e-mail: solly@mec.puc-rio.br

Jorge Luiz Fontanella

Universidade Católica de Petrópolis

Estrada Presidente Sodr , 302 BL16/206 - Sim ria – Petrópolis – CEP: 25645-710

e-mail: jorge.fontanella@ucp.br

Resumo. *O presente trabalho concerne o desenvolvimento de um prot tipo de traçadora gr fica para desenhar sobre uma superf cie r gida na posiç o vertical. A principal aplicaç o desta traçadora   no segmento de comunicaç o visual para an ncios em grandes superf cies. Com a intenç o de atingir alto grau de flexibilidade nesta utilizaç o, a traçadora   concebida para trabalhar em coordenadas polares, dispensando um quadro fixo composto de v rias guias lineares como seria o caso numa traçadora x, y. Este trabalho discute o layout b sico, a construç o do prot tipo, a cinem tica e a estrutura do programa de acionamento. Resultados experimentais ainda preliminares s o apresentados com foco especial na movimentaç o do braço. A parte final do trabalho inclui sugest es para melhoria do projeto proposto.*

Palavras chave: *Traçadora de coordenadas polares, Traçadora gr fica computadorizada, Plotter.*

1.Introduç o.

A constante necessidade de traçagem de grandes imagens para aplicaç o em propaganda, tem levado os fabricantes a desenvolver traçadoras de grandes capacidades. A forma construtiva destas m quinas quase sempre segue o mesmo padr o, traçando sobre superf cie horizontal ou sobre papel em rolo. Com a intenç o de desenvolver e analisar uma traçadora para impress o em superf cie vertical, capaz de imprimir sobre espaços de dimens es relativamente grandes (do tamanho de um outdoor por exemplo), foi projetado e constru do um prot tipo que   objeto deste trabalho.

O conceito adotado para construç o desta traçadora, levou em conta que a estrutura deveria ser simples, leve e com suportaç o em uma  nica base, visando facilitar o transporte, instalaç o e manuseio do equipamento e que permitisse que a pena traçadora alcance grandes dist ncias.

A id ia de usar apenas um pilar como apoio de toda traçadora, permite utilizar, por exemplo, um ve culo tipo camioneta, que al m de servir para transportar o braço, serve como base de sustentaç o da traçadora quando em uso no campo. Por m, neste trabalho n o ser o abordados estes detalhes.

Seguindo o conceito adotado, foi constru do um braço mec nico, contendo uma guia linear, com um ponto de rotaç o fixo em uma das extremidades. O dispositivo porta-caneta corre sobre esta guia linear. Assim, a traçadora   posicionada segundo coordenadas polares r e ϕ como mostrado na fig. 1.

A movimentaç o do braço e do carro   executada por dois motores de passo. Adotou-se esta opç o pois barateia e simplifica todo o processo de controle e acionamento. Uma inconveni ncia para o uso dos motores de passo   a vibraç o que eles apresentam, e acabam transmitindo para toda estrutura. Para eliminar estas vibraç es, foi adotado um acionamento para os motores operando em regime de micropasso, Segenreich et al (1997,2000).

O controle da traçadora   executado a partir de um microcomputador, via porta paralela. Al m da construç o do prot tipo, foi desenvolvido como parte integrante do trabalho, um programa em C para o controle da traçadora.

Rob s e plotters utilizando coordenadas polares tem sido amplamente investigadas na literatura e utilizadas em algumas aplicaç es. Um trabalho que utiliza um mecanismo pantogr fico para ampliar a  rea  til de trabalho foi apresentado por Segenreich (1991,1993)

2.Layout b sico do dispositivo

A fig.(1), ilustra diagramaticamente o layout da traçadora. O braço est  ligado a uma  rvore no ponto O. O movimento angular ϕ , do braço,   produzido pelo atuador n mero 1. J  o carro porta-pena, desloca-se sobre o braço, movimentado pelo atuador n mero 2, que est  preso a uma extremidade do braço. A coordenada r , representa a dist ncia do carro porta-pena ao ponto de rotaç o O.

Este layout atende a id ia pela qual este trabalho originou-se. Tendo um  nico ponto de apoio, o braço pode ser colocado sobre uma base e ser manuseado em campo com facilidade, eliminando estruturas mais complexas que precisariam de espaços maiores e mais peças para montagem.

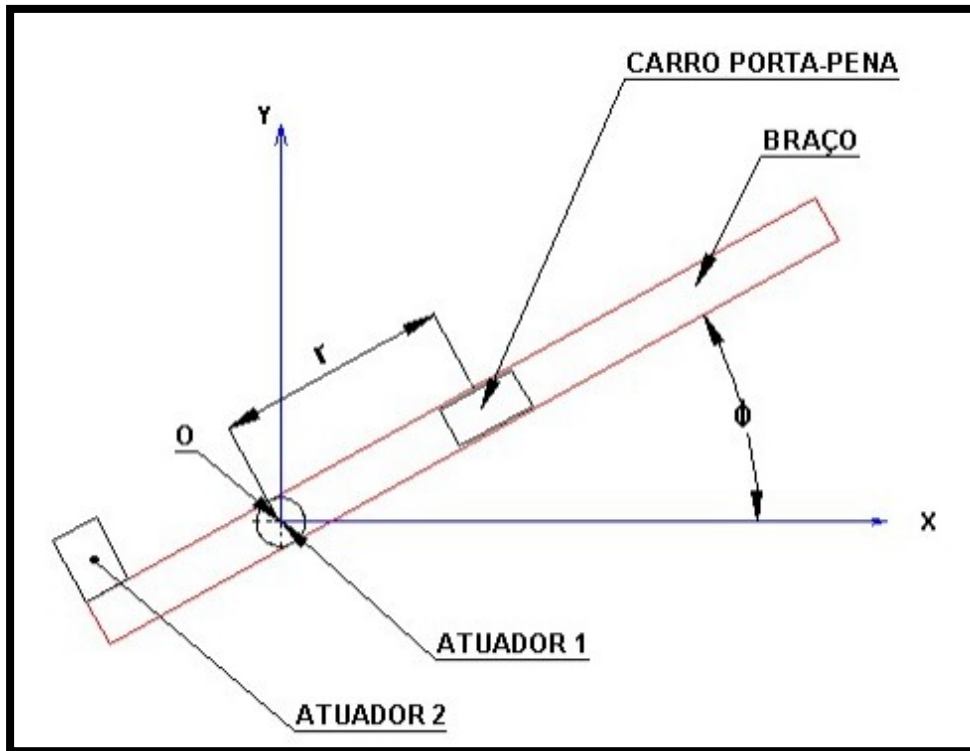


Figura 1. Layout básico da traçadora.

As dimensões do braço do protótipo, são indicadas na fig. (2). Adotaram-se essas dimensões para poder atingir uma área de traçagem que cobrisse todo o quadro que serve de superfície de traçagem.

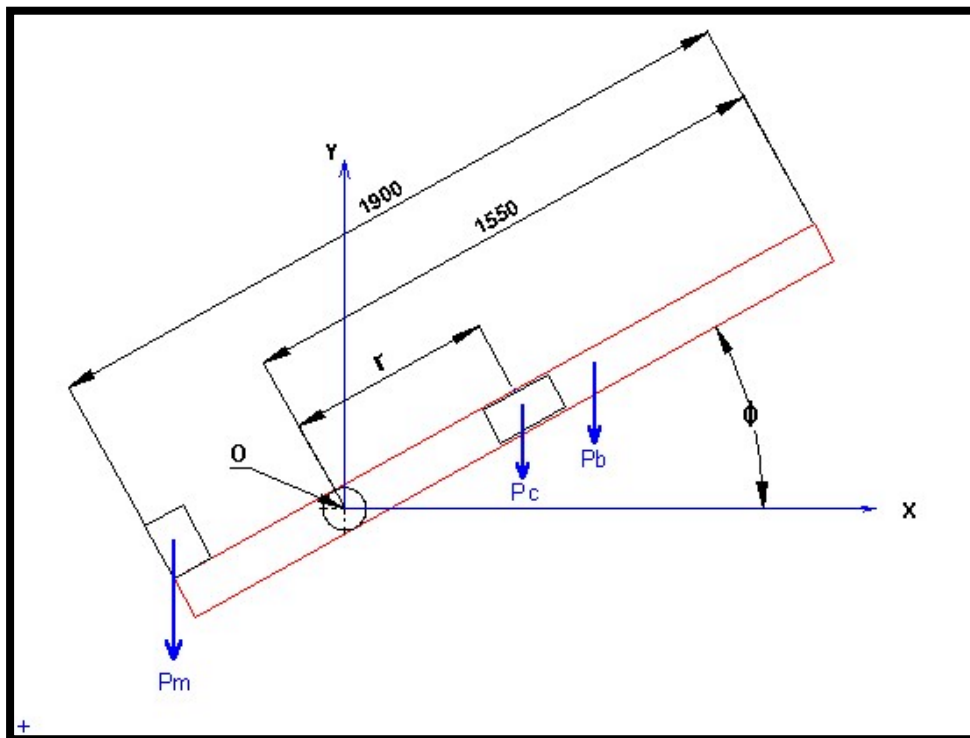


Figura 2. Dimensões e resultantes de peso dos principais componentes (unidades em milímetro).

Os pesos dos componentes móveis mostrados na fig. (2), estão abaixo relacionados.

P_m = peso do motor $\Rightarrow 18,0$ N

P_c = peso do carrinho $\Rightarrow 6,0$ N
 P_b = peso do braço $\Rightarrow 23,0$ N

Como estes pesos determinam momentos diferentes em relação a O de acordo com a inclinação do braço e a posição do carro, consideramos a situação mais crítica, com o carro na extremidade e o braço na horizontal, para determinar o momento estático máximo do sistema ($(M_o)_{max} = 20,82$ Nm).

3. Equações cinemáticas para a trajetória da pena.

Consideremos a pena em x_1, y_1 , como mostrado na fig. (3). Este ponto é determinado em coordenadas polares pelo ângulo ϕ e pelo raio r

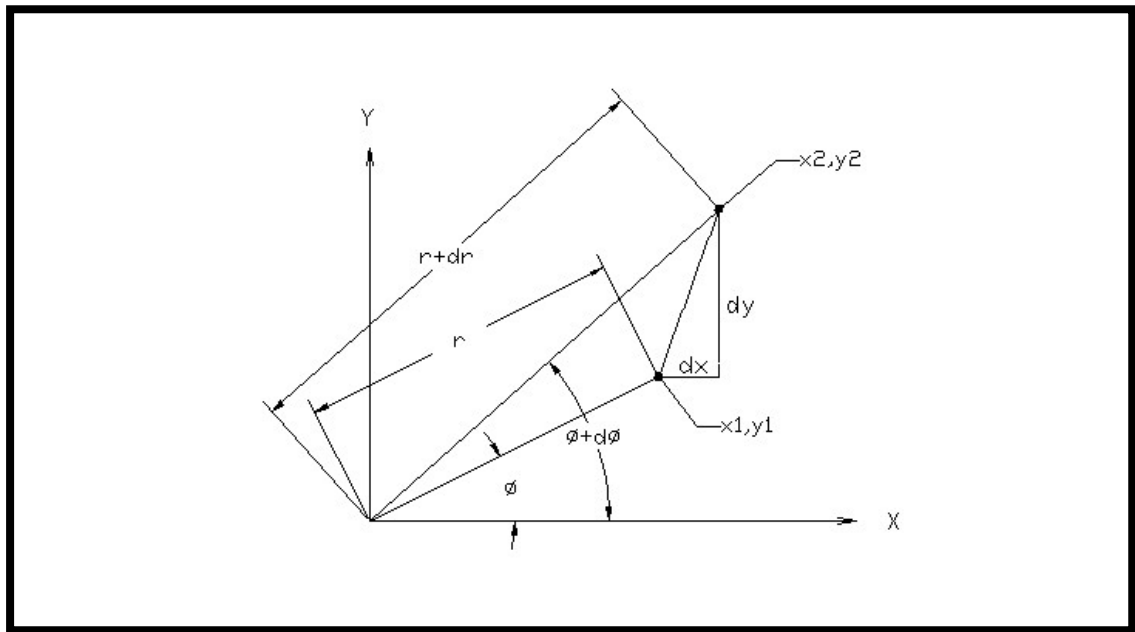


Figura 3. Trajetória da pena para um deslocamento infinitesimal.

Seja um deslocamento infinitesimal dx, dy . Queremos determinar $d\phi$ e dr que provocam este deslocamento. Assim,

$$x = r \cos \phi \tag{1}$$

$$y = r \sin \phi \tag{2}$$

Diferenciando,

$$dx = dr \cos \phi - r d\phi \sin \phi \tag{3}$$

$$dy = dr \sin \phi + r d\phi \cos \phi \tag{4}$$

Resolvendo o sistema acima,

$$dr = \cos \phi \cdot dx + \sin \phi \cdot dy \tag{5}$$

$$d\phi = (dy \cdot \cos \phi - dx \cdot \sin \phi) / r \tag{6}$$

Substituindo nas Eq. (5) e na Eq. (6) os valores de $\cos \phi, \sin \phi$ e r , tem-se:

$$\cos \phi = \frac{x}{\sqrt{x^2+y^2}}$$

$$\sin \phi = \frac{y}{\sqrt{x^2+y^2}}$$

$$r = \sqrt{x^2 + y^2}$$

$$dr = \frac{x}{\sqrt{x^2 + y^2}} dx + \frac{y}{\sqrt{x^2 + y^2}} dy \quad (7)$$

$$d\phi = \frac{x}{x^2 + y^2} dy - \frac{y}{x^2 + y^2} dx \quad (8)$$

Pelas equações (7) e (8) podemos determinar dr e $d\phi$, que são diretamente controlados pelos atuadores, em função de um deslocamento dx , dy desejado.

As equações (7) e (8) são não lineares, já que dr e $d\phi$ dependem da própria posição dada por x e y . Assim, para traçar um segmento de reta, este deve ser dividido em pequenos segmentos elementares para os quais x e y podem ser considerados constantes. No presente trabalho, o comprimento do segmento elementar é fornecido pelo usuário.

As equações (7) e (8) determinam os valores $\Delta\phi$ e Δr . A partir destes valores, o número de passos em cada motor é dado pelo produto destes valores pelo fator 1 e fator 2, respectivamente, que incorporam as reduções mecânicas e a própria resolução do motor.

4. Detalhamento construtivo da traçadora

Como superfície de traçagem, foi utilizado um quadro de fórmica branco, de dimensões 100cm x 80cm. A estrutura de sustentação do quadro e do braço foi construída em perfil metalon.

Como braço da traçadora foi utilizado um perfil retangular de alumínio. Uma árvore fixada no ponto de rotação do braço é sustentada por um mancal, sendo responsável pelo movimento de rotação.

Adicionalmente, foram fixados dois tubos de alumínio ao perfil retangular, servindo de guia linear para a movimentação do carro porta-pena. O contato do carro com a guia é feito por rodízios de nylon, como visto na Fig. (7), que reduzem o atrito e praticamente eliminam o ruído. O carro porta-pena é por sua vez deslocado por intermédio de uma correia sincronizadora aberta. Esta correia é fixada em ambas as extremidades do carro, passando por uma polia acoplada ao atuador 2 e a outra polia louca na extremidade oposta do braço.

O torque para rotação do braço, advém do atuador número 2, que aciona um redutor tipo sem-fim coroa, que está acoplado diretamente a árvore do braço.

Foram utilizados motores de passo Parker, OEM83-135. Para virtualmente eliminar a trepidação comum em motores de passo, foram utilizados acionamentos, marca Parker, OEM750, que tem a capacidade de operar em micropasso. A resolução adotada foi de 5.000 passos por volta.

As fig. (4), (5), (6) e (7), mostram detalhes do protótipo construído.

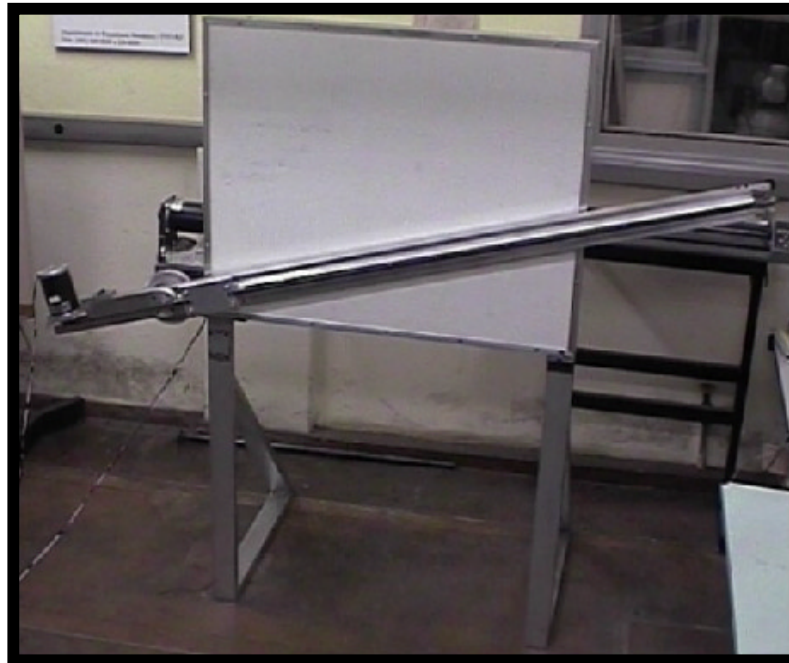


Figura 4 – Vista geral do protótipo

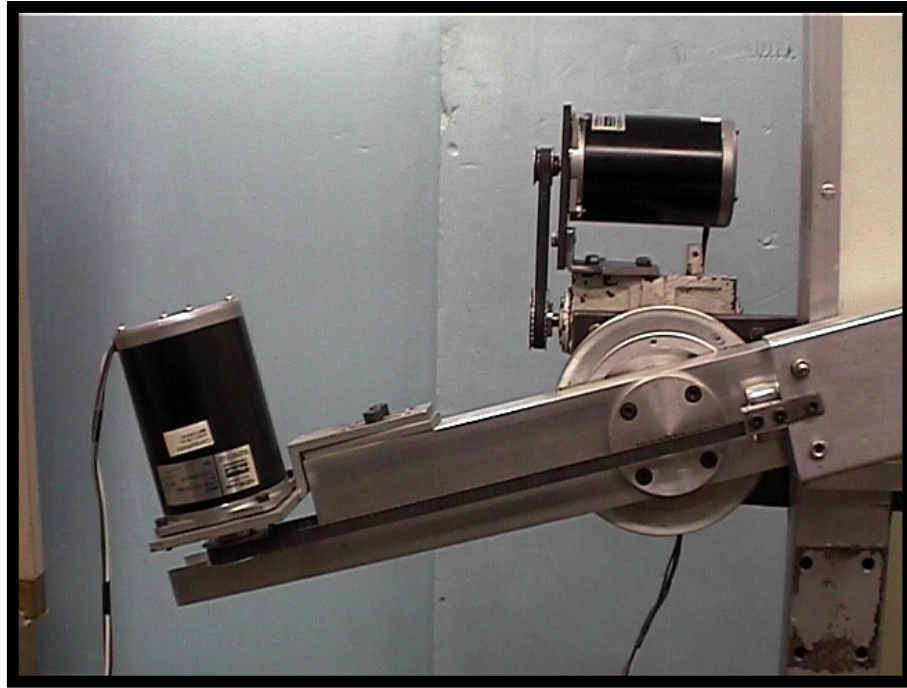


Figura 5 – Detalhe dos atuadores

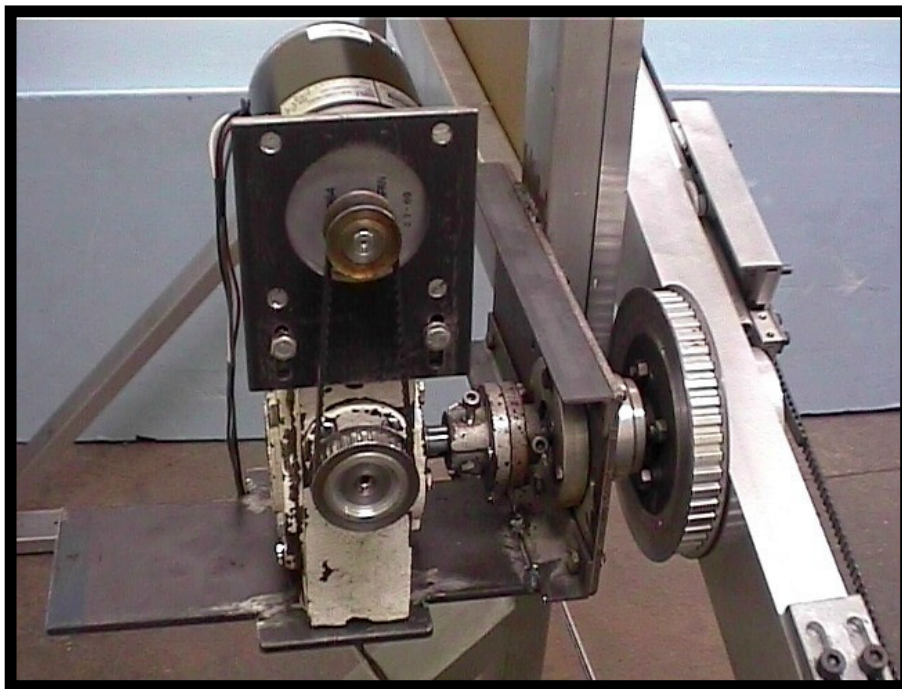


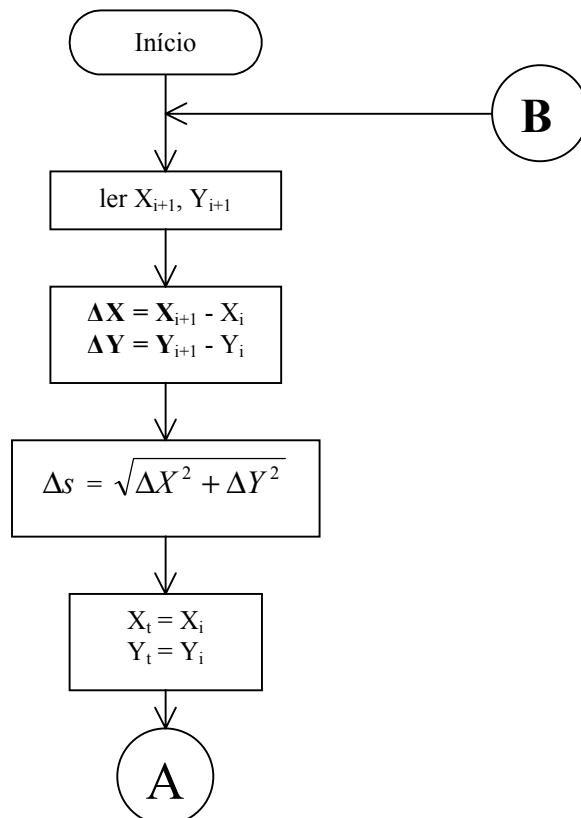
Figura 6 – Detalhe do sistema de giro do braço

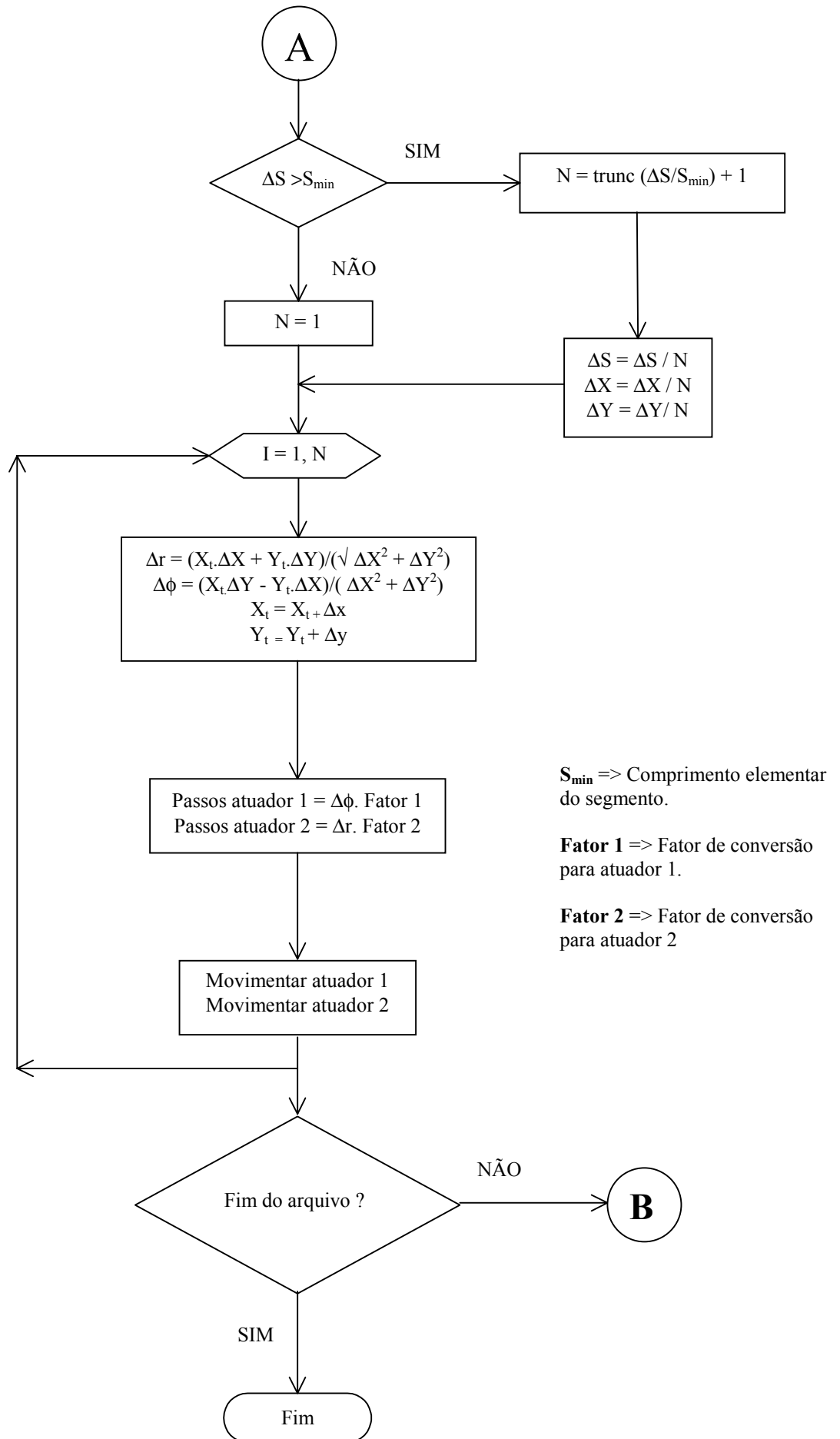


Figura 7 – Detalhe do carro e trilho do carro.

5. Fluxograma do sistema de acionamento dos atuadores.

Apresentamos nesta seção, o fluxograma básico de acionamento dos atuadores para a traçagem de um segmento de reta da posição (X_i, Y_i) , para a posição (X_{i+1}, Y_{i+1}) . Cabe destacar que a posição inicial (X_0, Y_0) é uma posição de “reset”, padronizada e conhecida.





6. Testes experimentais.

No presente estágio da pesquisa, estão sendo feitos os primeiros testes de traçagem de segmentos de reta, em diversas velocidades, para o levantamento dos eventuais problemas de vibração.

Tem-se verificado que a movimentação do carrinho (direção radial) numa faixa de velocidade, variando de 5cm/s até 30cm/s, ocorre de modo suave e preciso. Já a movimentação do braço em torno de O, tem sido acompanhada de oscilações quando o braço é freado na posição desejada, devido ao elevado momento polar de inércia do conjunto braço/carro porta pena e de pouca rampa de desaceleração. A solução deste problema está em enrijecer a árvore que liga o braço ao conjunto sem-fim coroa e aumentar os passos nas rampas de aceleração e desaceleração. Cabe destacar que não tem sido observado nenhum problema quanto aos torques estáticos, ficando o conjunto em equilíbrio nas mais diversas configurações ao longo do quadro. Os testes de traçagem seguindo linhas quaisquer, serão retomados e relatados tão logo fique concluída a modificação da árvore.

7. Conclusões.

Os primeiros ensaios revelam que o conceito da traçadora funcionando ancorada em um pilar e operando com controle em coordenadas polares é viável e pode evoluir para um produto industrializável com mercado definido.

O maior problema nesta fase, é representado pelas oscilações do braço quando aplicada uma velocidade angular. Este problema deverá ser reduzido com o aumento do diâmetro da árvore que transmite este movimento e uma melhoria nas rampas de aceleração-desaceleração. Um aumento no atrito do mancal, também poderá mostrar-se desejável, contribuindo ainda mais para o amortecimento das oscilações. Por outro lado a solução coroa sem-fim provou ser essencial para compatibilizar o momento do braço em relação ao seu eixo e o torque disponível no motor de passo.

A movimentação no sentido radial não apresenta nenhum problema, seja de posicionamento, seja de indução de oscilações.

Como última sugestão, deve-se reprojeter o desenho do braço, visando uma apreciável redução em sua massa.

8. Bibliografia.

- Catálogo Parker Compumotor Division, "Step Motor & Servo Motor Systems and Controls", 1996/1997, USA.
- Segenreich, S. A., 1991, "Posicionamento de uma Traçadora Gráfica de Hastes Articuladas", XI Congresso Brasileiro de Mecânica, ABCM, São Paulo, Brasil.
- Segenreich, S.A., 1993, "Conceito e Aplicação de um Posicionador Pantográfico Bidimensional", DINAME 93, Santo Amaro da Imperatriz – SC, Brasil.
- Segenreich, S.A., Viveiros Lima, R. 1997, "Análise e Desenvolvimento Experimental de um Acionamento em Micropasso", COBEM 97, Baurú – São Paulo, Brasil.
- Segenreich, S.A., Paniagua Vargas, J., 2000, "Uma nova Tecnologia de Acionamento de Motores de Passo Operando em Micropasso", CONEM 2000, Natal-RN, Brasil.

ANALYSIS AND EXPERIMENTAL DEVELOPMENT OF A PLOTTER FOR DRAWING ON VERTICAL SURFACES

Solly A. Segenreich
Pontifícia Universidade Católica do Rio de Janeiro – PUC-RIO
Rua Marquês de São Vicente 225 – Gávea – RJ – CEP: 22453-900
e-mail: solly@mec.puc-rio.br

Jorge Luiz Fontanella
Universidade Católica de Petrópolis
Estrada Presidente Sodr , 302 BL16/206 - Sim ria – Petrópolis – CEP: 25645-710
e-mail: jorge.fontanella@ucp.br

Abstract – The work presented in this paper describes the analysis and experimental development of a prototype of a plotter designed to draw on large scale vertical surfaces. This type of plotter is particularly useful for the production of outdoors. In order to achieve high degree of flexibility, the plotter uses polar coordinates in order to have a single fixed point. The present work discusses the basic lay-out of the system, kinematics of motion, as well as the structure of the driving program.

Key words: Plotters, Polar Coordinates Plotters

ABORDAGEM NUMÉRICA PARALELA E DISTRIBUÍDA PARA MODELAGEM NO ESPAÇO DE ESTADO DE SÉRIES TEMPORAIS MULTIVARIÁVEIS

Celso Pascoli Bottura

bottura@dmcsi.fee.unicamp.br

Gilmar Barreto

gbarreto@dmcsi.fee.unicamp.br

Mauricio José Bordon

bordon@dmcsi.fee.unicamp.br

Annabell Del Real Tamariz

annabell@dmcsi.fee.unicamp.br

Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e Computação
Cx. P. 6101 - 13083-970 - Campinas, SP, Brasil

***Resumo.** Neste trabalho, o algoritmo de Masanao Aoki para modelagem multivariada de séries temporais em subespaços do espaço de estado é abordado em aspectos numéricos e computacionais paralelos e distribuídos. A solução paralela da equação de Ricatti por ser uma etapa de maior demanda de processamento é destacada. Resultados de predições de um passo à frente pelo modelo no espaço de estado utilizando Filtro de Kalman para realizações calculadas são apresentados para avaliação da modelagem.*

***Palavras-chave:** Método de Subespaço, Séries Temporais, Identificação, Realização, Processamento Paralelo, Computação de Alto Desempenho.*

1. Introdução

Uma sequência de dados observados de um sinal resultante de um teste sobre um sistema mecânico, por exemplo um veículo, obtidos a intervalos constantes e ordenados temporalmente, constitui uma série temporal monovariada. Este sinal mecânico ou série temporal discreta pode ser processado por razões várias, dentre elas para se obter o seu modelo matemático discreto no tempo, com suas múltiplas utilidades. Raciocínio semelhante nos leva à séries temporais multivariadas obtidas por exemplo de múltiplos testes de uma grande e complexa estrutura aeroespacial. Podemos processar e ou modelar estes sinais multivariados por inúmeras razões seja para realizar regressões, seja para realizar predições, seja para detectar falhas, seja para tomar decisões, seja para seu monitoramento, seja para alguma forma de análise de dados e ou de controle, dentre outras. Quando é possível modelar com acuidade um sinal vetorial torna-se possível executar importantes tarefas de processamento de sinais, dentre elas extrapolações e interpolações.

O problema de modelagem de séries temporais é muito importante nas mais variadas áreas do conhecimento. Gauss dispo de observações sobre o movimento de um asteróide elaborou o famoso e importante método dos mínimos quadrados. Nobert Wiener (1949) realizou contribuição importante para a solução do problema de predição de trajetória futura de avião a partir de séries temporais e consequentemente de problemas de estimação e controle associados. Problemas de processamento de sinais e de modelagem de séries temporais estão intimamente relacionados e em muitas situações se confundem entre si. Algoritmos para modelagem de sinais multivariados discretos no tempo para múltiplas finalidades e aplicações constituem uma área muito fundamental e útil neste campo.

Uma série temporal multivariada constitui um processo estocástico vetorial e modelos destes sinais, que variam discretamente e aleatoriamente no tempo, podem ser tratados no espaço de estado, pelo emprego da teoria de realizações estocásticas de sistemas lineares, Akaike (1974), Caines (1988).

De um ponto de vista metodológico, neste trabalho desenvolvemos e implementamos um procedimento computacional paralelo e distribuído original para a modelagem de séries temporais multivariadas no espaço de estado de um algoritmo originalmente proposto por Masanao Aoki e o aplicamos a um sistema dinâmico que permite comprovar a qualidade e a utilidade do trabalho que realizamos sobre tal algoritmo e bem como

abre muitas possibilidades para sua aplicação e para muitas outras pesquisas teóricas e práticas nesta área.

A modelagem paralela de séries temporais multivariadas no espaço de estado requer múltiplos experimentos com sequências de saídas. Tal modelo pode ser representado por um sistema discreto estocástico linear multivariável e invariante no tempo, com ruído branco na entrada :

$$\begin{cases} x_{t+1} = \mathbf{A}x_t + \mathbf{B}e_t \\ y_t = \mathbf{C}x_t + e_t \end{cases} \quad (1)$$

onde : $x_t \in \mathfrak{R}^n$: vetor de estado estacionário no sentido fraco; $e_t \in \mathfrak{R}^p$: vetor de ruído com média zero, serialmente não correlato, estacionário no sentido fraco; $y_t \in \mathfrak{R}^p$: vetor de observação (saída); $\mathbf{A} \in \mathfrak{R}^{n \times n}$: matriz dinâmica do sistema; $\mathbf{B} \in \mathfrak{R}^{n \times p}$: matriz de entrada do sistema; $\mathbf{C} \in \mathfrak{R}^{p \times n}$: matriz de saída. Para a obtenção da tripla $(\mathbf{A}, \mathbf{B}$ e $\mathbf{C})$ a partir das séries temporais de saída , considera-se que o par de matrizes $\{\mathbf{A}, \mathbf{C}\}$ é observável e que o par de matrizes $\{\mathbf{A}, \mathbf{B}\}$ é atingível e ainda que $\{\mathbf{A}, \mathbf{B}, \mathbf{C}\}$ é uma realização mínima balanceada para o sistema dado na equação (1), isto é ; $\mathbf{G}_c = \mathbf{G}_o$; onde \mathbf{G}_o é o gramiano de observabilidade e \mathbf{G}_c é o gramiano de atingibilidade:

$$\mathbf{G}_o = \sum_{k=0}^{\infty} (\mathbf{A}^T)^k \mathbf{C}^T \mathbf{C} \mathbf{A}^k \quad \text{e} \quad \mathbf{G}_c = \sum_{k=0}^{\infty} \mathbf{A}^k \mathbf{B} \mathbf{B}^T (\mathbf{A}^T)^k \quad (2)$$

Neste trabalho o algoritmo paralelo e distribuído apresentado em Bottura (1997) é executado de forma paralela em 7 etapas distintas :

- Etapa 1 - Geração simultânea das matrizes \mathbf{H}^A , \mathbf{H}^M , \mathbf{H}^C , \mathbf{H} , \mathbf{Y}_- e \mathbf{Y}_+ . A partir dos dados de saída do sistema podemos gerar duas matrizes, \mathbf{Y}_- e \mathbf{Y}_+ dadas por:

$$\mathbf{Y}_- = \begin{bmatrix} \mathbf{y}_1 & \mathbf{y}_2 & \mathbf{y}_3 & \dots & \mathbf{y}_{N-1} \\ 0 & \mathbf{y}_1 & \mathbf{y}_2 & \dots & \mathbf{y}_{N-2} \\ 0 & 0 & \mathbf{y}_1 & \dots & \mathbf{y}_{N-3} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & \mathbf{y}_{N-K} & \mathbf{y}_{N-K+1} \\ 0 & 0 & \dots & \mathbf{y}_{N-K-1} & \mathbf{y}_{N-K} \end{bmatrix} \quad \text{onde } \mathbf{Y}_- \in \mathfrak{R}^{(Kp) \times (N-1)} \quad (3)$$

$$\mathbf{Y}_+ = \begin{bmatrix} \mathbf{y}_2 & \mathbf{y}_3 & \mathbf{y}_4 & \dots & \mathbf{y}_{N-1} & \mathbf{y}_N \\ \mathbf{y}_3 & \mathbf{y}_4 & \mathbf{y}_5 & \dots & \mathbf{y}_N & 0 \\ \mathbf{y}_4 & \mathbf{y}_5 & \mathbf{y}_6 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}_J & \mathbf{y}_{J+1} & \mathbf{y}_{J+2} & \dots & 0 & 0 \\ \mathbf{y}_{J+1} & \mathbf{y}_{J+2} & \mathbf{y}_{J+3} & \dots & 0 & 0 \end{bmatrix} \quad \text{onde } \mathbf{Y}_+ \in \mathfrak{R}^{(Jp) \times (N-1)} \quad (4)$$

O produto das matrizes \mathbf{Y}_+ e \mathbf{Y}_-^T permite gerar uma matriz de Hankel , \mathbf{H} , sem calcular explicitamente as covariâncias $\mathbf{\Lambda}_k$. Assim, obtemos uma matriz de Hankel tal que :

$$\mathbf{H}_{J,K} = \frac{\mathbf{Y}_+ \mathbf{Y}_-^T}{N} = \begin{bmatrix} \mathbf{\Lambda}_1 & \mathbf{\Lambda}_2 & \dots & \mathbf{\Lambda}_K \\ \mathbf{\Lambda}_2 & \mathbf{\Lambda}_3 & \dots & \mathbf{\Lambda}_{K+1} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{\Lambda}_J & \mathbf{\Lambda}_{J+1} & \dots & \mathbf{\Lambda}_{J+K} \end{bmatrix} \quad (5)$$

Dos dados de saída do sistema três novas matrizes são geradas:

$$\mathbf{H}^A = \begin{bmatrix} \mathbf{\Lambda}_2 & \mathbf{\Lambda}_3 & \dots & \mathbf{\Lambda}_{K+1} \\ \mathbf{\Lambda}_3 & \mathbf{\Lambda}_4 & \dots & \mathbf{\Lambda}_K \\ \mathbf{\Lambda}_4 & \mathbf{\Lambda}_5 & \dots & \mathbf{\Lambda}_K \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{\Lambda}_{J+1} & \mathbf{\Lambda}_{J+2} & \dots & \mathbf{\Lambda}_{J+K+1} \end{bmatrix}, \quad \mathbf{H}^M = \begin{bmatrix} \mathbf{\Lambda}_1 \\ \mathbf{\Lambda}_2 \\ \mathbf{\Lambda}_3 \\ \vdots \\ \mathbf{\Lambda}_J \end{bmatrix} \quad \text{e} \quad (6)$$

$$\mathbf{H}^C = [\mathbf{\Lambda}_1 \quad \mathbf{\Lambda}_2 \quad \dots \quad \mathbf{\Lambda}_K]. \quad (7)$$

bem como $\mathbf{\Lambda}_0 = E(y_t y_t^T) = \mathbf{C} \mathbf{\Pi} \mathbf{C}^T + \mathbf{\Delta}$, onde $\mathbf{\Lambda}_0$ é a autocovariância do vetor de saída, e onde considera-se que o ruído de entrada tenha uma matriz de covariância dada por $E(e_t e_s^T) = \mathbf{\Delta} \delta_{t,s}$, $\mathbf{\Delta} > 0$ (condição de regularidade) e que o estado possua uma matriz de covariância dada por $E(x_t x_t^T) = \mathbf{\Pi}$, com a matriz $\mathbf{\Pi}$ constante.

- Etapa 2 - Decomposição em valores singulares da matriz $\mathbf{H}_{J,K} = \mathbf{U}\Sigma^{1/2}\Sigma^{1/2}\mathbf{V}^T$. A determinação dos autovalores e autovetores de uma matriz possibilita realizar a decomposição em valores singulares de $\mathbf{H}_{J,K}$. Este problema possui uma formulação teórica simples, mas sua solução computacional apresenta dificuldades consideráveis, Jainandusing *et al.* (1989) e Janakiraman *et al.* (1982).

Dada uma matriz $\mathbf{\Gamma} \in \mathfrak{R}^{m \times n}$, sempre existe a decomposição

$$\mathbf{U}^T \mathbf{\Gamma} \mathbf{V} = \text{diag}(\sigma_1, \sigma_2, \dots, \sigma_r) \in \mathfrak{R}^{m \times n} \quad r = \min\{m, n\} \quad (8)$$

com $\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_r \geq 0$ e sendo \mathbf{U} e \mathbf{V} matrizes unitárias.

Os números σ_i são valores singulares de $\mathbf{\Gamma}$, as colunas de \mathbf{U} são vetores singulares de $\mathbf{\Gamma}$ à esquerda e formam uma base para o espaço coluna de $\mathbf{\Gamma}\mathbf{\Gamma}^T$; as colunas de \mathbf{V} são vetores singulares de $\mathbf{\Gamma}$ à direita e formam uma base para o espaço coluna de $\mathbf{\Gamma}^T\mathbf{\Gamma}$. Por inspeção do conjunto de valores singulares de uma matriz, determina-se seu posto numérico, visto que seu menor valor singular é uma medida da sua distância ao espaço vetorial das matrizes reais $m \times n$ com posto incompleto, Golub *et al.* (1989).

- Etapa 3 - Neste nível são calculadas em paralelo as matrizes do sistema (1): $\mathbf{A} = \Sigma^{-1/2}\mathbf{U}^T\mathbf{H}^A\mathbf{V}\Sigma^{-1/2}$; $\mathbf{C} = \mathbf{H}^C\mathbf{V}\Sigma^{-1/2}$ e a matriz \mathbf{M} é dada por : $\mathbf{M} = \Sigma^{-1/2}\mathbf{U}^T\mathbf{H}^M$, onde \mathbf{M} é definida por : $\mathbf{M} = \mathbf{A}\mathbf{\Pi}\mathbf{C}^T + \mathbf{B}\Delta$.

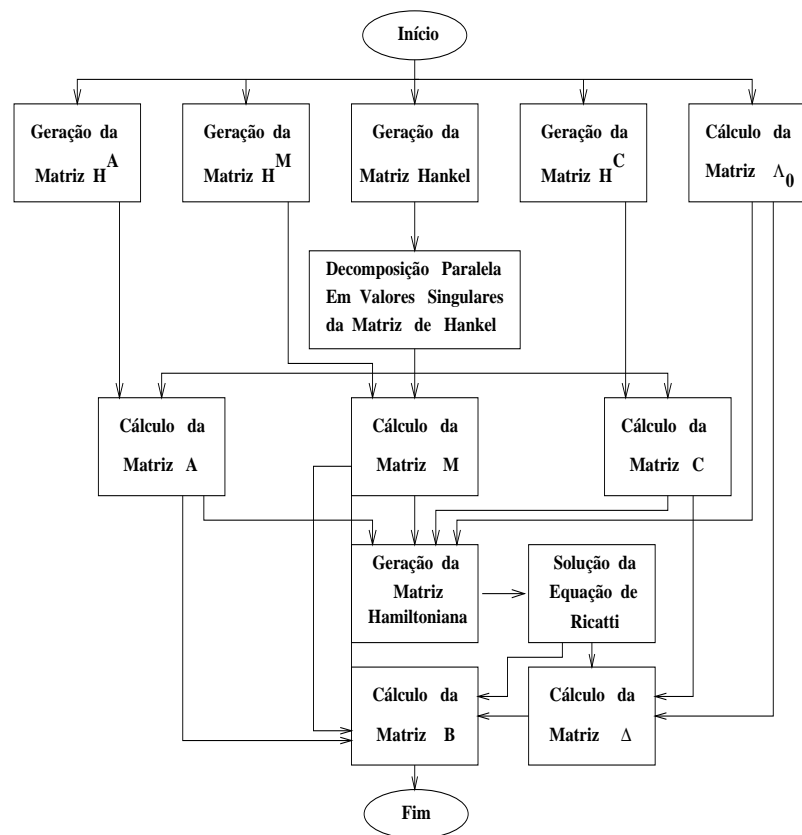


Figura 1: Implementação Paralela do Algoritmo

- Etapa 4 - Construção da matriz simplética :

$$\Phi = \begin{bmatrix} \Psi - \hat{Q}\Psi^{-T}\mathbf{D} & \hat{Q}\Psi^T \\ -\Psi^T\mathbf{D} & \Psi^{-T} \end{bmatrix}, \quad (9)$$

onde as matrizes auxiliares são dadas por :

$$\Psi = \mathbf{A}^T - \mathbf{C}^T\Lambda_0^{-1}\mathbf{M}^T, \quad \hat{Q} = \mathbf{C}^T\Lambda_0^{-1}\mathbf{C}, \quad \text{e } \mathbf{D} = \mathbf{M}\Lambda_0^{-1}\mathbf{M}^T. \quad (10)$$

- Etapa 5 - Determinação da matriz $\mathbf{\Pi}$, solução da equação de Ricatti, a partir da equação: $\mathbf{\Pi} = \mathbf{A}\mathbf{\Pi}\mathbf{A}^T + \mathbf{K}(\mathbf{\Pi})$ onde,

$$\mathbf{K}(\mathbf{\Pi}) = (\mathbf{M} - \mathbf{A}\mathbf{\Pi}\mathbf{C}^T)(\Lambda_0 - (\mathbf{C}\mathbf{\Pi}\mathbf{C}^T)^{-1}(\mathbf{M} - \mathbf{A}\mathbf{\Pi}\mathbf{C}^T))^T. \quad (11)$$

A solução pode ser determinada por meio da decomposição de Schur real da matriz Φ : $\Phi = \mathbf{W}\mathbf{T}\mathbf{W}^T$. Esta decomposição é obtida a partir do método **QR**. Particionando a matriz unitária \mathbf{W} conforme :

$$\mathbf{W} = \begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix}, \quad (12)$$

enunciamos a seguinte proposição, Vaughan (1970):

Proposição : A matriz $\mathbf{\Pi} = \mathbf{W}_{21}\mathbf{W}_{11}^{-1}$ é a solução da Equação de Ricatti.

- Etapa 6 - Determinação da matriz $\mathbf{\Delta} = \mathbf{\Lambda}_0 - \mathbf{C}\mathbf{\Pi}\mathbf{C}^T$.
- Etapa 7 - Obtenção da matriz $\mathbf{B} = (\mathbf{M} - \mathbf{A}\mathbf{\Pi}\mathbf{C}^T)\mathbf{\Delta}^{-1}$

As etapas da decomposição em valores singulares da matriz e da solução paralela da equação de Ricatti por serem de maior demanda de processamento são destacadas nas próximas seções.

2. Solução Paralela da Decomposição em Valores Singulares

A decomposição em valores singulares, *SVD*, de uma matriz real $\mathbf{A}_{m \times n}$ é escrita como $\mathbf{A} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ onde $\mathbf{\Sigma}$ é uma matriz $m \times n$ que é zero exceto para seus $\min(m, n)$ elementos de diagonal, \mathbf{U} é uma matriz ortogonal $m \times m$, e \mathbf{V} é uma matriz ortogonal $n \times n$. Os elementos da diagonal de $\mathbf{\Sigma}$ são os valores singulares de \mathbf{A} e as colunas de \mathbf{U} e \mathbf{V} são os correspondentes vetores singulares, esquerdos e direitos, respectivamente.

Os valores singulares são retornados em $\mathbf{\Sigma}$ em ordem decrescente e só as primeiras $\min(m, n)$ colunas de \mathbf{U} e linhas de \mathbf{V}^T são calculadas. Se $m \gg n$, um melhor desempenho computacional é obtido por meio da fatorização **QR** de \mathbf{A} ; então a decomposição em valores singulares é realizada em \mathbf{R} que é uma matriz $n \times n$, desta forma se $\mathbf{A} = \mathbf{Q}\mathbf{R}$ e $\mathbf{R} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ então a *SVD* de \mathbf{A} e tal que $\mathbf{A} = (\mathbf{Q}\mathbf{U})\mathbf{\Sigma}\mathbf{V}^T$. Observe que nesta transformação os valores singulares da matriz \mathbf{A} são preservados.

Para realizar a implementação paralela da subrotina, as linhas, ou colunas da matriz são distribuídas em uma grade de processadores de duas dimensões, pois as subrotinas da *Scalapack* utilizadas, Blackford (1997), são baseadas em algoritmos com particionamento em blocos, o que minimiza a frequência de movimento de dados entre os diferentes níveis de memória hierárquica. Ou seja, em cada passo do algoritmo um bloco de linhas ou colunas distribuídas na grade é atualizada, e a maioria da computação é executada por operações matriz-matriz nestes blocos.

3. Solução Paralela da Equação Algébrica de Ricatti

A obtenção da solução paralela da equação algébrica de Riccati foi dividida em duas partes: A primeira corresponde a formação da matriz em forma de Hessenberg via *TSEE*,¹ utilizando como plataforma de programação paralela o *MPI*², com memória distribuída, comunicação síncrona, e a semântica das funções para enviar e receber mensagens é bloqueante com modo de programação *SPMD*; a segunda parte corresponde à obtenção da matriz em Forma Real de Schur que foi implementada utilizando as subrotinas da *Scalapack*.

A biblioteca de alto desempenho com rotinas de Álgebra Linear, *Scalapack*, para computadores *MIMD*³ com paradigma de troca de mensagens e memória distribuída em redes de estações de trabalho suporta *PVM*⁴ ou *MPI*. Utiliza também como parte de suas rotinas a biblioteca sequencial *BLAS - Basic Linear Algebra Subroutines*, a paralela *PBLAS - Parallel BLAS*, os subprogramas de comunicação da algebra linear básica *BLACS - Basic Linear Algebra Communication Subroutines* e as rotinas da *LAPACK - Linear Algebra PACKage*.

Similarmente ao *LAPACK*, as rotinas da *Scalapack* são baseadas em algoritmos com particionamento em blocos, o que minimiza a frequência de movimento de dados entre os diferentes níveis da memória hierárquica. Os algoritmos são apresentados como processos e a comunicação pode ser ponto-a-ponto ou em grupo quando necessário. De forma geral, a *Scalapack* foi desenvolvida e testada para o caso de um processo por processador.

Nos interessa a implementação paralela do método de Schur para o caso de matrizes reais não simétricas utilizando *TSEE*: método de Schur-Modificado paralelo.

Consideramos a paralelização do algoritmo de Schur-Modificado para matrizes densas, não simétricas sobre um sistema de memória distribuída; o modo de programação paralela utilizado é *SPMD*, ou seja todos

¹TSEE : transformações de similaridade elementares estabilizadas

²MPI : Message Passing Interface

³Esta arquitetura combina paralelismo nos fluxos de dados e de instruções; é composta de unidades de processamento, cada qual capaz de realizar operações lógicas e aritméticas padrões. As unidades operam assincronamente sob o controle de fluxo de instruções individuais.

⁴PVM : Parallel Virtual Machine

os processadores vão executar o mesmo programa, porém podem executar tarefas diferentes sobre dados diferentes. A estratégia de paralelização aplicada segue um esquema assíncrono; descreve-se as principais subrotinas utilizadas na solução do problema apresentado.

A implementação paralela da primeira parte da solução da equação algébrica de Ricatti que corresponde a formação da matriz de Hessenberg via *TSEE* é baseada em operações do tipo produto matriz-matriz, que representam maior custo computacional; essas operações são efetuadas acessando os elementos da matriz por linhas; desta maneira formamos blocos que são distribuídos entre os diferentes processadores participantes do processamento, Figura 2. Neste caso é possível paralelizar o produto tal que cada processador calcula os valores dos elementos diferentes do vetor resultante. O método para acessar os elementos da matriz é a *Distribuição por linhas*: os P processadores que participam do processamento paralelo podem ser apresentados como um arranjo linear uni-dimensional, onde os elementos são distribuídos em blocos. Assim ao distribuir uma matriz (6×6) em 4 processadores, numeramos os processadores de 0 até $P - 1$ e as linhas da matriz de 1 até N ($N = 6$). A Tabela 1 apresenta uma distribuição de blocos por linha, onde cada linha ou submatriz formada é nomeada por um número de processador. Neste caso, cada processador armazena no máximo (NB) número de linhas, onde $NB = \lceil N/P \rceil$ e a linha k é armazenada no processador $\lfloor k/NB \rfloor$. A distribuição atribui blocos de linhas de tamanho (NB) aos processadores sucessivos. Se o valor de P é divisível pelo valor de N , então cada processador vai receber blocos do mesmo tamanho, de forma a garantir um melhor balanceamento de carga.

0
1
2
3

Tabela 1: Armazenagens por linha

Com o modo de programação *SPMD*, cada processador participante realiza uma operação de multiplicação ($C = A * B$) com os dados enviados pelo processador principal, ou seja, os dados são divididos e cada processador calcula no máximo (NB) blocos da matriz resultante na multiplicação matriz-matriz que realiza. O processador principal, encarregado de distribuir os dados e realizar seus cálculos envia no máximo (NB) blocos da matriz A (NB, N) e a matriz B (N, N) para cada processador participante do processo e recebe uma matriz de dimensão (NB, N) com o produto final de cada processador. A Figura 2, ilustra o procedimento descrito para um sistema de ordem (6×6) onde os processadores recebem as linhas seguintes:

- P_0 = linha A_{1n} da matriz (processador principal)
- P_1 = linhas A_{2n} e A_{3n} da matriz
- P_2 = linhas A_{4n} e A_{5n} da matriz
- P_3 = linha A_{6n} da matriz

A implementação paralela da segunda parte do algoritmo proposto para solução da equação algébrica de Ricatti, a correspondente a achar a *Forma Real de Schur*, é baseada na utilização das subrotinas definidas na biblioteca *Scalapack*.

Na implementação estão presentes os mecanismos básicos *Scalapack*, tais como criação da topologia malha, comunicação entre tarefas e entrada-saída de dados. A *Scalapack* utiliza uma distribuição de dados com blocos cíclicos de dimensão-2, na qual a matriz, dividida em blocos de linhas e colunas ($MB \times NB$), é atribuída ao mesmo processador. A distribuição de dados cíclica é parametrizada por quatro valores: P , Q , r e c , onde ($P \times Q$) é a definição da topologia malha utilizada e ($r \times c$) é a dimensão do bloco. Supõe-se que os processadores estão arranjados logicamente como uma malha de P linhas e Q colunas; assim o número de processadores utilizados corresponde ao produto $R = P \times Q$.

Após o particionamento da matriz ($N \times N$) em blocos ($MB \times NB$), o primeiro bloco é associado ao processador com coordenadas ($RSRC, CSRC$), onde $RSRC$ e $CSRC$ correspondem respectivamente aos números dos processos linha e coluna aos quais o primeiro bloco da matriz é atribuído. Seja a entrada (I, J); cada elemento da matriz global está localizado no processador definido pelas coordenadas (p_r, p_c), do bloco local (l, m), na posição (x, y), onde:

$$\begin{aligned}
 (l, m) &= (\lfloor (I - 1)/(P * MB) \rfloor, \lfloor (J - 1)/(Q * NB) \rfloor), \\
 (p_r, p_c) &= ((RSRC + \lfloor (I - 1)/MB \rfloor) \bmod P, (CSRC + \lfloor (J - 1)/NB \rfloor) \bmod Q), \\
 (x, y) &= (\bmod(I - 1, MB) + 1, \bmod(J - 1, NB) + 1).
 \end{aligned}$$

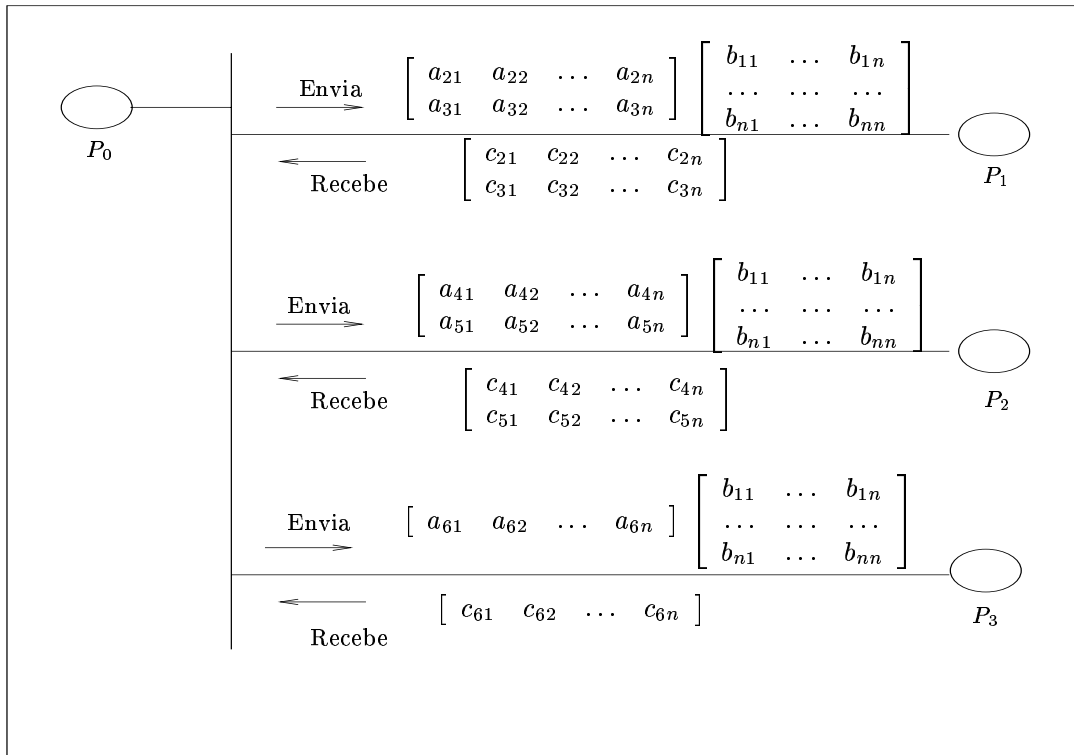


Figura 2: Processo Paralelo de Distribuição de matrizes

Apresentamos na Figura 3 o mapeamento de uma matriz A (6×6) particionada em blocos (2×2) e na Figura 4 o ponto de vista do processador para a distribuição da matriz (6×6) em uma topologia malha (2×2) processadores, onde podemos observar que a entrada local $A(2, 3)$, definida no processador de coordenadas $(0, 0)$, contém o valor da entrada da matriz global a_{25} .

a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}
a_{21}	a_{22}	a_{23}	a_{24}	a_{25}	a_{26}
a_{31}	a_{32}	a_{33}	a_{34}	a_{35}	a_{36}
a_{41}	a_{42}	a_{43}	a_{44}	a_{45}	a_{46}
a_{51}	a_{52}	a_{53}	a_{54}	a_{55}	a_{56}
a_{61}	a_{62}	a_{63}	a_{64}	a_{65}	a_{66}

Figura 3: Matriz 6×6 particionada em blocos 2×2

Existe uma subrotina da *Scalapack* denominada *BLACS-GRIDINIT* que realiza o mapeamento dos elementos da matriz global em cada processador por linha, ou seja vai distribuindo os elementos ordenados por linha, veja a definição na Tabela 2, onde quatro processos são distribuídos em uma malha (2×2) com uma distribuição por linha:

0	1
2	3

Tabela 2: Quatro processos mapeados em uma malha (2×2)

Para um exemplo de ordem (20×20), os parâmetros da subrotina que inicializam a topologia malha vão estar definidos como: $MB=NB=6$, $P=Q=2$ e $LLD=12$, onde LLD corresponde aos números máximos de linhas e ou colunas em cada bloco da submatriz; para o exemplo que estamos apresentando os parâmetros estão definidos como: $MB = NB = 6$, $P = Q = 2$ e $LLD = 6$.

	0				1	
0	a_{11}	a_{12}	a_{15}	a_{16}	a_{13}	a_{14}
	a_{21}	a_{22}	a_{25}	a_{26}	a_{23}	a_{24}
	a_{51}	a_{52}	a_{55}	a_{56}	a_{53}	a_{54}
	a_{61}	a_{62}	a_{65}	a_{66}	a_{63}	a_{64}
1	a_{31}	a_{32}	a_{35}	a_{36}	a_{33}	a_{34}
	a_{41}	a_{42}	a_{45}	a_{46}	a_{43}	a_{44}

Figura 4: Matriz 6 × 6 em uma Topologia Malha 2 × 2

4. Resultados Experimentais

O algoritmo proposto foi aplicado na modelagem de um série temporal multivariada, $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$, $\mathbf{y}_t \in \mathbb{R}^2$, gerando a seguinte tripla de matrizes :

$$\mathbf{A}_N = \begin{bmatrix} -0.2497 & -0.6753 & -0.0392 \\ 0.7490 & 0.3505 & -0.0535 \\ 0.0045 & 0.0649 & -0.4679 \end{bmatrix}; \quad \mathbf{B}_N = \begin{bmatrix} -0.2225 & 0.2349 \\ 0.3285 & 0.6982 \\ 0.0597 & 0.1756 \end{bmatrix} \quad \text{e} \quad \mathbf{C}_N = \begin{bmatrix} -2.6917 & 1.6029 & -0.1129 \\ -0.2622 & -1.0398 & -0.0745 \end{bmatrix}$$

Da sequência $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N\}$ tomamos os q primeiros vetores $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_q\}$ com $q \ll N$ e calculamos a tripla de matrizes do modelo pelo algoritmo descrito. As matrizes assim obtidas são chamadas de \mathbf{A}_q , \mathbf{B}_q e \mathbf{C}_q . O mesmo procedimento é repetido para $2q, 3q, \dots, N$ e assim obtém-se uma sequência de triplas formadas com os dados disponíveis desde $t = 1$ até $t = rq$ sendo $r = N/q$.

$$\mathbf{A}_1 = \begin{bmatrix} -0.1795 & -0.7008 & -0.0684 \\ 0.7542 & 0.4025 & -0.1320 \\ 0.0529 & -0.0127 & -0.5074 \end{bmatrix}; \quad \mathbf{B}_1 = \begin{bmatrix} -0.2243 & 0.1940 \\ 0.2599 & 0.6136 \\ 0.2930 & 0.3757 \end{bmatrix} \quad \text{e} \quad \mathbf{C}_1 = \begin{bmatrix} -2.8609 & 1.6482 & -0.3140 \\ -0.1056 & -0.9674 & -0.3234 \end{bmatrix};$$

$$\mathbf{A}_2 = \begin{bmatrix} -0.2322 & -0.6571 & -0.1596 \\ 0.7278 & 0.3926 & -0.1402 \\ 0.1428 & -0.1056 & 0.5753 \end{bmatrix}; \quad \mathbf{B}_2 = \begin{bmatrix} -0.2187 & 0.2190 \\ 0.2821 & 0.6441 \\ 0.2501 & 0.3228 \end{bmatrix} \quad \text{e} \quad \mathbf{C}_2 = \begin{bmatrix} -2.8214 & 1.6826 & -0.2861 \\ -0.1910 & -1.0127 & -0.1833 \end{bmatrix};$$

$$\mathbf{A}_3 = \begin{bmatrix} -0.2388 & -0.6636 & -0.1326 \\ 0.7331 & 0.3790 & -0.1149 \\ 0.1240 & -0.0921 & 0.6399 \end{bmatrix}; \quad \mathbf{B}_3 = \begin{bmatrix} -0.2086 & 0.2220 \\ 0.2911 & 0.6670 \\ 0.2039 & 0.2647 \end{bmatrix} \quad \text{e} \quad \mathbf{C}_3 = \begin{bmatrix} -2.8339 & 1.6876 & -0.2486 \\ -0.2097 & -1.0485 & -0.1816 \end{bmatrix}$$

sendo $N = 8192$ e $q = 512$.

Na seção 4.1 são avaliados os tempos computacionais correspondentes à solução da equação algébrica de Ricatti de formas sequencial e paralela. Na seção 4.2 os modelos obtidos com dados intermediários são usados no projeto de preditores de um passo a frente por filtro de Kalman, e as covariâncias dos erros de predição em cada intervalo são avaliadas pelo seu traço conforme a Figura 6.

4.1 Resultados Computacionais : Equação Algébrica de Ricatti

Os resultados que apresentamos foram obtidos com diferentes valores dos parâmetros. Por exemplo, para uma matriz de ordem (6 × 6) que definimos a seguir os parâmetros são os seguintes: N=6, P=3, NB=2. Com este tamanho de matriz, obtemos tempos de execução paralela de cada uma das partes do programa maiores do que os tempos respectivos obtidos na implementação sequencial, veja Tabela 3; mas isso é diferente quando utilizamos P=2.

Atividade	SEQUENCIAL	PARALELO	
	1-proc	2-proc	3-proc
Leitura dos dados	0.1256	0.0296	0.0323
Matriz Hessenberg	0.1699	0.1060	0.2049
Forma Real de Schur	0.2228	0.1412	0.3623
Tempo Total de Execução	0.5183	0.2768	0.5995

Tabela 3: Tempos de Execução das partes do Algoritmo em uma matriz (6 × 6)

Com nossa implementação obtemos uma solução simétrica, definida não negativa da equação algébrica de Riccati com *TSEE*.

Quando testamos com matrizes de ordens maiores, por exemplo (50×50) obtemos, Tabela 4, que os tempos de execução paralela de cada uma das partes do programa são bem menores do que os tempos respectivos obtidos na implementação sequencial; assim podemos concluir que a paralelização surte melhores resultados quando o problema envolve cálculos com matrizes de ordens bem maiores.

Atividade	SEQUENCIAL	PARALELO	
	1-proc	2-proc	3-proc
Leitura dos dados	0.3092	0.2642	0.2869
Matriz Hessenberg	59.0045	8.3596	10.3933
Forma Real de Schur	1.8914	0.9707	1.3932
Tempo Total de Execução	61.2051	9.5945	12.0734

Tabela 4: Tempos de Execução das partes do Algoritmo em uma matriz (50×50)

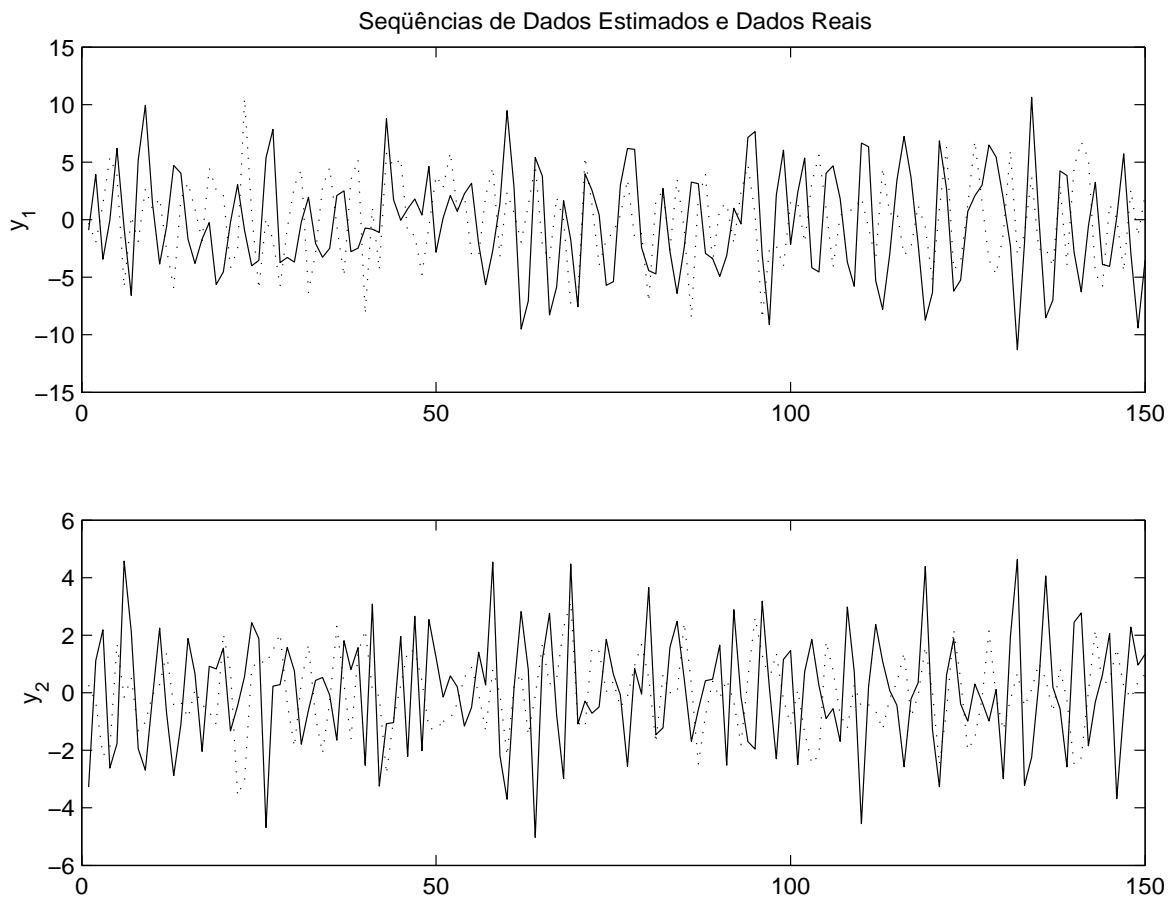


Figura 5: Valores medidos e valores estimados. A linha sólida representa os dados reais e a pontilhada os preditos

Contrariamente, temos um exemplo teste com uma matriz de ordem (100×100) , na qual os resultados não são favoráveis, ou seja obtemos tempos de execução paralela de cada uma das partes do programa bem maiores do que os tempos respectivos obtidos na implementação sequencial, veja Tabela 5; achamos que a causa deste problema esteja na definição do tamanho dos pacotes no sistema operacional ou na grande comunicação entre os processadores para obter o resultado final do problema.

4.2. Avaliação da Modelagem de Séries Temporais Multivariadas no Espaço de Estado

As triplas de matrizes obtidas são avaliadas por meio de um procedimento de comparação entre dados preditos pelo modelo e reais. A Figura 5 mostra o traçado de observações originais da série e de observações estimadas das últimas 150 amostras da sequência $\{y_1, y_2, \dots, y_N\}$. O teste de predição para avaliação de modelos obtidos é descrito a seguir. Dada a sequência $\{y_1, y_2, \dots, y_q\}$ calcula-se a tripla $\mathbf{A}_q, \mathbf{B}_q$ e \mathbf{C}_q . Um estimador de um passo à frente é construído e usado para projetar o vetor de saída no subespaço gerado pelas

Atividade	SEQUENCIAL	PARALELO	
	1-proc	2-proc	3-proc
Leitura dos dados	0.5088	0.2939	0.3293
Matriz Hessenberg	73.2661	121.6329	132.3660
Forma Real de Schur	1.2397	2.1775	3.2429
Tempo Total de Execução	75.6775	125.1755	137.1508

Tabela 5: Tempos de Execução das partes do Algoritmo em uma matriz (100 × 100)

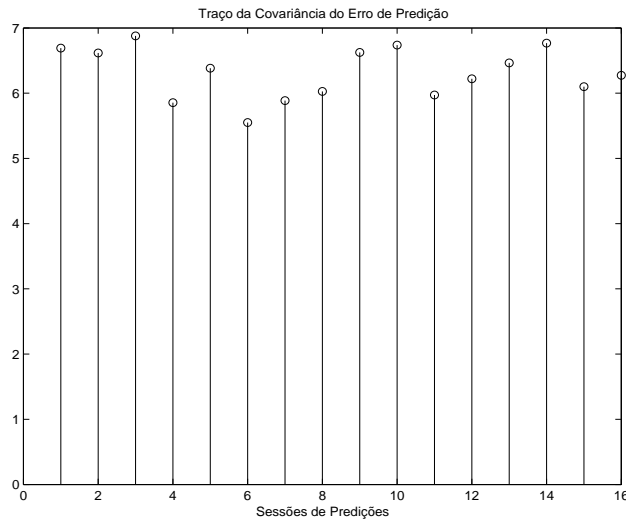


Figura 6: Traço da Covariância do Erro de Predição

saídas anteriores, dadas por $\{\mathbf{y}_{2|1}, \mathbf{y}_{3|2}, \dots, \mathbf{y}_{q|q-1}\}$; se o estimador for consistente, o vetor erro de predição $\mathbf{e}_{t|t-1} = \mathbf{y}_{t|t-1} - \mathbf{y}_t$ terá uma matriz de covariância que se aproxima da covariância da inovação, conforme a quantidade $q \rightarrow \infty$. Nesta avaliação do modelo usamos os primeiros q vetores de observações para calcular as matrizes do modelo \mathbf{A}_q , \mathbf{B}_q e \mathbf{C}_q . A predição é feita sobre os vetores $\{\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_q\}$. Isto é feito para verificar se o modelo obtido pelas matrizes \mathbf{A}_q , \mathbf{B}_q e \mathbf{C}_q representa a sequência de vetores originais \mathbf{y}_t no intervalo de 1 a q . Este procedimento de predição é repetido nos intervalos $q + 1$ a $2q$, $2q + 1$ a $3q$ e assim sucessivamente até $(r - 1)q + 1$ a rq sendo $rq = N$. A Figura 6 mostra os traços das matrizes de covariâncias dos erros de predição conforme o intervalo de $q = 512$ vetores aproxima-se da N -ésima observação. Vemos que o estimador formado com as matrizes calculadas mantém o traço da covariância do erro de estimação dentro de um intervalo fixo de 5,5 a 7,0.

5. Conclusões

Em relação a comparação em tempo de execução dos algoritmos implementados em formas sequencial e paralela, podemos dizer que o processamento paralelo nem sempre foi melhor que o processamento sequencial quanto ao tempo de execução, pois no processamento paralelo existe uma alta comunicação de dados entre os diferentes processadores, o que implica que a solução possa ser mais demorada; contudo o processamento paralelo propiciou uma menor carga computacional para os processadores envolvidos.

Em nossa implementação devido a utilização de uma rede de estações de trabalho, e a forma de implementar o algoritmo paralelo, o tempo de execução obtido, para alguns casos foi bem maior do que o tempo de execução obtido no algoritmo sequencial, por causa da comunicação entre os diferentes processadores participantes do procedimento. Em muitos algoritmos e sistemas paralelos e ou distribuídos o tempo consumido pela comunicação entre processadores é uma fração do tempo total necessário para resolver o problema.

Além da redução no tempo de execução do algoritmo pelo uso de processamento paralelo e distribuído, notamos que o uso de filtros de Kalman contruídos com matrizes sistêmicas obtidas na modelagem produzem estimativas próximas dos vetores originais da série, conforme Figura 5. Uma avaliação das covariâncias dos erros de predição em cada intervalo também é mostrada na Figura 6. Vemos que os traços das covariâncias dos erros de predição permanecem dentro de um intervalo limitado indicando estacionariedade do processo de estimação e que o estimador é consistente.

Na continuidade deste trabalho pretendemos : testar nosso algoritmo em uma máquina paralela; realizar implementação diferente da que fizemos para o nosso algoritmo, procurando reduzir a comunicação em favor

da computação; utilizar uma arquitetura paralela com memória compartilhada e comparar os resultados quanto ao tempo de execução global do algoritmo paralelo; estudar o efeito das transformações de similaridade não ortogonais em matrizes não simétricas em relação ao condicionamento das matrizes e estudar o efeito do processo de balanceamento no método de solução proposto neste trabalho.

6. Agradecimento

FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo.

7. Referências

- Akaike, Hirotugu, 1974, " Stochastic Theory of Minimal Realization " , IEEE Transactions on Automatic Control, Vol. AC-19, No.6, 667-674, December.
- Aoki, Masanao, 1987, " State Space Modeling of Time Series " , Springer-Verlag.
- Bertsekas, D. and Tsitsiklis, J., 1989, " Parallel and Distributed Computations " , Prentice Hall.
- Blackford, L. S.; J. Choi, A. Cleary, E. D'Ázevedo, J. Demmel, I. Dhillon, J. Dongarra, S. Hammarling, G. Henry, A. Petitet, K. Stanley, D. Walker; R. C. Whaley, 1997, " ScaLAPACK Users' Guide " , SIAM Publications, Philadelphia.
- Bottura, C.P.; Barreto, G.; Bordon, M.J. e Costa Filho, J.T. , 1997, " State Space Modelling and High Performance Computing of Multivariate Time Series " , Eighth SIAM Conference on Parallel Processing for Scientific Computing.
- Bottura, C.P.; Barreto, G.; Bordon, M.J. e Costa Filho, J.T., 1997, " Séries Temporais No Espaço de Estado: Modelamento E Computação De Alto Desempenho " , XIV - Congresso Brasileiro de Engenharia Mecânica, Bauru.
- Bottura, Celso P.; Tamariz, Annabell D.R.; Barreto, Gilmar & Fonseca Neto, Joao Viana da, 1999, " A modified-schur method for solving algebraic Riccati equations " , 38th IEEE Conference on Decision and Control.
- Caines, Peter E., 1988 " Linear Stochastic Systems" , Wiley.
- Golub, Gene H. and Ortega, James M., 1993, " Scientific Computing : An Introduction with Parallel Computing " , Academic Press.
- Jainandung K. and Deprettrere, E. F., 1989, " A New Class Of Parallel Algorithms For Solving Systems Of Linear Equations " , SIAM J.Sci.Stat.Comput. Vol.10, No.5, 880-912.
- Moonen, Marc; Moor, Bart De ; Vandenberghe, Lieven and Vandewalle, Joos, 1989, " On-and Off-Line Identification Of Linear State-Space Models " , INT.J. Control, Vol.49, No.1, 219-232.
- Overschee, Peter van and Moor, Bart De, 1996, " Subspace Identification For Linear Systems " , Kluwer Academic Publishers.
- Vaughan, David R., 1970, " A Nonrecursive Algebraic Solution for the Discrete Ricatti Equation " , IEEE Transactions on Automatic Control, Vol. AC-15, No.5, 597-599.
- Verhaegen, Michel and Dewilde, Patrick, 1992, " Subspace Model Identification " , Int.J. Control, Vol.56, n.5, 1187-1210.
- Wiener, Nobert, 1949 " Extrapolation, interpolation, and smoothing of stationary time series, with engineering applications " , New York, Wiley.
- Willems, Jan C., 1986, " From Time Series to Linear System - Part I. Finite Dimensional Linear Time Invariant Systems, Automatica, Volume 22, No.5, 561-580 and " Part II. Exact Modelling " , Automatica, Volume 22, No.6, 675-694.
- Willems, Jan C., 1987, " From Time Series to Linear System - Part III. Approximate Modelling " , Automatica, Volume 23, No.1, 87-115.

An Approach to State Space Computational Modeling and Prediction of Time Series in Parallel and Distributed Computers

Abstract. *In this work the algorithm of Masanao Aoki for state space modeling of time series in subspaces of the state space is treated in numerical and parallel and distributed computational aspects. The parallel solution of the Riccati equation due to the huge computational effort it requires receives a special attention. For model evaluation, short time predictions, where a central role is played by a Kalman filtering approach are tested and some results are presented.*

Keywords: *Subspace Method, Time Series, System Identification, Realization, High Performance Computation*

PETRI NET TOOLS: UM SISTEMA GRÁFICO E ABERTO DE EDIÇÃO E SIMULAÇÃO

João Borsoi Soares

Universidade de São Paulo – USP
Escola de Engenharia de São Carlos – EESC
Núcleo de Manufatura Avançada - NUMA
Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
E-mail: joao.soares@bigfoot.com

Andrea Ribari Yoshizawa da Silva

Universidade de São Paulo – USP
Escola de Engenharia de São Carlos – EESC
Núcleo de Manufatura Avançada - NUMA
Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
E-mail: arys@sc.usp.br

Patrícia Ferraz

Universidade de São Paulo – USP
Escola de Engenharia de São Carlos – EESC
Núcleo de Manufatura Avançada - NUMA
Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
E-mail: pferraz@sc.usp.br

Ricardo Yassushi Inamasu

Centro Nacional de Pesquisa e Desenvolvimento de Instrumentação Agropecuária – CNPDIA
Empresa Brasileira de Pesquisa Agropecuária – EMBRAPA
Rua Quinze de Novembro, 1452 - São Carlos/SP – Brasil CEP 13561-160
E-mail: ricardo@cnpdia.embrapa.br

Arthur José Vieira Porto

Universidade de São Paulo – USP
Escola de Engenharia de São Carlos – EESC
Núcleo de Manufatura Avançada - NUMA
Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
E-mail: ajvporto@sc.usp.br

Resumo. Este trabalho apresenta a mais recente versão do software Petri Net Tools 2000, desenvolvido pelo Grupo de Simulação do Núcleo de Manufatura Avançada (NUMA), na EESC-USP. É composto de uma breve descrição das características do sistema e ambiente de funcionamento, e uma revisão dos editores e simuladores mais comuns encontrados no mercado. A característica principal do novo editor é baseada no conceito de frameworks orientados a objetos, o qual habilita o usuário a desenvolver novas funcionalidades em diferentes modelos de funcionamento denominados templates. Desta forma, o editor prove um ambiente poderoso para o desenvolvimento e validação de redes de Petri interpretadas.

Palavras chave. Petri, framework, simulador.

1. Introdução

A rede de Petri e suas extensões consistem em poderosas ferramentas de modelagem gráfica e matemática, aplicáveis a uma grande variedade de sistemas, inclusive para a análise e controle de Sistemas e Eventos Discretos (SEDs), conforme pode ser visto em Miyagi (1993).

Dentre as diversas aplicações relacionadas, podemos citar a utilização para modelar componentes de Sistemas Flexíveis de Manufatura (FMS) (Cecil et al., 1992) e processos de fabricação (Porto, 1990). Através dela é possível representar graficamente os estados dinâmicos (estados que podem ser assumidos durante o processo em questão) dos elementos de um sistema e seus eventos relacionados, como intertravamento de máquinas, coordenação de máquinas (robô x máquinas CN x AGV), sequenciamento da produção, sincronismo de operações de montagem, etc.

Algumas das características mais interessantes da rede de Petri, apresentadas por Murata (1989), são:

- Visualização através de representação gráfica;
- Modelagem através de vários níveis de abstração;
- Modelagem de sistemas paralelos e concorrentes;
- Sincronização de eventos;
- Verificação de propriedades de forma sistemática.

Entretanto, a rede de Petri possui algumas limitações tais como a grande profusão de elementos gráficos em modelagem de sistemas complexos. Para contornar as limitações, existem muitas propostas de extensões, tais como: rede de Petri colorida (Jensen, 1980; Jensen, 1992 e Jensen, 1994); rede de Petri modular (Inamasu, 1995; Inamasu e Porto, 1997; Jeng, 1997; Proth et al, 1997 e Volgler, 1992); rede de Petri interpretada como SFC (Sequential Flow Chart), MFG (Marked Flow Graph) e PFS (Production Flow Schema) (Home, 2001; Miyagi et al, 1989; Miyagi, 1993 e Zaytoon, 1996); dentre outras.

Apesar da relativa facilidade de uso das redes de Petri, implementar softwares de edição, análise e simulação de modelos representados em rede de Petri e suas extensões é uma tarefa árdua. No entanto, existe uma grande semelhança entre todos os tipos de rede de Petri e extensões que viabiliza esforços em se reutilizar estruturas de implementação.

Do ponto de vista matemático, as redes de Petri interpretadas podem, por exemplo, ser descritas enquanto redes de Petri Lugar/Transição. Desta forma, ferramentas de análise e simulação implementadas para rede de Petri Lugar/Transição podem ser reaproveitadas para as redes de Petri interpretadas. Algumas características relativas aos softwares de edição também podem ser reaproveitadas, como o manuseio e visualização de objetos gráficos, controle de zoom, controle de linhas de grade, armazenamento do modelo em arquivos, impressão, etc.

Em função disto, torna-se de grande importância e necessidade a construção de uma ferramenta de software flexível, que promova o reaproveitamento de código e projeto, e que permita a inclusão de novas funcionalidades, como por exemplo, a implementação de novas propostas de extensões. A esse tipo de ferramenta denomina-se software aberto.

Os principais benefícios de um software aberto aplicado ao desenvolvimento de ferramentas para edição e análise de redes de Petri e suas extensões são:

- Promove o processo contínuo de desenvolvimento do software, a partir da inclusão e aperfeiçoamento de funcionalidades do software;
- Reduz o tempo e custo de futuras implementações, a partir do reaproveitamento de funcionalidades teoricamente já testadas e validadas;
- Viabiliza a implementação da edição de redes de Petri interpretadas e encoraja o desenvolvimento de futuras propostas baseadas em redes de Petri;
- Viabiliza a integração com sistemas externos.

No campo da engenharia de software, a metodologia de desenvolvimento de sistemas orientados a objeto tem se mostrado uma poderosa ferramenta para implementação de sistemas complexos devido às seguintes características, apresentadas por Khoshafian (1990):

- Modelagem do mundo real mais próxima à ótica do usuário;
- Facilidade de interação com ambiente computacional, usando metáforas familiares;
- Construção de componentes reutilizáveis de software e facilidade de expandir bibliotecas nos módulos do software;
- Facilidade de implementar modificações e extensões em componentes sem a necessidade de uma exaustiva re-codificação.

Além disso, a orientação a objeto tem sido vista como uma poderosa ferramenta de auxílio para promover a reutilização de código e projeto, na qual técnicas de utilização de padrões de projeto (design patterns) (Gamma et al, 1997) e *frameworks* (Johnson e Foote, 1988; Johnson e Russo, 1991) viabilizam a implementação de softwares abertos.

Este trabalho realiza uma breve revisão da rede de Petri e das principais ferramentas de edição e análise existente no mercado. Posteriormente apresenta-se a mais recente versão do editor e simulador Petri Net Tools 2000, desenvolvido pelo Grupo de Simulação do Núcleo de Manufatura Avançada (NUMA), da Escola de Engenharia de São Carlos (EESC), Universidade de São Paulo (USP). O Petri Net Tools 2000 consiste num *framework* de edição e análise gráfica de redes de Petri, e possui características de um software aberto, o qual viabiliza o processo contínuo de desenvolvimento deste, promovendo a inclusão, reutilização e aperfeiçoamento de funcionalidades.

2. Rede de Petri

A rede de Petri foi introduzida em 1962 por Carl Adam Petri, e consiste numa ferramenta gráfica e matemática extremamente efetiva para a modelagem e análise de Sistemas de Eventos Discretos (SEDs).

Além de permitir a modelagem e representação gráfica de uma grande variedade de sistemas, a rede de Petri possui uma formalização matemática bem definida e técnicas de análise já consagradas, fundamentais na validação do modelo. Dentre as técnicas existentes pode se destacar a verificação de propriedades comportamentais do modelo, determinação da árvore de alcançabilidade, uso da matriz de incidência e equação de estado, análise de invariantes, entre outras (Murata, 1989; Al-jaar e Desrochers, 1995).

2.1. Elementos estruturais e suas inter-relações

A rede de Petri consiste num grafo direcionado, com peso e bipartido, composto por dois elementos estruturais: lugares e transições. O lugar é representado graficamente por um círculo e a transição por uma barra. Os elementos estruturais são utilizados para criar o modelo, no qual arcos orientados conectam lugares a transições e transições a lugares. Estes arcos podem ser rotulados com um valor inteiro positivo, indicando o peso do arco. Um arco de peso k pode ser interpretado como k arcos paralelos.

2.2. Marcação e seu comportamento dinâmico

Cada lugar pode possuir marcas (fichas), indicando um estado. A marcação do sistema (estado), é denotada por um vetor M de dimensão igual ao número de lugares do modelo. O p -ésimo componente de M , indicado por $M(p)$ consiste no número de fichas do lugar p . O vetor M_0 consiste no vetor de marcação inicial, isto é, estado inicial do sistema.

Na modelagem, usando o conceito de condições e eventos, o lugar pode ser interpretado como uma condição e a transição como um evento. Uma transição (evento) possui um certo número de lugares de entrada e saída, representando as pré e pós-condições, respectivamente. A presença de uma ficha num lugar indica que a condição deste é verdadeira.

O comportamento de diversos sistemas pode ser descrito em termos de seus estados e de sua respectiva mudança ou transição para outros possíveis estados. Na rede de Petri, a mudança de estado ocorre de acordo com a regra de habilitação e disparo das transições.

A ocorrência de um evento é denominada disparo de transição. A regra de habilitação e disparo de transições é:

- Uma transição está habilitada para disparar se para todo lugar de entrada, que possui arco para a transição, possuir um número de marcas maior ou igual ao peso do arco;
- Uma transição habilitada pode disparar ou não, dependendo se o evento realmente ocorrer;
- No disparo de uma transição habilitada, todo lugar que possui um arco para a transição tem seu número de marcas reduzido pelo valor do peso deste arco, e todo lugar que possui um arco vindo da transição tem seu número de marcas acrescido do valor do peso deste arco.

Uma transição sem lugares de entrada é denominada transição fonte, e a sem lugares de saída é chamada de transição sumidouro. Uma transição fonte está sempre habilitada, e o disparo de uma transição sumidouro consome fichas, mas não produz nenhuma.

A Fig. (1) esboça um exemplo de modelo feito em rede de Petri representando a montagem de uma peça a partir de outras duas peças (peça 1 e peça 2). Segundo o modelo, para montar uma peça são utilizadas duas peças do tipo 1 e uma peça do tipo 2.

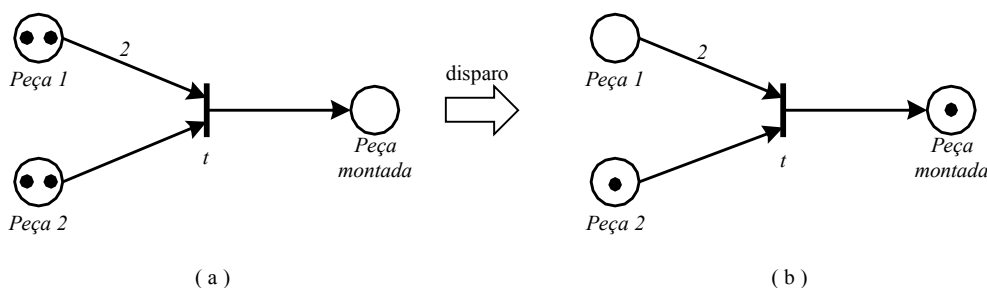


Figura 1. Exemplo de rede de Petri Lugar/Transição e disparo.

Os lugares “Peça 1” e “Peça 2” representam respectivamente o número de peças do tipo 1 e do tipo 2 disponíveis. De forma análoga o lugar “Peça montada” representa o número de peças montadas. A transição “t” indica o evento de montagem.

A Fig. (1) mostra o modelo em dois estados distintos. A Fig. (1a) representa o estado inicial, no qual existem duas peças do tipo 1 e do tipo 2, e nenhuma peça montada. A Fig. (1b) representa o estado do sistema após a montagem de uma peça, indicando o disparo da transição “t” a partir do estado inicial.

2.3. Definição formal

A rede de Petri é definida por Murata (1989) como uma quintupla, $PN = (P, T, F, W, M_0)$ onde:

$P = \{p_1, p_2, \dots, p_m\}$ é um conjunto finito de lugares,

$T = \{t_1, t_2, \dots, t_n\}$ é um conjunto finito de transições,

$F \subseteq (P \times T) \cup (T \times P)$ é um conjunto de arcos (relação de fluxo),

$W : F \rightarrow \{1, 2, 3, \dots\}$ é a função peso,

$M_0 : P \rightarrow \{0, 1, 2, 3, \dots\}$ é a marcação inicial,

$P \cap T = \emptyset$ e $P \cup T \neq \emptyset$.

2.3. Ferramentas de edição e análise

Atualmente existem diversas ferramentas de edição e análise de redes de Petri no mercado, com diferentes características e funcionalidades. Na *World Wide Web* a página do *Petri Nets Tool Database* (disponível em <http://www.daimi.aau.dk/PetriNets/tools/db.html>) relaciona as seguintes ferramentas:

Petri Nets Tool Database (2001) relaciona as seguintes ferramentas:

ALPHA/Sim: Esta ferramenta foi desenvolvida para trabalhar com redes de Petri de Alto Nível, Estocásticas e Temporais, nos ambientes Windows e Sun. Ela permite ao usuário construir graficamente um modelo de rede de Petri, entrar com dados, executar a simulação e verificar os resultados obtidos. Não é necessário programar para construir os modelos, mas estes podem ser ligados a códigos externos, escritos em qualquer linguagem que possa ser compilada na plataforma em que se estiver trabalhando. Tem sido utilizada em um grande número de aplicações, por exemplo: operações militares, sistemas de manufatura, controle de filas, e está disponível comercialmente.

ANARCO: Trabalha com redes de Petri de Lugar/Transição e Alto Nível, no ambiente MS-DOS. Consiste numa ferramenta de análise de “liveness”, “cleanness” e “safeness”, em redes de Petri Coloridas. É disponível gratuitamente para instituições acadêmicas.

Artifex: Modela e analisa sistema de eventos discretos, com base em redes de Petri de Alto Nível e Temporais, através de uma linguagem de modelação gráfica que adota conceitos de orientação a objetos. Códigos em C ou em C++ podem ser escritos como parte do modelo, na descrição de dados e operações, o que permite virtualmente a integração dessa ferramenta com qualquer outro software já existente. Funciona em diversos ambientes, permitindo reduções significantes no tempo de desenvolvimento de sistemas e aumento na qualidade do planejamento, visto que fornece uma maneira fácil e confiável de apresentar validação no comportamento e nas funcionalidades de um sistema. Está disponível comercialmente no mercado.

CodeSign: Disponível para os ambientes Windows, Sun e Macintosh gratuitamente para instituições acadêmicas, essa ferramenta trabalha com redes de Petri de Alto Nível e Temporais. Utiliza transições temporais para modelar componentes do sistema. Os dados e a hierarquia são tratados estritamente com orientação a objetos. Durante a simulação, utiliza a semântica da “pilha ordenada”. Foi implementada em “Parcplace VisualWorks Smalltalk” e, os arquivos binários necessários devem ser obtidos na página da empresa.

DaNAMICS: Desenvolvida em Java, essa ferramenta suporta redes de Petri de Alto Nível e Estocásticas, e foi criada para auxiliar e facilitar a criação de modelos GSPN ou CPN. Possui características importantes, como, por exemplo, modelar hierarquicamente sistemas através de sub-redes. Está disponível comercialmente.

Design/CPN: Suporta redes de Petri de alto nível e temporais. Foi desenvolvida para os ambientes Linux, HP, Sun e Silicon Graphics. O editor da ferramenta possibilita a construção e modificação de modelos CPN, além de fazer a verificação da sintaxe dos mesmos, enquanto que o simulador permite uma simulação dos modelos, de maneira interativa e automática. Possui a capacidade de construir e analisar as árvores de alcançabilidades (grafos) dos modelos CPN.

EDS Petri Net Tool: Esta ferramenta, disponível comercialmente, trabalha com redes de Petri de Lugar/Transição, Temporais, Estocásticas e Coloridas, em ambiente Windows e OS/2. Fornece as facilidades usuais de edição gráfica, e permite o controle da complexidade da rede, através da inclusão de nós que representam sub-redes. O uso da linguagem é auxiliado por um ambiente integrado de edição e compilação de subprogramas, e o exame do comportamento da rede é feito através dos modos de operação de Monte Carlo. A interface dessa ferramenta permite que dados sejam lidos ou gravados em arquivos, ou com o auxílio de bases de dados comuns à PC's.

ELSIR: Disponível para os ambientes Sun, HP e Linux, de maneira comercial, essa ferramenta serve para redes de Petri de Alto Nível, de Lugar/Transição, Estocásticas e Temporais. A ELSIR permite análises estruturadas, descrição de dados e de tempo, além de fornecer instâncias dos processos genéricos modelados. Pode ser usada para: análise de performance de um sistema, estudos de dimensionamento de recursos e de confiabilidade e, pode gerar relatórios com texto e curvas.

GDToolkit: Serve para redes de Petri de Alto Nível e de Lugar/Transição, e trabalha em ambientes Windows, Sun e Linux. É uma biblioteca para layout automático, capaz de desenhar vários tipos de diagramas além das redes de Petri. É gratuito para as instituições acadêmicas.

Moses Tools Suite: Desenvolvida em Java, está disponível gratuitamente. Trabalha com redes de Petri de Alto Nível, Estocásticas e Temporais. Facilita a especificação de notações visuais, onde estas são utilizadas para configurar o editor, o compilador e o animador gráfico. As redes de Petri são compiladas em classes de Java, e podem ser executadas em um animador gráfico ou, combinadas com outro software também desenvolvido em Java.

PetriSim: Trabalha em ambiente MS-DOS, e suporta redes de Petri de Alto Nível, Temporais e de Lugar/Transição. Esta ferramenta é capaz de trabalhar em mais de uma rede simultaneamente e possui algumas opções como: cor, tamanho, estilos de linha, etc. O usuário pode criar procedimentos em Turbo Pascal 7, a fim de modificar o comportamento da rede e, gerar randomicamente a duração do disparo das transições. É gratuita.

PNtalk: Suporta redes de Petri de Alto Nível e Temporais, e trabalha em ambiente Windows e Sun. O sistema e a linguagem PNtalk são baseados em redes de Petri orientadas a objetos. Essa ferramenta é implementada em Smalltalk, e é possível juntar os dois tipos de objetos (da ferramenta e do Smalltalk) num mesmo modelo. O usuário pode trabalhar separadamente com redes particulares de classes particulares, durante a criação e depuração de uma rede. Está disponível gratuitamente no mercado.

PN-tools: Esta ferramenta serve para redes de Petri de Lugar/Transição e Estocásticas, além de também suportar redes que se modificam automaticamente, redes com arcos inibidores, redes de prioridade e redes relacionais. Está disponível gratuitamente para ambientes MS-DOS e Windows. O PN-tools é um pacote computacional totalmente integrado que auxilia projetistas e analistas nas funções de editar, validar, analisar e executar redes de Petri padrão ou estocásticas generalizadas. As redes podem ser estruturadas hierarquicamente de maneira a facilitar sua leitura e entendimento.

TimeNET: Funciona em plataforma Windows e Linux, e suporta redes de Petri de Alto Nível, de Lugar/Transição, Estocásticas e Temporais. É um pacote computacional para modelagem e avaliação de SPNs, nas quais os tempos de disparo das transições podem ser determinísticos, ou distribuídos de maneira exponencial ou mais geral. É possível criar uma interface gráfica com o usuário. Um componente especial de modelagem e análise de sistemas de manufatura está disponível para redes de Petri Coloridas. A análise estacionária é baseada na teoria Markoviana, mas também existe um componente de simulação para redes não-Markovianas. A ferramenta oferece técnicas diferentes para acelerar a simulação dos experimentos. Está disponível gratuitamente para instituições acadêmicas.

Visual Object Net ++: Disponível gratuitamente para ambiente Windows. Trabalha com redes de Petri de Lugar/Transição e Temporais. É uma ferramenta CAE que suporta redes que são uma mistura de redes de Petri contínuas com redes de Petri de eventos discretos. O objetivo dessa ferramenta é estudar o comportamento e as características de uma classe de redes de Petri híbridas.

Visual SimNet: Suporta redes de Petri de Alto Nível, de Lugar/Transição, Estocásticas e Temporais. É uma ferramenta gratuita e, funciona nas plataformas MS-DOS e Windows. Basicamente, é um sistema de análise de sistemas de eventos discretos baseado em redes de Petri Coloridas Estocásticas. Possui um editor gráfico com simulação animada, e é capaz de apresentar resultados de distribuição e de alcançabilidade da rede graficamente. Pode exportar esses resultados para outros softwares, como por exemplo, o Látex e o PostScript.

3. Petri Net Tools 2000

O Petri Net Tools 2000 consiste num *framework* (Johnson e Foote, 1988; Johnson e Russo, 1991) de edição e análise gráfica de redes de Petri, desenvolvido pelo Grupo de Simulação do Núcleo de Manufatura Avançada (NUMA) da Escola de Engenharia de São Carlos (EESC), da Universidade de São Paulo (USP). O editor encontra-se disponível na *World Wide Web* na página <http://www.petrinet.sc.usp.br>.

Este editor possui características de um software aberto, o qual viabiliza o processo contínuo de desenvolvimento deste, promovendo a inclusão, reutilização e aperfeiçoamento de funcionalidades.

3.1. Arquitetura aberta

De uma forma geral, o funcionamento do editor é composto de dois módulos, conforme pode ser visualizado na Fig. (2).

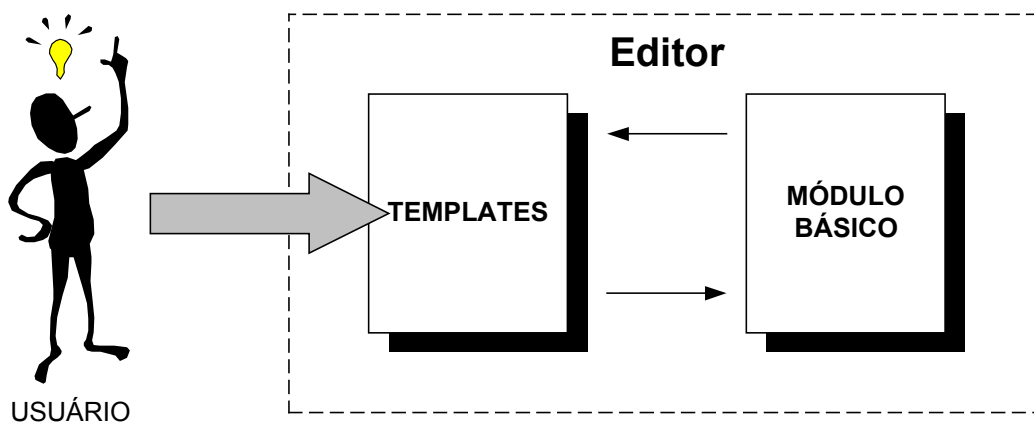


Figura 2. Modelo geral de funcionamento do editor.

O módulo básico é composto por um conjunto de ferramentas genéricas de edição de modelos gráficos, e por ferramentas de análise de redes de Petri Lugar/Transição. Estas ferramentas são utilizadas pelo usuário indiretamente através dos *templates*.

Cada *template* consiste numa forma diferente e personalizada de uso destas ferramentas, isto é, os *templates* consistem em diferentes modos de funcionamento do editor. O sistema possui a característica de sistema aberto, uma vez que o usuário pode implementar, alterar, adicionar e remover *templates* do editor.

Os *templates* podem ser utilizados em diversos contextos. Podem ser utilizados para fornecer novas funcionalidades ao editor, como por exemplo, novos recursos gráficos, ou novas metodologias de análise. Mas eles também podem ser

utilizados para incorporar novas ferramentas de modelagem, que podem utilizar elementos gráficos diferentes e utilizar as ferramentas de análise e simulação de redes de Petri já implementadas.

Neste contexto, os *templates* podem ser utilizados para fornecer ao editor a possibilidade de trabalhar com redes de Petri interpretadas, como o MFG/PFS e SFC (Home, 2001; Miyagi et al, 1989; Miyagi, 1993 e Zaytoon, 1996).

O Petri Net Tools 2000, por se tratar de um sistema aberto, possui dois tipos distintos de usuários. Um deles é o usuário comum, que utiliza os *templates* já implementados do editor para elaborar um modelo com finalidades de análise e simulação. O outro consiste num programador que utiliza o editor para implementar, modificar ou reaproveitar funcionalidades, através da geração de novos *templates*.

3.2. Aspectos operacionais

Os *templates*, como relatado anteriormente, consistem em modos de funcionamento do editor. Ao iniciar o editor, o usuário deve escolher com qual *template* deseja trabalhar. O funcionamento do editor varia de acordo com o *template* escolhido. De uma forma geral, o editor consiste num aplicativo do tipo MDI (Multiple Document Interface), capaz de trabalhar com diversos documentos simultaneamente. Cada documento constitui num modelo representável por uma rede de Petri Lugar/Transição. A Figura 3 mostra a tela do Petri Net Tools 2000 executando um *template* para edição de modelos de MFG/PFS.

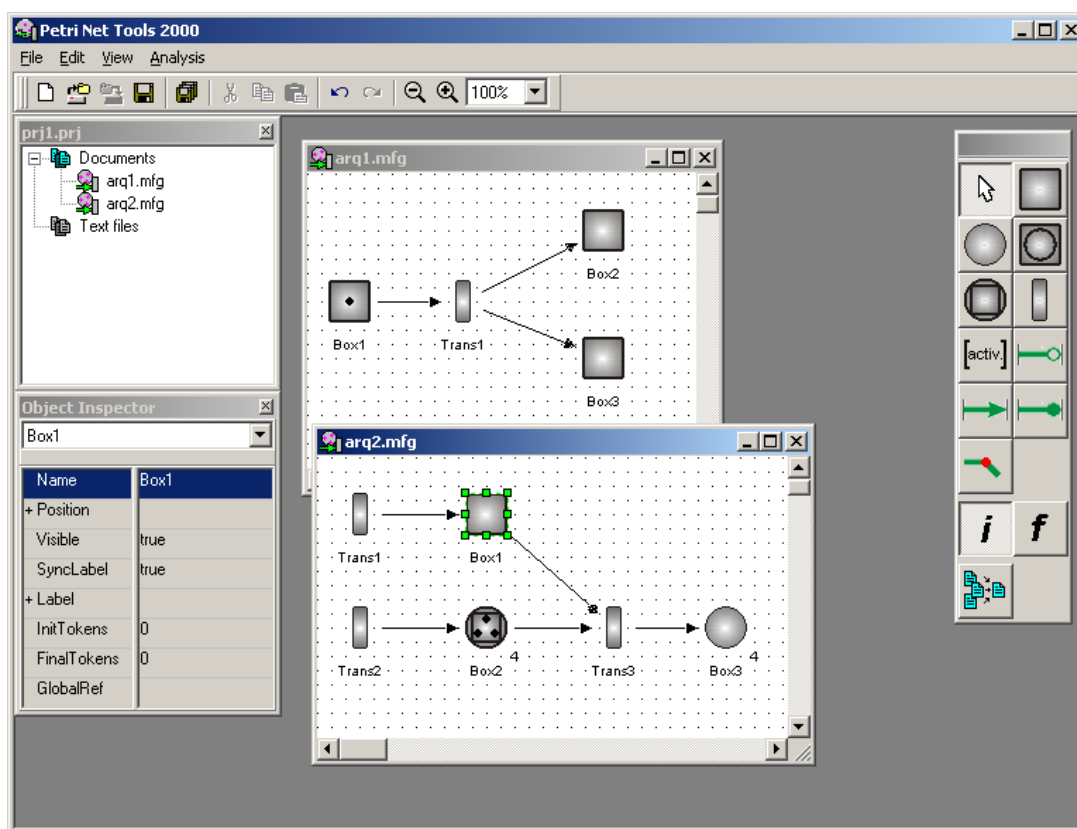


Figura 3. Tela Petri Net Tools 2000 com um *template* para MFG/PFS

Os aspectos operacionais de funcionamento do editor são formados por um grupo de características genéricas, compartilhadas ou não pelos *templates*, e pelas características específicas de cada *template*. Dentre as características genéricas pode-se citar:

- Ferramentas de edição gráfica, compostas pela barra de ferramentas, janela de inspeção (acesso as variáveis dos objetos gráficos) e janelas de edição (posicionamento dos objetos com auxílio de controle de zoom e pontos de grade);
- Controle de documentos, no qual os modelos podem ser salvos, abertos, copiados e impressos;
- Controle de projeto, no qual documentos relacionados podem ser organizados na forma de um projeto;
- Ferramentas de análise, incluindo a geração da matriz de incidência, geração de árvore de alcançabilidade, análise de propriedades comportamentais (alcançabilidade, limitabilidade, e vivacidade) e análise usando invariantes.

Um exemplo de uso desta funcionalidade pode ser visto no *template* para rede de Petri Lugar/Transição, onde o conceito de rede de Petri modular (Inamasu, 1995; Inamasu e Porto, 1997; Jeng, 1997; Proth et al, 1997 e Volgler, 1992) é implementado de forma que todos os arquivos do projeto constituem num único modelo. A Fig. (4) exhibe um

exemplo de rede de Petri modular editado no Petri Net Tools 2000, com três arquivos e dois lugares comuns a mais de um arquivo, denominados “Entrada”, e “Saída”.

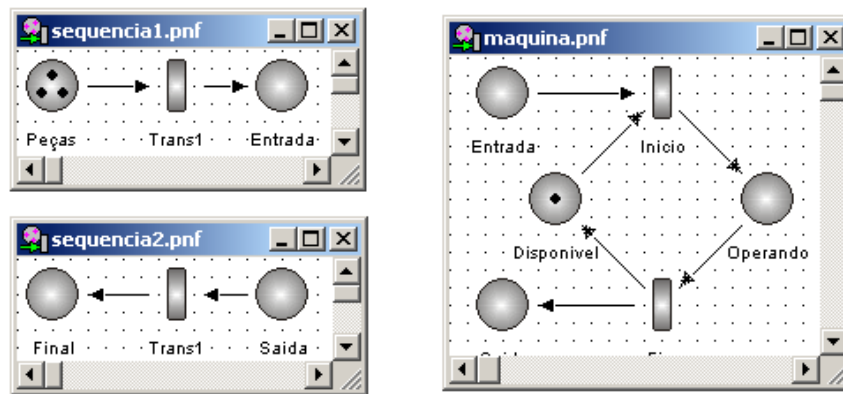


Figura 4. Exemplo de rede de Petri modular

Neste exemplo, o arquivo “maquina.pnf” modela a operação de uma máquina sobre uma peça, e os arquivos “seqüência1.pnf” e “seqüência2.pnf” modelam a disponibilidade e saída das peças na máquina.

4. Conclusões

A rede de Petri, por se tratar de uma poderosa ferramenta de modelagem e análise, é utilizada numa grande variedade de sistemas. Desta forma, conforme relatado na Seção 2.3, existem diversas ferramentas de edição e análise, cada qual apresentando funcionalidades específicas para um âmbito de solução. Por exemplo, algumas dessas ferramentas trabalham com redes de Petri Coloridas, outras permitem redes de Petri Temporais, redes de Petri para sistemas contínuos, muitas permitem a programação e conseqüente interação com outros ambientes, e assim por diante.

Neste contexto, o Petri Net Tools 2000 apresenta uma grande contribuição uma vez que consiste numa ferramenta aberta que permite a inclusão de novas funcionalidades. Além disso, a estrutura de *templates* permite trabalhar de diversas formas no mesmo editor. Assim, o grande potencial desta ferramenta não está nas funcionalidades que ela apresenta e sim no processo contínuo de desenvolvimento, no qual cada usuário pode personalizar a ferramenta de acordo com o âmbito de aplicação desejado.

Apesar do grande potencial desta ferramenta, algumas considerações são necessárias. A implementação dos *template* requer o domínio da metodologia de desenvolvimento de sistema orientado a objetos, e também das técnicas de reutilização de código, padrões de projeto e *frameworks*. Além disso, o conhecimento do próprio *framework* implementado, isto é, detalhes da implementação do próprio editor também são necessários.

5. Agradecimentos

À FAPESP e CAPES pelo apoio dado ao presente trabalho.

6. Referências

- Al-Jaar, R.Y., Desrochers, A., 1995, “Applications of petri nets in manufacturing systems: modeling, control, and performance analysis”, Proceedings of IEEE, New York.
- Cecil, J.A., Srihari, K., Emerson, C.R., 1992, “A review of petri-net applications in manufacturing”, The International Journal of Advanced Manufacturing Technology, Vol. 7, pp. 168-177.
- Gamma, E.; Helm, R.; Johnson, R.; Vlissides, J. “Design patterns: elements of reusable object-oriented software”, Addison-Wesley, MA, 1997.
- Inamasu, R. Y. “Modelo de FMS: uma plataforma para simulação e planejamento. São Carlos, 1995”, Tese (Doutorado), Escola de Engenharia de São Carlos, Universidade de São Paulo.
- Inamasu, R. Y.; Porto, A. J. V. “Plataforma para sequenciamento de processo alternativo de fabricação baseado em rede de Petri”, Congresso iberoamericano de ingeniería mecánica, Instituto Superior Politécnico José Antonio Echeverría, La Habana, Cuba, 1997.
- Jeng, M. D. “Petri nets for modeling automated manufacturing systems with error recovery”, IEEE Transactions on Robotics and Automation, Vol. 13, No. 5, pp. 752-760, 1997.
- Jensen, K. “Coloured petri nets and the invariant methods”, Theoretical Computer Science, Vol. 14, pp. 317-336, Amsterdam, 1980.
- Jensen, K. “Coloured petri nets: basic concepts, analysis methods and practical use”, Springer-Verlag, Berlin, 1992, Vol. 1.

- Jensen, K. "Coloured petri nets: basic concepts, analysis methods and practical use", Springer-Verlag, Berlin, 1994, Vol. 2.
- Johnson, R.; Foote, B. "Designing reusable classes", Journal of Object-Oriented Programming Vol. 1, No. 2, pp. 22-35, 1988.
- Johnson, R.; Russo, V. "Reusing object-oriented designs", United States, Urbana, Department of Computer Science, University of Illinois, 1991. (Tech Report UIUCDCS 91-1696)
- Khoshafian, A. R. "Object orientation: concepts, languages, databases, user interfaces", John Wiley & Sons, New York, 1990.
- Miyagi, P.E., 1993, "Controle de sistemas de eventos discretos: aplicação em automação industrial", Tese (Livre-Docência), Escola Politécnica da Universidade de São Paulo, São Paulo, Brasil.
- Miyagi, P.E., Hasegawa, K., Takahashi, K., 1989, "Mark flow graph (mfg) para modelamento e controle de sistemas de eventos discretos", Monografias em Automação e Inteligência Artificial - PMC/PEL/EPUSP, Vol. 1, No. 1, São Paulo, Brasil, pp.1-10.
- Murata, T., 1989, "Petri net: properties, analysis and applications", Proceedings of the IEEE, Vol. 77, No. 4, pp. 541-579.
- Porto, A.J.V., 1990, "Desenvolvimento de um método de integração do planejamento do processo de fabricação e do planejamento e controle da produção, baseado na flexibilidade do processo de fabricação", Tese (Doutorado) – Escola de Engenharia de São Carlos – Universidade de São Paulo, São Carlos, Brasil.
- Proth, J. M.; Wang, L.; Xie, X. "A class of petri nets for manufacturing system integration", IEEE Transaction on Robotics and Automation, Vol. 13, No. 3, pp. 317-326, 1997.
- Volgler, W. "Modular construction and partial order semantics of petri nets", Lecture Notes in Computer Science, Springer-Verlag, 1992.
- Zaytoon, J. "Specification and design of logic controllers for automated manufacturing systems", Robotics & Computer-Integrated Manufacturing, Vol. 12, No. 4, pp. 353-366, 1996.

PETRI NET TOOLS: A GRAPHICAL OPEN SYSTEM EDITOR AND SIMULATOR

João Borsoi Soares

Universidade de São Paulo – USP
 Escola de Engenharia de São Carlos – EESC
 Núcleo de Manufatura Avançada - NUMA
 Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
 E-mail: joao.soares@bigfoot.com

Andrea Ribari Yoshizawa da Silva

Universidade de São Paulo – USP
 Escola de Engenharia de São Carlos – EESC
 Núcleo de Manufatura Avançada - NUMA
 Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
 E-mail: arys@sc.usp.br

Patrícia Ferraz

Universidade de São Paulo – USP
 Escola de Engenharia de São Carlos – EESC
 Núcleo de Manufatura Avançada - NUMA
 Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
 E-mail: pferraz@sc.usp.br

Ricardo Yassushi Inamasu

Centro Nacional de Pesquisa e Desenvolvimento de Instrumentação Agropecuária – CNPDIA
 Empresa Brasileira de Pesquisa Agropecuária – EMBRAPA
 Rua Quinze de Novembro, 1452 - São Carlos/SP – Brasil CEP 13561-160
 E-mail: ricardo@cnpdia.embrapa.br

Arthur José Vieira Porto

Universidade de São Paulo – USP
 Escola de Engenharia de São Carlos – EESC
 Núcleo de Manufatura Avançada - NUMA
 Av. Trabalhador Sancarlene, 400 – São Carlos/SP – Brasil CEP 13566-590
 E-mail: ajvporto@sc.usp.br

Abstract. *This work presents the most recent version of the Petri Net Tools 2000 software, developed by the Advanced Manufacturing Center (NUMA) Simulation Group, at EESC-USP. It is composed by a brief description of the system's characteristics and working environment, and a revision of the most common editors and simulators encountered in the market. The*

new editor main characteristic is based in the concept of oriented object frameworks, which enables the user to develop new functionalities as different working templates. Therefore, the editor provides a powerful environment for the developing and validating of interpreted Petri nets.

Keywords. *Petri, framework, simulator.*

MONITORAMENTO DE DESGASTE EM FRESAS DE TOPO ATRAVÉS DA MEDIÇÃO DE TORQUE

Vinicius Leal Carneiro da Cunha

Escola Politecnica da USP
Alameda Prof . Mello de Moraes 2231
05508-900,SP,SP,Brasil

Lucas Antônio Moscato

Escola Politecnica da USP
Alameda Prof . Mello de Moraes 2231
05508-900,SP,SP,Brasil
lamoscat@usp.br 1

Resumo: A completa automação das operações de usinagem depende de um sistema capaz de monitorar as condições do processo. Neste trabalho foi desenvolvido um sistema para medição do torque na operação de fresamento de topo. Foi utilizado um sistema ótico modulado em frequência para a transmissão do sinal, garantindo grande imunidade à ruídos. A modulação em frequência permitiu a coleta do sinal usando apenas a porta paralela de impressora, através de um software escrito em assembler. Com este sistema, foram coletados sinais de usinagem e foi analisado o relacionamento do sinal de torque com o desgaste da ferramenta

Palavras chave: monitoramento, desgaste da ferramenta, fresamento, torque, modulação em frequência

1.Introdução

A elevada competitividade tornou a redução de custos uma atividade chave para o sucesso de qualquer atividade industrial. A completa automatização das operações de usinagem depende de um sistema capaz de monitorar o processo eficientemente, o que ainda depende intensa supervisão de operadores. A experiência tem mostrado que o monitoramento do processo, mesmo quando feito por operadores preparados, é muito sujeito a erros que interferem negativamente na qualidade e nos custos dos produtos. Grandes melhorias poderiam ser obtidas com o monitoramento automático destes processos, principalmente no que se refere ao estado de desgaste da ferramenta, evitando desperdícios com trocas prematuras da ferramenta ou falta de qualidade nos produtos com a troca tardia Byrne (1996), Dornfeld (1990), Rangwala (1989), Teixeira (1998) e Tlusty (1983).

2. Desenvolvimento do Sistema de Medição de Torque:

Os sistema se baseia no sistema de transmissão óptica do sinal modulado em frequência desenvolvido por [Ferreira 1988]. Este sistema foi aprimorado para permitir uma recepção absolutamente clara do sinal modulado em frequência, e assim conseguir a medição dinâmica do torque.

A figura 2 mostra o circuito utilizado para implementar a parte girante do sistema. Os elementos sensores são extensômetros colados na superfície da ferramenta, uma fresa de topo de aço rápido com duas entradas de 25 mm de diâmetro. Todo este circuito foi acondicionado em um invólucro metálico resistente e impermeável fixado

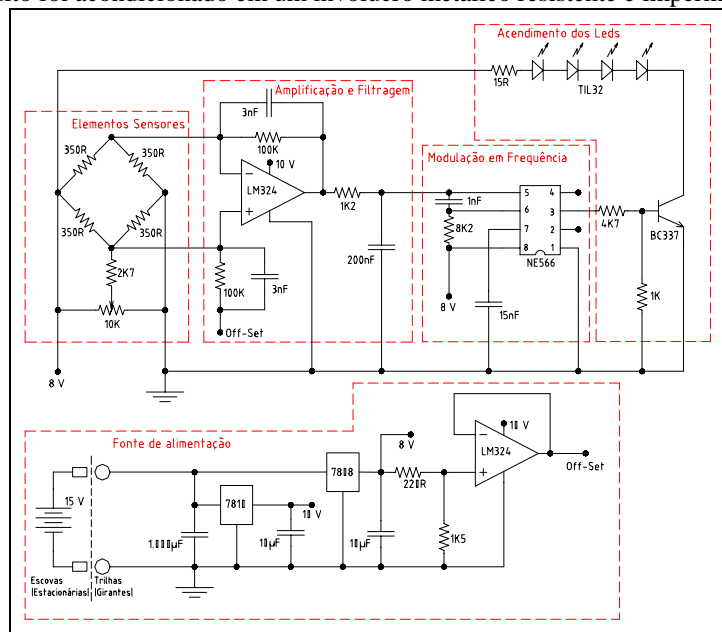


Figura 2: Circuito da Parte Girante

na própria fresa. O invólucro também forneceu proteção mecânica para os extensômetros e ajudou a proteger o sistema de ruídos externos. O circuito girante foi alimentado por escovas. A figura 3 mostra a estrutura da conexão óptica, baseada em um espelho elíptico. Este espelho tem a propriedade de focalizar os feixes luminosos vindos de um dos focos da elipse no outro foco. Num dos focos está o centro de rotação da ferramenta, no outro está o elemento receptor. Os leds emissores são instalados radialmente ao centro de rotação da ferramenta e defasados 90°. O posicionamento das escovas e do captador foi feito por uma estrutura auxiliar fixada ao canhão da fresadora. A figura 5 mostra toda a estrutura mecânica utilizada. O circuito receptor é mostrado na figura 4. O circuito receptor foi acondicionado em outra caixa metálica que também abrigou a fonte de alimentação para os dois circuitos. O elemento receptor, um fototransistor fixado no captador, foi conectado ao circuito receptor por um cabo protegido contra ruídos.

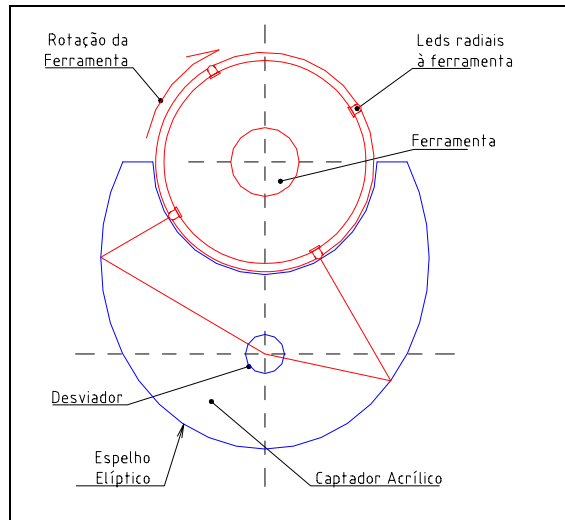


Figura 3: Conexão Óptica

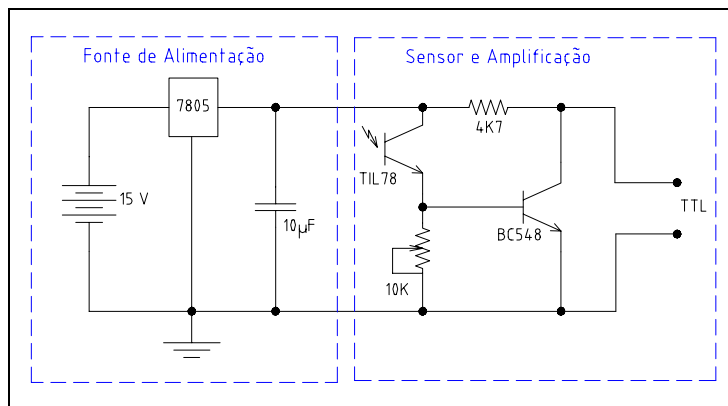


Figura 4: Circuito Receptor Estacionário

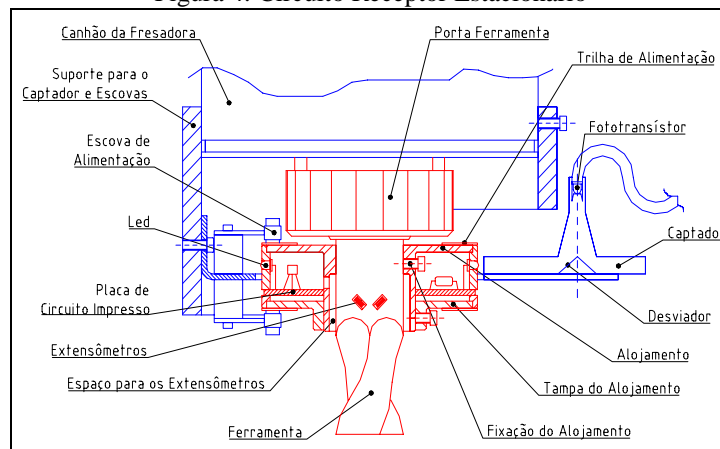


Figura 5: Estrutura mecânica utilizada para suportar os elementos

3. Amostragem do Sinal:

O circuito receptor fornece na sua saída um sinal com forma de onda quadrada no padrão TTL, com frequência proporcional ao torque na fresa. Este sinal pode ser lido diretamente pela porta paralela do PC, através do pino 10. Para

a leitura do sinal foi desenvolvido um software em assembler capaz de tratar a interrupção gerada a cada borda de descida do sinal colocado na porta paralela. O software registra o instante em que ocorre cada borda de descida do sinal através da gravação do valor de um contador de 16 bits que existe no PC, e que é decrementado a uma frequência constante de aproximadamente 1,2 MHz.

Sabendo-se o instante em que ocorreu cada borda de descida, é possível determinar a frequência entre cada par de bordas consecutivas. A frequência é proporcional ao torque. Estes dados não são suficientes para a análise dinâmica do sinal, pois precisa-se de dados amostrados a uma frequência constante. Para a análise dinâmica foi utilizado o seguinte método (figura 6): Como não existe um instante específico entre duas bordas onde se possa afirmar que ocorreu a frequência calculada com o intervalo de tempo entre as bordas, convencionou-se localizar a frequência calculada no ponto médio do intervalo. Com base na frequência de amostragem, define-se uma seqüência de instantes onde deve-se obter uma amostra do sinal. Estes instantes podem cair num ponto qualquer entre as duas bordas de descida consecutivas. Para obter a amostra neste ponto é feita a interpolação entre a frequência calculada com base em dois intervalos consecutivos delimitados por três bordas de descida consecutivas, adotando-se o ponto médio de cada intervalo como referência para as frequências que são a base para a interpolação.

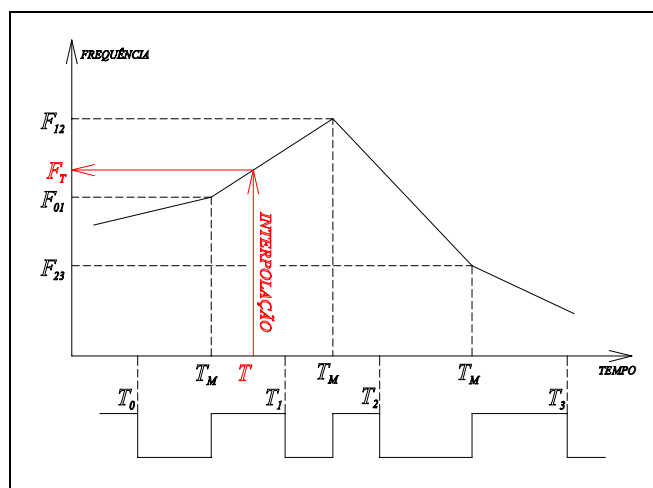


Figura 6: Método para amostragem do sinal modulado em frequência

4. Resultados e Análise do Desempenho:

A frequência do contador (1,2 MHz) limita a frequência de modulação do sinal, pois quanto mais próxima a frequência de modulação da frequência do contador, menor a resolução conseguida. Testes preliminares mostraram que seria suficiente observar a dinâmica do sinal até 500 Hz, ou seja, amostragem a 1.000 Hz. O sinal foi modulado a frequência acima de 2.500 Hz, obtendo-se um compromisso entre a dinâmica do sinal e a resolução, conforme demonstraram os testes de recuperação do sinal realizados introduzindo o sinal de um gerador de sinais antes de todo o sistema de modulação, transmissão e coleta do sinal. Outros testes foram realizados para analisar o desempenho do sistema no que se refere a nível de ruído, linearidade dinâmica e estática. Todos os testes demonstraram a efetividade do sistema de medição e de amostragem do sinal.

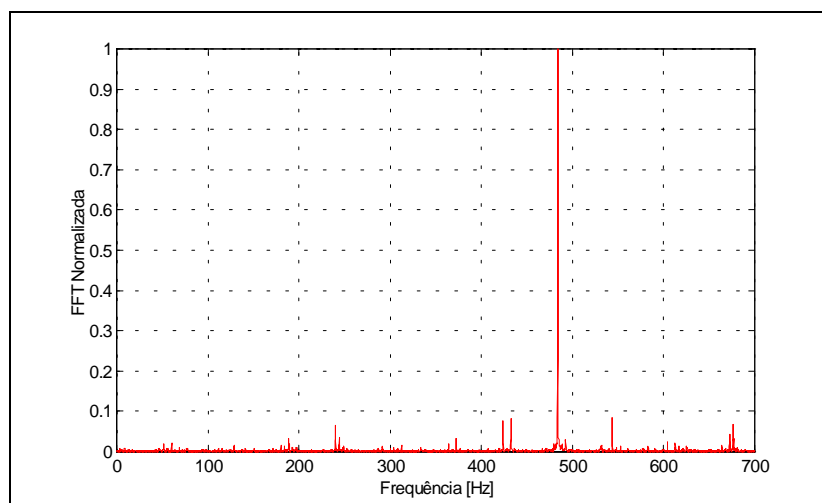


Figura 7: Recuperação do sinal senoidal introduzido por gerador antes da modulação em frequência

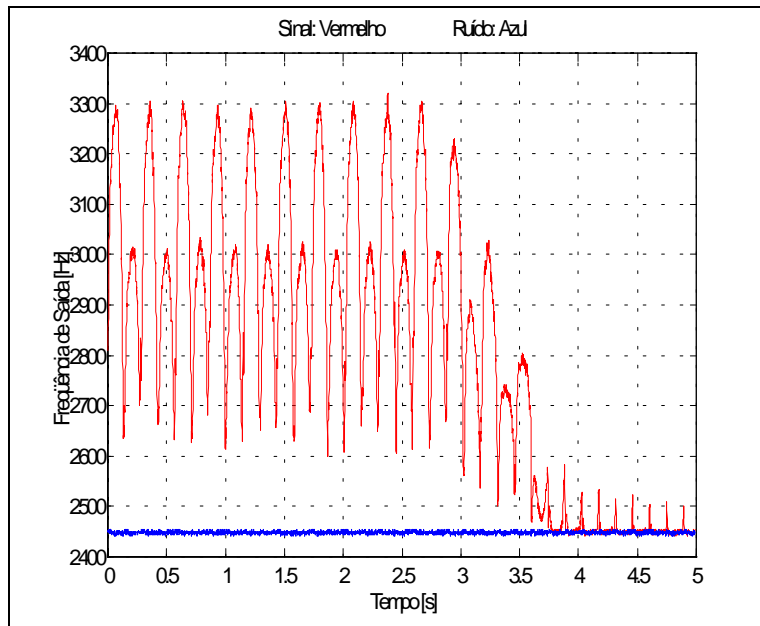


Figura 8: Sinal(vermelho) de torque em operação e ruído(azul) (obtido com a ferramenta girando em vazio)

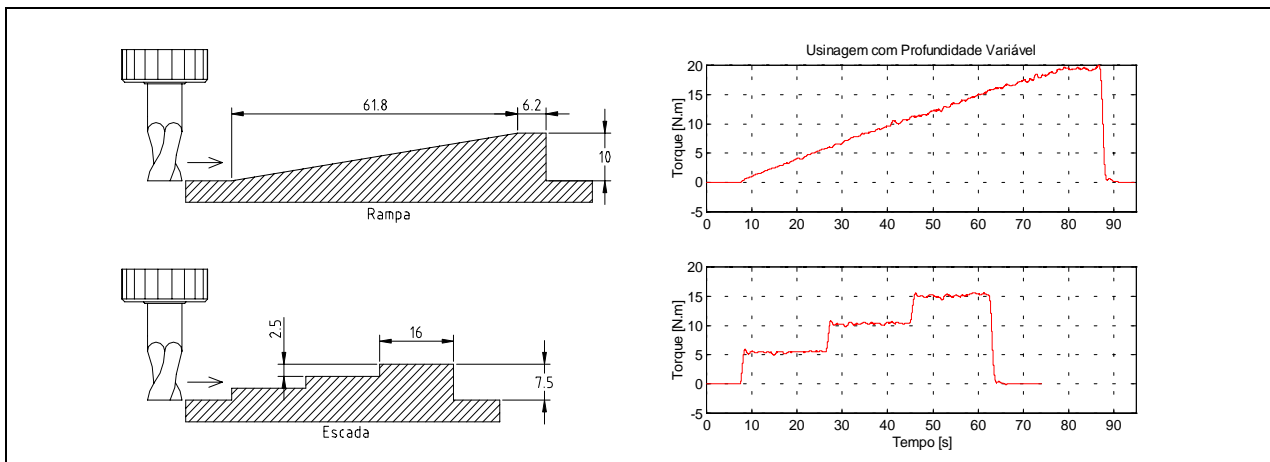


Figura 9: Linearidade Dinâmica – Testes usinando com profundidade variável

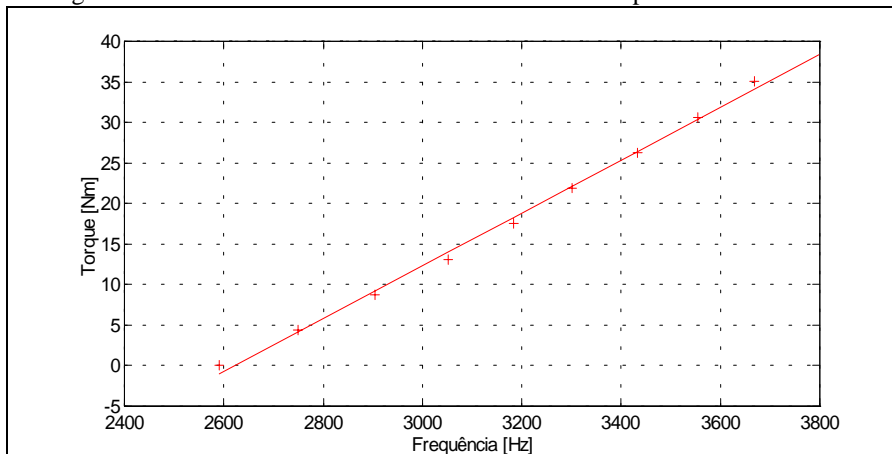


Figura 10: Curva de calibração estática obtida com a aplicação de torque conhecido

5. Evolução do Sinal com o Desgaste da Ferramenta

5.1. Procedimento Experimental e Associação como Desgaste:

Deseja-se observar a evolução do sinal de torque conforme ocorre o desgaste da ferramenta. Para isso, foram feitos testes de usinagem com a ferramenta instrumentada, coletando todos os dados durante todo o ciclo de vida da ferramenta, desde nova até desgastada. Os testes foram feitos utilizando mais de um parâmetro de usinagem, fazendo sempre um trecho com cada parâmetro. Foi utilizada fresa de aço rápido com 25 mm. de diâmetro e duas entradas. Os

corpos de prova eram barras de aço quadradas, SAE 1020, trefiladas. Os experimentos foram feitos com imersão total da ferramenta e a profundidade foi mantida em 8 mm. Os demais parâmetros conforme tabela abaixo:

Operação	Rotação [RPM]	Vel. de Avanço [mm/mim]
1	280	39
2	280	50
3	355	50
4	355	64
5	710	99

Foram utilizados sempre os conjuntos de parâmetros de 1 a 4. O conjunto 5 só foi utilizado no final dos testes para acelerar o desgaste da ferramenta e para testar a capacidade do transdutor numa situação bastante severa.

Para relacionar os dados coletados com um indicador de desgaste, foi utilizada a fórmula de vida de uma ferramenta de usinagem dada por Bonassi (1994):

$$C = V_C \cdot T_V^n$$

C : constante empírica; V_C : velocidade de corte;
 T_V : vida útil da ferramenta; n : expoente empírico de correção.

A associação com o desgaste foi feita assumindo que cada trecho usinado consome um pouco da vida útil da ferramenta, conforme:

$$t_C = t_R \cdot \frac{T_R}{2 \cdot T_V}$$

t_R : tempo real de cada trecho; T_R : tempo real total de usinagem;
 t_C : tempo corrigido, equiivale a vida da ferramenta consumida no trecho;
 2 : fator de correção, pois no fresamento com uma fresa de duas arestas, cada aresta só trabalha metade do tempo.
 T_C : tempo corrigido total de usinagem

5.2. Evolução do Sinal com o Desgaste no Domínio do Tempo:

No sinal de torque se percebe uma desigualdade entre a amplitude dos lobos. Como cada lobo representa a ação de uma das arestas da ferramenta, esta desigualdade é provocada pela desigualdade de forças. A variação mais marcante ao observar a evolução dos sinais com o desgaste, é variação na desigualdade entre as amplitudes dos lobos. Entretanto, esta variação não se mostrou correlacionada com o desgaste. Percebeu-se que outros eventos foram os maiores responsáveis pela variação na excentricidade, como a necessidade de retirada e reinstalação da ferramenta. Estes eventos afetam essencialmente a excentricidade existente na ferramenta, percebendo-se que é esta a principal causa da desigualdade de forças. O desgaste irregular entre as duas arestas também provoca desigualdade de forças, porém na ferramenta utilizada pode-se perceber, pela simples observação do estado similar das arestas, que este não é o fator decisivo. Chega-se a conclusão que a variação observada na desigualdade entre os lobos é provocada principalmente pela variação na excentricidade da ferramenta, provocada principalmente por fatores que não o desgaste.

No que se refere a amplitude média do sinal, é de se esperar que esta sofra um aumento devido ao desgaste. Entretanto, nos experimentos realizados, esta característica não pode ser controlada com precisão, devido a variações na profundidade de corte e devido a uma instabilidade que se percebeu no sistema de medição de torque. Dessa forma, este não foi um parâmetro que pudesse ser relacionado com o desgaste da ferramenta. A figura 11 mostra a evolução com o desgaste dos parâmetros ligados a amplitude média do sinal. Nesta figura, o máximo e o mínimo foram calculados por uma estimativa robusta que desconsidera 10% da amostra acima e abaixo, respectivamente, do máximo e do mínimo obtido. O desgaste é expresso através de T_C .

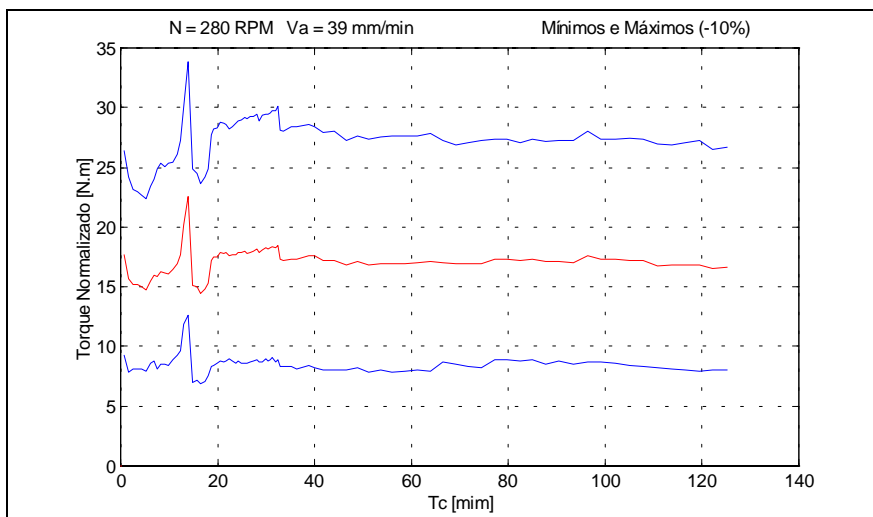


Figura 11: Amplitude das Amostras

5.3. Evolução do Sinal com o Desgaste no Domínio da Freqüência:

Para observar melhor a evolução da dinâmica do sinal, foi feita a normalização dos dados no domínio do tempo, utilizando a seguinte transformação:

$$\begin{cases} y_i = x_i - \bar{X} \\ z_i = \frac{y_i}{iqr(Y)} \end{cases} \quad \begin{array}{l} x_i : \text{Ponto de dado original;} \\ \bar{X} : \text{Média dos pontos } x_i ; \\ y_i : \text{Ponto de dado após 1ª transformação;} \\ z_i : \text{Ponto de dado após 2ª transformação;} \\ iqr(Y) : \text{Variação interquartos dos pontos } y_i. \end{array}$$

A variação interquartos é uma forma robusta de estimar variação, onde o limite inferior é o ponto imediatamente maior que 25% da amostra, e o limite superior é o ponto imediatamente maior que 75% da amostra. A figura 12 mostra a amplitude das amostras após a normalização.

O espectro do sinal foi obtido da seguinte forma:

- uma amostra de 8192 pontos foi separada em dois blocos menores de 4096 pontos;
- cada bloco foi normalizado, conforme descrito;
- foi calculada a FFT de cada bloco de dados e calculada a FFT média entre eles.

Para visualização, foram separadas as seis maiores freqüências do espectro de cada amostra, que são as seis primeiras. A primeira delas correspondente à rotação da ferramenta. Para a separação selecionou-se os seis pontos correspondentes do espectro, formado por um total de 2048 pontos. Para as amostras coletadas com 280 RPM, estes pontos e suas respectivas freqüências são mostrados na tabela:

N = 280 RPM		Espectro com 2048 pontos				
Ponto do Espectro	21	41	60	80	100	120
Freqüência [Hz]	4,90	9,79	14,44	19,34	24,24	29,13

O resultado desta análise pode ser observado na figura 13. Não se observou nenhuma característica no espectro dos sinais que evoluísse de forma correlacionada com o desgaste da ferramenta. Com base nos relatórios experimentais, observou-se que as grandes variações na evolução aconteciam exatamente onde houve a necessidade de retirada e reinstalação da ferramenta. Na reinstalação os parâmetros do processo e demais condições são mantidas. Entretanto, ocorrem variações na excentricidade da ferramenta. Conclui-se que a variação na excentricidade foi a principal causa da evolução observada do espectro do sinal, a exemplo do que observado no domínio do tempo.

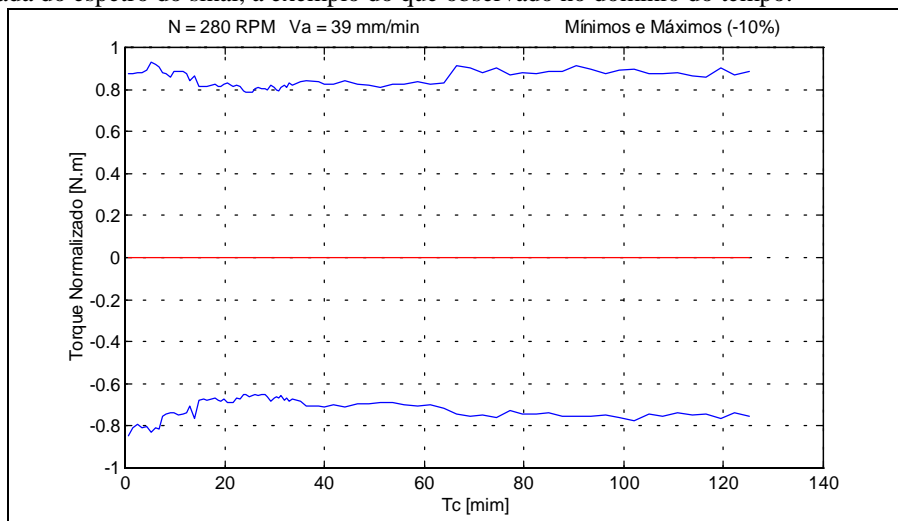


Figura 12: Amplitude Normalizada das Amostras

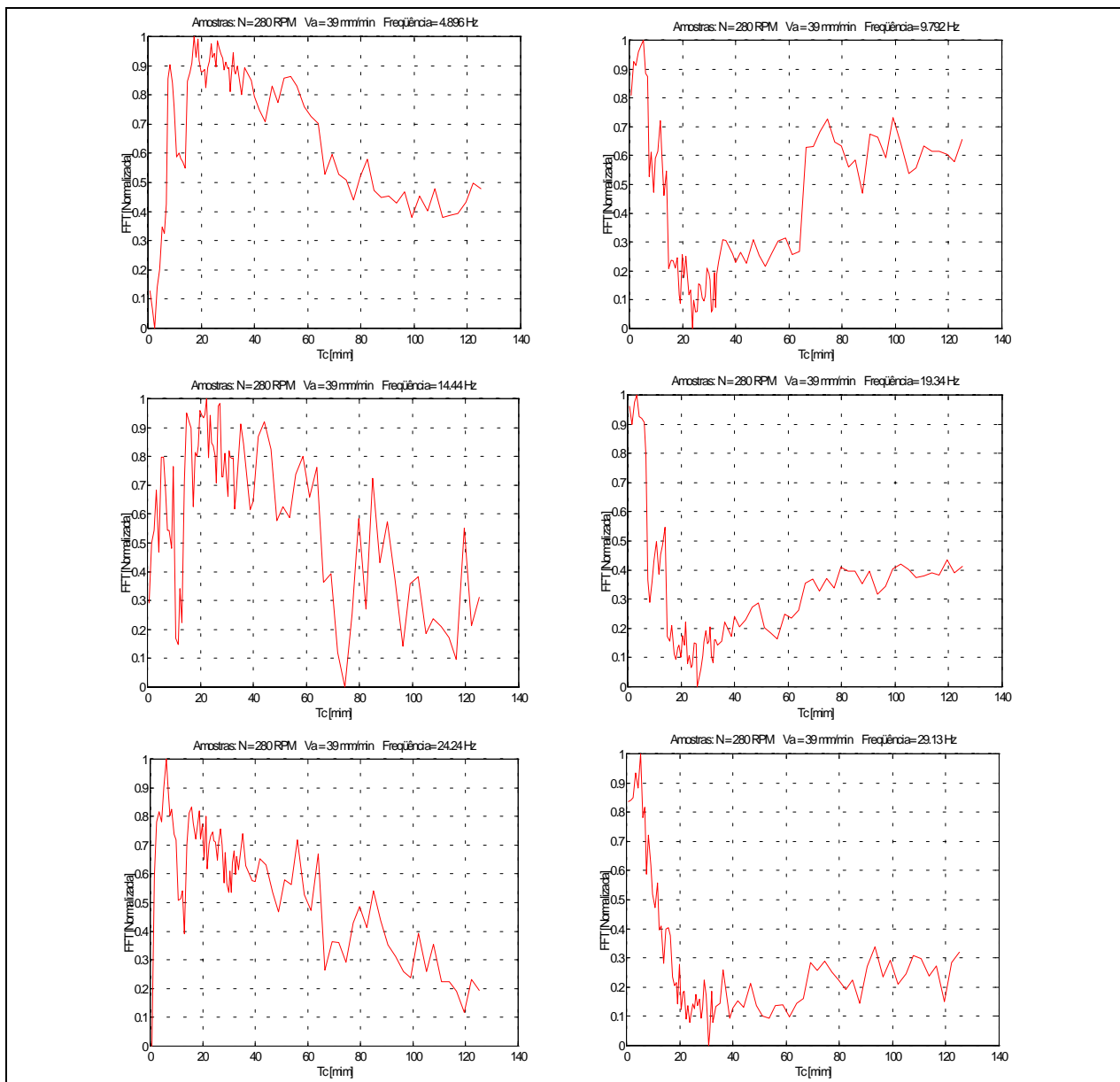


Figura 13: Evolução dos Picos do Espectro com o Desgaste expresso por T_C

6. Aplicação de Redes Neurais para Reconhecimento do Desgaste:

Do espectro calculado, conforme descrito na seção anterior, foram separadas as seis maiores frequências de cada amostra, que são as seis primeiras, que também foram normalizadas de 0 a 1.

Utilizou-se redes com estrutura *feed forward perceptron multicamada* com função de ativação *logaritmo sigmoidal*. Esta estrutura de rede foi treinada através do algoritmo *back propagation*. A rede utilizada também possui em cada neurônio uma constante de soma que pode, assim como os pesos, ser alterada pelo algoritmo de treinamento. Esta constante é chamada de “*bias*”.

O treinamento supervisionado foi feito utilizando-se 42 do total de 84 amostras de dados obtidas nos experimentos realizados com rotação da fresa a 280 RPM e velocidade de avanço a 39 mm/mim. As 42 amostras restantes nestas condições foram separadas para teste. A separação foi feita seqüencialmente, sendo uma amostra para treinamento, uma para teste e assim sucessivamente. A rede foi estruturada para seis entradas. Adotou-se uma única saída para a rede, obtida normalizando-se de 0,01 a 0,99 o tempo corrigido de usinagem (T_C), que foi o parâmetro escolhido para representar o desgaste. Várias estruturas de rede foram testadas. Dentre todas, a estrutura que apresentou melhores resultados era formada por 12 neurônios na primeira camada, uma camada intermediária com 7 neurônios e 1 neurônio na camada de saída.

Na figura 14 pode-se observar o resultado do treinamento, com a rede colocada diante dos mesmos dados utilizados para o treinamento. O eixo horizontal é o valor esperado de saída da rede, e o vertical é o valor da saída da rede após treinamento. Nas amostras coletadas com os mesmos parâmetros que as amostras de treinamento, pela grande similaridade e proximidade era de se esperar que não ocorressem desvios tão grandes quanto os observados na figura 15. Isto significa que a rede não foi capaz de uma boa generalização, adaptando-se apenas as características específicas

das amostras de treinamento. Pode-se interpretar este resultado, sob o ponto de vista dos dados, como a falta de uma boa correlação dos dados utilizados na rede com o que se deseja observar. Na figura 16 pode-se ver o desempenho da rede quando colocada diante de dados não utilizados para o treinamento e, além disso, com parâmetros de usinagem diferentes. Observa-se uma degradação crescente do desempenho quanto mais distintos são os parâmetros dos utilizados para o treinamento.

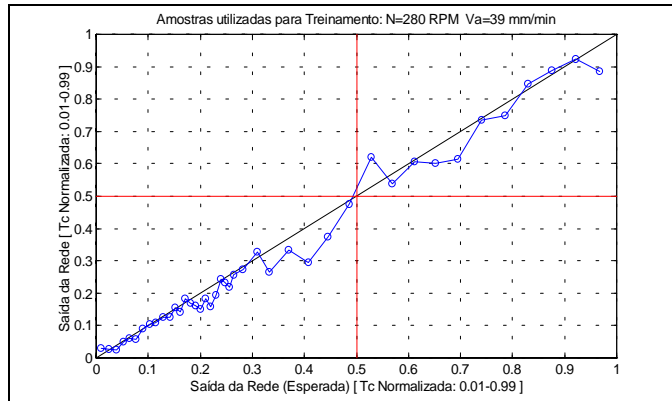


Figura 14: Saída da rede treinada sobre amostras utilizadas no treinamento

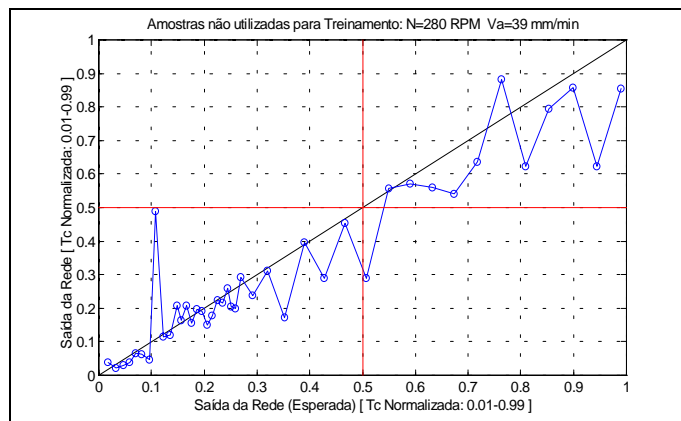
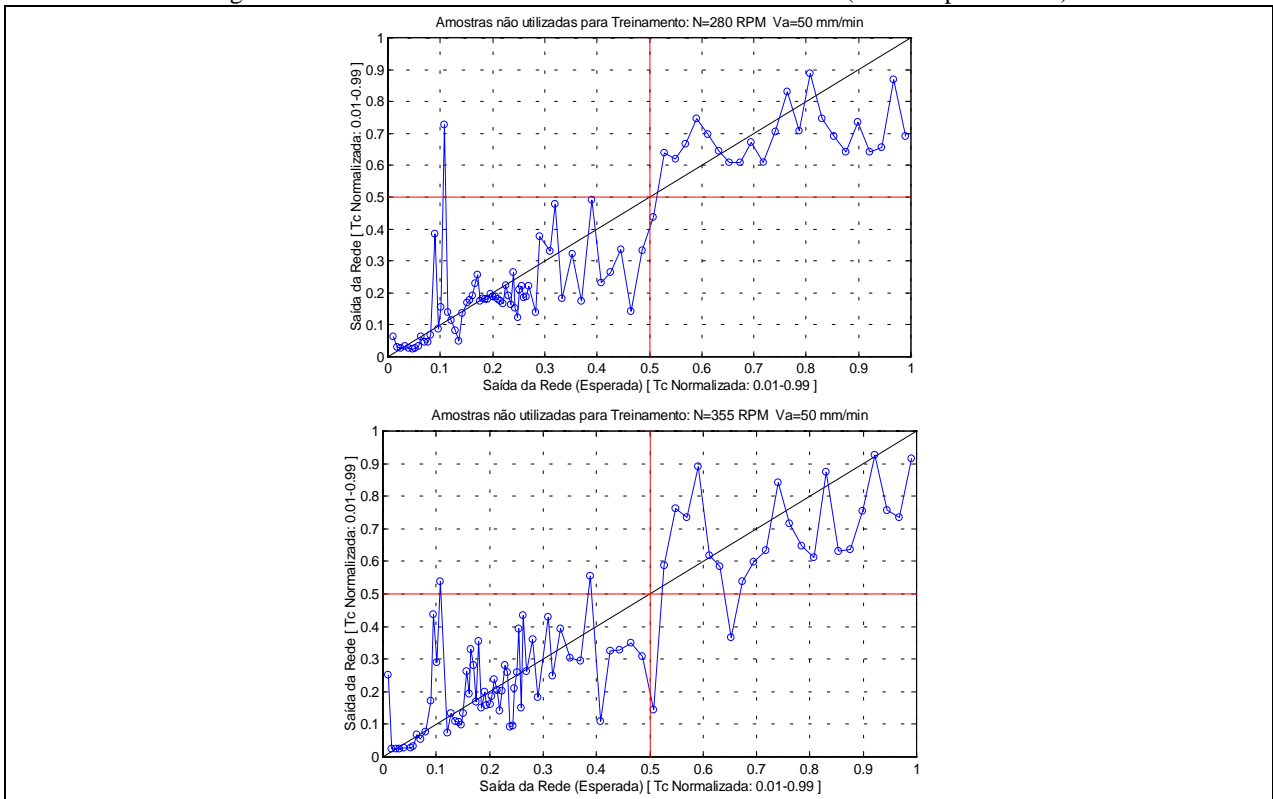


Figura 15: Saída da rede sobre amostras fora do treinamento (mesmos parâmetros)



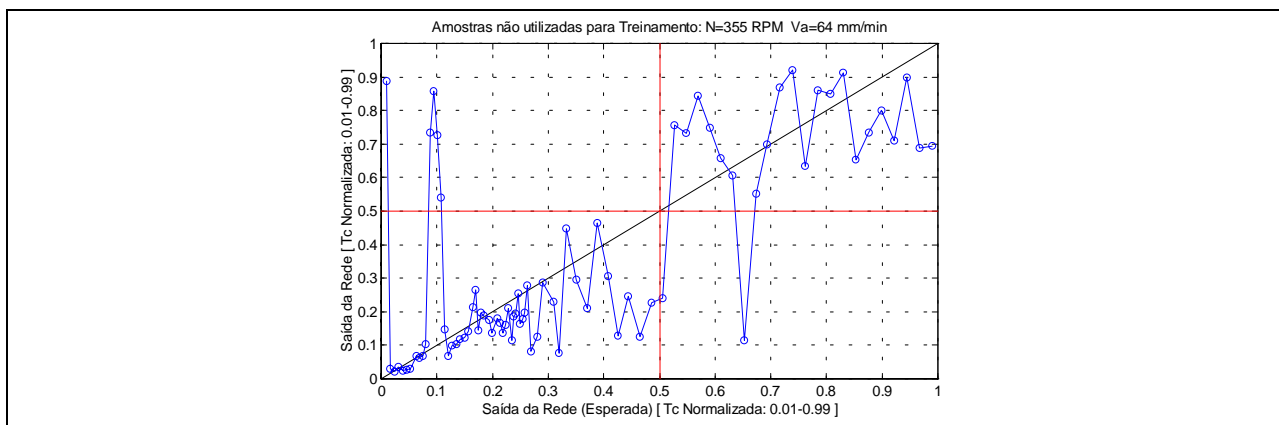


Figura 16: Saída da rede sobre amostras fora do treinamento (parâmetros diferentes)

Até aqui, o desempenho da rede na identificação foi analisado considerando o desgaste da ferramenta uma variável contínua. Nesta seção, passa-se a considerar o desgaste uma variável discreta. Com base no tempo corrigido de usinagem (T_C), que foi o parâmetro escolhido para representar o desgaste, podemos considerar dois estados discretos de desgaste da ferramenta, em método semelhante ao utilizado por Rangwala e Dornfeld Rangwala (1990):

Nova: se ($T_C < 0,5$)

Gasta: se ($T_C \geq 0,5$)

O resultado desta discretização está representado nas figuras 14 a 16 na forma de linhas, dividindo a área dos gráficos em quadrantes. Todos os pontos que estão no 1º e no 3º quadrante obedecem os critérios estabelecidos, sendo portanto considerados classificações corretas. A tabela mostra os resultados observados.

Parâmetros de Usinagem	Total de Amostras	Classificações Corretas	Sucesso
N=280 RPM Va=39 mm / min	42	41	97,6%
N=280 RPM Va=50 mm / min	83	81	97,6%
N=355 RPM Va=50 mm / min	81	77	95,1%
N=355 RPM Va=64 mm / min	79	72	91,1%
Geral	285	271	95,1%

Conforme discutido anteriormente, existe uma forte correlação das amostras com a variação da excentricidade que ocorre durante o desgaste da ferramenta, e também devido a retirada e reinstalação da ferramenta. Assim, apesar de a rede ter conseguido acompanhar razoavelmente bem a variação do desgaste dentro dos limites estabelecidos, este resultado não pode ser generalizado. Uma ferramenta igual à utilizada, submetida aos mesmos ensaios poderá apresentar evolução da excentricidade distinta da observada nestas amostras, o que certamente resultaria na total deterioração da capacidade de classificação observada com estas amostras.

7. Conclusões:

A técnica de transmissão óptica do sinal modulado em frequência mostrou sua efetividade em uma nova aplicação, permitindo desenvolver um equipamento de baixo custo.

Desejava-se observar também a dinâmica do sinal. Para isso foi desenvolvida e comprovada uma técnica que permitiu utilizar o sinal modulado em frequência para a análise digital de sinais.

O software desenvolvido foi capaz de coletar o sinal modulado em frequência diretamente a partir da porta paralela do PC. Sistema eficiente e barato que pode ser utilizado em outras aplicações.

A principal evolução observada no sinal de torque foi a variação da diferença que existe entre o esforço realizado por cada uma das arestas de corte da ferramenta. No domínio do tempo, esta variação se expressa através da variação da diferença que existe entre a amplitude dos cumes do sinal no tempo. No domínio da frequência esta evolução pode ser observada ao acompanhar a evolução da amplitude relativa do pico na frequência de rotação da ferramenta e o seu primeiro múltiplo.

A diferença entre os esforços das arestas de corte pode ser gerada tanto pela excentricidade da ferramenta, como também pode ser provocada por desgaste irregular entre as arestas de corte. Nos sinais apresentados neste trabalho, sabe-se, pela simples observação da ferramenta gasta, que a diferença entre os esforços foi gerada principalmente pela excentricidade.

Não foi observada, no sinal, nenhuma característica marcante que pudesse ser correlacionada com o desgaste da ferramenta.

Uma rede neural conseguiu acompanhar razoavelmente bem a evolução do desgaste, baseadas em um conjunto relativamente pequeno de dados. Quando o desgaste foi analisado de uma maneira discreta, o resultado foi excelente, alcançando 95% de classificações corretas. Porém, este resultado deve ser observado com grande cuidado, pois como as principais variações nos sinais que serviram de entrada para a rede foram geradas pela variação na excentricidade, é

possível afirmar que esta performance não se repetiria se a rede fosse testada em dados de uma outra ferramenta, que teria uma evolução da excentricidade com o desgaste, diferente da que foi utilizada para o aprendizado da rede.

Referências

- Bonassi, V. L. "Usinagem dos Metais." São Bernardo do Campo, 1994. Notas do curso ministrado na Faculdade de Engenharia Industria
- Byrne, G.; Dornfeld, D.; Inasaki, I.; Ketteler, G.; König, W. e Teti, R. "Tool Condition Monitoring (TCM)- The Status of Research and Industrial Application 1996l
- Dornfeld, D. A. "Neural Network Sensor Fusion for Tool Condition Monitoring." Annals of the CIRP, v. 39/1, 1990
- Ferreira, W. P. e Chieng-Ching, C. T. "Torquímetro Dinâmico Óptico em Frequência Modulada." Relatório Técnico, 1988
- Rangwala, S. e Dornfeld, D. "Learning and Optimization of Machining Operations Using Computing Abilities of Neural Networks." IEEE Transactions on Systems, Man and Cybernetics v. 19 n.2 Março/Abril 1989
- Rangwala, S. e Dornfeld, D. "Sensor Integration Using Neural Networks for Intelligent Tool Condition Monitoring." Journal of Engineering for Industry, v. 112, Agosto de 1990
- Teixeira, M. "Monitoramento do Desgaste de Ferramentas de Corte através de um Sistema Multi-Sensorial." São Paulo, 1998. Tese (Doutorado) - Escola Politécnica, Universidade de São Paulo
- Tlusty, J e Andrews, G. C. "A Critical Review of Sensors for Unmanned Machining." Annals of the CIRP, v. 32/2, 1983

END MILL TOOL WEAR MONITORING THROUGH TORQUE MEASUREMENT

Vinicius Leal Carneiro da Cunha

Escola Politécnica -USP
Alameda Prof . Mello de Moraes 2231
05508-900,SP,SP,Brazil

Lucas Antônio Moscato

Escola Politécnica -Universidade de São Paulo
Alameda Prof . Mello de Moraes 2231
05508-900,SP,SP,Brazil
lamoscat@usp.br

Abstract: The complete automation of the machining operations depends on a system able to monitor the process conditions. In this work, it was developed a torque measurement system to end milling operation. A frequency modulated optical system was used to the signal transmission, assuring high noise immunity. The frequency modulation allowed the signal acquisition using only the parallel printer port, through a software written in assembler. With this system, machining signals were collected and the relation of the torque signal with the tool wear was addressed.

Keywords: monitoring, tool wear, milling, torque, frequency modulation.

MÉTODO DE IDENTIFICAÇÃO NOS SUBESPAÇOS PARA SISTEMAS MULTIVARIÁVEIS COM ESTRUTURA INTERCONECTADA

Celso Pascoli Bottura

Departamento de Máquinas Componentes e Sistemas Inteligentes
Faculdade de Engenharia Elétrica e de Computação - UNICAMP
e-mail bottura@dmcsi.fee.unicamp.br

Angel Fernando Torrico Cáceres

Departamento de Engenharia de Eletricidade - Centro Tecnológico
Universidade Federal do Maranhão
e-mail ftorrico@dmcsi.fee.unicamp.br

Resumo Neste trabalho é proposto um método de identificação nos subespaços para sistemas lineares multivariáveis com estrutura de interconexão serial. A identificação do sistema global é feita por meio da identificação individual dos seus subsistemas. Nesta metodologia de identificação são necessários dados de entrada/saída de cada subsistema. Uma proposta de aplicação ao modelamento de sistemas da qualidade da água de rios é feita e seus resultados são apresentados..

Palavras chave: Identificação de sistemas, modelamento no espaço de estados, sistemas interconectados.

1. Introdução

Os métodos de identificação nos subespaços tem-se mostrado eficientes e atraentes nos últimos anos, principalmente devido à habilidade de providenciar precisos modelos multivariados no espaço de estados diretamente de dados entrada-saída, (Verhaegen e Dewilde, 1992), (Verhaegen, 1994), (Viberg, 1995), (Overschee e de Moor, 1996). Neste trabalho utilizamos esses métodos na identificação de sistemas serialmente interconectados, como mostrado na figura 1; a natureza serial do sistema permite estabelecer uma metodologia de identificação do sistema global por meio da identificação separada dos seus subsistemas. O comportamento de um subsistema é devido à entrada própria do subsistema e à saída do subsistema anterior, portanto os dados de saída do subsistema anterior são essenciais e devem ser utilizados para a identificação isolada do subsistema. Existem vários exemplos onde esta metodologia pode ser aplicada como por exemplo, um sistema de reservatórios em cascata; neste trabalho apresentamos um exemplo de aplicação no modelamento da qualidade da água de rios.

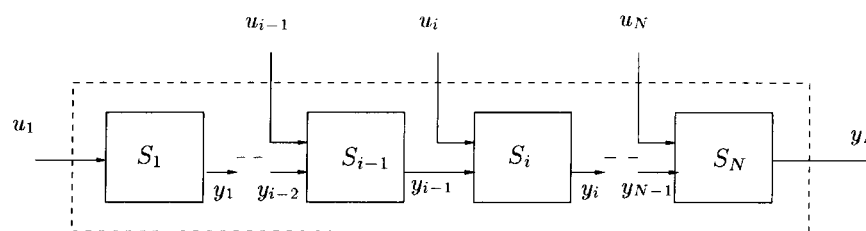


Figura 1: Sistema serial interconectado.

2. Sistemas serialmente interconectados.

Representamos um sistema interconectado serial de N subsistemas por meio da figura 1, onde u_i é a entrada própria do subsistema e y_i a sua saída. Notemos que existe uma ligação direta entre eles por meio das suas saídas, isto é, a saída de um subsistema influi diretamente no comportamento do seguinte. Na figura 2 mostramos dois subsistemas, S_{i-1} e S_i , onde consideramos incertezas no modelamento interno dos subsistemas e erros nas medidas das suas saídas. Denotamos as saídas medidas por z_{i-1} e z_i , isto é, supomos que as saídas verdadeiras y_{i-1} e y_i são perturbadas pelos processos aleatórios ruído branco, v_{i-1} e v_i . As incertezas no modelamento interno estão caracterizadas por meio das entradas w_{i-1} e w_i que também consideramos serem

processos aleatórios ruído branco. Para efeitos de identificação consideramos um subsistema isolado, como mostrado na figura 3, isto é, consideramos u_i e y_{i-1} como entradas conhecidas sem erro, e z_i como saída.

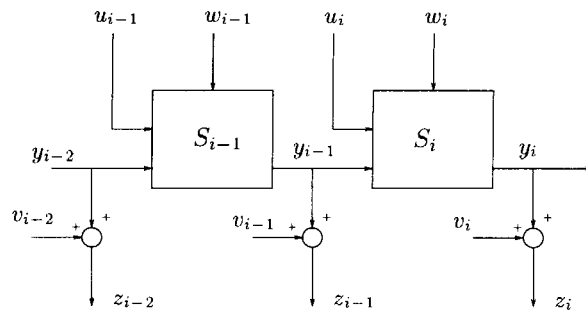


Figura 2: Dois subsistemas do sistema global.

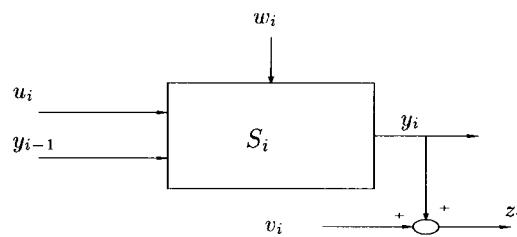


Figura 3: Subsistema Isolado.

Representamos o subsistema i no espaço de estados pelas equações:

$$x_i(t+1) = A_i x_i(t) + B_i u_i(t) + M_i y_{i-1}(t) + w_i(t) \tag{1}$$

$$y_i(t) = C_i x_i(t) + D_i u_i(t) + N_i y_{i-1}(t) \tag{2}$$

$$z_i(t) = y_i(t) + v_i(t), \tag{3}$$

onde $y_i(t)$, $u_i(t)$ e $x_i(t)$ são respectivamente os vetores de entrada, saída e estado do subsistema i , Rearranjamos esta representação, para efeitos de identificação, como:

$$x_i(t+1) = A_i x_i(t) + \begin{bmatrix} B_i & M_i \end{bmatrix} \begin{bmatrix} u_i(t) \\ y_{i-1}(t) \end{bmatrix} + w_i(t) \tag{4}$$

$$z_i(t) = C_i x_i(t) + \begin{bmatrix} D_i & N_i \end{bmatrix} \begin{bmatrix} u_i(t) \\ y_{i-1}(t) \end{bmatrix} + v_i(t). \tag{5}$$

O problema resume-se em encontrar as matrizes A_i, B_i, C_i, D_i, M_i e N_i e as covariâncias

$$E \left[\begin{pmatrix} w_i(s) \\ v_i(s) \end{pmatrix} \begin{pmatrix} w_i(t)' & v_i(t)' \end{pmatrix} \right] = \begin{pmatrix} Q_i & S_i \\ S_i' & R_i \end{pmatrix} \delta_{st} \geq 0, \tag{6}$$

a partir dos dados medidos. O modelo global, sem erro de leitura, obtido para um sistema serial interconectado de 3 subsistemas é

$$\begin{bmatrix} x_1(t+1) \\ x_2(t+1) \\ x_3(t+1) \end{bmatrix} = \begin{bmatrix} A_1 & 0 & 0 \\ M_2 C_1 & A_2 & 0 \\ M_3 N_2 C_1 & M_3 C_2 & A_3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} B_1 & 0 & 0 \\ M_2 D_1 & B_2 & 0 \\ M_3 N_2 D_1 & M_3 D_2 & B_3 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} + \begin{bmatrix} w_1(t) \\ w_2(t) \\ w_3(t) \end{bmatrix} \tag{7}$$

$$\begin{bmatrix} y_1(t) \\ y_2(t) \\ y_3(t) \end{bmatrix} = \begin{bmatrix} C_1 & 0 & 0 \\ N_2 C_1 & C_2 & 0 \\ N_3 N_2 C_1 & N_3 C_2 & C_3 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} D_1 & 0 & 0 \\ N_2 D_1 & D_2 & 0 \\ N_3 N_2 D_1 & N_3 D_2 & D_3 \end{bmatrix} \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix}, \quad (8)$$

a generalização para N subsistemas é feita de modo similar.

Os modelos obtidos nesta forma possibilitam o seu uso para efeitos de monitoramento e controle, seja para um controle global seja para um controle descentralizado.

3. Metodologia de identificação

A identificação do sistema global é realizada por meio da identificação isolada de cada subsistema, isto é tratamos um subsistema como um sistema que tem como entradas a entrada própria do subsistema e os dados de saída do subsistema anterior, $\begin{bmatrix} u_i \\ y_{i-1} \end{bmatrix}$, e z_i como sua saída. Como o vetor y_{i-1} não esta diretamente disponível para leitura, consideramos a estimativa dele, \hat{y}_{i-1} , que é obtida por meio de um filtro de Kalman aplicado no subsistema anterior, como dada. Esta metodologia pode ser aplicada a sistemas serialmente interconectados devido a que as saídas do subsistema anterior não estão correlacionadas com a saída do subsistema seguinte.

4. Métodos de Identificação nos Subespaços

Nesta seção apresentamos, de forma breve, um metodo de identificação nos subespaços; para facilidade de notação denotamos a dependência do tempo discreto da seguinte forma: $y(t) \triangleq y_t, u(t) \triangleq u_t, \dots$; os modelos lineares obtidos são da forma:

$$x_{t+1} = Ax_t + Bu_t + w_t \quad (9)$$

$$y_t = Cx_t + Du_t + v_t, \quad (10)$$

onde w_t e v_t são processos aleatórios ruído branco de covariância

$$E \left[\begin{pmatrix} w(s) \\ v(s) \end{pmatrix} \begin{pmatrix} w(t)' & v(t)' \end{pmatrix} \right] = \begin{pmatrix} Q & S \\ S' & R \end{pmatrix} \delta_{st} \geq 0. \quad (11)$$

Os vetores $u_t \in R^{m \times 1}$ e $y_t \in R^{l \times 1}$ são respectivamente as medidas das m entradas e das l saídas do processo no instante t . Podemos escrever as equações (9) e (10) na seguinte forma inovativa (Viberg, 1995):

$$\begin{bmatrix} x_{t+1} \\ y_t \end{bmatrix} = \begin{bmatrix} Ax_t + Bu_t + Ke_t \\ Cx_t + Du_t + e_t \end{bmatrix}, \quad (12)$$

onde e_t é um processo inovativo ruído branco, K é o ganho do filtro de Kalman.

Definimos o vetor

$$\mathbf{y}(t) = [y_t^T \ y_{t+1}^T \ \dots \ y_{t+\alpha-1}^T]^T, \quad (13)$$

onde $\alpha > n$ (onde n é a ordem do sistema), definimos também vetores $\mathbf{u}(t)$ e $\mathbf{e}(t)$ de forma similar. Então de (12) e (13) podemos escrever

$$\mathbf{y}(t) = \Gamma_\alpha x_t + \Phi_\alpha \mathbf{u}(t) + \Psi_\alpha \mathbf{e}(t) \quad (14)$$

onde

$$\Gamma_\alpha = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{\alpha-1} \end{bmatrix} \quad (15)$$

é a matriz de observabilidade, e as matrizes

$$\Phi_\alpha = \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{\alpha-2}B & \dots & CB & D \end{bmatrix} \quad (16)$$

$$\Psi_\alpha = \begin{bmatrix} I & 0 & \cdots & 0 \\ CK & I & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{\alpha-2}K & \cdots & CK & I \end{bmatrix} \quad (17)$$

são matrizes Toeplitz.

Suponhamos que temos observações de y_t e u_t , para $t = 1, 2, \dots, N + \alpha - 1$; formamos a partir dos dados de entrada e saída as seguintes matrizes de Hankel

$$\mathbf{Y} = [\mathbf{y}(1) \ \cdots \ \mathbf{y}(N)], \quad \mathbf{U} = [\mathbf{u}(1) \ \cdots \ \mathbf{u}(N)], \quad (18)$$

então de (13) e (14) temos que

$$\mathbf{Y} = \Gamma_\alpha \mathbf{X} + \Phi_\alpha \mathbf{U} + \Psi_\alpha \mathbf{E}, \quad (19)$$

onde $\mathbf{X} = [x_t \ \cdots \ x_N]$ é uma matriz composta pelas trajetórias do estado.

A equação (19) é fundamental nos métodos de subespaços. Se matriz de observabilidade (15) for conhecida, podemos determinar a matriz C pegando somente o primeiro bloco linha de Γ_α . A matriz A pode ser encontrada da relação

$$\Gamma_{2:\alpha+1} = \begin{bmatrix} CA \\ \vdots \\ CA^\alpha \end{bmatrix} = \begin{bmatrix} C \\ \vdots \\ CA^{\alpha-1} \end{bmatrix} A \quad (20)$$

com $A = \Gamma_\alpha^\dagger \Gamma_{2:\alpha+1}$, onde $(\cdot)^\dagger$ denota a pseudoinversa de Moore-Penrose. A obtenção da matriz de observabilidade é feita por meio de projeções ortogonais dentro de subespaços formados pelos vetores da matriz \mathbf{U} . Suponhamos que o sistema não tenha a parte estocástica, isto é $e_t = 0$, então temos

$$\mathbf{Y} = \Gamma_\alpha \mathbf{X} + \Phi_\alpha \mathbf{U}, \quad (21)$$

como Φ_α é desconhecido, podemos estimá-lo de forma a satisfazer o seguinte problema de mínimos quadrados não estruturado

$$\min_{\Phi_\alpha} \|\mathbf{Y} - \Phi_\alpha \mathbf{U}\|_F^2, \quad (22)$$

onde $\|\cdot\|_F$ denota a norma de Frobenius. A matriz $\hat{\Phi}_\alpha$ solução deste problema obedece a seguinte equação:

$$\mathbf{Y} - \hat{\Phi}_\alpha \mathbf{U} = \mathbf{Y} \Pi_{\mathbf{U}^\perp}, \quad (23)$$

onde $\Pi_{\mathbf{U}^\perp}$ é a projeção ortogonal no espaço nulo de \mathbf{U} , dado por

$$\Pi_{\mathbf{U}^\perp} = \mathbf{I} - \mathbf{U}^T (\mathbf{U} \mathbf{U}^T)^{-1} \mathbf{U}.$$

como $\mathbf{U} \Pi_{\mathbf{U}^\perp} = \mathbf{0}$ temos de (21)

$$\mathbf{Y} \Pi_{\mathbf{U}^\perp} = \Gamma_\alpha \mathbf{X} \Pi_{\mathbf{U}^\perp}. \quad (24)$$

Se o posto de $\mathbf{X} \Pi_{\mathbf{U}^\perp}$ é completo e igual a n , podemos recuperar Γ_α através de uma decomposição em valores singulares truncada da matriz $\mathbf{Y} \Pi_{\mathbf{U}^\perp}$:

$$\mathbf{Y} \Pi_{\mathbf{U}^\perp} = \hat{\mathbf{Q}}_s \hat{\mathbf{S}}_s \hat{\mathbf{V}}_s^T + \hat{\mathbf{Q}}_n \hat{\mathbf{S}}_n \hat{\mathbf{V}}_n^T, \quad (25)$$

onde $\hat{\mathbf{Q}}_s$ contem os n vetores correspondentes aos valores singulares mais significativos. Na ausência de ruído teremos $\hat{\mathbf{S}}_n = \mathbf{0}$. A estimativa da matriz de observabilidade pode ser considerada como $\hat{\Gamma}_\alpha = \hat{\mathbf{Q}}_s$. Uma implementação mais eficiente é proposta em (Verhaegen, 1994), que calcula a matriz $\mathbf{Y} \Pi_{\mathbf{U}^\perp}$ por meio de uma fatoração QR da seguinte matriz

$$\begin{bmatrix} \mathbf{U} \\ \mathbf{Y} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \end{bmatrix}; \quad (26)$$

podemos observar que \mathbf{Q}_2 gera a parte do espaço nulo de \mathbf{U} que não é compartilhada por \mathbf{Y} , então

$$\mathbf{Y} \Pi_{\mathbf{U}^\perp} = \mathbf{Y} \mathbf{Q}_2 \mathbf{Q}_2^T = \mathbf{R}_{22} \mathbf{Q}_2^T. \quad (27)$$

Sabendo que $\mathbf{Q}_2^T \mathbf{Q}_2 = \mathbf{I}$, os vetores singulares \mathbf{R}_{22} que estão à esquerda, são aqueles de $\mathbf{R}_{22} \mathbf{Q}_2^T$, assim como também os vetores singulares de $\mathbf{Y} \mathbf{\Pi}_{\mathbf{U}_T}^\perp$. O método de Verhaegen e Dewilde calcula \mathbf{R}_{22} da seguinte maneira:

$$\mathbf{R}_{22} = \hat{\mathbf{Q}}_s \hat{\mathbf{S}}_s (\hat{\mathbf{V}}_s^T \mathbf{Q}_2) + \hat{\mathbf{Q}}_n \hat{\mathbf{S}}_n (\hat{\mathbf{V}}_n^T \mathbf{Q}_2). \quad (28)$$

Observe que (28) é compatível com (25). Somente a parte determinística pode ser identificada desta maneira; qualquer dinâmica adicional causada por perturbação que não seja ruído branco não é considerada.

Em (Verhaegen, 1994) a organização estrutural dos dados permite (no limite quando $N \rightarrow \infty$) eliminar a parte estocástica de (19). Isto é feito como segue: a saída é particionada em termos passado e futuro, como

$$\mathbf{y}(t) = \begin{bmatrix} \mathbf{y}_p(t) \\ \mathbf{y}_f(t) \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} \mathbf{Y}_p \\ \mathbf{Y}_f \end{bmatrix}.$$

Da mesma forma são particionadas as entradas e os ruídos. A equação para a saída futura é dada por:

$$\mathbf{Y}_f = \Gamma_\gamma \mathbf{X}_f + \Phi_\gamma \mathbf{U}_f + \Psi_\gamma \mathbf{E}_f, \quad (29)$$

onde $\mathbf{X}_f = [x_{\beta+1} \cdots x_{\beta+N}]$, $\gamma = \alpha - \beta > n$, e β , é o número de bloco filas em \mathbf{Y}_p . Podemos escolher agora $\mathbf{u}_p(t)$ como uma variável instrumental de modo a remover o termo \mathbf{U}_f de (29) e decorrelacionar o ruído. Verhaegen considera a seguinte projeção ortogonal

$$\mathbf{Y}_f \mathbf{\Pi}_{\mathbf{U}_f}^\perp \mathbf{U}_p^T = \Gamma_\gamma \mathbf{X}_f \mathbf{\Pi}_{\mathbf{U}_f}^\perp \mathbf{U}_p^T + \Psi_\gamma \mathbf{E}_f \mathbf{\Pi}_{\mathbf{U}_f}^\perp \mathbf{U}_p^T,$$

onde $\mathbf{\Pi}_{\mathbf{U}_f}^\perp$ é a projeção ortogonal do espaço nulo de \mathbf{U}_f . Pode-se provar que

$$\lim_{N \rightarrow \infty} \frac{1}{N} \mathbf{E}_f \mathbf{\Pi}_{\mathbf{U}_f}^\perp \mathbf{U}_p^T = \mathbf{0}, \quad \text{w.p.1.}$$

O algoritmo de Verhaegen calcula a seguinte fatoração QR:

$$\begin{bmatrix} \mathbf{U}_f \\ \mathbf{U}_p \\ \mathbf{Y}_p \\ \mathbf{Y}_f \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{21} & \mathbf{R}_{22} & \mathbf{0} & \mathbf{0} \\ \mathbf{R}_{31} & \mathbf{R}_{32} & \mathbf{R}_{33} & \mathbf{0} \\ \mathbf{R}_{41} & \mathbf{R}_{42} & \mathbf{R}_{43} & \mathbf{R}_{44} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^T \\ \mathbf{Q}_2^T \\ \mathbf{Q}_3^T \\ \mathbf{Q}_4^T \end{bmatrix},$$

onde \mathbf{R}_{ii} é uma matriz triangular superior.

No proximo passo do algoritmo é calculada a seguinte SVD

$$\begin{bmatrix} \mathbf{R}_{32} & \mathbf{R}_{33} \end{bmatrix} = \hat{\mathbf{Q}}_s \hat{\mathbf{S}}_s \hat{\mathbf{V}}_s^T + \hat{\mathbf{Q}}_n \hat{\mathbf{S}}_n \hat{\mathbf{V}}_n^T.$$

A matriz de observabilidade é estimada como $\hat{\mathbf{Q}}_s$, e as matrizes \mathbf{A} e \mathbf{C} podem ser extraídas, como em (20). Para encontrar \mathbf{B} e \mathbf{D} uma solução do problema de mínimos quadrados

$$\begin{bmatrix} \mathbf{R}_{31} & \mathbf{R}_{42} \end{bmatrix} \approx \Phi_\gamma \begin{bmatrix} \mathbf{R}_{11} & \mathbf{R}_{22} \end{bmatrix},$$

dá uma estimação consistente de Φ_γ . Com esta estimação, \mathbf{B} e \mathbf{D} podem ser calculados.

5. Aplicação à qualidade da água de rios

Utilizamos a metodologia de identificação acima descrita, para encontrar modelos no espaço de estados de sistemas de qualidade da água de rios. Devido à natureza complexa do problema, dividimos em segmentos o comprimento total do rio, como mostrado na figura 4. Cada um deles constitui um subsistema do sistema global, com entradas e saídas próprias. As entradas podem ser, por exemplo, as descargas controladas de estações de tratamento de água; as saídas são parâmetros da qualidade da água de rios, por exemplo, as concentrações da Demanda Bioquímica de Oxigênio (DBO) e do Oxigênio Dissolvido (OD).

Para propósitos de simulação consideramos que cada subsistema (segmento de rio) pode ser considerado como um reator de fluxo contínuo, (CFSTR), como citado em (Young e Beck, 1974) e (Cáceres, 1992). O modelo de simulação tem as seguintes equações dinâmicas:

DBO:

$$\frac{dL_i}{dt} = -k_{L_i} L_i + \frac{Q_{i-1}}{V_i} L_{i-1} - \frac{Q_i + Q_{ei}}{V_i} L_i + \frac{m_i Q_{ei}}{V_i} \quad (30)$$

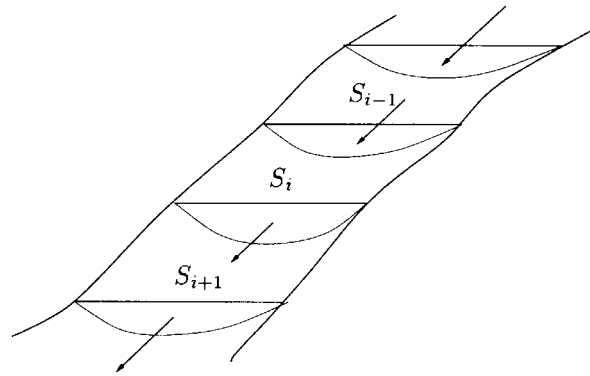


Figura 4: Segmentos de rio.

OD:

$$\frac{dq_i}{dt} = k_{ri}(q_i^s - q_i) + \frac{Q_{i-1}}{V_i} q_{i-1} - \frac{Q_i + Q_{ei}}{V_i} q_i - k_{Li} L_i - \frac{\eta_i}{V_i} \quad (31)$$

onde Q_i e Q_{ei} são a vazão do i -ésimo segmento e a vazão do seu efluente respectivamente, L_i e q_i são as concentrações de DBO e OD do segmento, q_i^s é o valor de saturação do OD, k_{Li} é taxa de decaimento de DBO, m_i é a concentração de DBO do efluente, k_{ri} é a taxa de reoxigenação, V_i é o volume do segmento, η_i é a taxa de variação na concentração de OD devida ao lodo do segmento.

O modelo não considera os efeitos da fotossíntese, da adição de matéria orgânica devida às chuvas e da matéria orgânica depositada no fundo do rio. Também não são considerados os atrasos de transporte entre os subsistemas. Para a simulação utilizamos dados de um segmento de 4.7km do rio Cam na Inglaterra e que podem ser encontrados em (Young e Beck, 1974) e são: $k_{ri} = 0.2$ [dia⁻¹], $k_{Li} = 0.32$ [dia⁻¹], $\frac{\eta_i}{V_i} = 0.1$ [$\frac{mg}{litro}$][dia⁻¹], $q_i^s = 10$ [mg/litro], $\frac{Q_{ei}}{V_i} = 0.1$, $\frac{Q_i}{V_i} = \frac{Q_{i-1}}{V_i} = 0.9$. Processos ruído branco gaussianos δL_i e δq_i de média zero e covariâncias ΔL e Δq foram adicionados nas equações dos subsistemas. As concentrações de DBO de entrada, m_i , são determinísticas e as supomos constantes. Para esses valores o modelo de cada segmento é:

$$\dot{\mathbf{x}}_i(t) = \mathbf{A}_i \mathbf{x}_i(t) + \mathbf{B}_i \mathbf{u}_i(t) + \mathbf{u}_{i-1}(t) + \omega_i(t) \quad (32)$$

$$\text{onde } \mathbf{A}_i = \begin{bmatrix} -1.32 & 0 \\ -0.32 & -1.2 \end{bmatrix}, \mathbf{B}_i = \begin{bmatrix} 0.1 \\ 0 \end{bmatrix},$$

$$\mathbf{u}_{i-1}(t) = \begin{bmatrix} 0.9L_{i-1} \\ 0.9q_{i-1} + 1.9 \end{bmatrix}, \omega_i(t) = \begin{bmatrix} \delta L_i \\ \delta q_i \end{bmatrix}$$

$$\mathbf{u}_i(t) = m_i \text{ and } \mathbf{x}_i(t) = \begin{bmatrix} L_i \\ q_i \end{bmatrix}.$$

Construímos uma plataforma de simulação que considera um sistema global formado por três segmentos de rio iguais, com os dados e as equações dinâmicas dadas acima. Consideramos que as entradas próprias dos subsistemas são determinísticas e fixas. Temos então como dados somente as séries temporais dos parâmetros da qualidade da água, DBO e OD, de cada segmento. Aplicamos a metodologia previamente exposta a essas séries.

6. Resultados

Foram geradas na simulação 100 amostras de medidas para cada subsistema. As tendências devido às respostas de regime foram tratados e eliminadas por meio de uma primeira diferença. Utilizamos o método de Verhaegen, (Verhaegen, 1994), na identificação de cada subsistema. As figuras 6, 7 e 8 mostram, respectivamente, os valores medidos versus os valores estimados de DBO e OD do primeiro, segundo e terceiro segmentos. Observamos que os valores estimados seguem razoavelmente os valores medidos. As dimensões dos modelos no espaço de estados são: 6 para o primeiro, 6 para o segundo e 7 para o terceiro segmentos. Observamos que foram necessários modelos com maior número de estados que os utilizados na simulação. Isto é devido principalmente às perturbações inseridas na simulação.

Os valores estimados foram encontrados utilizando o seguinte filtro de Kalman:

$$\begin{aligned} \hat{\mathbf{x}}_i(\mathbf{t} + 1) &= \mathbf{A}_i \hat{\mathbf{x}}_i(\mathbf{t}) + \mathbf{M}_i \hat{\mathbf{y}}_{i-1}(\mathbf{t}) + \mathbf{K}_i [\mathbf{y}_i(\mathbf{t}) - \mathbf{C}_i \hat{\mathbf{x}}_i(\mathbf{t})] \\ \hat{\mathbf{y}}_i(\mathbf{t}) &= \mathbf{C}_i \hat{\mathbf{x}}_i(\mathbf{t}) + \mathbf{D}_i \hat{\mathbf{y}}_{i-1}(\mathbf{t}), \end{aligned} \quad (33)$$

onde A_i, M_i, D_i e $K_i, i = 1, 2, 3$ são as matrizes dos subsistemas obtidas pela metodologia de identificação. A matriz B_i é nula dado que modelamos somente as séries temporais dos segmentos. \hat{y}_{i-1} é o vetor de saída estimado do subsistema $i-1$. As matrizes abaixo são modelos de espaço de estado obtidos para os 3 subsistemas que constituem o trecho de rio, cuja qualidade é modelada por esta proposta.

$$A_1 = \begin{bmatrix} -0.6285 & 0.3269 & -0.2807 & -0.0572 & -0.0008 & 0.2800 \\ -0.0708 & 0.6635 & -0.2498 & 0.4078 & 0.3772 & -0.8251 \\ -0.1935 & 0.2544 & 0.4427 & 0.6958 & -0.1131 & 0.2417 \\ -0.6336 & -0.3038 & -0.3744 & -0.1538 & -0.2622 & 0.2866 \\ -0.3205 & 0.2897 & 0.6623 & -0.3410 & 0.5333 & -0.1190 \\ 0.0100 & 0.2595 & -0.0957 & 0.0258 & 0.0016 & 0.9896 \end{bmatrix}, K_1 = \begin{bmatrix} -0.0085 & 0.1626 \\ 0.1878 & 0.3836 \\ -0.0064 & 0.1490 \\ -0.1050 & 0.0281 \\ 0.1428 & -0.0738 \\ 0.0636 & -0.0752 \end{bmatrix}$$

$$C_1 = \begin{bmatrix} 0.8785 & 1.0493 & -1.1512 & -0.4299 & 0.5639 & 1.9624 \\ -2.5241 & 1.0865 & 0.6654 & 2.0324 & -0.2483 & 0.4207 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0.5202 & -0.4052 & 0.6514 & 0.0762 & -0.0857 & -0.1513 \\ 0.3486 & 0.5738 & 0.3644 & 0.1640 & 0.2888 & 0.4224 \\ 0.1496 & -0.1314 & 0.1677 & 0.1452 & 0.1934 & 0.6684 \\ 0.0507 & -0.5159 & -0.1944 & -0.3830 & 0.7416 & 0.0867 \\ -0.0392 & 0.2357 & -0.1386 & -0.4390 & -0.0239 & 0.1867 \\ 0.1868 & -0.2305 & -0.1873 & -0.0842 & -0.5959 & 0.5975 \end{bmatrix}, M_2 = \begin{bmatrix} 0.2873 & -0.7123 \\ 0.0638 & 0.5145 \\ 0.5566 & 0.2486 \\ -0.2192 & 0.3349 \\ 0.0064 & -0.2930 \\ 0.0884 & 0.0783 \end{bmatrix}$$

$$C_2 = \begin{bmatrix} 0.1213 & 0.2021 & 0.3597 & -0.7437 & -0.0942 & -0.0867 \\ -0.7445 & -0.0795 & 0.4635 & 0.0942 & 0.0459 & 0.1762 \end{bmatrix}, D_2 = \begin{bmatrix} 0.0334 & 0.0728 \\ -0.3485 & 0.0963 \end{bmatrix}$$

$$K_2^T = \begin{bmatrix} 0.1347 & 0.3854 & 0.1931 & -0.2721 & 0.1355 & -0.0469 \\ 0.1974 & -0.0759 & 0.2479 & 0.0934 & 0.1720 & -0.1105 \end{bmatrix}.$$

$$A_3 = \begin{bmatrix} 0.7223 & -0.2689 & -0.2610 & -0.1061 & -0.3884 & 0.0242 & -0.4267 \\ 0.4971 & 0.5500 & 0.1773 & 0.2553 & 0.1389 & -0.4834 & -0.2295 \\ 0.1687 & -0.4508 & 0.7625 & -0.0330 & 0.3635 & 0.1577 & 0.1049 \\ -0.0099 & 0.2808 & 0.1389 & 0.6981 & -0.1024 & 0.5036 & 0.0265 \\ 0.3728 & -0.0632 & -0.4117 & 0.0378 & 0.2875 & 0.3512 & 0.6035 \\ -0.0474 & 0.1833 & 0.1602 & -0.2568 & -0.3782 & -0.1994 & 0.3166 \\ 0.1132 & 0.0131 & -0.2092 & -0.0278 & 0.5465 & -0.3468 & -0.1268 \end{bmatrix}$$

$$M_3 = \begin{bmatrix} 0.3369 & -0.4269 \\ -0.7249 & -0.0248 \\ 0.0043 & -0.0065 \\ 0.2658 & -0.0361 \\ -0.1591 & 0.4391 \\ 0.5044 & 0.1037 \\ 0.0294 & -0.2786 \end{bmatrix}, D_3 = \begin{bmatrix} 0.0247 & -0.3439 \\ -0.0030 & -0.2566 \end{bmatrix}$$

$$C_3 = \begin{bmatrix} -0.2328 & -0.3635 & -0.3999 & 0.4349 & 0.0740 & -0.3497 & -0.2346 \\ -0.1094 & 0.4279 & -0.2960 & -0.3756 & 0.2405 & 0.1778 & -0.1404 \end{bmatrix}$$

$$K_3^T = \begin{bmatrix} 0.0795 & -0.1835 & -0.1768 & -0.0450 & -0.0990 & -0.0399 & 0.0641 \\ -0.2812 & 0.2779 & 0.1060 & -0.1336 & 0.0562 & -0.0094 & 0.1234 \end{bmatrix}$$

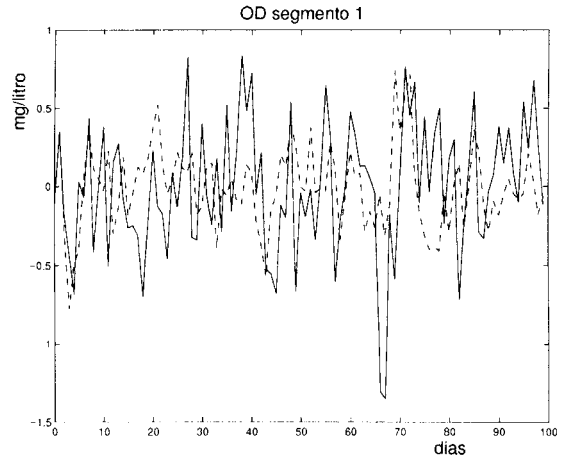
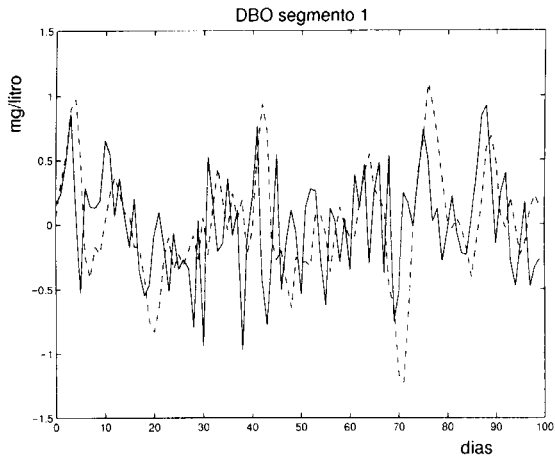


Figura 5: Concentrações de DBO e OD do primeiro segmento. A linha sólida representa os dados medidos e a pontilhada os estimados

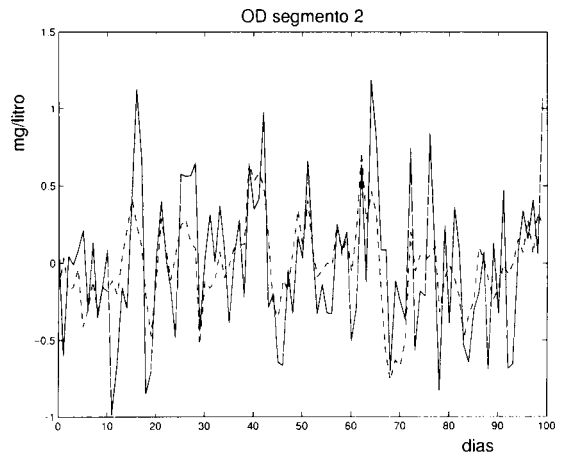
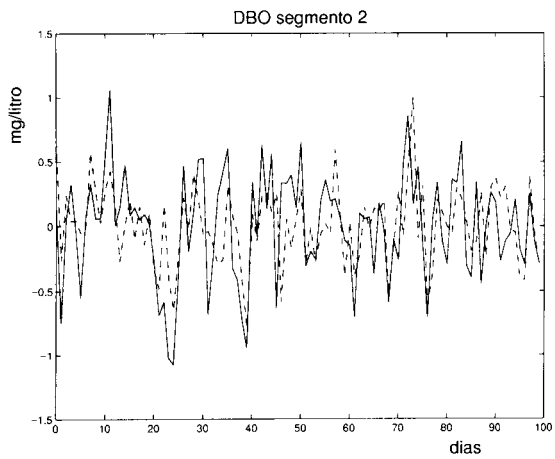


Figura 6: Concentrações de DBO e OD do segundo segmento. A linha sólida representa os dados medidos e a pontilhada os estimados

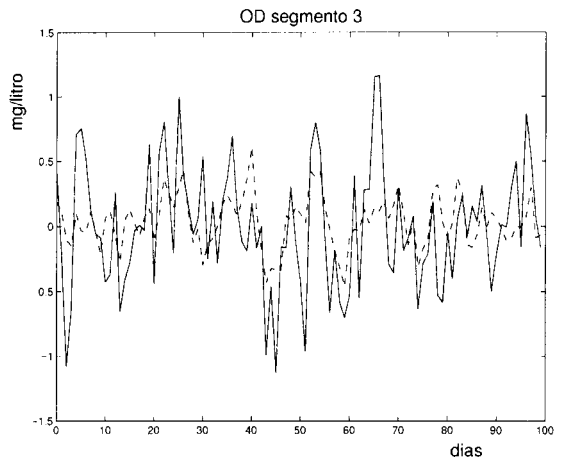
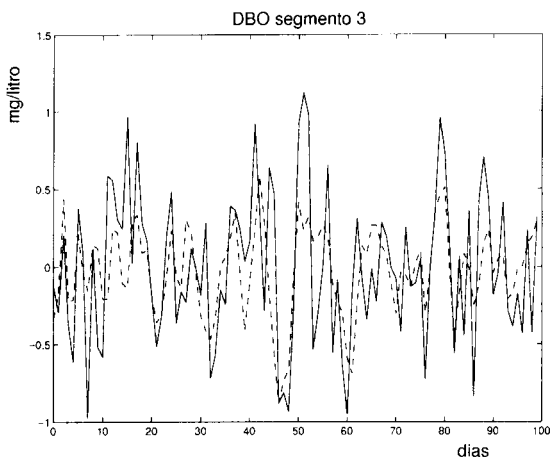


Figura 7: Concentrações de DBO e OD do terceiro segmento. A linha sólida representa os dados medidos e a pontilhada os estimados

7. Conclusões

Neste trabalho apresentamos uma proposta de método geral para identificação de sistemas multivariáveis serialmente interconectados. Métodos de identificação nos subespaços são utilizados como uma boa ferramenta de identificação de sistemas dinâmicos multivariáveis. Os resultados obtidos são preliminares, mas mostram que a metodologia pode ser utilizada com bons resultados. Propomos e apresentamos uma aplicação desta metodologia ao modelamento da qualidade da água de rios. Estudos mais avançados referentes às propriedades desta metodologia e desta proposta de aplicação devem ser e estão sendo feitos e serão objeto de próximos trabalhos.

Referências

- Cáceres, A. F. Torrico (1992). *Controle descentralizado: Técnicas e Aplicação ao Controle da Qualidade da Água de Rios*. Tese de Mestrado FEE-UNICAMP.
- Overschee, P. Van e B de Moor (1996). *Subspace Identification for Linear Systems*. Kluwer Academic Publishers.
- Verhaegen, M. (1994). Identification of the deterministic part of *mimo* state space models given in innovations form from input-output data. *Automatica* **30**(1), 61–74.
- Verhaegen, M. e P. Dewilde (1992). Subspace model identification - part 1 : The output-error state space model identification class algorithms. *International Journal of Control* **53**(5), 1187–1210.
- Viberg, M. (1995). Subspace-based methods for the identification of linear time-invariant systems. *Automatica* **31**(12), 1835–1851.
- Young, P. e B. Beck (1974). The modelling and control of water quality in a river system. *Automatica* **10**, 455–468.

SUBSPACE IDENTIFICATION METHOD FOR MULTIVARIABLE SYSTEMS WITH INTERCONNECTED STRUCTURE

Celso Pascoli Bottura

Departamento de Máquinas Componentes e Sistemas Inteligentes
Faculdade de Engenharia Elétrica e de Computação - UNICAMP
e-mail bottura@dmcsi.fee.unicamp.br

Angel Fernando Torrico Cáceres

Departamento de Engenharia de Eletricidade - Centro Tecnológico
Universidade Federal do Maranhão
e-mail ftorrico@dmcsi.fee.unicamp.br

Abstract *In this work a subspace identification approach for multivariable linear systems with serial interconnected structure is proposed. The global system identification is made through the individual identification of his subsystems. In this identification methodology, input/output data for each subsystem is necessary. A proposal of application to a river water quality modeling problem is made and its results are presented.*

Keywords *System identification, state space modeling, interconnected systems.*

IDENTIFICAÇÃO MULTIVARIADA PARALELA E DISTRIBUÍDA VIA TRATAMENTO MOESP

Celso Pascoli Bottura

bottura@dmcsi.fee.unicamp.br

Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e Computação
Cx. P. 6101 - 13083-970 - Campinas, SP, Brasil**Gilmar Barreto**

gbarreto@dmcsi.fee.unicamp.br

Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e Computação
Cx. P. 6101 - 13083-970 - Campinas, SP, Brasil**Annabell Del Real Tamariz**

annabell@dmcsi.fee.unicamp.br

Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e Computação
Cx. P. 6101 - 13083-970 - Campinas, SP, Brasil**João Viana da Fonseca Neto**

jviana@dee.ufma.br

Universidade Federal do Maranhão, Departamento Engenharia de Eletricidade
Av. dos Portugueses, s/n. - 65080-040 - São Luís, MA, Brasil

Resumo. Neste trabalho apresentamos um procedimento numérico paralelo para determinar uma realização no espaço de estado para um sistema dinâmico linear cuja representação é obtida a partir de sequências de dados multivariados de entrada - saída. A metodologia apresentada enfatiza a implementação paralela para a utilização de subrotinas que fazem a fatorização **QR** e a decomposição em valores singulares. A execução sequencial do algoritmo é feita e uma implementação paralela em um sistema de memória distribuída com uma estratégia assíncrona de paralelização sobre uma rede de estações de trabalho é proposta e é executada. Comparações entre os processamentos sequencial e paralelo são apresentadas.

Palavras chave: Método de Subespaço, Espaço de Estado, Identificação de Sistemas, Realização, Computação de Alto Desempenho.

1. Introdução

A representação de um sistema em variáveis no espaço de estado se torna muito conveniente em engenharia pois processos industriais multivariados podem ser descritos com exatidão por esta classe de modelos, além disso, hoje, as ferramentas de projeto de sistemas de controle que estão disponíveis estão baseadas neste tipo de tratamento. Por estas razões a classe de modelos no espaço de estado vem a ser muito importante.

Neste trabalho, o problema de identificação de sistemas multivariáveis pelo emprego da teoria da realização de sistemas lineares no espaço de estado é o objetivo central. Através da metodologia aqui abordada problemas de modelagem da dinâmica de sistemas aero-espaciais complexos, de dinâmica de estruturas, de vibrações mecânicas dentre outros, podem ser resolvidos de forma original e computacionalmente avançada. Problemas deste tipo são e serão em número cada vez maior estudados e resolvidos pela teoria de identificação em subespaços do espaço de estado, especialmente quando estiverem envolvidos com a solução de problemas de controle automático de sistemas dinâmicos.

O algoritmo apresentado é classificado como um esquema de identificação de modelo em subespaço do espaço de estado a partir de dados de entrada - saída. Esta estrutura é então explorada no cálculo da realização. Uma característica comum na organização destes algoritmos é a execução de uma fatorização, por exemplo **QR**, seguida por uma decomposição em valores singulares e a solução de um conjunto superdeterminado de equações. O tratamento computacional de alto desempenho em método de subespaços para modelagem de dados foi apresentado em Bottura (1999) e esta baseado no algoritmo *MOESP - MIMO Output-Error State Space model identification* apresentado em Verhaegen (1992).

O esquema apresentado supõe que o sistema tem uma sequência de medidas de entrada - saída disponível. Matematicamente o sistema dinâmico pode ser descrito pelo seguinte conjunto de equações :

$$x_{k+1} = \mathbf{A}x_k + \mathbf{B}u_k \tag{1}$$

$$y_k = \mathbf{C}x_k + \mathbf{D}u_k \tag{2}$$

onde os vetores $u_k \in \mathfrak{R}^m$ e $y_k \in \mathfrak{R}^l$ são as medidas no instante k , com entradas, m e saídas, l do processo. O vetor $x_k \in \mathfrak{R}^n$ é o vetor de estado do processo para o instante de tempo discreto k e contém os n valores numéricos do estado. As matrizes $\mathbf{A} \in \mathfrak{R}^{n \times n}$, $\mathbf{B} \in \mathfrak{R}^{n \times p}$, $\mathbf{C} \in \mathfrak{R}^{m \times n}$ e $\mathbf{D} \in \mathfrak{R}^{m \times p}$ terão dimensões adequadas com a ordem do sistema. O problema investigado neste trabalho é o de identificar o modelo no espaço de estado, incluindo a ordem do sistema e a quádrupla de matrizes, a partir de um algoritmo que explora as propriedades de subespaços. Para obter a ordem do sistema e a quádrupla de matrizes utiliza-se a aproximação de um subespaço definido pelo espaço gerado por colunas de matrizes determinadas por dados de entrada-saída do sistema.

Uma representação particularmente útil no estudo das propriedades de realização no espaço de estado, baseada na seqüência de entrada-saída, foi obtida por Moonen e outros (1989). Se $\{\mathbf{u}, \mathbf{y}\}$ e $\{\mathbf{x}^n\}$ podem ser representados pelas equações 1 e 2, então estes vetores também podem ser representados por uma equação de entrada e saída dada por :

$$\mathbf{Y}_{k,i,j} = \Gamma_i \mathbf{X}_{k,j} + \mathbf{H}_i \mathbf{U}_{k,i,j} \tag{3}$$

A matriz $\mathbf{Y}_{k,i,j}$ é bloco Hankel com i blocos linhas e j colunas e terá consecutivos vetores de saída $\mathbf{y}[k]$, (de dimensão $l \times 1$, onde l é o número de saídas), ordenados da seguinte maneira :

$$\mathbf{Y}_{k,i,j} = \begin{bmatrix} \mathbf{y}[k] & \mathbf{y}[k+1] & \dots & \mathbf{y}[k+j-1] \\ \mathbf{y}[k+1] & \mathbf{y}[k+2] & \dots & \mathbf{y}[k+j] \\ \mathbf{y}[k+2] & \mathbf{y}[k+3] & \dots & \mathbf{y}[k+j+1] \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}[k+i-1] & \mathbf{y}[k+i] & \dots & \mathbf{y}[k+j+i-2] \end{bmatrix} \tag{4}$$

A matriz $\mathbf{U}_{k,i,j}$ é bloco Hankel com as mesmas dimensões de bloco da matriz $\mathbf{Y}_{k,i,j}$ e contém consecutivos vetores de entrada $\mathbf{u}[k]$, (de dimensão $m \times 1$, onde m é o número de entradas), ordenados da seguinte maneira :

$$\mathbf{U}_{k,i,j} = \begin{bmatrix} \mathbf{u}[k] & \mathbf{u}[k+1] & \dots & \mathbf{u}[k+j-1] \\ \mathbf{u}[k+1] & \mathbf{u}[k+2] & \dots & \mathbf{u}[k+j] \\ \mathbf{u}[k+2] & \mathbf{u}[k+3] & \dots & \mathbf{u}[k+j+1] \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{u}[k+i-1] & \mathbf{u}[k+i] & \dots & \mathbf{u}[k+j+i-2] \end{bmatrix} \tag{5}$$

\mathbf{X} contém os vetores de estado:

$$\mathbf{X}_{k,j} = [\mathbf{x}[k] \quad \mathbf{x}[k+1] \quad \mathbf{x}[k+2] \quad \dots \quad \mathbf{x}[k+j-1]] \tag{6}$$

Γ_i é uma matriz com estrutura semelhante à da matriz de observabilidade é chamada de matriz de observabilidade estendida e dada por :

$$\Gamma_i = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} \tag{7}$$

e finalmente, \mathbf{H}_i é uma matriz triangular inferior dada por:

$$\mathbf{H}_{k,i,j} = \begin{bmatrix} \mathbf{D} & 0 & 0 & \dots & 0 \\ \mathbf{CB} & \mathbf{D} & 0 & \dots & 0 \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \dots & 0 \\ \mathbf{CA}^2\mathbf{B} & \mathbf{CAB} & \mathbf{CB} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{CA}^{i-2}\mathbf{B} & \mathbf{CA}^{i-3}\mathbf{B} & \mathbf{CA}^{i-4}\mathbf{B} & \dots & \mathbf{D} \end{bmatrix} \tag{8}$$

Observa-se que \mathbf{D} está na diagonal principal da matriz \mathbf{H}_i .

As matrizes expressas anteriormente foram obtidas através de substituição direta nas matrizes apresentadas nas equações 1 e 2 propiciando uma representação muito útil para estudar as propriedades de uma realização no espaço estado a partir das seqüências multivariadas de entrada e de saída:

$$\begin{bmatrix} \mathbf{y}[k] & \mathbf{y}[k+1] & \dots & \mathbf{y}[k+j-1] \\ \mathbf{y}[k+1] & \mathbf{y}[k+2] & \dots & \mathbf{y}[k+j] \\ \mathbf{y}[k+2] & \mathbf{y}[k+3] & \dots & \mathbf{y}[k+j+1] \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{y}[k+i-1] & \mathbf{y}[k+i] & \dots & \mathbf{y}[k+j+i-2] \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} [\mathbf{x}[k] \quad \mathbf{x}[k+1] \quad \mathbf{x}[k+2] \quad \dots \quad \mathbf{x}[k+j-1]] + \begin{bmatrix} \mathbf{D} & 0 & 0 & \dots & 0 \\ \mathbf{CB} & \mathbf{D} & 0 & \dots & 0 \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \dots & 0 \\ \mathbf{CA}^2\mathbf{B} & \mathbf{CAB} & \mathbf{CB} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{CA}^{i-2}\mathbf{B} & \mathbf{CA}^{i-3}\mathbf{B} & \mathbf{CA}^{i-4}\mathbf{B} & \dots & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{u}[k] & \mathbf{u}[k+1] & \dots & \mathbf{u}[k+j-1] \\ \mathbf{u}[k+1] & \mathbf{u}[k+2] & \dots & \mathbf{u}[k+j] \\ \mathbf{u}[k+2] & \mathbf{u}[k+3] & \dots & \mathbf{u}[k+j+1] \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{u}[k+i-1] & \mathbf{u}[k+i] & \dots & \mathbf{u}[k+j+i-2] \end{bmatrix} \quad (9)$$

Definindo uma matriz \mathcal{O}_i como o produto da matriz de observabilidade estendida, $\mathbf{\Gamma}_i$, pela seqüência de vetores de estado, tem-se a projeção das saídas futuras no subespaço das saídas e entradas passadas na direção das entradas futuras:

$$\mathcal{O}_i = \mathbf{\Gamma}_i \mathbf{X}_{k,j} \stackrel{\text{def}}{=} \mathbf{Y}_f / \mathbf{U}_f \begin{bmatrix} \mathbf{U}_p \\ \mathbf{Y}_p \end{bmatrix} \quad (10)$$

onde \mathbf{Y}_f e \mathbf{U}_f são submatrizes de $\mathbf{Y}_{k,i,j}$ e $\mathbf{U}_{k,i,j}$ respectivamente conforme definida em Overschee & De Moor (1996).

2. Implementação do algoritmo MOESP

A implementação do algoritmo para obtenção da quádrupla de matrizes, \mathbf{A} , \mathbf{B} , \mathbf{C} e \mathbf{D} , consiste de um pré-processamento dos dados através de uma fatorização \mathbf{RQ} seguida de uma decomposição em valores singulares, SVD, e da solução de um conjunto de equações lineares superdeterminado.

2.1 Cálculo das Matrizes \mathbf{A} e \mathbf{C}

Suponha que aplicamos ao sistema uma seqüência de j entradas independentes $\mathbf{u}_i, i = 0, 1, 2, \dots, j-1$, mas que temos apenas um segmento finito de uma seqüência de saída, por exemplo de $t = 0$ até $t = L$ para $d \leq L \ll j$. Podemos então através de uma escolha adequada de uma combinação linear relacionar as entradas conhecidas com a seqüência de saída medida de tal forma que :

$$\begin{bmatrix} \mathbf{u}_0 & \mathbf{u}_1 & \dots & \mathbf{u}_{j-1} \\ \mathbf{y}_0 & \mathbf{y}_1 & \dots & \mathbf{y}_{j-1} \end{bmatrix} \mathbf{Q} = \begin{bmatrix} \mathbf{u}'_0 & \mathbf{u}'_1 & \dots & \mathbf{u}'_{j-1} \\ \mathbf{y}'_0 & \mathbf{y}'_1 & \dots & \mathbf{y}'_{j-1} \end{bmatrix} \quad (11)$$

Através de transformações apropriadas, por exemplo fatorização \mathbf{QR} , podemos reescrever a equação 11 como :

$$\begin{bmatrix} \mathbf{u}_0 & \mathbf{u}_1 & \dots & \mathbf{u}_{j-1} \\ \mathbf{y}_0 & \mathbf{y}_1 & \dots & \mathbf{y}_{j-1} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & 0 \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^* \\ \mathbf{Q}_2^* \end{bmatrix} \quad (12)$$

onde \mathbf{R}_{11} e \mathbf{R}_{22} são matrizes triangulares inferiores de dimensão $(L+1) \times (L+1)$ e $[(\mathbf{Q}_1^*)^T \quad (\mathbf{Q}_2^*)^T]$ são as primeiras $2(L+1)$ colunas de uma matriz unitária \mathbf{Q} tendo dimensão $j \times j$.

Logo,

$$[\mathbf{y}'_0 \quad \mathbf{y}'_1 \quad \dots \quad \mathbf{y}'_{j-1}] = [\mathbf{R}_{22} \quad 0] \quad (13)$$

Devido a propriedade de invariância ao deslocamento da matriz Γ , a partir da decomposição em valores singulares de \mathbf{R}_{22} obtemos :

$$\mathbf{R}_{22} = [\mathbf{U}_n \quad | \quad \mathbf{U}_n^\perp] = \begin{bmatrix} \mathbf{S}_n & 0 \\ 0 & \mathbf{S}_2 \end{bmatrix} \begin{bmatrix} \mathbf{V}_n^T \\ (\mathbf{V}_n^\perp)^T \end{bmatrix} \quad (14)$$

Então a matrizes \mathbf{A} e \mathbf{C} serão dadas por :

$$\mathbf{U}_n^{(1)} \mathbf{A}_T = \mathbf{U}_n^{(2)} \quad (15)$$

$$\mathbf{C}_T = \mathbf{U}_n(1:l,:) \quad (16)$$

onde $\mathbf{U}_n^{(1)}$ é uma submatriz composta pelas primeiras $(i-1)$ colunas da matriz \mathbf{U}_n e $\mathbf{U}_n^{(2)}$ é uma submatriz composta pelas últimas $(i-1)$ colunas da matriz \mathbf{U}_n .

2.2 Cálculo das Matrizes \mathbf{B} e \mathbf{D}

O desenvolvimento algébrico para o cálculo das matrizes \mathbf{B} e \mathbf{D} é obtido a partir da fatorização \mathbf{QR} da matriz de dados de entrada e saída :

$$\begin{bmatrix} \mathbf{U}_{1,i,j} \\ \mathbf{Y}_{1,i,j} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{11} & | & 0 \\ \mathbf{R}_{21} & | & \mathbf{R}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_1^j \\ \mathbf{Q}_2^j \end{bmatrix} \quad (17)$$

Da equação 17 obtemos :

$$\mathbf{U}_{1,i,j} = \mathbf{R}_{11} \mathbf{Q}_1^j \quad (18)$$

Retomando a equação 3 $\mathbf{Y}_{k,i,j} = \Gamma_i \mathbf{X} + \mathbf{H}_{k,i,j} \mathbf{U}_{k,i,j}$ podemos substituir $\mathbf{Y}_{1,i,j}$, para $k=1$, obtido da equação 17 deste modo encontramos :

$$\mathbf{Y}_{1,i,j} = \Gamma_i \mathbf{X} + \mathbf{H}_{1,i,j} \mathbf{U}_{1,i,j} = \mathbf{R}_{21} \mathbf{Q}_1^j \mathbf{R}_{22} \mathbf{Q}_2^j \quad (19)$$

Como o espaço coluna de \mathbf{R}_{22} é igual ao espaço coluna de Γ_i e se a decomposição em valores singulares de $\mathbf{R}_{22} = \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^T$, então haverá uma matriz de transformação de tal forma que $\Gamma_i \mathbf{T} = \mathbf{U}_n$. Então podemos escrever que :

$$\mathbf{U}_n \mathbf{T}^{-1} \mathbf{X} + \mathbf{H}_{1,i,j} \mathbf{R}_{11} \mathbf{Q}_1^j = \mathbf{R}_{21} \mathbf{Q}_1^j + \mathbf{U}_n \mathbf{S}_n \mathbf{V}_n^T \mathbf{Q}_2^j \quad (20)$$

Como $\mathbf{U}_n^\perp \mathbf{U}_n = 0$ obtemos da equação 20:

$$[(\mathbf{U}_n^\perp)^T \mathbf{H}_{1,i,j} \mathbf{R}_{11} - (\mathbf{U}_n^\perp)^T \mathbf{R}_{21}] \mathbf{Q}_1^j = 0 \quad (21)$$

ou

$$(\mathbf{U}_n^\perp)^T \mathbf{H}_{1,i,j} \mathbf{R}_{11} - (\mathbf{U}_n^\perp)^T \mathbf{R}_{21} = 0 \quad (22)$$

Como \mathbf{R}_{11} tem inversa :

$$(\mathbf{U}_n^\perp)^T \mathbf{H}_{1,i,j} - (\mathbf{U}_n^\perp)^T \mathbf{R}_{21} \mathbf{R}_{11}^{-1} = 0 \quad (23)$$

como $\mathbf{H}_{1,i,j} = \text{func}[\mathbf{B} \mid \mathbf{D}]$ temos a quádrupla de matrizes determinada.

3. Ambiente de Programação

Neste trabalho a obtenção da solução paralela para determinação de uma realização no espaço de estado é feita em um ambiente que utiliza como plataforma de programação paralela o *MPI : Message Passing Interface*, com memória distribuída, comunicação síncrona, e a semântica das funções para enviar e receber mensagens é bloqueante com modo de programação *SPMD : Single Program, Multiple Data*, ou seja todos os processadores vão executar o mesmo programa, porém podem executar tarefas diferentes sobre dados diferentes, que foi implementada utilizando as subrotinas da biblioteca pública *Scalapack*.

A *Scalapack* é uma biblioteca de alto desempenho com rotinas de Álgebra Linear para computadores *MIMD*¹ com paradigma de troca de mensagens e memória distribuída em redes de estações de trabalho suportando *PVM* e ou *MPI*. Utiliza também como parte de suas rotinas a biblioteca sequencial *BLAS* -

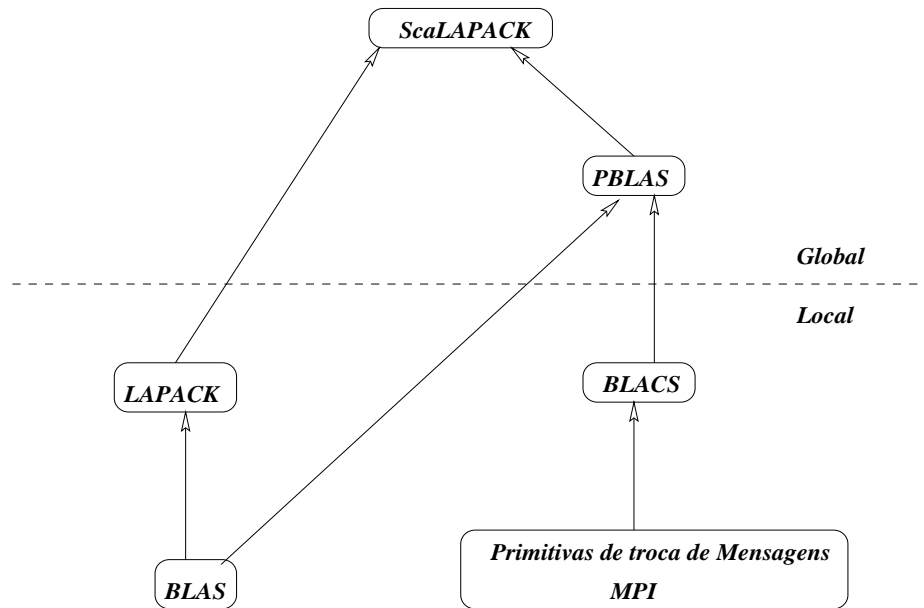


Figura 1: Estrutura do Ambiente Computacional Scalapack

Basic Linear Algebra Subroutines, a paralela *PBLAS - Parallel BLAS*, os subprogramas de comunicação da álgebra linear básica *BLACS - Basic Linear Algebra Communication Subroutines* e as rotinas da *LAPACK - Linear Algebra PACKage* a estrutura do *Scalapack* é mostrada na Figura 1.

Similarmente ao *LAPACK*, as rotinas da *Scalapack* são baseadas em algoritmos com particionamento em blocos, o que minimiza a frequência de movimento de dados entre os diferentes níveis da memória hierárquica. Os algoritmos são apresentados como processos e a comunicação pode ser ponto-a-ponto ou em grupo quando necessário. De forma geral, a *Scalapack* foi desenvolvida e testada para o caso de um processo por processador.

Dos vários algoritmos paralelos que têm sido propostos, um dos que receberam maior atenção recentemente é o baseado em multiplicação de matrizes, pois a multiplicação de matrizes de alta ordem é forte candidata à paralelização, Henry & Watkins & Dongarra (1997).

3.1. Aspectos Paralelos Para Processamento Computacional

Em todas as rotinas os elementos locais de uma matriz são armazenados em arranjos uni-dimensionais. Por exemplo, os elementos locais de uma matriz A ($m \times n$) podem ser armazenados num arranjo vetorial real $A(*)$. Embora seja mais conveniente considerar A como um arranjo de duas dimensões ($LDA, *$), onde LDA deve ser maior ou igual ao número de linhas de A armazenadas na linha específica da grade de processadores e o arranjo A deve ter um número de colunas maior ou igual ao número de colunas de A armazenadas na coluna específica da grade dos processadores. Maiores informações da distribuição da matriz sobre os processadores participantes no processo são encapsuladas num arranjo denominado "descriptor de arranjo". Tal descriptor é associado a cada uma das matrizes distribuídas. As coordenadas do descriptor unicamente determinam o mapa das coordenadas da matriz sobre a memória dos processadores locais. Por exemplo, na fatoração **QR** e rotinas associadas, o descriptor, *DESCA*, tem dimensão (9), e armazena as seguintes informações:

- DESCA(1) - Tipo do descriptor (para a distribuição de bloco cíclica de duas dimensões deve ser igual a 1);
- DESCA(2) - Encadeamento para a biblioteca (variável retornada pela rotina de inicialização para identificar a matriz **A**);
- DESCA(3) - m , número de linhas de **A**;
- DESCA(4) - n , número de colunas de **A** ;
- DESCA(5) - M_b , fator bloqueante usado para distribuir as linhas de **A**, isto é, o número de linhas armazenadas num bloco;

¹Esta arquitetura combina paralelismo nos fluxos de dados e de instruções; é composta de unidades de processamento, cada qual capaz de realizar operações lógicas e aritméticas padrões. As unidades operam assincronamente sob o controle de fluxo de instruções individuais.

- DESCA(6) - N_b , fator bloqueante usado para distribuir as colunas de \mathbf{A} , isto é, o número de colunas armazenadas num bloco;
- DESCA(7) - Índice do processador linha sobre o qual a primeira linha de \mathbf{A} é armazenada;
- DESCA(8) - Índice do processador coluna sobre o qual a primeira coluna de \mathbf{A} é armazenada;
- DESCA(9) - Dimensão (LDA) do arranjo local \mathbf{A} armazenando os blocos locais de \mathbf{A} .

Em geral, o arranjo descritor não muda através do ciclo de vida da matriz para a qual ele esta associado.

3.2. Estrutura da rede de estações de trabalho

As simulações foram realizadas num computador paralelo virtual formado por computadores conectados numa rede de área local com máquinas *workstations-Sun* do tipo SPARC-20, SPARC-4, SPARC-Classic e SPARC-2. Esta rede pertence ao DMCSI : *Departamento de Máquinas, Componentes e Sistemas Inteligentes*; e sua configuração esta apresentada na figura 2. Esta rede é constituída de duas sub redes que possuem como ponto de ligação (*Gateway*) entre elas a máquina *Papagaio*. As simulações foram realizadas na sub rede LCSi : *Laboratório de Controle e Sistemas Inteligentes*, utilizando a máquina *aguia*, que é a servidora desta rede e é o ponto de ligação com a rede da FEEC : *Faculdade de Engenharia Elétrica e Computação*, *Saturno*, *Urano* e outras.

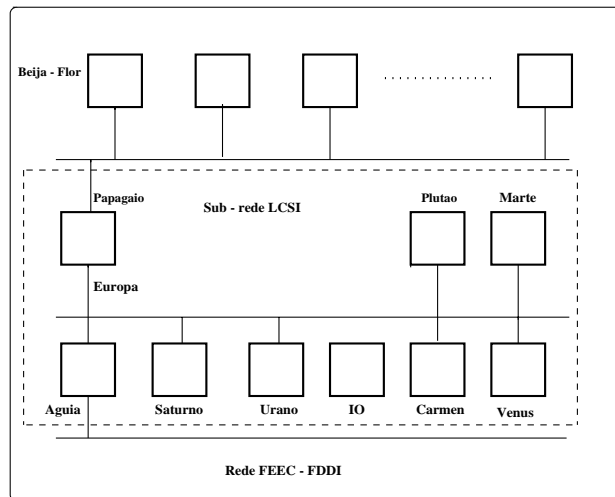


Figura 2: Estrutura da rede de estações de trabalho

3.3. Subrotina para Fatorização QR

Para o processamento usamos o algoritmo **QR**, aplicado em uma matriz de Hankel, \mathbf{H} , construída com os dados de entrada - saída do sistema. Uma fatorização **QR** é usada para comprimir os dados. O algoritmo para fatorização **QR** explora a estrutura da matriz de Hankel. O método implementado é numericamente estável e eficiente.

3.3.1. Descrição

A subrotina para fatorização **QR** de uma matriz real, \mathbf{A}_s , $m \times n$, calcula $\mathbf{A}_s = \mathbf{QR}$, onde \mathbf{A}_s é uma submatriz de tamanho $m_{\mathbf{A}} \times n_{\mathbf{A}}$ da matriz \mathbf{A} , isto é,

$$\mathbf{A}_s(1 : m, 1 : n) = \mathbf{A}(i_{\mathbf{A}} : i_{\mathbf{A}} + m - 1, j_{\mathbf{A}} : j_{\mathbf{A}} + n - 1)$$

Se $i_{\mathbf{A}} = j_{\mathbf{A}} = 1, m = m_{\mathbf{A}}, n = n_{\mathbf{A}}$ então $\mathbf{A}_s = \mathbf{A}$.

A matriz ortogonal \mathbf{Q} não é formada explicitamente, mas é representada como o produto de refletorees elementares

$$\mathbf{Q} = \mathbf{H}_1 \mathbf{H}_2 \dots \mathbf{H}_k, \text{ onde } k = \min(m, n) \tag{24}$$

Cada refletor elementar \mathbf{H}_l têm a forma

$$\mathbf{H}_l = \mathbf{I} - \tau_l v_l v_l^T, \quad (25)$$

onde τ_l um escalar real e v_l um vetor real de tamanho m .

3.4. Subrotina para Decomposição em Valores Singulares

A subrotina calcula a decomposição em valores singulares (*SVD*) de uma matriz real cujas colunas são distribuídas numa grade de duas dimensões.

A *SVD* de uma matriz A real retangular $m \times n$, onde $m \geq n$, é definida como

$$A = \begin{bmatrix} U & \tilde{U} \end{bmatrix} \begin{bmatrix} \Sigma \\ 0 \end{bmatrix} V^T = U \Sigma V^T$$

onde U é uma matriz $m \times n$ de vetores singulares esquerdos, \tilde{U} é uma matriz $m \times (m - n)$, V é uma matriz ortogonal $n \times n$ de vetores singulares direitos e Σ é uma matriz diagonal $n \times n$ de valores singulares. Os valores singulares $\sigma_1, \sigma_2, \dots, \sigma_n$ são não negativos e decrescentes em ordem de magnitude. A matriz $\begin{bmatrix} U & \tilde{U} \end{bmatrix}$ é ortogonal.

Para o caso $m < n$ a *SVD* pode ser definida na forma

$$A = U \begin{bmatrix} \Sigma & 0 \end{bmatrix} \begin{bmatrix} V & \tilde{V} \end{bmatrix}^T = U \Sigma V^T$$

onde U é uma matriz $m \times m$ ortogonal, \tilde{V} é uma matriz $n \times (n - m)$, V é uma matriz ortogonal $n \times m$ de vetores singulares direitos e Σ é uma matriz diagonal $m \times m$ de valores singulares. A matriz $\begin{bmatrix} V & \tilde{V} \end{bmatrix}$ é ortogonal.

Em geral, matrizes \tilde{U} e \tilde{V} não são únicas.

3.5 Especificações para Subrotinas

3.5.1 Especificação para Subrotina QR

```
SUBROUTINE PSGEQRF(M, N, A, IA, JA, DESCA, TAU, WORK, LWORK, INFO)
REAL          A(*), TAU(*), WORK(*)
INTEGER       M, N, IA, JA, DESCA(*), LWORK, INFO
```

A matriz A deve ser particionada em $M_b^A \times N_b^A$ blocos retangulares os quais são armazenados num arranjo A com uma distribuição de bloco cíclica de duas dimensões.

3.5.2 Especificação para Subrotina SVD

```
SUBROUTINE PSGESVD( JOBU, JOBVT, M, N, A, IA, JA, DESCA, S, U, IU,
$                 JU, DESCU, VT, IVT, JVT, DESCVT, WORK, LWORK,
$                 INFO )
INTEGER          DESCA( * ), DESCU( * ), DESCVT( * )
REAL            A( * ), S( * ), U( * ), VT( * ), WORK( * )
CHARACTER       JOBU, JOBVT
INTEGER         IA, INFO, IU, IVT, JA, JU, JVT, LWORK, M, N
```

As colunas da matriz A são alocadas a processadores lógicos da grade, linha por linha, isto é, na linha de maior ordem da grade, começando no processador lógico $\{0, 0\}$. Cada processador lógico que contém colunas da matriz, contém $N_b = \left\lceil \frac{n}{p} \right\rceil$ colunas, exceto para o último processador que contém dados e o número de colunas deve ser menor que N_b . Este processador deverá conter $\text{mod}(n, N_b)$ colunas se $\text{mod}(n, N_b) \neq 0$ e poderia conter N_b colunas. Alguns processadores lógicos podem não conter qualquer coluna da matriz se n não é relativamente maior que p , mas se $n > (p - 1)^2$ então todos os processadores podem certamente conter colunas da matriz.

O número de processadores lógicos que contém colunas da matriz é dado por $p_d = \left\lceil \frac{n}{N_b} \right\rceil$.

O seguinte exemplo ilustra o caso onde o último processador com dados é o último processador da grade. Além do mais, o número de colunas no último processador com dados não é igual ao número de colunas nos demais processadores.

Se $m_p = 2$, $n_p = 2$ então $p = m_p \times n_p = 4$. Se $n = 11$ então $N_b = \left\lceil \frac{n}{p} \right\rceil = \lceil 2.75 \rceil = 3$, $mod(n, N_b) = 2 \neq 0$ e $p_d = \left\lceil \frac{n}{N_b} \right\rceil = \lceil 3.6667 \rceil = 4$.

Processador {0, 0}	Processador {0, 1}
$N_x = 3$	$N_x = 3$
colunas (1 : 3)	colunas (4 : 6)

Processador {1, 0}	Processador {1, 1}
$N_x = 3$	$N_x = 2$
colunas (7 : 9)	colunas (10 : 11)

4. Resultados e Comentários

Nesta seção apresentamos alguns dos resultados obtidos com uma versão paralela do método MOESP utilizando a biblioteca *ScaLAPACK*.

Todos os experimentos foram realizados utilizando Fortran 77 e aritmética com dupla precisão, padrão IEEE.

A partir de uma seqüência de entrada-saída, com vetores de entrada multivariada de 3 elementos e vetores de saída multivariada de 2 elementos tem-se uma realização no espaço de estado com as seguintes matrizes: $\mathbf{A}_{4 \times 4}$; $\mathbf{B}_{4 \times 3}$; $\mathbf{C}_{2 \times 4}$ e $\mathbf{D}_{2 \times 3}$.

As matrizes do modelo no espaço de estado representantes do conjunto de dados original são apresentadas a seguir:

$$\mathbf{A} = \begin{bmatrix} 0.2128 & 0.1360 & 0.1979 & -0.0836 \\ 0.1808 & 0.4420 & -0.3279 & 0.2344 \\ -0.5182 & 0.1728 & -0.5448 & -0.3083 \\ 0.2252 & -0.0541 & -0.4679 & 0.8290 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} -0.0101 & 0.0317 & -0.9347 \\ -0.0600 & 0.5621 & 0.1657 \\ -0.3310 & -0.3712 & -0.5846 \\ -0.2655 & 0.4255 & 0.2204 \end{bmatrix},$$

$$\mathbf{C} = \begin{bmatrix} 0.6557 & -0.2502 & -0.5188 & -0.1229 \\ 0.6532 & -0.1583 & -0.0550 & -0.2497 \end{bmatrix} \quad \text{e} \quad \mathbf{D} = \begin{bmatrix} -0.4326 & 0.1253 & -1.1465 \\ -1.6656 & 0.2877 & 1.1909 \end{bmatrix}$$

Depois de executar nossa primeira versão de programa em paralelo, obtivemos os tempos de execução correspondentes a 3 experimentos apresentados na Tabela 1.

Leitura de Dados	0.1283	0.1280	0.1280
Achando QR	0.2969	0.0862	0.3326
Formando \mathbf{R}_{32}	0.3062	0.1974	0.1867
Achando SVD	0.0407	0.0372	0.3999
Achando $\mathbf{U}_1, \mathbf{U}_2$	0.0640	0.0635	0.0652
Achando SVD de \mathbf{U}_1	0.0421	0.0273	0.3749
Achando $inv(\mathbf{S}_1)$	0.0093	0.0099	0.0065
Formando \mathbf{Y}_{ij}	0.2060	0.2171	0.0363
Formando $\mathbf{\Gamma}$	0.0851	0.2520	0.0434
Formando P	0.3975	0.3178	0.1864
Achando QR	0.3893	0.1275	0.3395
Formando Rnew	0.3400	0.1475	0.1703
Achando SVD de Rnew	0.0549	0.0590	0.4334
Achando $inv(\mathbf{S})$	0.0424	0.0422	0.0469

Tabela 1: *Tempos de Execução: 1, 1 e 2 Processadores*

Através da modelagem computacional aqui desenvolvida para um conjunto com 490 amostras de valores

entrada - saída, obtivemos o seguinte modelo no espaço de estado:

$$\mathbf{A}_c = \begin{bmatrix} 0.3386 & 0.4402 & -0.5891 & -0.0241 \\ -0.3428 & -0.6781 & -0.5043 & -0.3215 \\ -0.0519 & 0.0062 & 0.7988 & -0.5805 \\ 0.0003 & 0.0043 & 0.0103 & 0.4796 \end{bmatrix}, \quad \mathbf{B}_c = \begin{bmatrix} -0.0940 & 0.2708 & 1.0339 \\ 0.2965 & 0.2712 & 0.3916 \\ 0.1043 & -0.1980 & 0.1669 \\ 0.0088 & 0.0115 & -0.0037 \end{bmatrix},$$

$$\mathbf{C}_c = \begin{bmatrix} -0.5725 & 0.5875 & -0.1078 & -0.4545 \\ -0.6617 & 0.0679 & -0.0000 & 0.5835 \end{bmatrix} \quad \text{e} \quad \mathbf{D}_c = \begin{bmatrix} -0.4326 & 0.1253 & -1.1465 \\ -1.6656 & 0.2877 & 1.1909 \end{bmatrix}$$

A verificação da qualidade do modelo foi realizada através dos parâmetros de Markov, como os do sistema original $[\mathbf{D}, \mathbf{CB}, \mathbf{CAB}, \mathbf{CA}^2\mathbf{B}, \dots]$ coincidem com os do modelo obtido $[\mathbf{D}_c, \mathbf{C}_c\mathbf{B}_c, \mathbf{C}_c\mathbf{A}_c\mathbf{B}_c, \mathbf{C}_c\mathbf{A}_c^2\mathbf{B}_c, \dots]$ concluímos que o modelo obtido está correto.

5. Agradecimento

Os autores agradecem os vários anos de colaboração frutífera e agradável com o Prof. Maurício José Bordon que cedeu muitas visões e idéias expressadas neste trabalho e à FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo.

6. Referências

- Bertsekas, D. and Tsitsiklis, J., 1989, "Parallel and Distributed Computations", Prentice Hall.
- Bottura, Celso P.; Barreto, Gilmar; Bordon, Mauricio José and Tamariz, Annabell D.R., 1999, "Tratamento Computacional de Alto Desempenho em Método de Subespaços para Modelagem de Dados", XV Congresso Brasileiro de Engenharia Mecânica - Cobem 99, Águas de Lindóia, São Paulo, Brasil.
- Golub, Gene H. and Ortega, James M., 1993, "Scientific Computing: An Introduction with Parallel Computing", Academic Press.
- Henry, Greg; Watkins, David and Dongarra, Jack, 1997, "A Parallel Implementation of the Nonsymmetric QR Algorithm for Distributed Memory Architectures", University of Tennessee Computer Science Technical Report, CS-97-352, March 1997 and LAPACK Working Note 121.
- Jainandung K. and Deprettere, E. F., 1989, "A New Class Of Parallel Algorithms For Solving Systems Of Linear Equations", SIAM J.Sci.Stat.Comput. Vol.10, No.5, 880-912.
- Moonen, Marc; De Moor, Bart ; Vandenberghe, Lieven and Vandewalle, Joos, 1989, "On-and Off-Line Identification Of Linear State-Space Models", INT.J. Control, Vol.49, No.1, 219-232.
- Overschee, Peter van and De Moor, Bart , 1996, "Subspace Identification For Linear Systems", Kluwer Academic Publishers.
- Verhaegen, Michel and Dewilde, Patrick, 1992, "Subspace Model Identification", Int.J. Control, Vol.56, n.5, 1187-1210.

Parallel and Distributed Multivariable Identification Via the MOESP Approach.

In this work a parallel numerical procedure for determining a state space realization for a linear dynamic system representing input-output multivariate data sequences is developed. The methodology presented in the paper: "An High Performance Computational Approach In Subspaces Methods For Data Modelling - COBEM 99" is here utilized to parallelize the "MOESP : Multivariable Output Error State Space" algorithm for computational data modeling in the state space.

For data processing using QR algorithm, a matrix H is constructed, where; $\mathbf{H} = [\mathbf{U}_f \mathbf{U}_p \mathbf{Y}]$; and \mathbf{U}_p , \mathbf{U}_f and \mathbf{Y} are block Hankel matrices defined in terms of the input and output data. A QR factorization is used to compress the data. The fast QR algorithm uses a QR factorization which exploits the block-Hankel structure.

The implemented method is numerically stable reliable and efficient. Parallel implementations for the QR and SVD algorithms are also developed.

In this work a sequential execution of the MOESP algorithm is made and a parallel implementation on a distributed memory system with an asynchronous parallelization strategy over a workstations network is proposed and executed.

Comparisons between the sequential and the parallel and distributed implementations are made.

Keywords: *Subspace Method, State Space, System Identification, Realization, High Performance Computing, Computational Data Modelling .*

SUSPENSÃO SEMI-AATIVA DE VEÍCULOS DE PASSEIO EMPREGANDO CONTROLADORES *FUZZY*

Rafael Luís Teixeira

Universidade Federal de Uberlândia - Faculdade de Engenharia Mecânica - Campus Santa Mônica 38400-089, Uberlândia-MG
rafael@mecanica.ufu.br

José Francisco Ribeiro

Universidade Federal de Uberlândia - Faculdade de Engenharia Mecânica - Campus Santa Mônica 38400-089, Uberlândia-MG
jribeiro@mecanica.ufu.br

Resumo. Este trabalho apresenta o projeto de controle de um sistema de suspensão semi-ativa de um veículo de passeio. Fundamentado na teoria de controle da lógica *fuzzy*, o controlador proposto tem seus parâmetros obtidos através da análise intuitiva do sistema multi-corpos. É desenvolvido o modelo do veículo, considerando quatro suspensões acopladas por um corpo rígido, configurando um sistema não linear com 7 graus de liberdade, sujeito a excitação da superfície da pista. A partir deste modelo são feitas as simulações. O controlador proposto é submetido a um conjunto de testes que procura analisar a eficiência, a robustez e a estabilidade da solução sugerida. Uma comparação também é feita com um controlador passivo não linear no domínio do tempo e da frequência.

Palavras chaves. Controladores *fuzzy*; Suspensão semi-ativa.

1. Introdução

Os controladores baseados em lógica *fuzzy* foram introduzidos em 1965 por Lofti A. Zadeh. Esses controladores se aplicam em áreas onde decisões objetivas precisam ser tomadas a partir de um conjunto de informações muitas vezes vagas, imprecisas e não adequadamente quantificadas. Nos sistemas *fuzzy* as incertezas são expressas em variáveis lingüísticas (Ex: muito fraco, pouco forte, etc) e a tecnologia *fuzzy* é natural e intuitiva na sua formulação, pois tenta imitar o comportamento consciente ou a estratégia de controle de um operador humano. Pode ser aplicada a sistemas lineares e não lineares, é de baixo custo e de fácil implementação, apresentando boas características de robustez a incertezas ou variações paramétricas.

Este trabalho apresenta o projeto de controle de uma suspensão semi-ativa de um veículo de passeio, que procura minimizar as vibrações indesejáveis e melhorar o conforto dos passageiros. O controlador proposto é baseado na técnica de controle da lógica *fuzzy*. Recentemente, as investigações de suspensões semi-ativa e ativa com controladores *fuzzy*, têm se destacado na indústria automobilística (Wu & Xu, 1999; Yoshimura, 1998; Li et al. 1998). Mendel (1995) reporta a aplicação da lógica *fuzzy* em outros sistemas veiculares, como: piloto automático (Nissam), freio ABS (Mitsubishi e Nissam) e transmissão automática (Subaru e Nissam). A eficiência dos modelos *fuzzy*, ou seja, a capacidade de mapeamento entre as entradas e saídas de um sistema, depende da escolha de uma série de parâmetros que definem, entre outras coisas, a natureza dos conjuntos *fuzzy* envolvidos, o universo de variação das variáveis de entrada e saída, a base de regras empregada, os mecanismos de inferência utilizados, etc. Todavia no projeto de um controlador *fuzzy* a maior dificuldade está no estabelecimento do conjunto de regras, que normalmente são construídas com base na experiência e intuição de um especialista. Uma questão crítica é o grande número de combinações que podem ocorrer no projeto de controladores *fuzzy*, especialmente quando o sistema apresenta muitas variáveis lingüísticas e muitas funções de pertinência. Quanto maior o número de regras, maior será o tempo computacional de processamento, podendo inviabilizar o uso em tempo real do controlador. Há, portanto, um compromisso entre o número de regras, a velocidade de processamento e o desempenho esperado para o controlador. O veículo de passeio é modelado de forma completa configurando um sistema dinâmico com vários graus de liberdade. São feitas simulações numéricas que comparam um sistema de suspensão passivo com o semi-ativo proposto. Quando passivo, as forças de amortecimento são consideradas não lineares como os amortecedores reais, que apresentam coeficientes de amortecimento que variam com a velocidade. Quando semi-ativo, as forças de amortecimento são substituídas por forças obtidas a partir do controlador *fuzzy*. As respostas dos principais parâmetros dos sistemas passivo e semi-ativo são comparadas no tempo e na frequência. Desta forma, este artigo faz uma comparação entre a suspensão comercial com amortecedor passivo e a suspensão semi-ativa com controlador *fuzzy*. Este trabalho está assim organizado: na seção 2 é apresentado o modelo do veículo completo, na seção 3 é apresentado alguns fundamentos da lógica *fuzzy* bem como o controlador projetado, na seção 4 são apresentados os resultados dos ensaios realizados e, na última seção, as conclusões e futuros desdobramentos são apresentados.

2. O modelo matemático do veículo

O veículo foi modelado como um sistema dinâmico com 7 graus de liberdade, quais sejam: as translações verticais das quatro rodas e do centro de gravidade do veículo, e as rotações de mergulho e rolagem do veículo.

A figura 1 mostra o diagrama esquemático do veículo e o sistema de referência utilizado no desenvolvimento das equações.

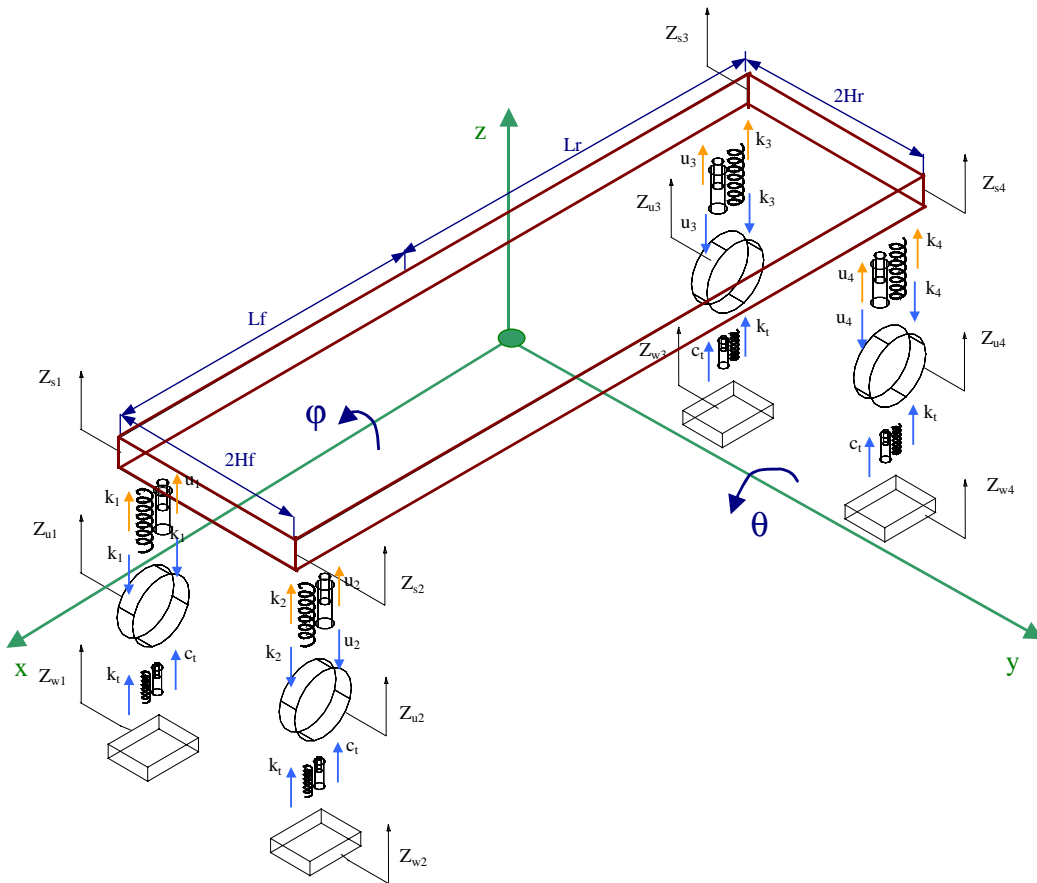


Figura 1. Diagrama do corpo livre do sistema com os 7 graus de liberdade e o sistema de referência xyz.

No desenvolvimento do modelo, supõe-se que:

- (i) as molas apresentam rigidez constante;
- (ii) os pneus apresentam rigidez e amortecimento constantes e
- (iii) o sistema é excitado através dos pneus por um sinal que modela a pista de rolamento.

As equações que descrevem o movimento são obtidas a partir da 2ª lei de Newton que, após um arranjo adequado dos termos, são assim descritas:

$$M_s \cdot \ddot{Z}_s + k_1 \cdot (Z_{s1} - Z_{u1}) + k_2 \cdot (Z_{s2} - Z_{u2}) + k_3 \cdot (Z_{s3} - Z_{u3}) + k_4 \cdot (Z_{s4} - Z_{u4}) + u_1 + u_2 + u_3 + u_4 = 0 \quad (1)$$

$$I_{xx} \cdot \ddot{\phi} - L_f [k_1 \cdot (Z_{s1} - Z_{u1}) + k_2 \cdot (Z_{s2} - Z_{u2})] + L_r [k_3 \cdot (Z_{s3} - Z_{u3}) + k_4 \cdot (Z_{s4} - Z_{u4})] - L_f (u_1 + u_2) + L_r (u_3 + u_4) = 0 \quad (2)$$

$$I_{yy} \cdot \ddot{\theta} + H_f [k_2 \cdot (Z_{s2} - Z_{u2}) - k_1 \cdot (Z_{s1} - Z_{u1})] + H_r [k_4 \cdot (Z_{s4} - Z_{u4}) - k_3 \cdot (Z_{s3} - Z_{u3})] + H_f (u_2 - u_1) + H_r (u_4 - u_3) = 0 \quad (3)$$

$$m_1 \cdot \ddot{Z}_{u1} + k_t \cdot (Z_{u1} - Z_{w1}) + c_t \cdot (\dot{Z}_{u1} - \dot{Z}_{w1}) - k_1 \cdot (Z_{s1} - Z_{u1}) - u_1 = 0 \quad (4)$$

$$m_2 \cdot \ddot{Z}_{u2} + k_t \cdot (Z_{u2} - Z_{w2}) + c_t \cdot (\dot{Z}_{u2} - \dot{Z}_{w2}) - k_2 \cdot (Z_{s2} - Z_{u2}) - u_2 = 0 \quad (5)$$

$$m_3 \cdot \ddot{Z}_{u3} + k_t \cdot (Z_{u3} - Z_{w3}) + c_t \cdot (\dot{Z}_{u3} - \dot{Z}_{w3}) - k_3 \cdot (Z_{s3} - Z_{u3}) - u_3 = 0 \quad (6)$$

$$m_4 \cdot \ddot{Z}_{u4} + k_t \cdot (Z_{u4} - Z_{w4}) + c_t \cdot (\dot{Z}_{u4} - \dot{Z}_{w4}) - k_4 \cdot (Z_{s4} - Z_{u4}) - u_4 = 0 \quad (7)$$

As equações que definem as posições dos cantos da carroceria do veículo, podem facilmente ser obtidas através de uma matriz de transformação, que considera uma translação em z e duas rotações em x e y. Essas equações são:

$$Z_{s1} = -\sin(\theta)L_f - \cos(\theta)\sin(\varphi)H_f + Z_s \tag{8}$$

$$Z_{s2} = -\sin(\theta)L_f + \cos(\theta)\sin(\varphi)H_f + Z_s \tag{9}$$

$$Z_{s3} = \sin(\theta)L_r - \cos(\theta)\sin(\varphi)H_f + Z_s \tag{10}$$

$$Z_{s4} = \sin(\theta)L_r + \cos(\theta)\sin(\varphi)H_f + Z_s \tag{11}$$

As funções trigonométricas envolvidas revelam as não linearidades no modelo veicular multi- corpos, o que dificulta o projeto de controladores pelos métodos convencionais.

As Tabelas 1 e 2 mostram os significados das variáveis do modelo e os valores numéricos adotados.

Tabela 1. Parâmetros dinâmicos do modelo do veículo.

Z_s	Deslocamento do centro de gravidade em relação a origem
θ	Ângulo de mergulho em relação a origem
φ	Ângulo de rolagem em relação a origem
$Z_{s1} = Z_{s2} = Z_{s3} = Z_{s4}$	Deslocamentos dos cantos da carroceria em relação a origem
$Z_{u1} = Z_{u2} = Z_{u3} = Z_{u4}$	Deslocamentos das rodas em relação a origem
$Z_{w1} = Z_{w2} = Z_{w3} = Z_{w4}$	Deslocamentos da pista em relação a origem
$u_1 = u_2 = u_3 = u_4$	Forças de amortecimento

Tabela 2. Parâmetros físicos adotados para o veículo [Wu e Xu , 1999].

M_s	Massa do corpo rígido principal	1.110	[Kg]
$m_1 = m_2$	Massa das rodas dianteiras	22	[Kg]
$m_3 = m_4$	Massa das rodas traseiras	34	[Kg]
$k_1 = k_2$	Rigidez das molas dianteiras	20.140	[N/m]
$k_3 = k_4$	Rigidez das molas traseiras	15.100	[N/m]
k_t	Rigidez dos pneus	156.200	[N/m]
c_t	Coefficiente de amortecimento dos pneus	40	[Ns/m]
I_θ	Momento de inércia em x	1240	[N m s ²]
I_φ	Momento de inércia em y	310	[N m s ²]
L_r	Distância do centro de gravidade até a traseira	1,455	[m]
L_f	Distância do centro de gravidade até a dianteira	1,006	[m]
H_r / H_f	Bitola traseira/dianteira e do veículo	0,625	[m]

Supôs-se que a fonte de excitação vibratória do veículo deriva-se de irregularidades na pista. Assim modelou-se a força de excitação através de um perfil de deslocamento/velocidade para a pista definido pelos valores de Z_w / \dot{Z}_w , que aparecem nas equações de 4 a 7. Para a excitação do veículo, foram considerados três sinais:

- (i) para simulação de um “piso asfáltico”: um sinal de ruído branco filtrado em 5 Hz conforme o gráfico da Figura 2;
- (ii) para simulação de uma “subida de estrada”: um sinal senoidal na frequência de 0,005 ciclos / m e
- (iii) para simulação de uma “lombada”: um sinal senoidal na frequência de 5 ciclos /m.

Assim, ao compor estes sinais, a excitação resultante total é a mostrada na Figura 3. Salienta-se que existem dois sinais de excitação resultante, um para cada eixo do veículo, ou seja, tomou-se cuidado em distanciar as lombadas exatamente da distância entre eixos do veículo.

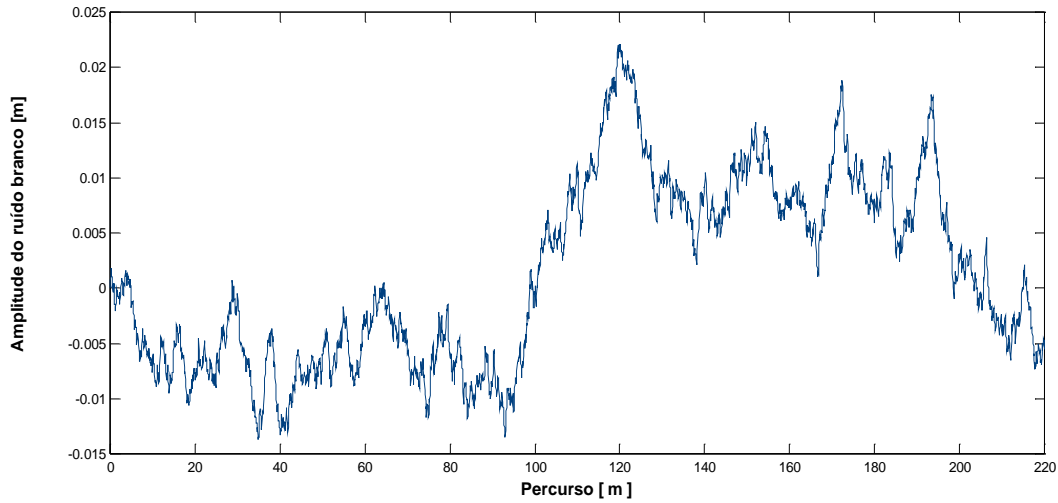


Figura 2. Sinal do ruído branco filtrado em 5 Hz adicionado aos sinais da pista.

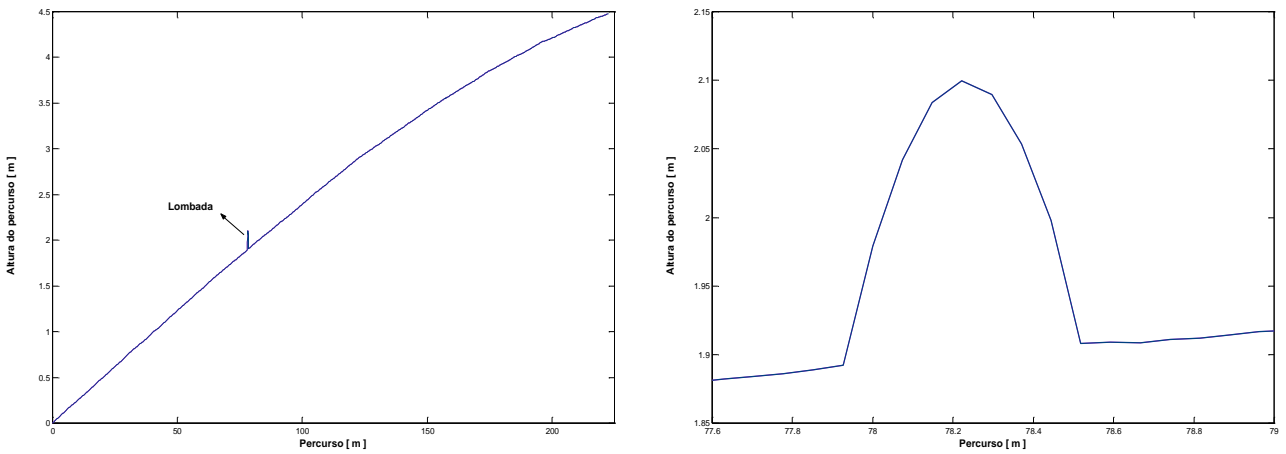


Figura 3a. Pista resultante ou excitação externa ao sistema.

3b. Detalhamento da lombada.

Para as simulações do sistema de suspensão passivo foi considerado um amortecimento não linear, de acordo com Yoshimura (1998). Observe-se, na Figura 4, que os amortecedores dianteiros são distintos dos traseiros tanto no que se refere ao coeficiente de amortecimento quanto às amplitudes das forças de tração e compressão. A velocidade relativa da suspensão é calculada pela derivada da diferença entre a posição do canto do veículo e a posição da roda, isto é, $(Z_{s1} - Z_{u1})$, $(Z_{s2} - Z_{u2})$, $(Z_{s3} - Z_{u3})$ e $(Z_{s4} - Z_{u4})$. Uma vez obtidas as velocidades relativas, as forças de amortecimento são derivadas a partir da Figura 4, que descrevem o comportamento típico de amortecedores convencionais. Nas simulações as forças desenvolvidas pelos amortecedores foram sempre saturadas no intervalo $[-500 \text{ N}, 1500 \text{ N}]$.

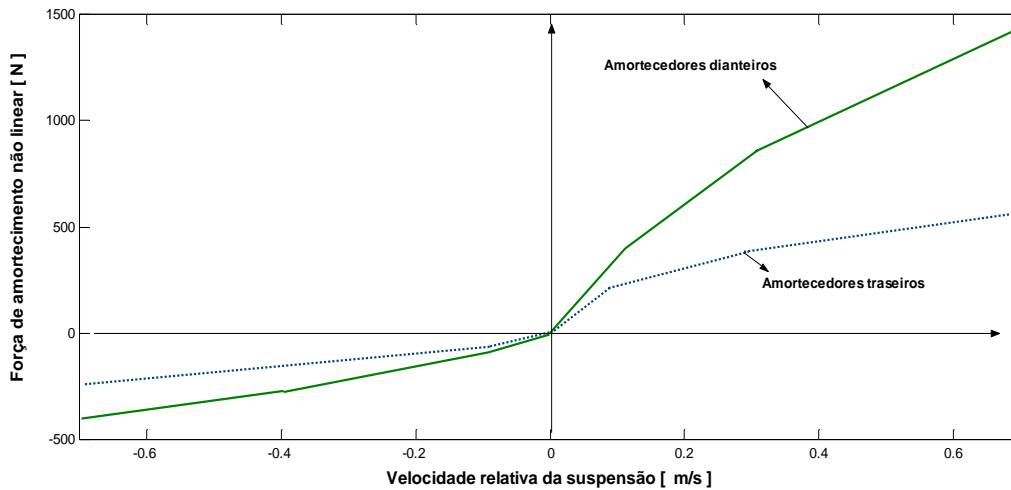
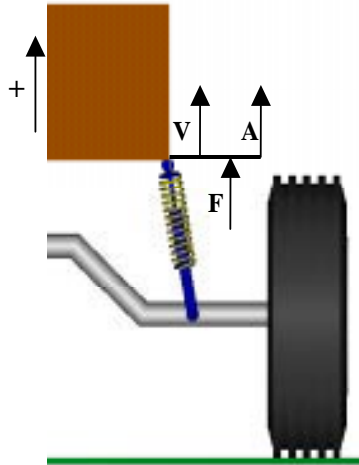


Figura 4. Curvas dos amortecedores não lineares do sistema de suspensão passiva, Yoshimura (1998).

3. O algoritmo da lógica fuzzy e o controlador fuzzy projetado

Em oposição à lógica tradicional, onde as proposições assumem somente dois valores: falso (0) ou verdadeiro (1), a lógica fuzzy trata as proposições com diferentes graus de valores verdadeiros ou falsos. Em Teixeira (2001) e em Lee (1990) é feito um apanhado da teoria dos conjuntos fuzzy.

A questão central dos controladores fuzzy é o método de inferência, que é construído e formulado por meio dos princípios da teoria dos conjuntos fuzzy. O método de inferência é baseado em um conjunto de regras *se-sentão* e de funções de pertinência que descrevem as variáveis do sistema. Para ilustrar, suponha o problema de controle de apenas um quarto de uma suspensão veicular, conforme esquematizado na Figura 5, onde **V** é a velocidade da massa suspensa; **A** é a aceleração da massa suspensa e **F** é a força resultante do sistema mola e amortecedor. O controlador fuzzy para esse caso é formulado considerando a aceleração e a velocidade como entradas e a força resultante como saída. Para cada uma das entradas e saídas são definidas três variáveis linguísticas: Negativo (N), Zero (Z) e Positivo (P). Cada uma destas variáveis é representada por um conjunto fuzzy com função de pertinência triangular.



Base de regras:

R₁: Se **V** é Negativa e **A** é Negativa então **F** é Positiva.

R₂: Se **V** é Positivo e **A** é Positiva então **F** é Negativa.

R₃: Se **V** é Zero e **A** é Zero então **F** é Zero.

Figura 5. Desenho esquemático da suspensão de um quarto de veículo e base de regras.

A partir das três regras anunciadas é possível, graficamente, mostrar como o algoritmo fuzzy funciona quando o método de inferência é *min-max* e a *defuzzificação* se faz pelo método do centróide, ilustrados na Figura 6.

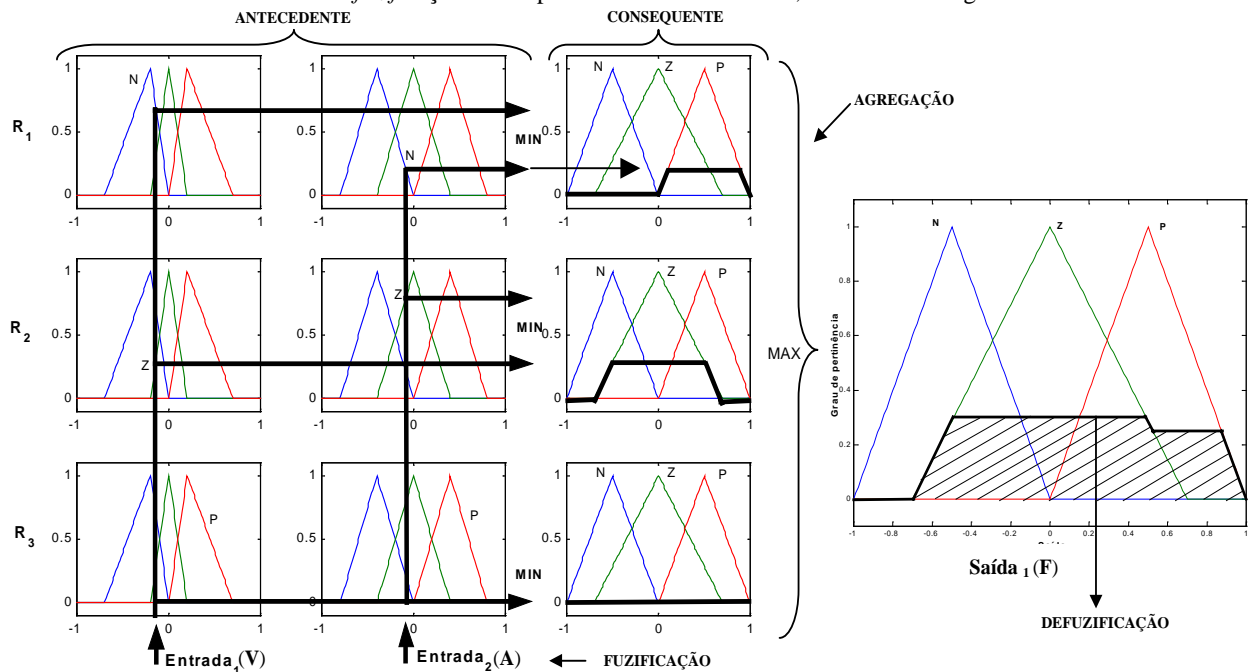


Figura 6. Exemplo gráfico do algoritmo fuzzy para somente três regras.

Para entendimento, considere a primeira regra: as entradas são *fuzzificadas* nas curvas N, isto é, o cálculo do antecedente da regra. O mínimo dos valores (no caso 0,2) é o valor de pertinência que irá truncar a curva da saída definida no consequente da primeira regra, a curva P. Este processo é repetido para todas as regras envolvidas. Percebe-se que a terceira regra não é atingida para estas entradas. Após esta fase é feita a agregação de todas as regras, que para o caso estudado, escolheu-se o operador MAX, que faz a união dos sub-conjuntos fuzzy. O cálculo do centróide da área agregada corresponde ao valor de controle da saída, o que é denominado de *defuzzificação* da saída.

No primeiro passo na construção do controlador fuzzy definiu-se as variáveis físicas de entrada e saída. Como variáveis de entrada adotou-se o vetor de variação do erro: $E = X - X_{ref}$ e a sua derivada, e para a saída do controlador o vetor $U = [u_1 u_2 u_3 u_4]$ que são as forças de amortecimento, conforme mostra o diagrama de bloco da Figura 7.

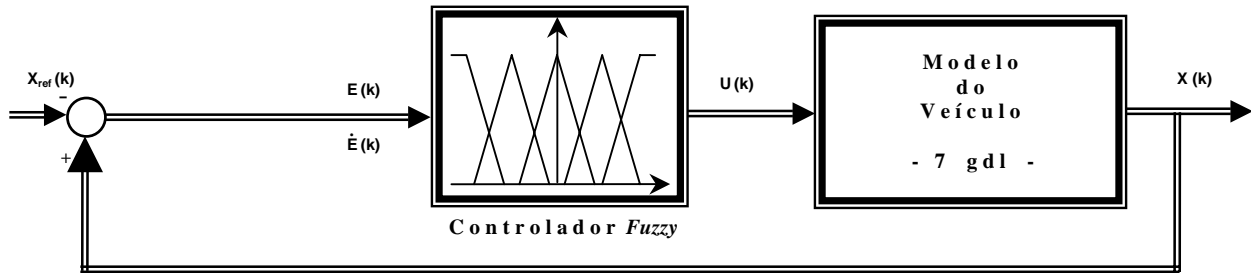


Figura 7. Diagrama de bloco com o sistema de controle implementado.

Para o projeto do controlador fuzzy foram considerados seis entradas e quatro saídas, configurando desta forma em um sistema MIMO (mult-input mult-output). O vetor de estado de saída do sistema, X, e o vetor de referência são:

$$X = [\dot{Z}_s \quad \dot{\theta} \quad \dot{\phi}] \tag{12}$$

$$X_{ref} = [\dot{Z}_w \quad \dot{\theta}_{pista} \quad 0] \tag{13}$$

Uma vez escolhidas as variáveis de entrada e saída, definiu-se o range de variação das mesmas observando o comportamento das variáveis envolvidas durante simulações. Em seguida, particionou-se os universos do discurso das entradas e saídas em funções de pertinência, conforme mostrado nas Figuras 8 e 9. A base de regras, descritas na Tabela 03 e considerada como a última etapa de construção do controlador fuzzy, foi obtida a partir da intuição do projetista e após uma série de ensaios numéricos.

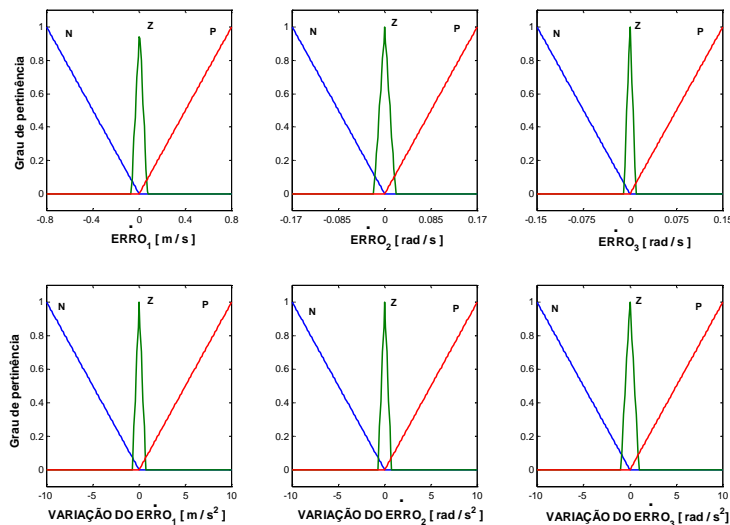


Figura 8. Curvas de pertinência das 6 entradas do controlador fuzzy.

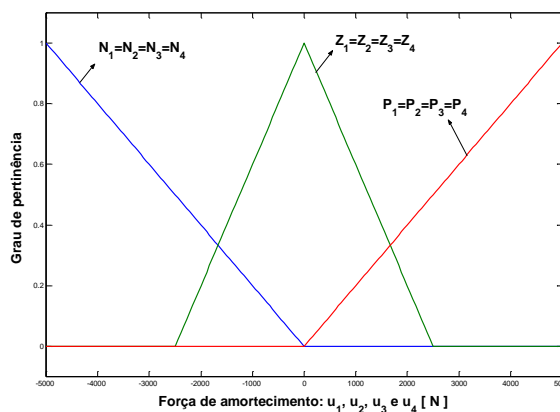


Figura 9. Curvas de pertinência das 4 saídas do controlador fuzzy.

Tabela 3. Base de regras do controlador *fuzzy* proposto.

		\dot{e}_1			\dot{e}_2			\dot{e}_3		
		N	Z	P	N	Z	P	N	Z	P
\ddot{e}_1	N	P1-P2-P3-P4	P1-P2-P3-P4	Z1-Z2-Z3-Z4	Z1-Z2-P3-P4	Z1-Z2-P3-P4	Z1-Z2-Z3-Z4	Z1-P2-Z3-P4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4
	Z	P1-P2-P3-P4	Z1-Z2-Z3-Z4	N1-N2-N3-N4	Z1-Z2-P3-P4	Z1-Z2-Z3-Z4	Z1-Z2-N3-N4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4
	P	Z1-Z2-Z3-Z4	N1-N2-N3-N4	N1-N2-N3-N4	Z1-Z2-Z3-Z4	Z1-Z2-N3-N4	Z1-Z2-N3-N4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4	Z1-N2-Z3-N4
\ddot{e}_2	N	Z1-Z2-P3-P4	Z1-Z2-P3-P4	Z1-Z2-Z3-Z4	N1-N2-P3-P4	N1-N2-P3-P4	Z1-Z2-Z3-Z4	Z1-P2-Z3-P4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4
	Z	Z1-Z2-P3-P4	Z1-Z2-Z3-Z4	Z1-Z2-N3-N4	N1-N2-P3-P4	Z1-Z2-Z3-Z4	P1-P2-N3-N4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4
	P	Z1-Z2-Z3-Z4	Z1-Z2-N3-N4	Z1-Z2-N3-N4	Z1-Z2-Z3-Z4	P1-P2-N3-N4	P1-P2-N3-N4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4	Z1-N2-Z3-N4
\ddot{e}_3	N	Z1-P2-Z3-P4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4	Z1-P2-Z3-P4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4	N1-P2-N3-P4	N1-P2-N3-P4	Z1-Z2-Z3-Z4
	Z	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4	Z1-P2-Z3-P4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4	N1-P2-N3-P4	Z1-Z2-Z3-Z4	P1-N2-P3-N4
	P	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4	Z1-N2-Z3-N4	Z1-Z2-Z3-Z4	Z1-N2-Z3-N4	Z1-N2-Z3-N4	Z1-Z2-Z3-Z4	P1-N2-P3-N4	P1-N2-P3-N4

4. Resultados das simulações numéricas

Com os sistemas passivo e semi-ativo projetado, as simulações foram feitas sob as mesmas condições iniciais, quais sejam: $Z_s=0,075$ m ; $\theta= 1^\circ$ e $\varphi = 1^\circ$. A velocidade adotada para o veículo foi de 80 Km/h e, conforme mostrado na Figura 3, no instante $t=3,75$ s (78 m de percurso) há uma perturbação configurando uma lombada. Utilizou-se para simulação um integrador Range Kutta de quarta ordem e de passo variável.

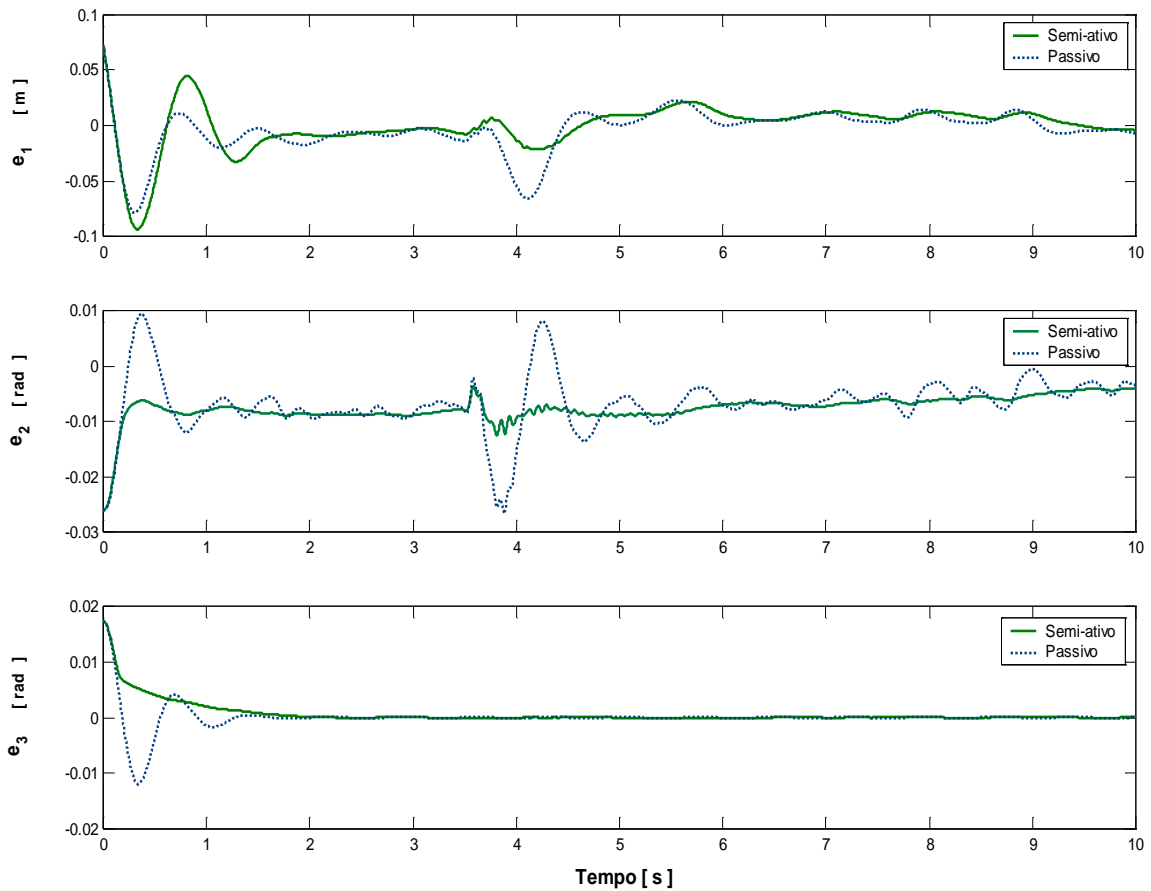


Figura 10. Respostas do erro do deslocamento do centro de gravidade e dos erros dos ângulos de mergulho e rolagem para simulações com os sistemas semi-ativo e passivo.

Os resultados no domínio do tempo mostram que o sistema semi-ativo apresentou um melhor comportamento quando comparado ao sistema passivo para as três variáveis dinâmicas escolhidas para análise.

Em virtude da semelhança dos esforços de controle nas quatro suspensões do veículo, seja no sistema passivo ou no sistema semi-ativo, optou-se em apresentar apenas uma força de controle de cada sistema, conforme mostram as Figuras 11 e 12.

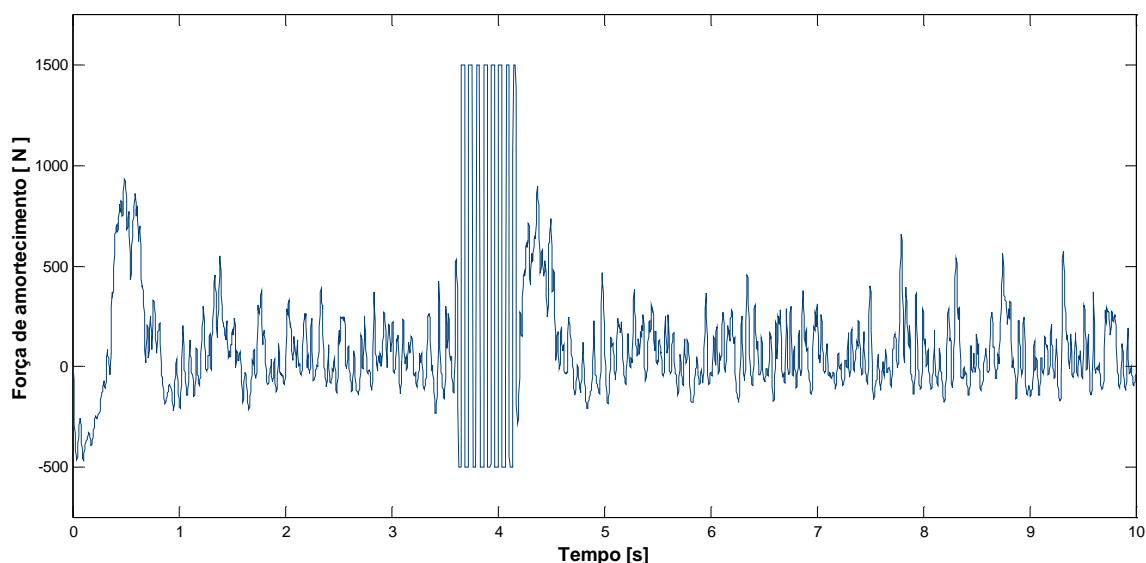


Figura 11. Resposta do esforço de controle ou da força de amortecimento para o sistema passivo.

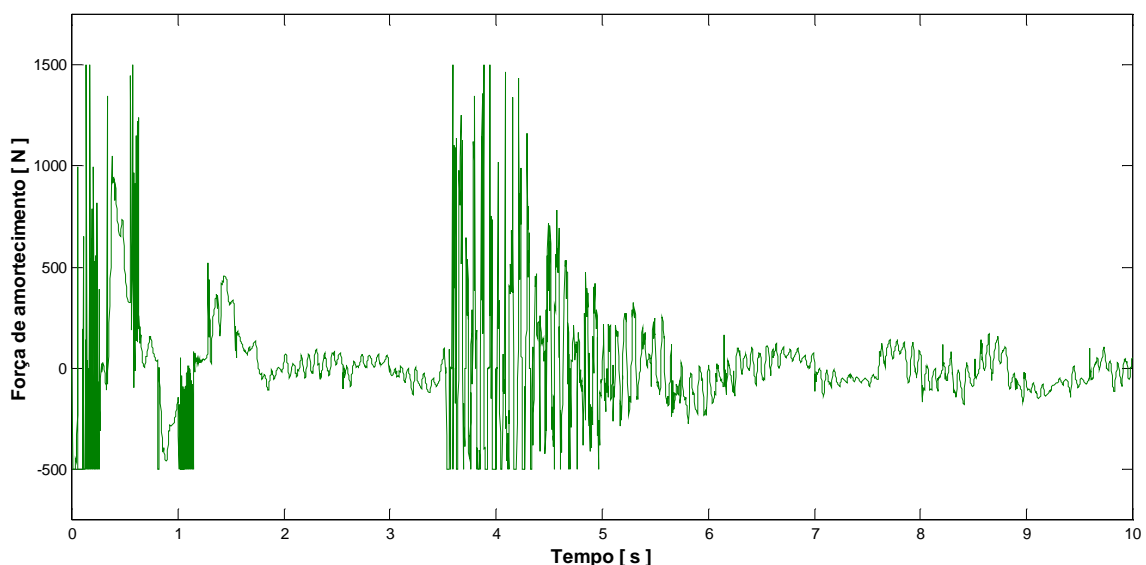


Figura 12. Resposta do esforço de controle ou da força de amortecimento para o sistema semi-ativo.

Devido ao modelo não linear utilizado para os amortecedores, conforme mostrado na Figura 4, observa-se a saturação nos esforços de controle para o sistema passivo. Já o sistema semi-ativo saturou-se apenas quando submetido à compressão.

Observa-se ainda que os esforços de controle no sistema semi-ativo são menores, o que pode implicar na construção de amortecedores de menor porte, mais baratos portanto.

Outro aspecto importante é que o sistema semi-ativo apresentou um melhor comportamento em regime permanente. Observa-se que este comportamento é bastante sensível à função de pertinência das entradas, o que leva a escolha de funções triangulares com base bastantes estreitas.

Para as análises no domínio da frequência, calculou-se a densidade espectral de potência dos sinais analisados observando o intervalo de 0 à 5 Hz. A Figura 13 mostram estes resultados

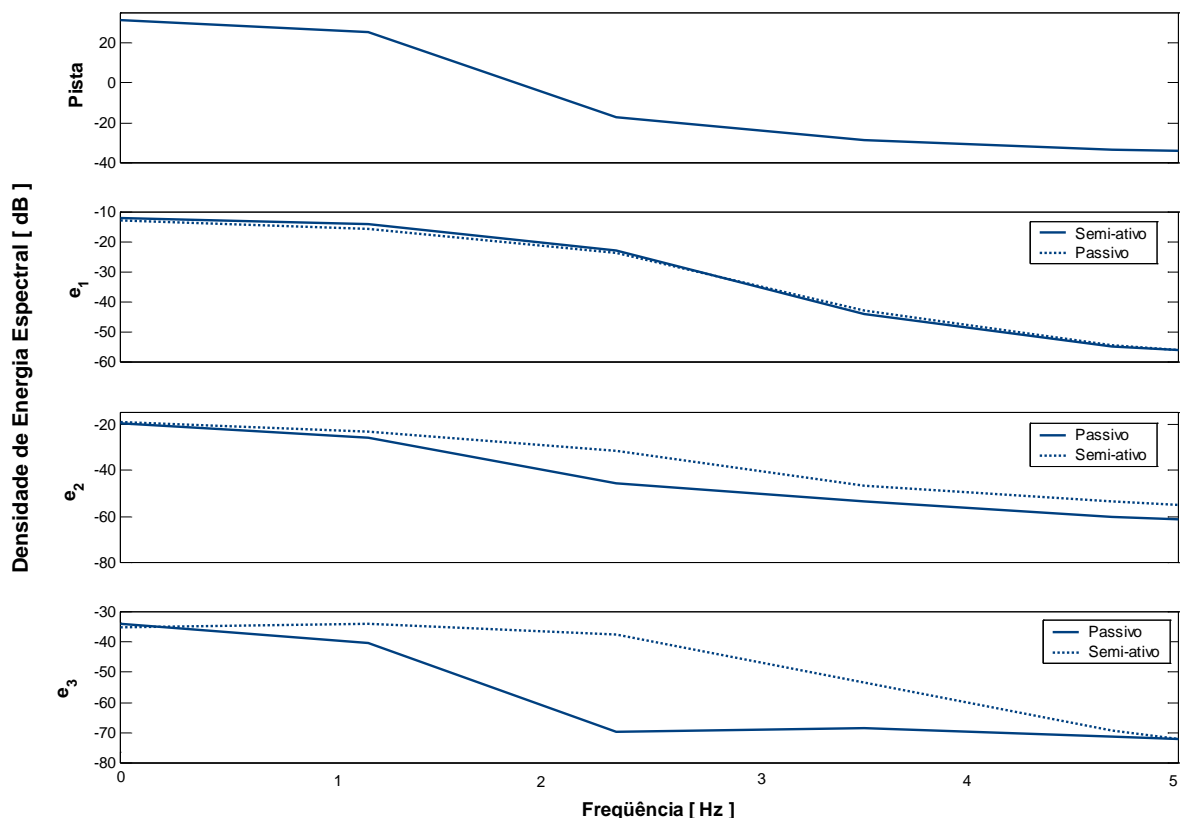


Figura 13. Caracterização das suspensões semi-ativa e passiva no domínio da frequência.

Observa-se na Figura 13 que o sistema semi-ativo apresentou melhores resposta em frequência, especialmente em baixas frequências, quando comparado com o sistema passivo. Isto é mais evidente quando se observa os ângulos de mergulho e rolagem. Quanto ao deslocamento do centro de gravidade, que relaciona-se com a estabilidade do veículo, os dois controladores tiveram comportamentos bastante semelhantes, tanto em termos de frequência quanto na resposta temporal (Figura 10), no entanto, sob o ponto de vista de conforto aos passageiros, o sistema semi-ativo é significativamente superior.

A Tabela 4 mostra os valores RMS dos sinais mostrados na Figura 10, e estes valores confirmam o que foi acima enunciado, quanto ao comportamento dos sistemas passivo e semi-ativo.

Tabela 4. Valores RMS dos sinais analisados e a razão entre eles.

	Semi-ativo (SA)	Passivo (P)	SA / P
Zs	0,0197 [m]	0,0193 [m]	1,0228
θ	0,0080 [m]	0,0084 [m]	0,9498
φ	0,0021 [m]	0,0024 [m]	0,9015

5. Conclusões

Este trabalho investigou a possibilidade do uso de controladores *fuzzy* no projeto de suspensão semi-ativa veicular. A proposta visa atenuar as vibrações presentes nos veículos, eliminando-se, desta forma, os inconvenientes gerados aos passageiros e motoristas.

Para as simulações numéricas, o veículo foi modelado como um sistema com 7 graus de liberdade e as não linearidades presentes no modelo justificam o uso da lógica *fuzzy*. Foi avaliado um sistema de suspensão passiva em contraposição com um sistema semi-ativo, construído a partir da lógica *fuzzy*.

As excitações ou a pista, propriamente dita, foi modelada por três sinais: um ruído branco para asfalto e dois sinais senoidais, um simulando uma subida de estrada e outro uma lombada.

As respostas dos sistemas passivo e semi-ativo foram comparadas no tempo e na frequência e os melhores resultados foram obtidos com o sistema de suspensão semi-ativa. A suspensão semi-ativa introduz maior conforto ao veículo, demanda menores esforços de controle, é robusta diante de perturbações inesperadas (lombada), pode ser construída a partir de um conhecimento intuitivo do sistema (não demanda a construção de modelos) e apresenta boas perspectivas de implementação no mundo real.

Em trabalho futuros pretende-se utilizar os algoritmos genéticos para otimização dos parâmetros do controlador *fuzzy*, visando uma diminuição do número de regras e a implementação do controle em tempo real em microcontroladores.

6. Referências Bibliográficas

- Lee, C.C. , 1990, “Fuzzy logic in control system: fuzzy logic controller – part i & ii”, IEEE Transactions on Systems, Man and Cybernetics, Vo. 20, No.02, pp. 404-418.
- Li, Q., Yoshimura, T. and Hino, J. , 1998, “Active suspension with preview of large-sized buses using fuzzy reasoning”. International Journal of Vehicle Design, Vol.29, No. 02, pp. 187-198.
- Mamdani, E.H. , 1974, “Applications of fuzzy algorithm for simple dynamic plant”, Proceedings IEE 121(12), pp. 1585-1588.
- Mendel, J.M., 1995, “Fuzzy Logic Systems for Engineering: A Tutorial”, Proceedings of the IEEE, vol. 83, No. 3, pp. 345-377.
- Teixeira, R.L., 2001, “Uma metodologia de projeto de controladores híbridos inteligentes com aplicações no controle ativo de vibrações mecânicas, Dissertação de Mestrado, Universidade Federal de Uberlândia – MG. <http://www.mecanica.ufu.br/>
- Wu, Y. and Xu, B. , 1999, “Study on the Damping Fuzzy Control of Semi-active Suspension”. Proceeding in IEEE 0-7803-5296-3/99, pp. 66-69.
- Yoshimura, T. , 1998, “A semi-active suspension of passenger cars using fuzzy reasoning and the field testing”. International Journal of Vehicle Design, Vol.29, No. 02, pp. 150-166.
- Zadeh, L. A., 1965, “Fuzzy Sets, Information and Control”, Vol. 8, pp. 338-353.

SEMI-ACTIVE SUSPENSION FOR PASSENGER VEHICLES USING FUZZY CONTROLLERS

Rafael Luís Teixeira

Universidade Federal de Uberlândia - Faculdade de Engenharia Mecânica - Campus Santa Mônica 38400-089, Uberlândia-MG
rafael@mecanica.ufu.br

José Francisco Ribeiro

Universidade Federal de Uberlândia - Faculdade de Engenharia Mecânica - Campus Santa Mônica 38400-089, Uberlândia-MG
jrribeiro@mecanica.ufu.br

Abstract. *This work presents the project of control of semi-activate suspension system of a passenger vehicle. Well-grounded on the fuzzy logic control theory, the proposed controller has its parameters obtained through the heuristic knowledge about the multi-body system. The vehicle model is developed considering four suspensions coupled by a rigid body, configuring a non-linear system with 7 degrees of freedom, subject to irregular excitation from road surface. From this model the simulation are done. The proposed controller are evaluated on the time and frequency domain, and during the numerical simulations, the efficiency, robustness and stability of the semi-active controller are verified by comparing with the non-linear passive suspension.*

Key words. *Fuzzy controllers; Semi-active suspension.*

CÁLCULO E ANÁLISE DO JACOBIANO DE UM MANIPULADOR PARALELO DE 3 GRAUS DE LIBERDADE BASEADO NA TEORIA DOS HELICÓIDES

Antonio Carlos Valdiero

Universidade Federal de Santa Catarina. Caixa Postal 476 – 88049 470 – Florianópolis - SC
valdiero@emc.ufsc.br

Alexandre Campos

Universidade Federal de Santa Catarina. Caixa Postal 476 – 88049 470 – Florianópolis - SC
alexandre@emc.ufsc.br

Raul Guenther

Universidade Federal de Santa Catarina. Caixa Postal 476 – 88049 470 – Florianópolis - SC
guenther@emc.ufsc.br

Daniel Martins

Universidade Federal de Santa Catarina. Caixa Postal 476 – 88049 470 – Florianópolis - SC
dmartins@lcmi.ufsc.br

Resumo. Este artigo apresenta o cálculo e análise do Jacobiano de um manipulador paralelo de três graus de liberdade baseado na teoria dos helicóides. Manipuladores paralelos são mecanismos robóticos contendo duas ou mais cadeias cinemáticas seriais conectando o efetuador final a base. Possuem alta rigidez, baixa inércia, alta relação carregamento/peso e são facilmente adaptáveis ao controle de posição e força. Uma componente crítica para maioria dos algoritmos de coordenação empregados nos manipuladores robóticos é a matriz Jacobiana. Esta matriz transforma as velocidades das juntas para o estado de velocidades do efetuador final. A matriz Jacobiana é usada para a identificação das singularidades do manipulador e é necessária para relacionar o estado de forças do efetuador final e as forças generalizadas nas juntas. Por outro lado, a teoria helicoidal induz uma matemática de par de vetores que pode ter várias interpretações geométricas e físicas. Neste trabalho, um heligiro e uma heliforça representam respectivamente a cinemática e a estática de um corpo rígido e podem ser utilizados na análise cinemática, estática e de rigidez do manipulador paralelo de 3 graus de liberdade (g.d.l.). O método proposto baseia-se no conceito de reciprocidade entre heliforças e heligiros. Este método possui um embasamento físico que não é encontrado nos métodos geométricos tradicionais. Além disso, permite uma gama de escolhas fisicamente viáveis para montagem das matrizes constituintes do Jacobiano. Estes resultados serão aplicados na construção do protótipo de um manipulador paralelo com atuadores pneumáticos.

Palavras chave: manipulador paralelo, matriz Jacobiana, teoria helicoidal.

1. Introdução

Os manipuladores paralelos tem sido objeto de muitas pesquisas nos últimos anos, tanto do ponto de vista teórico como de aplicações práticas (Merlet, 1996). Por serem mecanismos de cadeia cinemática fechada, apresentam vantagens em relação aos manipuladores seriais. Como exemplo destas vantagens, pode-se citar que a posição do efetuador final é menos sensível a erros e que o carregamento aplicado é distribuído entre os vários membros garantindo maior rigidez e mínima deformação dos elos. As desvantagens são o volume de trabalho reduzido e as dificuldades de projeto devido a arquitetura mais complexa das cadeias cinemáticas fechadas. Uma destas dificuldades é o cálculo da matriz Jacobiana. A matriz Jacobiana relaciona as velocidades das juntas atuadas com o estado de velocidades do efetuador final e permite a análise de configurações singulares. Também pode ser utilizada para obtenção das forças generalizadas nas juntas a partir do estado de forças do efetuador final e é importante nos cálculos de algoritmos de controle. Os trabalhos de Hunt (1987) e Tsai (1999) mostram que o cálculo e a análise da matriz Jacobiana de manipuladores seriais podem ser simplificados através da aplicação da teoria de helicóides. Lee et al (1999) apresenta um estudo da estabilidade de manipuladores paralelos planares através do Jacobiano baseado em helicóide.

Diante deste contexto, o presente trabalho visa aplicar o método dos helicóides recíprocos na determinação da matriz Jacobiana de um manipulador paralelo espacial de três graus de liberdade, a ser construído no Laboratório de Robótica da Universidade Federal de Santa Catarina. Os conceitos básicos e o procedimento geral do método dos helicóides recíprocos são apresentados na seção de fundamentos teóricos. Em seguida, descreve-se a concepção do manipulador paralelo e o cálculo de sua matriz Jacobiana baseada em helicóides. Apresenta-se também uma análise qualitativa dessa matriz, o que permite encontrar as várias configurações singulares.

2. Fundamentos teóricos

Nesta seção, pretende-se fornecer uma visão geral do método dos helicóides recíprocos para determinação da matriz Jacobiana de manipuladores paralelos.

Primeiramente, apresenta-se os conceitos básicos da teoria helicoidal aplicados na cinemática e estática de corpos rígidos. Em seguida, deduz-se a condição de reciprocidade a partir do princípio do trabalho virtual. Por fim, mostra-se uma metodologia geral para o cálculo do Jacobiano baseado em helicóides para manipuladores paralelos.

2.1. Conceitos básicos de helicóides aplicados na cinemática e estática

No estudo de manipuladores, trata-se com a localização de vários corpos rígidos no espaço. Estes corpos rígidos são, por exemplo, os elos do manipulador, as ferramentas, as peças de trabalho. O movimento tridimensional geral de cada um destes corpos rígidos pode ser representado por uma rotação em torno de uma reta no espaço e uma translação ao longo da mesma. O resultado de tal rotação mais translação é um movimento helicoidal (Selig, 1992). O eixo em torno do qual ele ocorre é chamado de eixo do helicóide. Na literatura inglesa, utiliza-se o termo "screw" para denominar helicóide.

Geometricamente, um helicóide pode ser definido como uma entidade composta por uma linha e um número λ , chamado passo, que possui dimensão de comprimento. Convenientemente representado em um sistema de coordenadas, chamadas coordenadas helicoidais, um helicóide unitário é definido por um par de vetores na Eq. (1) (Tsai, 1999).

$$\hat{\$} = \begin{bmatrix} \vec{s} \\ \vec{s}_0 \times \vec{s} + \lambda \vec{s} \end{bmatrix} = [S_1 \ S_2 \ S_3 \ S_4 \ S_5 \ S_6]^T \tag{1}$$

onde \vec{s} é um versor ao longo da direção do eixo helicóide e \vec{s}_0 é o vetor posição de qualquer ponto sobre o eixo helicóide. As componentes S_1, S_2, S_3, S_4, S_5 e S_6 são as coordenadas helicoidais.

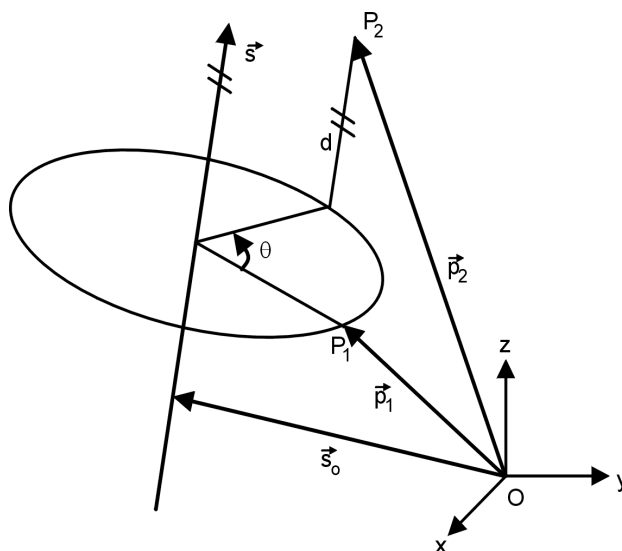


Figura 1. Movimento helicoidal de um ponto do corpo rígido de uma posição P_1 para P_2 .

Na cinemática, veja a Fig. (1), o teorema de Mozzi (Hunt, 1978) estabelece que o deslocamento infinitesimal de um corpo rígido pode ser expresso como uma rotação θ em torno de um eixo e uma translação d ao longo do mesmo eixo. Este movimento combinado é chamado deslocamento helicoidal ou heligiro, equivalente ao termo "twist" da literatura inglesa. Um heligiro pode ser representado como:

$$\$ = \dot{q} \hat{\$} = \dot{q} \cdot \begin{bmatrix} \vec{s} \\ \vec{s}_0 \times \vec{s} + \lambda \vec{s} \end{bmatrix} = \begin{bmatrix} \vec{\omega} \\ \vec{v} \end{bmatrix} \tag{2}$$

onde \dot{q} é a intensidade do heligiro, as três primeiras coordenadas representam a velocidade angular $\vec{\omega}$ e as três últimas a velocidade translacional \vec{v} . O passo λ é um escalar que representa a taxa de translação por rotação, $\lambda = d / \theta$, para deslocamentos infinitesimais.

As seis coordenadas do heligiro da Eq.(2) especificam completamente a cinemática instantânea de primeira ordem de um corpo rígido (Hunt, 1978). Para um corpo rígido com n graus de liberdade, $n \leq 6$, o movimento infinitesimal pode ser considerado como o movimento resultante de n heligiros, formando um sistema helicoidal de ordem n ou simplesmente chamado de sistema n .

Similarmente na estática, o teorema de Poinot (Hunt, 1978) estabelece que qualquer sistema de forças e binários atuando sobre um corpo rígido pode ser reduzido a uma força resultante \vec{f} ao longo de um eixo e um binário \vec{c} sobre o mesmo eixo. Quando escrita em coordenadas helicoidais, esta combinação de força e binário é chamada de heliforça, ou de "wrench" na literatura inglesa. Uma heliforça é definida como:

$$\$_r = \rho \hat{\$_r} = \rho \cdot \begin{bmatrix} \vec{s}_r \\ \vec{s}_{r0} \times \vec{s}_r + \lambda_r \vec{s}_r \end{bmatrix} = \begin{bmatrix} \vec{f} \\ \vec{c} \end{bmatrix} \quad (3)$$

onde ρ é a intensidade da heliforça. \vec{s}_r é um versor ao longo da direção única da força resultante, também chamada de eixo da heliforça, e \vec{s}_{r0} é o vetor posição de qualquer ponto sobre o eixo heliforça. O passo de uma heliforça, λ_r , é definido como sendo a relação entre o binário e a força ($\lambda_r = c/f$).

2.2. Trabalho virtual e o conceito de reciprocidade

Pelo princípio do trabalho virtual, considere um corpo rígido onde \vec{s}_0 e \vec{s}_{r0} são respectivamente os vetores posição de um ponto sobre o eixo do heligiros e de um ponto sobre o eixo da heliforça, ambos em relação a origem de um sistema de referência, o trabalho realizado por uma heliforça num movimento instantâneo representado por um heligiros é:

$$\delta W = \vec{f} \cdot \vec{v}_0 + \vec{c}_0 \cdot \vec{\omega} = \$_r^T \$_r \quad (4)$$

Onde \vec{v}_0 é a velocidade linear de um ponto do corpo rígido que é instantaneamente coincidente com a origem do sistema de referência e \vec{c}_0 é o conjugado resultante calculado na origem do mesmo sistema de referência. Os helicóides são definidos conforme as Eq. (5) e (6).

$$\$_r = \begin{bmatrix} \vec{\omega} \\ \vec{v}_0 \end{bmatrix} = \begin{bmatrix} \vec{\omega} \\ \vec{s}_0 \times \vec{\omega} + \vec{v} \end{bmatrix} = \dot{q} \cdot \begin{bmatrix} \vec{s} \\ \vec{s}_0 \times \vec{s} + \lambda \vec{s} \end{bmatrix} \quad (5)$$

$$\$_r = \begin{bmatrix} \vec{f} \\ \vec{c}_0 \end{bmatrix} = \begin{bmatrix} \vec{f} \\ \vec{s}_{r0} \times \vec{f} + \vec{c} \end{bmatrix} = \rho \cdot \begin{bmatrix} \vec{s}_r \\ \vec{s}_{r0} \times \vec{s}_r + \lambda_r \vec{s}_r \end{bmatrix} \quad (6)$$

O transposto de um helicóide é convenientemente definido por Tsai (1999) como:

$$\$_r^T = [S_4 \quad S_5 \quad S_6 \quad S_1 \quad S_2 \quad S_3] \quad (7)$$

Tal que o produto ortogonal de dois helicóides é dado por

$$\$_r^T \$_r = S_1 S_{r4} + S_2 S_{r5} + S_3 S_{r6} + S_4 S_{r1} + S_5 S_{r2} + S_6 S_{r3} \quad (8)$$

Considerando-se a Eq. (4), se uma heliforça atua sobre um corpo rígido de tal forma que o trabalho produzido durante um heligiros infinitesimal é nulo, então os dois helicóides são ditos recíprocos. A condição de reciprocidade pode ser deduzida como apresentado por Tsai (1999):

$$\delta W = \dot{q} \rho \left[\vec{s} \cdot (\vec{s}_{r0} \times \vec{s}_r + \lambda_r \vec{s}_r) + \vec{s}_r \cdot (\vec{s}_0 \times \vec{s} + \lambda \vec{s}) \right] = 0$$

$$(\lambda + \lambda_r) \vec{s} \cdot \vec{s}_r + \vec{s}_r \cdot (\vec{s}_0 \times \vec{s}) + \vec{s} \cdot (\vec{s}_{r0} \times \vec{s}_r) = 0 \quad (9)$$

Observando-se a Fig. (2), onde $a = |\vec{a}|$ é a perpendicular comum entre os eixos helicoidais $\$_r$ e $\$_r$, e α é o ângulo entre estes eixos medido a partir do heligiros, $\$_r$, de acordo com a regra da mão direita. A Eq. (9) pode ser rescrita como:

$$(\lambda + \lambda_r) \vec{s} \cdot \vec{s}_r - \vec{a} \cdot (\vec{s} \times \vec{s}_r) = 0$$

$$(\lambda + \lambda_r) \cos \alpha - a \sin \alpha = 0 \quad (10)$$

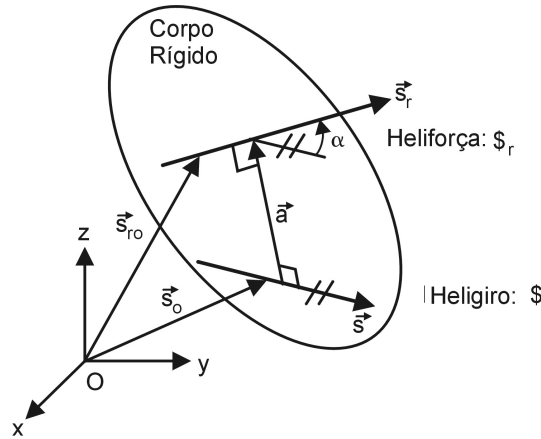


Figura 2. Corpo rígido sobre a atuação de uma heliforça durante um heligiro infinitesimal.

A Eq. (10) mostra que mediante a existência de uma heliforça sobre um corpo rígido sujeito a um heligiro, a condição de reciprocidade independe das intensidades destes helicóides, mas depende simetricamente dos passos, λ e λ_r , e da restrição geométrica definida pela distância e pelo ângulo entre eles. A Tabela (1) apresenta possíveis casos de reciprocidade entre dois helicóides. Note que quando ambos os passos dos helicóides são infinitos, o que é equivalente a um binário puro atuando sobre um corpo rígido sujeito apenas a um movimento de translação, a condição de reciprocidade é satisfeita para quaisquer distância a e ângulo α entre os helicóides.

Tabela 1. Relações entre o heligiro e a heliforça da Fig. (2) que satisfazem a condição de reciprocidade da Eq. (10).

Ângulo entre os helicóides	Distância entre os helicóides	
	(os helicóides interseccionam-se) $a = 0$	(helicóides não se interseccionam) $a \neq 0$
$\alpha = (2k+1) \cdot 90^\circ$, para $k = 0, 1$.	$\forall \lambda$ e $\forall \lambda_r$	$(\lambda = \infty$ e $\forall \lambda_r)$ ou $(\forall \lambda$ e $\lambda_r = \infty)$
$\alpha \neq k \cdot 180^\circ$, para $k = 0, 1, 2$.	$\lambda_r = -\lambda$	$\lambda_r = a \tan(\alpha) - \lambda$
$\alpha \neq (2k+1) \cdot 90^\circ$, para $k = 0, 1$.		
$\alpha = k \cdot 180^\circ$, para $k = 0, 1, 2$.	$\lambda_r = -\lambda$	$\lambda_r = -\lambda$

2.3. Cálculo do Jacobiano de manipuladores paralelos baseado na teoria dos helicóides

Como os manipuladores paralelos possuem cadeia cinemática fechada, nem todas juntas podem ser controladas independentemente. Algumas são acionadas por atuadores e as restantes são passivas. Assim, as restrições cinemáticas impostas pelos membros podem ser escritas na forma:

$$f_n(\vec{x}, \vec{q}) = 0 \tag{11}$$

onde o vetor \vec{x} descreve a localização da plataforma móvel, o vetor \vec{q} representa as variáveis das juntas atuadas e f_n é uma função n -dimensional implícita destes vetores. Diferenciando a Eq. (11) em relação ao tempo, obtém-se a razão entre as taxas das juntas atuadas, $\dot{\vec{q}}$, e a velocidade do efetuador final, $\dot{\vec{x}}$, dada por:

$$J_x \dot{\vec{x}} = J_q \dot{\vec{q}} \tag{12}$$

onde $J_x = \partial f_n / \partial \vec{x}$ e $J_q = -\partial f_n / \partial \vec{q}$. Assim, a matriz Jacobiana J pode ser escrita como:

$$\dot{\vec{q}} = J_q^{-1} J_x \dot{\vec{x}} = J \dot{\vec{x}} \tag{13}$$

Cabe ressaltar que esta matriz é a inversa da obtida para manipuladores seriais. A matriz J_q é quadrada de dimensão igual ao número de juntas atuadas e pode ser obtida através do método dos helicóides recíprocos apresentado na seqüência.

Considere um manipulador paralelo composto de m membros. Cada membro é definido como uma cadeia cinemática serial, constituída de vários elos ligados por juntas. Como sempre é possível substituir uma junta de grau de liberdade maior do que um pela combinação adequada de juntas rotativas e prismáticas, cada membro pode ser representado por uma cadeia cinemática aberta conectando a plataforma móvel à base fixa por l juntas de um grau de liberdade, conforme mostrado na Fig. (3).

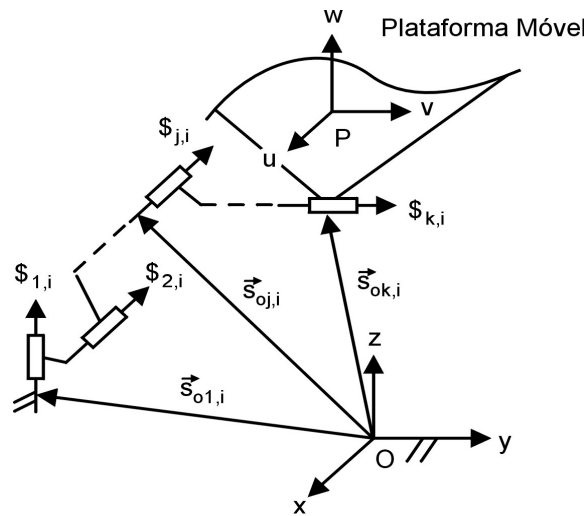


Figura 3. Desenho esquemático do i -ésimo membro de um manipulador paralelo (Tsai, 1999).

Assim, o heligiro instantâneo da plataforma móvel pode ser escrito como uma combinação linear de l heligiros instantâneos:

$$\dot{x}_p = \sum_{j=1}^l \dot{q}_{j,i} \hat{\$}_{j,i} = \dot{x} \quad \text{para } i = 1, 2, \dots, m. \tag{14}$$

onde $\dot{q}_{j,i}$ é a intensidade e $\hat{\$}_{j,i}$ é o helicóide unitário associado a j -ésima junta do i -ésimo membro. O método dos helicóides recíprocos baseia-se na eliminação das juntas não atuadas contidas na Eq. (14). Em geral, as juntas não atuadas são em maior número que as atuadas. Para isto, assume-se que os “ g ” primeiros termos representam as juntas atuadas em cada membro e identifica-se um conjunto de helicóides recíprocos a todos os helicóides das juntas não atuadas do i -ésimo membro. Esta identificação é feita conforme Hunt (1978). Em seguida, escolhe-se “ g ” helicóides unitários deste conjunto, dados por $\hat{\$}_{rj,i}$, para $j = 1, 2, \dots, g$, e representados na Eq. (14). Onde cada um dos quais é escolhido de forma a não ser recíproco a j -ésima junta atuada.

Efetuando o produto ortogonal, definido na Eq. (8), de ambos os lados da Eq. (14) para cada um dos helicóides recíprocos e aplicando-se a condição de reciprocidade, tem-se:

$$\begin{bmatrix} \hat{\$}_{r1,1}^T \\ \vdots \\ \hat{\$}_{rg,1}^T \\ \hat{\$}_{r1,2}^T \\ \vdots \\ \hat{\$}_{rg,2}^T \\ \vdots \\ \hat{\$}_{r1,m}^T \\ \vdots \\ \hat{\$}_{rg,m}^T \end{bmatrix} \dot{x} = \begin{bmatrix} J_{q,1} & 0 & \dots & 0 \\ 0 & J_{q,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & J_{q,m} \end{bmatrix} \begin{bmatrix} \dot{q}_{1,1} \\ \vdots \\ \dot{q}_{g,1} \\ \dot{q}_{1,2} \\ \vdots \\ \dot{q}_{g,2} \\ \vdots \\ \dot{q}_{1,g} \\ \vdots \\ \dot{q}_{g,g} \end{bmatrix} \tag{15}$$

onde $J_{q,i}$, para $i = 1, 2, \dots, m$, representa um bloco de elementos de dimensão $g \times g$, dado por:

$$J_{q,i} = \begin{bmatrix} \hat{\$}_{r1,i}^T \cdot \hat{\$}_{1,i} & \hat{\$}_{r1,i}^T \cdot \hat{\$}_{2,i} & \dots & \hat{\$}_{r1,i}^T \cdot \hat{\$}_{g,i} \\ \hat{\$}_{r2,i}^T \cdot \hat{\$}_{1,i} & \hat{\$}_{r2,i}^T \cdot \hat{\$}_{2,i} & \dots & \hat{\$}_{r2,i}^T \cdot \hat{\$}_{g,i} \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\$}_{rg,i}^T \cdot \hat{\$}_{1,i} & \hat{\$}_{rg,i}^T \cdot \hat{\$}_{2,i} & \dots & \hat{\$}_{rg,i}^T \cdot \hat{\$}_{g,i} \end{bmatrix} \quad \text{para } i = 1, 2, \dots, m.$$

Note que a Eq. (15) tem a forma da Eq. (12) e fornece um método para o cálculo das matrizes Jacobianas do manipulador paralelo. Para o caso mais comum em que cada membro possui apenas uma junta atuada ($g = 1$), a matriz J_q da Eq. (12) torna-se diagonal.

3. Descrição do manipulador paralelo de 3 graus de liberdade

O manipulador paralelo a ser construído ao Laboratório de Robótica da Universidade Federal de Santa Catarina consiste de uma plataforma móvel (1) conectada a uma base fixa (2) por três membros idênticos, conforme o desenho em perspectiva mostrado na Fig. (4). Cada membro é composto por um atuador pneumático linear (3) engastado na base fixa e ligado a um braço superior (4) através de uma junta rotativa. Cada braço superior é um paralelograma planar de quatro barras articuladas e é conectado na plataforma móvel através de uma junta rotativa. O braço superior é equivalente a um elo com duas juntas universais, uma em cada extremidade, ligadas ao atuador linear e à plataforma móvel.

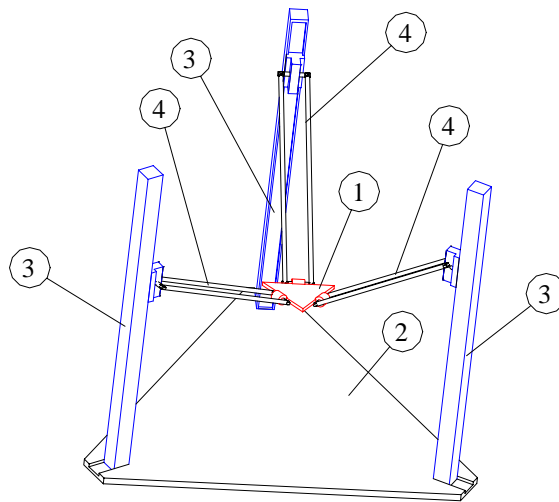


Figura 4. Desenho do manipulador paralelo de 3 graus de liberdade (g.d.l.) em perspectiva.

Este manipulador é dito simétrico, pois ele satisfaz as seguintes condições (Tsai, 1999): o número de membros é igual ao número de graus de liberdade da plataforma móvel, o tipo e o número de juntas em todos os membros são arranjadas de forma idêntica, o número e a localização das juntas atuadas em todos os membros é a mesma.

4. Cálculo do Jacobiano baseado em helicóide do manipulador paralelo de 3 graus de liberdade

Para a estrutura cinemática do manipulador paralelo mostrado na Fig. (4), a matriz Jacobiana é deduzida pela aplicação do conceito de helicóides recíprocos apresentado anteriormente. A Fig. (5) mostra a cadeia cinemática equivalente de um membro PUU (seqüência de juntas Prismática, Universal e Universal com dois elos), onde a junta prismática atuada é representada pelo helicóide $\$1$, a primeira junta universal pela interseção dos helicóides $\$2$ e $\$3$, e a segunda junta universal pela interseção dos helicóides $\$4$ e $\$5$.

Os sistemas de coordenadas (x,y,z) e (u,v,w) são fixados respectivamente na base fixa e na plataforma móvel, conforme representado na Fig. (5). Define-se um sistema de referência instantâneo (x',y',z') , com sua origem localizada no ponto P e os eixos x' , y' e z' paralelos aos eixos x , y e z do sistema fixo. Todos os helicóides das juntas são expressos em relação a este sistema de referência instantâneo. Os vetores \vec{a}_i , \vec{b}_i , \vec{d}_i e \vec{e}_i representam respectivamente o tamanho e a direção dos segmentos $\overline{OA_i}$, $\overline{A_iC_i}$, $\overline{PB_i}$ e $\overline{C_iB_i}$ do membro i .

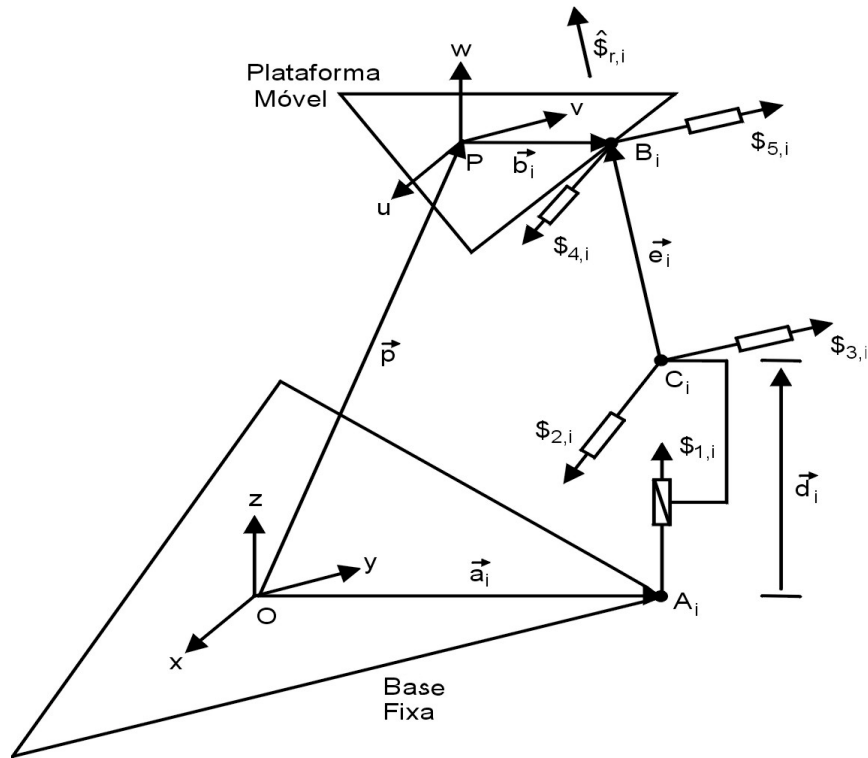


Figura 5. Estrutura cinemática equivalente a um membro i do manipulador paralelo.

Há cinco helicóides de juntas associadas com cada membro e mostrados na Fig. (5). A primeira junta (prismática) é a única junta atuada, as outras quatro são passivas. O helicóide da junta prismática atuada tem passo infinito ($\lambda = \infty$). Considerando que $\vec{s}_{j,i}$ é um versor na direção do eixo da j -ésima junta do i -ésimo membro, tem-se que os cinco helicóides associados às juntas do membro i podem ser escritos como:

$$\dot{\$}_{1,i} = \dot{\theta}_{1,i} \hat{\$}_{1,i} = \dot{\theta}_{1,i} \cdot \lambda \cdot \frac{\hat{\$}_{1,i}}{\lambda} = \dot{\theta}_{1,i} \cdot \frac{\dot{d}_i}{\dot{\theta}_{1,i}} \cdot \begin{bmatrix} \vec{s}_{1,i} \\ \frac{\vec{s}_{01,i} \times \vec{s}_{1,i}}{\lambda} + \vec{s}_{1,i} \end{bmatrix} = \dot{d}_i \begin{bmatrix} 0 \\ \vec{s}_{1,i} \end{bmatrix} \quad (16)$$

$$\dot{\$}_{2,i} = \dot{\theta}_{2,i} \hat{\$}_{2,i} = \dot{\theta}_{2,i} \begin{bmatrix} \vec{s}_{2,i} \\ \vec{s}_{02,i} \times \vec{s}_{2,i} + \lambda \vec{s}_{2,i} \end{bmatrix} = \dot{\theta}_{2,i} \begin{bmatrix} \vec{s}_{2,i} \\ \vec{s}_{02,i} \times \vec{s}_{2,i} \end{bmatrix} = \dot{\theta}_{2,i} \begin{bmatrix} \vec{s}_{2,i} \\ (\vec{b}_i - \vec{e}_i) \times \vec{s}_{2,i} \end{bmatrix} \quad (17)$$

$$\dot{\$}_{3,i} = \dot{\theta}_{3,i} \hat{\$}_{3,i} = \dot{\theta}_{3,i} \begin{bmatrix} \vec{s}_{3,i} \\ \vec{s}_{03,i} \times \vec{s}_{3,i} + \lambda \vec{s}_{3,i} \end{bmatrix} = \dot{\theta}_{3,i} \begin{bmatrix} \vec{s}_{3,i} \\ \vec{s}_{03,i} \times \vec{s}_{3,i} \end{bmatrix} = \dot{\theta}_{3,i} \begin{bmatrix} \vec{s}_{3,i} \\ (\vec{b}_i - \vec{e}_i) \times \vec{s}_{3,i} \end{bmatrix} \quad (18)$$

$$\dot{\$}_{4,i} = \dot{\theta}_{4,i} \hat{\$}_{4,i} = \dot{\theta}_{4,i} \begin{bmatrix} \vec{s}_{4,i} \\ \vec{s}_{04,i} \times \vec{s}_{4,i} + \lambda \vec{s}_{4,i} \end{bmatrix} = \dot{\theta}_{4,i} \begin{bmatrix} \vec{s}_{4,i} \\ \vec{s}_{04,i} \times \vec{s}_{4,i} \end{bmatrix} = \dot{\theta}_{4,i} \begin{bmatrix} \vec{s}_{4,i} \\ \vec{b}_i \times \vec{s}_{4,i} \end{bmatrix} \quad (19)$$

$$\dot{\$}_{5,i} = \dot{\theta}_{5,i} \hat{\$}_{5,i} = \dot{\theta}_{5,i} \begin{bmatrix} \vec{s}_{5,i} \\ \vec{s}_{05,i} \times \vec{s}_{5,i} + \lambda \vec{s}_{5,i} \end{bmatrix} = \dot{\theta}_{5,i} \begin{bmatrix} \vec{s}_{5,i} \\ \vec{s}_{05,i} \times \vec{s}_{5,i} \end{bmatrix} = \dot{\theta}_{5,i} \begin{bmatrix} \vec{s}_{5,i} \\ \vec{b}_i \times \vec{s}_{5,i} \end{bmatrix} \quad (20)$$

Considerando cada membro i como uma cadeia cinemática aberta, pela aplicação da Eq. (14), pode-se expressar o heligiro instantâneo da plataforma móvel em termos dos helicóides das juntas como:

$$\dot{\$}_p = \dot{d}_i \hat{\$}_{1,i} + \dot{\theta}_{2,i} \hat{\$}_{2,i} + \dot{\theta}_{3,i} \hat{\$}_{3,i} + \dot{\theta}_{4,i} \hat{\$}_{4,i} + \dot{\theta}_{5,i} \hat{\$}_{5,i} \quad (21)$$

Devido aos fatos dos eixos de todas as quatro juntas não atuadas interseccionarem a linha passando pelos pontos B_i e C_i , cuja direção é dada pelo versor $\vec{s}_{6,i}$, existe em cada membro do manipulador um sistema 2 de helicóides que são

recíprocos a todos os quatro helicóides de juntas não atuadas. Este conjunto é formado por todos os helicóides de qualquer passo ao longo da reta $\overline{C_i B_i}$, obtidos conforme Hunt (1978). Assim, pode-se escolher um destes que não seja recíproco ao helicóide da junta atuada, identificado como uma heliforça recíproca unitária de passo nulo ($\lambda_r = 0$):

$$\hat{\$}_{r,i} = \begin{bmatrix} \vec{s}_{6,i} \\ \vec{b}_i \times \vec{s}_{6,i} \end{bmatrix} \quad (22)$$

Tomando-se o produto ortogonal de ambos os lados da Eq. (21) pelo transposto do helicóide recíproco da equação Eq. (22), obtém-se:

$$\hat{\$}_{r,i}^T \cdot \$_p = \hat{\$}_{r,i}^T \cdot \hat{\$}_{1,i} \dot{d}_i \quad (23)$$

$$\hat{\$}_{r,i}^T \cdot \hat{\$}_{2,i} = \hat{\$}_{r,i}^T \cdot \hat{\$}_{3,i} = \hat{\$}_{r,i}^T \cdot \hat{\$}_{4,i} = \hat{\$}_{r,i}^T \cdot \hat{\$}_{5,i} = 0 \quad (24)$$

Rescrevendo-se a Eq. (23) para cada membro, obtém-se uma equação matricial na forma da Eq. (12), onde:

$$J_x = \begin{bmatrix} \hat{\$}_{r,1}^T \\ \hat{\$}_{r,2}^T \\ \hat{\$}_{r,3}^T \end{bmatrix} = \begin{bmatrix} (\vec{b}_1 \times \vec{s}_{6,1})^T & \vec{s}_{6,1}^T \\ (\vec{b}_2 \times \vec{s}_{6,2})^T & \vec{s}_{6,2}^T \\ (\vec{b}_3 \times \vec{s}_{6,3})^T & \vec{s}_{6,3}^T \end{bmatrix} \quad (25)$$

$$J_q = \begin{bmatrix} \hat{\$}_{r,1}^T \cdot \hat{\$}_{1,1} & 0 & 0 \\ 0 & \hat{\$}_{r,2}^T \cdot \hat{\$}_{1,2} & 0 \\ 0 & 0 & \hat{\$}_{r,1}^T \cdot \hat{\$}_{1,3} \end{bmatrix} \quad (26)$$

$$\dot{\vec{x}} = [\omega_x \ \omega_y \ \omega_z \ v_{px} \ v_{py} \ v_{pz}]^T \quad (27)$$

$$\dot{\vec{q}} = [\dot{d}_1 \ \dot{d}_2 \ \dot{d}_3]^T \quad (28)$$

Assim, dadas as velocidades \dot{d}_1 , \dot{d}_2 e \dot{d}_3 das juntas prismáticas atuadas dos três membros, pode-se encontrar o estado de velocidades da plataforma móvel no ponto P, definido pelas componentes ω_x , ω_y e ω_z da velocidade angular e pelas componentes v_{px} , v_{py} e v_{pz} da velocidade linear.

5. Análise do Jacobiano do manipulador paralelo de 3 graus de liberdade

A análise das configurações singulares de manipuladores paralelos não é uma tarefa simples (Lee et all, 1999; Tsai, 1999), mas algumas destas podem ser encontradas pela inspeção das matrizes J_q e J_x . Quando o determinante da matriz J_q é nulo, ocorrem as singularidades da cinemática inversa e o manipulador paralelo perde um ou mais graus de liberdade. As singularidades da cinemática direta acontecem quando a matriz J_x perde a característica (posto completo) e, neste caso, o manipulador ganha graus de liberdade. Se os determinantes de ambas matrizes são nulos, tem-se as singularidades combinadas, as quais dependem não apenas da configuração mas também da arquitetura do manipulador.

Para calcular o Jacobiano, considera-se os seguintes vetores em cada membro i do manipulador paralelo de 3 graus de liberdade:

$$\vec{n}_i = \vec{b}_i \times \vec{s}_{6,i} = [n_{ix} \ n_{iy} \ n_{iz}]^T \quad (29)$$

$$\vec{s}_{6,i} = \frac{\vec{e}_i}{|\vec{e}_i|} = [s_{6,ix} \ s_{6,iy} \ s_{6,iz}]^T \quad (30)$$

$$\vec{s}_{1,i} = [0 \ 0 \ 1]^T \quad (31)$$

Rescrevendo-se o heligiro unitário da junta atuada 1 e o transposto da heliforça recíproca unitária para cada membro em função dos vetores acima, tem-se:

$$\hat{\$}_{1,i} = [0 \ 0 \ 0 \ 0 \ 0 \ 1]^T \quad (32)$$

$$\hat{\$}_{r,i}^T = [n_{ix} \ n_{iy} \ n_{iz} \ s_{6,ix} \ s_{6,iy} \ s_{6,iz}]^T \quad (33)$$

Logo, $\hat{\$}_{r,i}^T \cdot \hat{\$}_{1,i} = s_{6,iz}$, e a matriz J_q resulta em:

$$J_q = \begin{bmatrix} s_{6,1i} & 0 & 0 \\ 0 & s_{6,2i} & 0 \\ 0 & 0 & s_{6,3i} \end{bmatrix} \quad (34)$$

Observando-se a Eq. (34), verifica-se que as singularidades da cinemática inversa do manipulador paralelo de 3 graus de liberdade ocorrem quando $s_{6,iz} = 0$ para $i = 1, 2, 3$. Ou seja, quando em um dos três membros, o seguimento $C_i B_i$ assume uma configuração paralela ao plano xy e o manipulador perde um grau de liberdade.

A matriz J_x pode ser escrita como:

$$J_x = \begin{bmatrix} n_{1x} & n_{1y} & n_{1z} & s_{6,1x} & s_{6,1y} & s_{6,1z} \\ n_{2x} & n_{2y} & n_{2z} & s_{6,2x} & s_{6,2y} & s_{6,2z} \\ n_{3x} & n_{3y} & n_{3z} & s_{6,3x} & s_{6,3y} & s_{6,3z} \end{bmatrix} \quad (35)$$

onde pode-se notar a ocorrência de uma singularidade de cinemática direta quando $n_{ix} = n_{iy} = n_{iz} = 0$, isto é, os vetores \vec{b}_i e \vec{e}_i estão alinhados em algum membro i e neste caso o manipulador ganha um grau de liberdade.

E a matriz Jacobiana J toma a seguinte forma:

$$J = \begin{bmatrix} \frac{n_{1x}}{s_{6,1z}} & \frac{n_{1y}}{s_{6,1z}} & \frac{n_{1z}}{s_{6,1z}} & \frac{s_{6,1x}}{s_{6,1z}} & \frac{s_{6,1y}}{s_{6,1z}} & 1 \\ \frac{n_{2x}}{s_{6,2z}} & \frac{n_{2y}}{s_{6,2z}} & \frac{n_{2z}}{s_{6,2z}} & \frac{s_{6,2x}}{s_{6,2z}} & \frac{s_{6,2y}}{s_{6,2z}} & 1 \\ \frac{n_{3x}}{s_{6,3z}} & \frac{n_{3y}}{s_{6,3z}} & \frac{n_{3z}}{s_{6,3z}} & \frac{s_{6,3x}}{s_{6,3z}} & \frac{s_{6,3y}}{s_{6,3z}} & 1 \end{bmatrix} \quad (36)$$

Observando a última coluna do Jacobiano, verifica-se que uma das parcelas da velocidade de cada junta é igual a componente de velocidade v_{zi} da plataforma móvel.

6. Conclusões e perspectivas futuras

O método para a construção do Jacobiano baseado na teoria helicoidal foi exposto e aplicado para um manipulador paralelo espacial de 3 graus de liberdade. A vantagem do método dos helicóides recíprocos é que ele mostra explicitamente o significado físico de anular as variáveis das juntas passivas, o que não é observado no método convencional do triângulo de velocidades (Tsai, 1999).

Para o manipulador paralelo descrito neste trabalho, observa-se a existência de uma solução única para as matrizes J_q e J_x , embora haja uma variedade de heliforças candidatas no conjunto de helicóides recíprocos aos helicóides das juntas passivas e não recíprocos ao helicóide de uma junta atuada. No caso geral, existe um leque de heliforças que resultam em diversos pares de matrizes J_q e J_x , sendo possível escolher o mais conveniente para a aplicação desejada.

Como perspectivas futuras, pretende-se construir o protótipo do manipulador paralelo espacial 3-dof no Laboratório de Robótica da Universidade Federal de Santa Catarina, utilizando-se dos resultados deste trabalho e dos obtidos no controle de um servoposicionador pneumático (Perondi e Guenther, 2000).

7. Referências

- Hunt, K.H., 1978, "Kinematic Geometric of Mechanism", Oxford University Press, Oxford, Great Britain, 465 p.
 Hunt, K.H., 1987, "Robot Kinematic – a Compact Analytic Inverse Solution for Velocities", Trans. ASME, Journal of Mechanisms, Transmissions and Automation in Design, Vol. 109, pp. 42-49.
 Lee, J., Duffy, J., Keler, M., 1999, "The Optimum Quality Index for The Stability of In-Parallel Planar Platform Devices", Journal of Mechanical Design of The ASME, Vol. 121, pp. 15-20.

- Merlet, J-P, 1996, "Parallel Manipulators: State of the Art and Perspectives", Research Report, INRIA, Sophia-Antipoles, France.
- Perondi, E., Guenther, R., 2000, "Control of a Servopneumatic Drive with Friction Compensation", 1st FPNI – PhD Symposium, Fluid Power Net International, Hamburgo, Germany.
- Selig, J.M., 1992, "Introductory Robotics", Prentice Hall, Great Britain, 152 p.
- Tsai, L.-W., 1999, "Robot Analysis: The Mechanics of Serial and Parallel Manipulators", John Wiley & Sons, New York, United States of America, 505 p.

SCREW-BASED JACOBIAN ANALYSIS OF A 3-DOF PARALLEL MANIPULATOR

Antonio Carlos Valdiero

Federal University of Santa Catarina. Postal Box 476 – 88049 470 – Florianópolis – SC – Brazil.
valdiero@emc.ufsc.br

Alexandre Campos

Federal University of Santa Catarina. Postal Box 476 – 88049 470 – Florianópolis – SC – Brazil.
alexandre@emc.ufsc.br

Raul Guenther

Federal University of Santa Catarina. Postal Box 476 – 88049 470 – Florianópolis – SC – Brazil.
valdiero@emc.ufsc.br

Daniel Martins

Federal University of Santa Catarina. Postal Box 476 – 88049 470 – Florianópolis – SC – Brazil.
dmartins@lcmi.ufsc.br

***Abstract.** This paper presents the screw-based Jacobian analysis of a 3-dof parallel manipulator. Parallel manipulators are robotic mechanisms containing two or more serial kinematics chains connecting the end-effector to the base. They possess a high stiffness, low inertia, high payload/weight ratio and easily adaptable to position and force control. A critical component for most coordination algorithms employed by robot manipulators is the Jacobian matrix. This matrix transforms the joint rates to the velocity state of end-effector, besides it shows the singularities of the manipulator and is necessary to relation the end-effector force state and the joint generalized forces. On the other hand, the screw theory is represented by a pair of vectors, these vectors can have several physical and geometrical interpretations. Twist and wrench represent the kinematics and static of a rigid body and can be used to formulate the Jacobian matrix. With the screw theory, the Jacobian matrix is greatly simplified to be used in the kinematic, static and stiffness analysis for the 3-dof parallel manipulator. The proposed method is based on the reciprocity between twists and wrenches. This method has a physical interpretation that is not found on classical geometrical methods. Moreover the constituent matrices of the Jacobian are not uniquely determined and can be conveniently chosen. These results will be applied to the construction of a parallel manipulator prototype with pneumatic actuators.*

***Keywords:** parallel manipulator, Jacobian matrix, screw theory.*

METODOLOGIA PARA A MODELAGEM DE SISTEMAS FLEXÍVEIS DE MOVIMENTAÇÃO DE MATERIAIS

Fabício Junqueira

Departamento de Engenharia Mecatrônica e Sistemas Mecânicos – Escola Politécnica da USP – Av. Prof. Mello Moraes, 2231, CEP 05508-900 – São Paulo - SP
fabri@usp.br

Paulo Eigi Miyagi

Departamento de Engenharia Mecatrônica e Sistemas Mecânicos – Escola Politécnica da USP – Av. Prof. Mello Moraes, 2231, CEP 05508-900 – São Paulo - SP
pemiyagi@usp.br

Resumo. Os sistemas de manufatura há muito vêm sendo objeto de estudos devido à constante busca de aumento da produtividade bem como de aumento da competitividade empresarial. Dentre os componentes de um processo de manufatura, o sistema de movimentação de materiais merece atenção especial pois, apesar de não incrementar o valor do produto, é responsável por manter o fluxo de materiais entre máquinas, células de manufatura, centros de custos ou mesmo entre empresas, que é imprescindível para qualquer sistema produtivo. Neste contexto, o presente trabalho propõe uma metodologia para a modelagem de sistemas flexíveis de movimentação de materiais e partes em ambiente fabril, focando-se em sistemas cuja movimentação é realizada por VATs (Veículos Autônomos de Transporte). Considerando o sistema de movimentação de materiais como sendo um sistema a eventos discretos (SED), emprega-se técnicas derivadas das Redes de Petri como o PFS (*Production Flow Schema*) e o E-MFG (*Enhanced Mark Flow Graph*) na modelagem de tais sistemas. O modelo resultante é efetivo para a análise e controle destes sistemas.

Palavras chave: *Sistemas a Eventos Discretos, Rede de Petri, Movimentação de Materiais, Simulação Discreta*

1. Introdução

Segundo Bolwijn & Kumpe (1990), as empresas têm passado por inúmeras mudanças a partir dos anos 60 pois nesta época ficou claro que a oferta de bens de consumo passou a ser superior à demanda. Para continuarem sobrevivendo e crescendo, as empresas adotaram estratégias competitivas. A primeira foi em relação aos preços, fazendo com que muitas empresas fossem reestruturadas, até mesmo mudassem de região ou país. No final dos anos 60 os consumidores passaram a se questionar sobre a qualidade dos produtos adquiridos, fazendo com que isto se tornasse um importante fator para o sucesso de uma empresa. Nos anos 70 as empresas tiveram que reduzir o intervalo de tempo entre o lançamento de novos produtos para poderem conquistar um público mais exigente em várias partes do mundo. Dos anos 90 em diante a inovação passa a ser o novo fator de sucesso, onde as empresas tinham que não apenas melhorar o produto existente, mas sim criar novos produtos.

Ainda com relação a estas estratégias, verifica-se que na prática das empresas consideradas como as mais capacitadas (competitivas), existe uma evolução temporal no emprego das estratégias, que é caracterizada pela seguinte seqüência (Muscat & Fleury, 1993): (1) custos, (2) qualidade, (3) tempo, (4) flexibilidade e (5) inovação e, conforme verificado nas empresas, a utilização das estratégias se dá cumulativamente (Muscat & Fleury, 1993). Assim, se a empresa adota a estratégia de qualidade, ela está adotando, ao mesmo tempo, a estratégia de custos. A acumulação de competências está altamente correlacionada com o sucesso da empresa em ambientes complexos e dinâmicos. Caso a mudança estratégica não ocorra cumulativamente, a empresa terá pontos fracos que a deixarão vulnerável com relação aos competidores.

De acordo ainda com o trabalho de Muscat & Fleury (1993), nota-se que enquanto as empresas estrangeiras estão entrando na era da competição por “inovação” a maioria das empresas no Brasil está ainda num processo de transição da era da “qualidade” para a era do “tempo”, sendo que muitas ainda não se encontram suficientemente capacitadas em “custos” para garantir uma maior competitividade.

Dentro deste contexto, este trabalho enfoca o problema e a necessidade de uma técnica para aprimorar o sistema de movimentação de partes / materiais em instalações industriais, que está diretamente relacionado tanto a “custos” quanto a “tempos” de produção e que é um pré-requisito fundamental para se assegurar a “flexibilidade” do sistema produtivo (isto é, competitividade da empresa), apresentando uma metodologia para a construção de modelos de sistemas de movimentação de materiais em instalações fabris, enfatizando-se a utilização de VATs¹ (os quais concedem maior flexibilidade ao processo produtivo), para que tanto possam ser utilizados em análises através de simulação como para a implementação de estratégias de controle. Para tanto, adota-se uma abordagem baseada na teoria de sistemas a eventos discretos e na aplicação de técnicas derivadas de Redes de Petri, como a metodologia PFS/E-MFG, mediante a qual são desenvolvidos modelos do objeto de controle, que servirão como suporte para a especificação do controle.

¹ VAT ∴ Veículos Autônomos de Transporte.

2. Sistema de movimentação de materiais

Com relação aos sistemas de movimentação de materiais a seguir apresentados (Martins & Laugeni, 1998), é possível classificá-los em:

- fixos – são sistemas de movimentação de materiais que possuem seu movimento limitado por trilhos ou guias e mudanças no arranjo físico das instalações implicam em mudanças nos equipamentos de transporte. Dentre os equipamentos disponíveis, pode-se citar entre outros: correias, correntes, fitas metálicas, roletes, pontes rolantes, guindastes, elevadores, etc.;
- flexíveis – são sistemas de movimentação de materiais onde mudanças no arranjo físico não implicam em mudanças nos equipamentos de transporte, pois os movimentos não sofrem limitações físicas por trilhos, por exemplo. Pode-se citar entre outros: empilhadeiras, transportadores, etc.

Atentando-se para a competitividade das empresas, seja pela produtividade, flexibilidade e / ou qualidade dos produtos e serviços, um dos sistemas mais complexos e reconhecido como de maior potencial dentre os sistemas de movimentação anteriormente expostos, são os sistemas de veículos industriais (empilhadeiras, transportadores, VATs, tratores, etc.), que podem ter seus caminhos e movimentos dinamicamente alterados e podem transportar uma grande diversidade de materiais, bastando para isto alterar a sua interface com o material (utilizando por exemplo carretas específicas).

Bodin et al. (1983) fazem uma revisão de artigos e livros publicados até aquela época sobre problemas de roteirização e/ou programação e que são também aplicáveis ao se abordar os atuais sistemas flexíveis de movimentação de materiais em ambiente industrial. Os sistemas apresentados por estes derivam basicamente de dois tipos de problemas:

- caixeiro viajante - procura-se determinar o menor custo de um ciclo de atividades do transportador, impondo que este: possui capacidade infinita e passe por todos os nós (Fig. (1)) apenas uma vez.
- carteiro chinês - procura-se determinar o menor custo impondo que o transportador passe por todos os segmentos (Fig. (2)) ao menos uma vez.

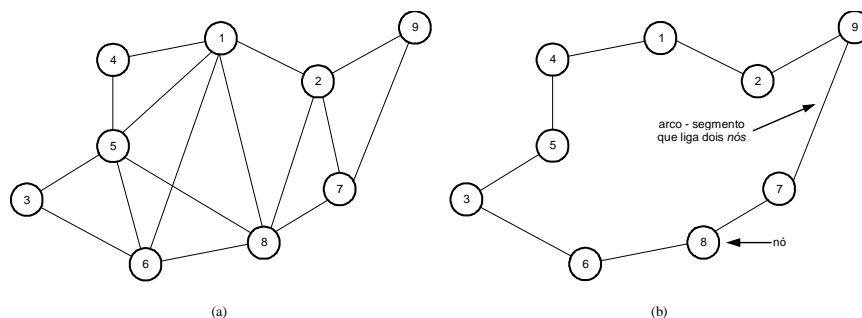


Figura (1) – Exemplo de uma possível solução (b) para um problema de caixeiro viajante (tendo como dado de entrada (a) ou um grafo densamente conexo) (Bodin et al., 1983).

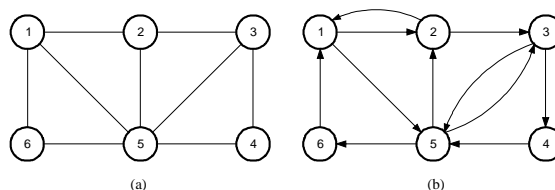


Figura (2) – Exemplo do problema de carteiro chinês: (a) grafo com as possíveis conexões entre os nós e; (b) uma possível solução (Bodin et al., 1983).

3. Ferramentas / metodologias empregadas na modelagem de sistemas a eventos discretos

Modelos baseados em sistemas a eventos discretos são intensamente usados para descrever, analisar e controlar processos em ambientes de rede de computadores e manufatura. O que caracteriza este processo é o fato de sua dinâmica ser caracterizada pela ocorrência de eventos instantâneos e estados discretos. Assim, a evolução de estados destes sistemas é baseado em regras que definem as condições para a ocorrência de eventos e o resultado destes eventos (Cassandras & Strickland, 1992).

A movimentação de materiais em sistemas produtivos, assim como outros processos concebidos pelo homem, pode ser abordado como uma sucessão de eventos e estados discretos e modelados como um SED (sistemas a eventos discretos) (Ho & Cao, 1991). Isto é, ferramentas como as Redes de Petri (Cardoso & Valette, 1997; Reisig, 1982; 1992)

e seus derivados como o PFS/MFG (Miyagi, 1996) (incluindo-se também suas variações como o PFS-R/E-MFG (Santos Filho, 1998)) podem ser consideradas para estudar o problema em questão.

Existem também outras ferramentas que podem ser utilizadas na análise destes sistemas tais como os programas específicos de simulação discreta (por exemplo: ProModel, Arena, AutoMod, etc. (Baird & Leavy, 1994; Banks, 1995)), porém estas ferramentas possuem restrições para a conversão dos modelos em especificações de estratégias de controle. O uso de programas específicos de simulação discreta envolvem a abstração de conceitos e decisões para a análise do sistema que não podem ser diretamente implementados ou interpretados em procedimentos de controle. Por outro lado, teorias como as Redes de Petri foram concebidas para a modelagem e especificação funcional de sistemas e do seu controle.

3.1. O PFS (*Production Flow Schema*)

Em sistemas de pequeno porte observa-se com facilidade o relacionamento entre as partes e o desenvolvimento de modelos destes sistema utilizando Redes de Petri é uma tarefa relativamente fácil. Porém, a medida em que os sistemas vão se tornando maiores e mais complexos, as relações entre os componentes deste deixam de ser tão visíveis. Para sanar esta dificuldade, em Miyagi (1996) é apresentada uma metodologia denominada PFS (*Production Flow Schema*), que ao invés de se desenvolver diretamente a estratégia de controle do sistema, isto é, o modelo em nível detalhado, adota-se primeiro a modelagem do sistema em nível conceitual e em seguida passa-se a detalhar gradativamente cada elemento, indo-se de um nível macro até o seu nível mais detalhado, ou seja, até se atingir um nível de detalhamento que se possa representar o sistema utilizando-se grafos específicos para controle como o MFG, SFC, GRAFCET, etc.

No PFS, apesar de ser derivado das Redes de Petri, não existe o conceito de *marcas* ou *marcações*. As inscrições em seus elementos indicam apenas quando (em relação a outros elementos) e como estas operam. As redes PFS têm como objetivo principal mostrar explicitamente os componentes que formam o sistema e quais relações existem entre cada um deles.

O PFS possui três elementos básicos (Fig. (3)), que são:

- elementos ativos – correspondente às atividades (macro eventos que representam a realização de operações como processamento, montagem, desmontagem);
- elementos passivos – correspondente às inter-atividades (lugares onde material, energia ou informações entram, ficam temporariamente armazenados e saem);
- *arcos* – que representam as relações entre os elementos ativos e passivos e indicam o sentido do fluxo de itens (materiais, informações, etc.). Este não representa necessariamente um elemento real do sistema, em geral, descreve uma relação lógica (eventualmente abstrata) entre os componentes.

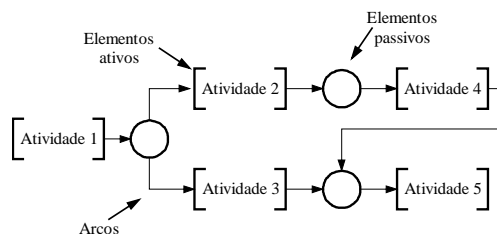


Figura (3) – Exemplo de um grafo PFS.

3.2 O E-MFG (*Enhanced - Mark Flow Graph*)

O E-MFG (Santos Filho, 1993; 1995) é uma extensão do MFG (Hasegawa, 1988; Miyagi, 1996). Este, assim com o MFG, é uma ferramenta de modelagem do comportamento dinâmico de sistemas de controle de equipamentos com capacidade para representar explicitamente relações de assincronismo, seqüencialização, paralelismo, concorrência e intertravamentos entre processos. O E-MFG é composto pelos seguintes elementos:

- elementos do MFG original (Hasegawa, 1988):
 - *box*: indica uma condição (Fig. (4a));
 - *transição*: representa a ocorrência de eventos (Fig. (4g));
 - *marcas*: indicam a manutenção de uma condição (Fig. (4i));
 - *arcos orientados*: estabelecem uma relação causal entre os eventos e as condições (Fig. (4k));
 - *arcos de sinal de saída*: podem transmitir informações aos dispositivos externos relativas ao estado dos *boxes* (Fig. (4n));
 - *portas habilitadoras / inibidoras*: portas que habilitam (Fig. (4l)) ou inibem (Fig. (4m)), respectivamente, a ocorrência de eventos.
- elementos derivados de macro-representações (Matsusaki, 1998):
 - *box capacidade*: é um macro elemento correspondendo a um grafo MFG com N elementos *box*, onde o símbolo N indica a sua capacidade (Fig. (4b));

- *box agrupador*: um macro elemento correspondendo a um grafo MFG de uma operação de montagem (Fig. (4c));
- *box dispersor*: um macro elemento correspondendo a um grafo MFG de uma operação de desmontagem (Fig. (4d));
- *box temporizador*: quando uma *marca* aparece neste tipo de *box*, ela deve necessariamente permanecer neste *box* um intervalo de tempo (Fig. (4f));
- *transição temporizada*: uma vez que todas as condições de disparo estejam satisfeitas, esta *transição* só dispara após decorrido um intervalo de tempo. Se durante este tempo uma das condições deixa de ser satisfeita, a contagem do tempo é anulada. Será reiniciada somente após todas as condições estarem novamente satisfeitas (Fig. (4h));
- elementos derivados de Redes de Petri Coloridas e outras extensões (Santos Filho, 1993). Neste caso as *marcas* podem ser individualizadas por *atributos* (Fig. (4j)) onde pode-se associar informações referentes ao produto, processo, etc. Desta forma além da necessidade de expandir os conceitos e representações associadas aos *boxes*, *transições*, *marcas* e *arcos* do MFG e das macro representações para a manipulação dos *atributos*, são introduzidos dois novos elementos:
 - *marca individual composta*: é uma *marca* que contém os *atributos* referentes à composição de várias *marcas* individuais;
 - *box controlador*: *box* onde se associa um conjunto de regras para a atualização dos *atributos* das *marcas*, as quais são regras de produção do tipo “se...então” referentes ao estado dos *atributos* e a atualização destes (Fig. (4e)).

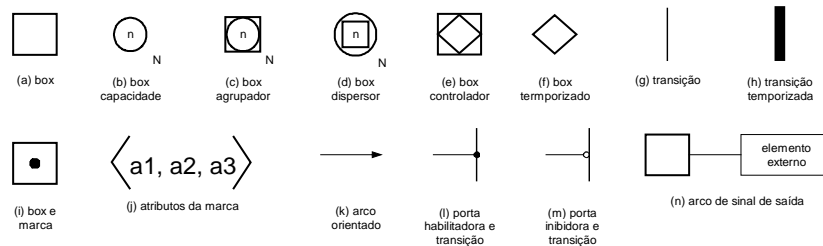


Figura (4) – Elementos do E-MFG.

Para efetuar a modelagem utilizando o E-MFG, devem ser observados os seguintes pontos:

- os *boxes* e as *transições* são necessariamente conectados de forma intercalada através dos *arcos orientados*.
- não existe limite para o número de *arcos* que entram ou saem dos *boxes* e das *transições*. Mas, em um par *transição-box* ou *transição-origem da porta externa*, pode existir apenas um *arco*.
- as *portas* e *arcos de sinal de saída* são formas de representação explícita das condições de controle.

A Fig. (5) ilustra a natureza da relação entre o PFS e o E-MFG, isto é, o modelo conceitual de um sistema é descrito em PFS cujos elementos são gradualmente detalhados até um modelo em E-MFG.

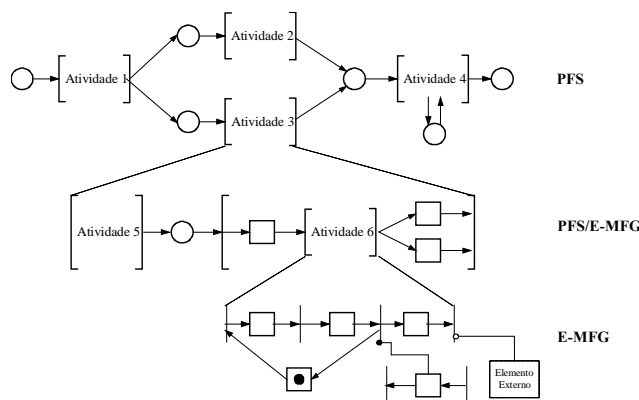


Figura (5) – Detalhamento de um modelo PFS até se obter o modelo em E-MFG.

4. Metodologia para a modelagem de sistemas flexíveis de movimentação de materiais em ambiente fabril

A metodologia para a modelagem de sistemas flexíveis de movimentação de materiais em ambiente fabril pode ser descrita em quatro passos / etapas como ilustra a Fig. (6).

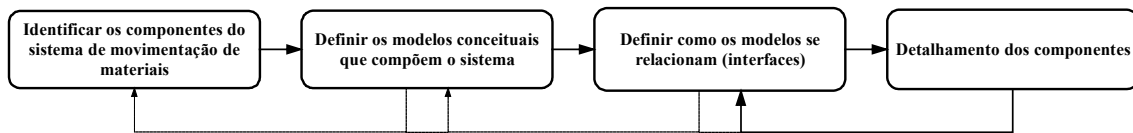


Figura (6) – Metodologia para a modelagem de sistemas de movimentação de materiais em ambiente fabril.

Passo 1: Identificação dos componentes do sistema de movimentação de materiais.

Nesta etapa identifica-se os componentes do sistema e suas características. Por exemplo:

- transportadores – características da frota; quantidade; capacidade de carga; etc.;
- pontos de distribuição de transportadores – características de sua distribuição, quantidade, imposições de retorno dos transportadores aos centros; etc.;
- características da demanda – natureza da demanda, possibilidade de satisfação parcial da demanda, etc.;
- características da malha de movimentação – número de centros de custos que compõem a malha; características dos caminhos (ligações) entre os centros de custos (unidirecional, bidirecional, ou a coexistência de ambas);
- custos – características dos custos envolvidos.

Passo 2: Definição dos modelos conceituais dos componentes que compõem o sistema.

Baseando-se nos problemas de caixeiro viajante e carteiro chinês e problemas derivados destes, destacam-se 2 elementos principais dos sistemas de movimentação de materiais, cujos modelos conceituais devem ser devidamente definidos:

- [Malha de Movimentação] definição dos *nós* (centros de custos) e segmentos (caminhos) que são representados, respectivamente, pelas atividades [Centro de Custos i] e [Transporte entre i e j];
- definição das *marcas* que percorrem a [Malha de Movimentação] e que podem representar instruções, transportadores, entre outros.

Passo 3: Definição de como os modelos se relacionam (interfaces).

Identificação dos relacionamentos dos componentes onde são explicitadas as informações que estão sendo trocadas entre os modelos. Como exemplos de relacionamentos citam-se:

- [Centro de Custos] e o [Sistema de Controle] (externo ao modelo) – informações para o transportador sobre as atividades que este deve desempenhar, informações sobre a posição do transportador;
- [Transporte entre i e j] e o [Sistema de Controle] (externo ao modelo) – informações sobre demanda nos *arcos*, informações sobre os estados dos segmentos, etc.;
- [Centro de Custos] e transportadores – informações sobre as atividades que o transportador deve desempenhar.

Passo 4: Detalhamento dos componentes.

Por fim, tem-se o detalhamento dos componentes do sistema partindo-se dos elementos conceituais até a obtenção dos modelos funcionais.

Os passos 3 e 4 são repetidos até se desenvolver o modelo funcional detalhado do sistemas. Em função do detalhamento funcional, caso haja necessidade, deve-se reconsiderar os passos anteriores no sentido de um contínuo aprimoramento do modelo.

5. Aplicação da metodologia em um estudo de caso

Como estudo de caso para demonstrar a aplicação da metodologia proposta no item 4, considerou-se uma parte da planta industrial da Mercedes Benz do Brasil:

Passo 1:

- transportadores – frota homogênea, composta por um ou múltiplos transportadores, e que não necessitam de base de distribuição;
- característica da rede – a rede é formada por centros de custos onde podem ser realizadas operações nos materiais (*nós*) e caminhos (*segmentos*) que podem ser utilizados para a movimentação de materiais entre os centros de custos. A movimentação nestes caminhos pode ocorrer nos dois sentidos.

Passo 2:

Aqui foram considerados a [Malha de Movimentação] composta por [Centros de Custos] e [Transporte entre i e j] e a *marca* que percorre a [Malha de Movimentação]. Além destes, incluiu-se um elemento externo que seria responsável pelo controle do sistema de movimentação ([Controle]) (Fig. (7)).

Passo 3:

- [Controle] (externo ao modelo) transfere instruções de movimentação (origem e destino dos materiais e a rota que o transportador deve utilizar) para a [Malha de Movimentação] e esta, por sua vez, informa a posição do transportador para o [Controle] (Fig. (7));
- a [Malha de Movimentação], através dos [Centros de Custos], transfere as instruções vindas do [Controle] para o [Transportador].

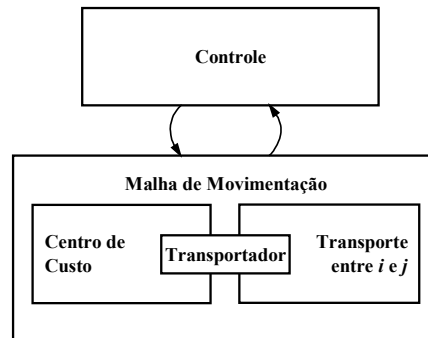


Figura (7) – Relacionamento entre os modelos do sistema.

Por fim, tem-se no quarto passo (e retornos ao passo 3) o detalhamento dos modelos conceituais que compõem o sistema.

5.1. Marca

A *marca*² é considerada como sendo composta por dois elementos:

- instrução – define as regras de movimentação como caminho a ser percorrido e a atividade a ser realizada;
- transportador – é o recurso responsável pela movimentação dos materiais.

$$\text{Marca} \infty \text{ .: } \text{Instrução} \infty, \text{Transportador} \infty \infty$$

A seguir, cada um destes é melhor detalhado:

A. Instrução

A *Instrução* ∞ é o *macro atributo* responsável por indicar ao transportador informações sobre as atividades que este deve realizar. Esta é composta por um conjunto de atributos (*origem* ∞ , *destino* ∞ e *rota a ser tomada* ∞) definidos no [Controle], responsável por designar os transportadores para atenderem as requisições de movimentação geradas pela produção.

$$\text{Instrução} \infty \text{ .: } \text{origem} \infty, \text{destino} \infty, \text{rota a ser tomada} \infty \infty$$

Seus *atributos* representam:

- origem – indica o centro de custo onde existe uma carreta a ser transportada;
- destino – local onde a carreta que está sendo transportada deve ser levada,
- rota a ser tomada – conjunto de *atributos* ordenados em forma de lista que indica o caminho que o transportador deve utilizar desde o instante em que lhe é solicitado a movimentação de uma carreta até a sua entrega no devido centro de custo;

B. O transportador

O transportador aqui considerado é um elemento do sistema que passa pelos centros de custo através de uma rota informada através do *macro atributo* *Instrução* ∞ .

Considera-se que o transportador possui quatro estados relevantes a serem modelados, que são:

- parado – o transportador está parado esperando por uma nova instrução de movimentação ou esperando para a realização do carregamento ou descarregamento de materiais;
- em movimento – o transportador está se movimentando ou para pegar materiais ou está transportando materiais para o centro de custo de destino;

² Como apresentado na seção 3.2, a *marca* indica a manutenção de uma condição no E-MFG. Utilizando conceitos de orientação a objetos [Junqueira, 2001] as partes móveis do sistema de movimentação de materiais (transportadores e carretas) são agrupadas (encapsuladas) em um mesmo elemento (*marca*). Assim, parte das regras de controle estão associadas à *marca* – neste caso, o *macro atributo* *Instrução* ∞ .

- carregando – ele está sendo carregado com materiais no centro de custo de origem para serem transportados até seu centro de custo de destino;
- descarregando – o transportador realizou a movimentação e está deixando os materiais em seu centro de custo de destino.

Chama-se a atenção para o fato de os tempos de carregamento e descarregamento estarem sendo considerados nas atividades do transportador pois estas são operações realizadas pelo seu operador ao passo que o tempo de movimentação está vinculado à rota – seu comprimento (já que a velocidade de movimentação é considerada constante).

O E-MFG da Fig. (8), descreve os estados do transportador. Quando a *marca* se encontra em B02 o transportador está parado. Caso a *transição* T01 seja disparada, a *marca* passa para B01, indicando que o transportador está em movimento e permanecerá até que a *transição* T02 seja disparada. Parado, o transportador pode também:

- carregando materiais (pelo disparo da *transição* T03 dando início ao carregamento (B03), e com o disparo de T04, o tempo de carregamento é considerado em B04; após o carregamento, a *transição* T05 dispara e o novo estado é indicado em B05; o disparo de T06 faz com que a *marca* retorne ao estado B02);
- descarregando (pelo disparo da *transição* T07 dando início ao descarregamento (B06) e com o disparo de T08, o tempo de descarregamento é considerado em B07; após o descarregamento, a *transição* T09 dispara e o novo estado é indicado em B08; o disparo de T10 faz com que a *marca* retorne ao estado B02). B09 indica que há material sendo rebocada pelo transportador.

Ainda no modelo da Fig. (8) tem-se duas *portas externas* (a1 e a2) e três *arcos de sinal de saída* (a3, a4 e a5):

- a1 – é a *porta* que habilita o carregamento e descarregamento de materiais. Ele assume o valor “0” para indicar que não há carregamento ou descarregamento, “1” quando este estiver no centro de custo especificado para que este carregue materiais e “2” no centro de custo especificado para que este descarregue materiais;
- a2 – *porta* que habilita a movimentação do transportador – “0” para manter o estado; “1” para habilitar a movimentação – ou porque este recebeu uma instrução para movimentar materiais de um centro de custo a outro ou acabou de carregar materiais e está se dirigindo ao centro de custo de destino; “2” para habilitar a sua parada ou porque chegou ao centro de custo para carregar materiais a serem transportados ou chegou ao centro de custo de destino para a descarregar os materiais;
- a3 – *arco de sinal* que indica que os materiais acabaram de ser carregados e que a movimentação já pode ser iniciada;
- a4 – *arco de sinal* que indica que os materiais acabaram de ser descarregados e o transportador pode atender a uma nova requisição de movimentação;
- a5 – *arco de sinal* que indica que o transportador está sendo utilizado na movimentação de materiais.

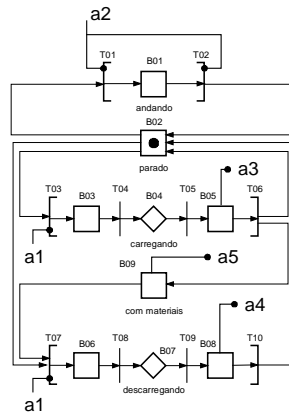


Figura (8) – Modelo do transportador.

No objeto *Transportador* ∞ , além dos *atributos* referentes a */a1* ∞ , */a2* ∞ , */a3* ∞ , */a4* ∞ e */a5* ∞ , existem outros dois *atributos* (*Ident.* ∞ e */grafo* ∞) que representam, respectivamente, a identificação do transportador (1, 2, ... n) e a estrutura de um grafo E-MFG que representa o elemento transportador (por exemplo: grafo da Fig. (8)). Este conjunto de sete *atributos* agrupados definem o macro atributo “transportador”:

$$/transportador\infty \therefore //ident.\infty, /a1\infty, /a2\infty, /a3\infty, /a4\infty, /a5\infty, /grafo\infty$$

5.2. Malha de movimentação

Apresenta-se aqui uma malha de movimentação simplificada afim de ilustrar o problema e obter um modelo para a análise do sistema. A Fig. (9) ilustra a malha considerada.

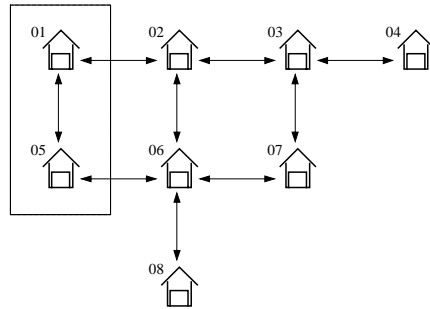


Figura (9) – Modelo representativo da malha de movimentação.

Utilizando-se a metodologia proposta, converte-se o modelo da Fig. (9) para uma representação utilizando-se o PFS (Fig. (10)) e dando início ao processo de detalhamento das atividades ([Centro de Custos i] e [Transporte entre i e j]).

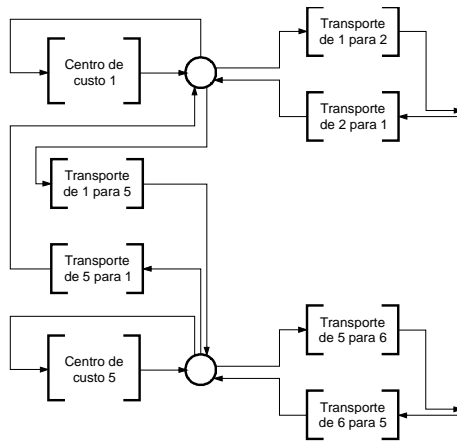


Figura (10) – Representação parcial da malha original utilizando-se o PFS.

No modelo em E-MFG da Fig. (11) está representada uma interligação (atividade de [transporte]) entre os centros de custo i e j , com capacidade para três transportadores. Esta poderia ser expandida ou mesmo ter sua capacidade reduzida caso necessário – no caso em estudo não se tem problema de restrição de capacidade nos *arcos*. O tempo de movimentação é o mesmo para todos quando se considera que o comprimento da interligação e a velocidade dos transportadores são as mesmas – frota homogênea.

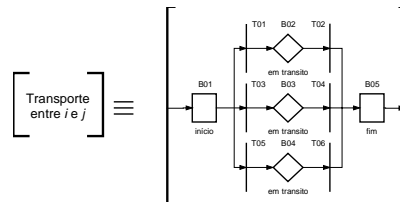


Figura (11) – Detalhamento das ligações de movimentação (atividade de [Transporte entre i e j]).

A Fig. (12) ilustra o detalhamento da atividade de um centro de custos i . Quando um transportador chega a um centro de custos, B01 é responsável por atualizar os *atributos* da *Instrução* e em seguida o transportador tem três possibilidades:

- passar direto (disparo de T06) – o centro de custos pelo qual ele está passando não é nem o local onde este deve pegar materiais nem onde os deve deixar – é apenas um ponto intermediário;
- carregar materiais (disparo de T08) – neste caso o material é carregado (B06) e o transportador segue para o centro de custos de destino;
- descarregamento de materiais (disparo de T02) – o material é descarregado (B03) e o transportador fica aguardando (em B05) nova *Instrução* (transmitidas através de T03 e armazenadas em B04) para movimentar outros materiais.

T05 disparando, há a combinação do macro *atributo* *Anstrução* vindo do [Controle] com a *marca* representando o transportador em B05

$$\text{Anstrução} + // \infty, \text{Transportador} \infty \rightarrow \text{Anstrução}, \text{Transportador} \infty$$

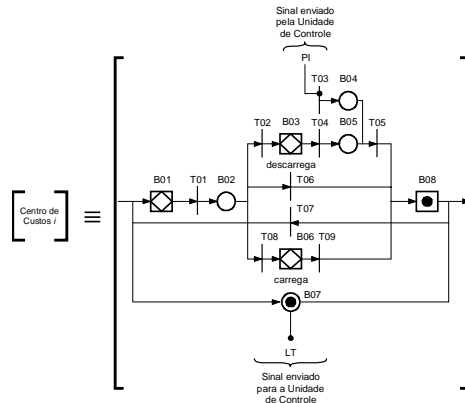


Figura (12) – Detalhamento da atividade [Centro de Custos i].

A Fig. (13) esquematiza o modelo resultante. Mais detalhes sobre o modelo resultante, a utilização da metodologia para o desenvolvimento de um sistema de controle, etc. podem ser obtidos em Junqueira [2001].

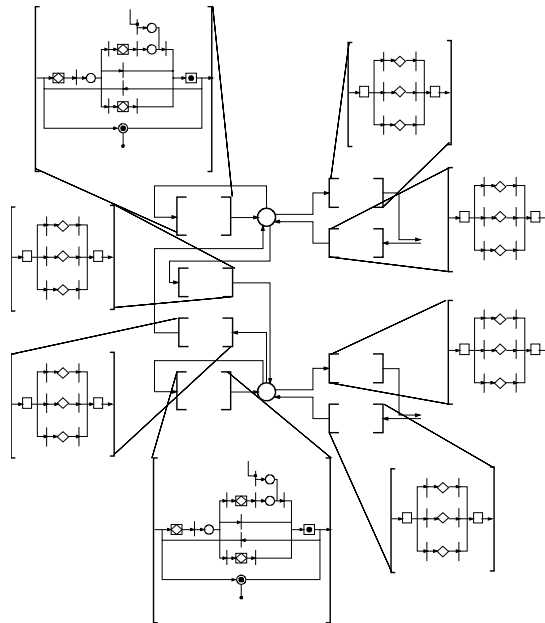


Figura (13) – Esquema do modelo resultante utilizando os modelos das Fig. (10), Fig. (11) e Fig. (12).

6. Comentários Finais e Conclusões

O estudo de um metodologia para a modelagem de sistemas de movimentação de materiais em ambiente fabril vem de encontro a uma necessidade por parte das empresas de dominar os processos e serem competitivas. Um modelo efetivo do sistema flexível de movimentação de materiais permite que se possa otimizar a frota de recursos de movimentação (por exemplo, transportadores e carretas) bem como estudar técnicas de controle que atendam suas necessidade produtivas e / ou institucionais (designar recursos de movimentação de forma a minimizar a distância percorrida e desta forma reduzir a emissão de poluentes, ou reduzir o tempo de espera para movimentação, etc.).

Focou-se o estudo em sistemas flexíveis de movimentação de materiais baseados em VATs (Veículos Autônomos de Transporte) cuja característica de não possuírem restrições quanto ao movimento (não existem guias ou trilhos limitando o seu movimento) agiliza mudanças de arranjo físico em instalações fabris, o que é um fator importante quando se procura adotar uma estratégia de “flexibilidade” e que subsidia uma evolução para a fase de “inovação” das empresas.

A abordagem do sistema de movimentação de materiais em ambiente fabril como uma classe de sistemas a eventos discretos, permite a utilização de técnicas como as Redes de Petri Interpretadas, dentre elas a metodologia PFS/E-MFG

(*Production Flow Schema / Enhanced Mark Flow Graph*), na sua modelagem. Esta metodologia provou ser eficiente como técnica de modelagem pois proporciona uma modelagem hierárquica onde se parte de um modelo conceitual em PFS até se obter um modelo funcional detalhado em E-MFG. Por exemplo, no estudo de caso apresentado, o modelo obtido foi fundamental para a análise e desenvolvimento das estratégias de controle do sistema

7. Agradecimentos

Os autores agradecem o apoio dado pela FAPESP, CNPq e CAPES ao presente trabalho.

8. Referências

- Baird, S.P.; Leavy, J.J., 1994, "Simulation Modeling Using Promodel for Windows", Proceedings of the 1994 Winter Simulation Conference.
- Banks, J., 1995, "Software for Simulation", Proceedings of the 1995 Winter Simulation Conference.
- Bodin, L., Golden, B., Assad, A., Ball, M., 1983, "Routing and Scheduling of Vehicles and Crews", Computers & Operations Research, Vol. 10, No. 1, New York, Pergamon Press.
- Bolwijn, P.; Kumpe, T., 1990, "Manufacturing in the 1990's - productivity, flexibility and innovation", Long Range Planning, Vol. 23, No.4, pp. 44-57.
- Cardoso, J.; Valette, R., 1997, "Redes de Petri", Editora da UFSC, Florianópolis.
- Cassandras, C. G., Strickland, S. G., 1992, "Sample Path Properties of timed Discrete Event Systems in Discrete Event Dynamic Systems – Analyzing Complexity and Performance in the Modern World", New York, IEEE Press.
- Hasegawa, K. et al., 1988, "Application of the Mark Flow Graph to Represent Discrete Event Production Systems and System Control", Transactions of the Society of Instrument and Control Engineers, Vol. 24, No. 1.
- Ho, Y.C.; Cao, X.R., 1991, "Perturbation Analysis of Discrete Event Dynamic Systems", Kluwer Academic Publishers.
- Junqueira, F., 2001, "Modelagem de Sistemas Flexíveis de Movimentação de Materiais Através de Redes de Petri Interpretadas", Dissertação de Mestrado, Escola Politécnica da USP, São Paulo.
- Martins, P. G., Laugeni, F. P., 1998, "Administração da Produção", São Paulo, Saraiva.
- Matsusaki, C. T. M., 1998, "Redes F-MFG (*Functional Mark Flow Graph*) e sua aplicação no projeto de sistemas antropocêntricos", Dissertação de Mestrado, Escola Politécnica da USP, São Paulo.
- Miyagi, P.E., 1996, "Controle Programável - Fundamentos do Controle de Sistemas a Eventos Discretos", São Paulo, Editora Edgard Blücher.
- Muscat, A.; Fleury, A., 1993, "Indicadores da Qualidade e Produtividade na Indústria Brasileira", Revista Indicadores da Qualidade, No. 2, pp. 82-107.
- Reisig, W., 1982, "Petri Nets - An Introduction", Springer-Verlag, Berlin.
- Reisig, W., 1992, "A Primer in Petri Design", Springer-Verlag, Berlin.
- Santos Filho, D.J., 1993, "Proposta do Mark Flow Graph Estendido para a Modelagem e Controle de Sistemas Integrados de Manufatura", Dissertação de Mestrado, Escola Politécnica da USP, São Paulo.
- Santos Filho, D.J., 1998, "Sistemas Antropocêntricos de Produção Baseado em Redes de Petri Interpretadas", Tese de Doutorado, Escola Politécnica da USP, São Paulo.
- Santos Filho, D.J.; Miyagi, P.E., 1996, "Enhanced MFG to Control Flexible Manufacturing Systems", Revista Brasileira de Ciências Mecânicas, Vol. 17, No. 2, pp. 232-248.

METHODOLOGY FOR MODELING FLEXIBLE SYSTEMS OF MATERIALS MOVEMENT

Fabício Junqueira

Departamento de Engenharia Mecatrônica e Sistemas Mecânicos – Escola Politécnica da USP – Av. Prof. Mello Moraes, 2231, CEP 05508-900 – São Paulo - SP
fabri@usp.br

Paulo Eigi Miyagi

Departamento de Engenharia Mecatrônica e Sistemas Mecânicos – Escola Politécnica da USP – Av. Prof. Mello Moraes, 2231, CEP 05508-900 – São Paulo - SP
pemiagi@usp.br

Abstract. Manufacturing systems have been object of study due to the continuous search of better methods for raising productivity and competitiveness. Among the manufacture process components, the material movement system deserves special attention because even not increasing the product value, it is responsible for keeping the flow of materials between machines, manufacturing cells, cost centers and also between companies, which is indispensable in any productive system. In this context, this work proposes a methodology for modeling flexible systems of materials and parts movement in the industrial environment, focusing on systems whose tasks can be performed by AGVs (Autonomous Guided Vehicles). Considering the materials movement system as a Discrete Event System (DES), techniques derived from Petri Nets as PFS (Production Flow Schema) and E-MFG (Enhanced Mark Flow Graph) are used for modeling those systems.

Keywords. *Discrete Event Systems, Petri Nets, Materials Movement Systems Modeling, Discrete Simulation.*

CONTROLE DE VIBRAÇÕES DE UM BRAÇO ROTATIVO FLEXÍVEL PELO MOVIMENTO DE UMA MASSA SOBRE O BRAÇO

Frederico Ricardo Ferreira de Oliveira

Departamento de Engenharia Mecânica; Escola Politécnica; Universidade de São Paulo; Caixa Postal 61548; CEP 05508-900; São Paulo, SP, Brasil; e-mail: fredrfo@usp.br

Agenor de Toledo Fleury

Agrupamento de Sistemas de Controle; Divisão de Mecânica e Eletricidade; Instituto de Pesquisas Tecnológicas do Estado de São Paulo - IPT; Caixa Postal 0141; CEP 01064-970; São Paulo, SP, Brasil; e-mail: agfleury@ipt.br

Resumo. O modelo de um braço rotativo flexível sobre o qual desliza uma massa móvel é deduzido pelo Princípio Estendido de Hamilton. Esse modelo é composto de equações íntegro-diferenciais acopladas, não-lineares e variáveis no tempo e no espaço. Com o fim de contornar as dificuldades matemáticas decorrentes, foi empregada a modelagem por subestruturação, considerando-se isoladamente o braço flexível e a massa móvel. Esse segundo modelo é genérico, no sentido de abranger um número arbitrário de autofunções. Duas abordagens distintas quanto ao movimento da massa são analisadas. Na primeira, admite-se pré-especificada sua trajetória sobre o braço durante a rotação. Na segunda abordagem, o movimento da massa é usado para fins de controle, empregando-se uma força atuante sobre ela como uma segunda variável de controle, além do torque sobre o braço; o modelo do controlador torna-se então não-linear. Um controlador LQ de ganho variável foi implementado para a primeira abordagem, e formulou-se para a segunda análise um Problema de Controle Ótimo. Ambos os casos foram simulados e analisados, mostrando que a proposta inicial pode gerar excelentes resultados no controle de vibrações de estruturas flexíveis.

Palavras chave: estruturas flexíveis, braço rotativo, controle, controle ótimo

1. Introdução

A Dinâmica e o Controle de estruturas flexíveis vêm despertando um crescente interesse nas últimas décadas, devido às amplas perspectivas que abririam em áreas importantes como a Engenharia Espacial e a Robótica. Demonstram-no os inúmeros trabalhos publicados, como os de Juang et al (1986) e Özgüner & Barbieri (1988). Book (1990) apresenta o estado da arte em Dinâmica e Controle de estruturas flexíveis. No estudo de sistemas mecânicos flexíveis, destaca-se o problema da modelagem e controle de vibrações. Diversos métodos de atenuação de vibrações têm sido examinados, tais como o emprego de materiais piezoelétricos na construção de atuadores (Warren et al, 1995). Para esse efeito, pouco proveito é tirado da interação entre as partes distintas de um sistema: quando se pretende projetar mecanismos de múltiplos corpos, é quase inevitável deparar-se com problemas de flexibilidade e vibrações.

Diversos trabalhos têm-se debruçado sobre o estudo e implementação de técnicas de controle otimizado. Schwartz & Polak (1997) desenvolvem o método das aproximações consistentes para a solução numérica do Problema de Controle Ótimo. Friedland et al (1987) demonstra que, para sistemas variantes no tempo, uma solução sub-ótima do problema de controle pode ser obtida pela resolução da equação algébrica de Riccati para uma discretização no tempo do sistema. Este trabalho propõe-se estudar o controle de vibrações de um braço rotativo flexível sobre o qual desliza uma massa móvel, empregando no controle o torque aplicado ao cubo e a força que posiciona o carro ao longo do braço. Pretende-se avaliar se a interação entre o carro e o braço pode ser empregada efetivamente no controle de vibrações.

2. Modelagem dinâmica do braço rotativo

A Fig. (1) mostra um diagrama esquemático da flexão provocada no braço, quando uma manobra de rotação é realizada. Admite-se que a manobra se realiza no plano horizontal Oxy , de modo que as forças-peso não são consideradas. As variáveis presentes na Fig. (1) estão definidas ao longo do texto.

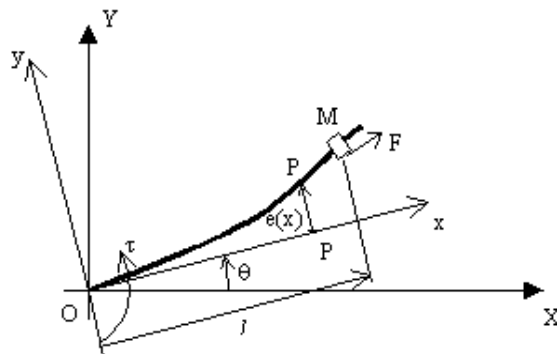


Figura 1. Flexão do braço rotativo durante manobra de rotação (vista de topo).

2.1 Energia potencial elástica e energia cinética do braço

O braço armazena uma energia potencial elástica devida à flexão igual a

$$U(t) = \frac{1}{2} \cdot \int_0^L E \cdot I_B \cdot \left(\frac{\partial^2 e}{\partial x^2} \right)^2 dx \quad (1)$$

E denota o módulo de elasticidade do material; I_B é momento de inércia de área do braço em relação ao eixo Oz, normal ao plano da rotação; e $e = e(x, t)$ é a flexão transversal da viga. $\partial^2 e / \partial x^2$ denota a derivada parcial segunda de $e(x, t)$ em relação a x .

A energia cinética total armazenada no braço é

$$T_B(t) = \int_0^L \frac{1}{2} \rho \cdot \left[\left(\frac{\partial e}{\partial t} \right)^2 + 2 \cdot \frac{\partial e}{\partial t} \cdot x \cdot \frac{d\theta}{dt} + (x^2 + e^2) \cdot \left(\frac{d\theta}{dt} \right)^2 \right] dx \quad (2)$$

onde L e ρ são o comprimento do braço e sua densidade linear de massa, respectivamente. $\theta(t)$ é o ângulo de rotação do sistema Oxyz, conforme a Fig. (1).

2.2 Energia cinética da massa móvel

A energia cinética da massa é igual a

$$T_M(t) = \frac{1}{2} \cdot M \cdot \left[\left(\frac{\partial e}{\partial t} \right)_{x=l}^2 + 2 \cdot \left(\frac{\partial e}{\partial t} \right)_{x=l} \cdot l \cdot \frac{d\theta}{dt} + [l^2 + e^2(l)] \cdot \left(\frac{d\theta}{dt} \right)^2 + \left(\frac{dl}{dt} \right)^2 - 2 \cdot \frac{dl}{dt} \cdot e(l) \cdot \frac{d\theta}{dt} \right] \quad (3)$$

onde M é a inércia da massa e $l(t)$, sua posição no sistema de referência girando com o braço, como mostra a Fig. (1).

2.3 Energia cinética do cubo rotativo

Seja L_C o raio; M_C , a massa, e J_C , o momento de inércia do cubo rotativo onde se fixa o braço flexível. A energia cinética armazenada no cubo é

$$T_C(t) = \frac{1}{2} \cdot \frac{M_C \cdot L_C^2}{2} \cdot \left(\frac{d\theta}{dt} \right)^2 = \frac{1}{2} \cdot J_C \cdot \left(\frac{d\theta}{dt} \right)^2 \quad (4)$$

2.4 Modelo contínuo do sistema

O Princípio de Hamilton estabelece que, entre dois instantes arbitrários t_1 e t_2 ,

$$\int_{t_1}^{t_2} \delta L dt + \int_{t_1}^{t_2} \delta W dt = 0 \quad (5)$$

onde, sem prejuízo da notação, L indica o lagrangiano do sistema, definido como

$$L = T - U = T_V + T_M + T_C - U \quad (6)$$

O trabalho virtual das forças não-conservativas é dado por

$$\delta W = \tau \cdot \delta\theta + F \cdot \delta l \quad (7)$$

τ é o torque aplicado ao cubo e F , a força atuante sobre a massa móvel, de direção tangencial ao braço no ponto l em que se encontra o carro.

Assim, o Princípio de Hamilton reduz-se a

$$\int_0^L \left\{ E \cdot I_B \cdot \frac{\partial^4 e}{\partial x^4} + \rho \cdot \left[\frac{\partial^2 e}{\partial t^2} + x \cdot \frac{d^2 \theta}{dt^2} - e \cdot \left(\frac{d\theta}{dt} \right)^2 \right] + M \cdot \left[\frac{\partial^2 e}{\partial t^2} + 2 \cdot \frac{dl}{dt} \cdot \frac{d\theta}{dt} + l \cdot \frac{d^2 \theta}{dt^2} - e \cdot \left(\frac{d\theta}{dt} \right)^2 \right] \cdot \Delta_l \right\} dx = 0 \quad (8)$$

$$\int_0^L \left[\rho \cdot \left(e^2 \cdot \frac{d^2 \theta}{dt^2} + 2 \cdot e \cdot \frac{\partial e}{\partial t} \cdot \frac{d\theta}{dt} + x \cdot \frac{\partial^2 e}{\partial t^2} + x^2 \cdot \frac{d^2 \theta}{dt^2} \right) + M \cdot \left(\frac{\partial^2 e}{\partial t^2} \cdot l - e \cdot \frac{d^2 l}{dt^2} + e^2 \cdot \frac{d^2 \theta}{dt^2} + 2 \cdot e \cdot \frac{\partial e}{\partial t} \cdot \frac{d\theta}{dt} \right) \cdot \Delta_l \right] dx + M \cdot \left(l^2 \cdot \frac{d^2 \theta}{dt^2} + 2 \cdot l \cdot \frac{\partial l}{\partial t} \cdot \frac{d\theta}{dt} \right) + J_c \cdot \frac{d^2 \theta}{dt^2} = \tau \quad (9)$$

$$M \cdot \left[\frac{d^2 l}{dt^2} - l \cdot \left(\frac{d\theta}{dt} \right)^2 \right] - \int_0^L M \cdot \left(e \cdot \frac{d^2 \theta}{dt^2} + 2 \cdot \frac{\partial e}{\partial t} \cdot \frac{d\theta}{dt} \right) \cdot \Delta_l dx = F \quad (10)$$

Δ_l assinala o delta de Dirac, introduzido por motivos de simplificação da notação. As Eq. (8), (9) e (10) compõem o modelo contínuo do sistema, restrito às condições de contorno

$$e(0, t) = 0, \quad \left(\frac{\partial e}{\partial x} \right)_{x=0, t} = 0, \quad \left(\frac{\partial^2 e}{\partial x^2} \right)_{x=L, t} = 0, \quad \left(\frac{\partial^3 e}{\partial x^3} \right)_{x=L, t} = 0 \quad (11)$$

3. Modelo por síntese de subestruturas

O modelo contínuo compõe-se, como se verifica, de um conjunto de equações íntegro-diferenciais acopladas, com derivadas parciais. Uma solução matemática direta para esse problema é extremamente difícil, como atesta Meirovitch (1997). Técnicas de subestruturação têm sido empregadas satisfatoriamente no estudo de estruturas mecânicas complexas. Trata-se de uma possível abordagem ao estudo em questão, alternativa à solução direta das equações do sistema. A interação entre o braço rotativo e a massa móvel, analisados isoladamente, é modelada como uma força $F_E(l)$, externa ao braço e pontual, aplicada sobre ele na posição da massa, l .

3.1 Primeiro subsistema: massa móvel

Admite-se que, para pequenas oscilações, a força $F_E(l)$ aplicada ao braço pela massa móvel é

$$F_E(l) = -M \cdot \left(\frac{\partial^2 e}{\partial t^2} \Big|_{x=l} + l \cdot \frac{d^2 \theta}{dt^2} + 2 \cdot \frac{dl}{dt} \cdot \frac{d\theta}{dt} \right) \quad (12)$$

Desprezando os termos quadráticos, deduz-se que a força externa responsável pelo posicionamento da massa móvel é

$$F = M \cdot \left[\frac{d^2 l}{dt^2} - l \cdot \left(\frac{d\theta}{dt} \right)^2 \right] \quad (13)$$

3.2 Segundo subsistema: braço flexível

Considera-se agora um subsistema composto do braço rotativo flexível e do cubo no qual ele se fixa. Duas forças externas atuam sobre o conjunto: o torque τ aplicado ao cubo; e a força $F_E(l)$, de direção normal ao braço no ponto l , onde é aplicada.

O trabalho virtual das forças não-conservativas, nesse caso, é dado por

$$\delta W = \tau \cdot \delta \theta + F_E(l) \cdot \delta e \quad (14)$$

Considerando a Eq. (6), o Princípio de Hamilton reduz-se então a

$$\int_0^L E \cdot I_B \cdot \frac{\partial^4 e}{\partial x^4} + \rho \cdot \left[\frac{\partial^2 e}{\partial t^2} + x \cdot \frac{d^2 \theta}{dt^2} - e \cdot \left(\frac{d\theta}{dt} \right)^2 \right] dx = F_E(l) \quad (15)$$

$$\int_0^L \rho \cdot \left(e^2 \cdot \frac{d^2 \theta}{dt^2} + 2 \cdot e \cdot \frac{\partial e}{\partial t} \cdot \frac{d\theta}{dt} + x \cdot \frac{\partial^2 e}{\partial t^2} + x^2 \cdot \frac{d^2 \theta}{dt^2} \right) dx + J_C \cdot \frac{d^2 \theta}{dt^2} = \tau \quad (16)$$

restritas às mesmas condições de contorno indicadas na Eq. (11). Na Eq. (11), J_B é o momento de inércia do braço flexível. No modelo descrito pelas Eq. (15) e (16), supõe-se que os termos quadráticos são pequenos e podem ser desconsiderados.

Seja $z(x, t)$ uma variável definida por

$$z(x, t) = e(x, t) + x \cdot \theta(t) \quad (17)$$

Decorre das Eq. (15) e (16) que

$$E \cdot I_B \cdot \frac{\partial^4 z}{\partial x^4} + \rho \cdot \frac{\partial^2 z}{\partial t^2} = F_E(l) \quad (18)$$

$$\int_0^L \rho \cdot x \cdot \frac{\partial^2 z}{\partial t^2} dx + J_C \cdot \frac{d^2 \theta}{dt^2} = \tau \quad (19)$$

Investigando a possibilidade de movimentos síncronos, a resposta do sistema pode ser aproximada por uma expansão de funções independentes no tempo e no espaço:

$$z(x, t) = \sum_{r=1}^{\infty} \phi_r(x) \cdot \eta_r(t) \quad (20)$$

$\phi_r(x)$ são funções admissíveis e $\eta_r(t)$, coordenadas generalizadas. Decorre então que, para a vibração livre do braço flexível,

$$E \cdot I_B \cdot \frac{d^4 \phi_r}{dx^4} - \omega_r^2 \cdot \rho \cdot \phi_r = 0 \quad (21)$$

Por inspeção, verifica-se que a forma geral

$$\phi_r(x) = a_r \cdot \text{sen}(\beta_r \cdot x) + b_r \cdot \text{cos}(\beta_r \cdot x) + c_r \cdot \text{senh}(\beta_r \cdot x) + d_r \cdot \text{cosh}(\beta_r \cdot x) \quad (22)$$

satisfaz a equação das autofunções. Os coeficientes a_r, b_r, c_r, d_r e β_r da Eq. (22) são função dos parâmetros do sistema, como demonstrado por Meirovitch (1997). Para modos normalizados, o problema da resposta reduz-se então a

$$\frac{d^2 \eta_r}{dt^2} + \omega_r^2 \cdot \eta_r = \int_0^L F_E(l) \cdot \phi_r dx + \tau \cdot \frac{d\phi_r}{dx} \Big|_{x=0} \quad (23)$$

equação que descreve a dinâmica do sistema em termos das coordenadas generalizadas.

3.3 Modelo do sistema por síntese de subestruturas

Substituindo a Eq. (12) da força aplicada ao braço pela massa na Eq. (23) das coordenadas generalizadas, conclui-se que

$$\frac{d^2 \eta_r}{dt^2} = \frac{1}{M} \cdot \sum_{s=1}^p \left[S_{rs} \cdot \left(-\omega_s^2 \cdot \eta_s - 2 \cdot M \cdot \frac{dl}{dt} \cdot \frac{d\theta}{dt} \cdot \int_0^L \phi_s dx + \frac{d\phi_s}{dx} \Big|_{x=0} \cdot \tau \right) \right] \quad (24)$$

onde os elementos S_{rs} dependem dos parâmetros físicos do conjunto e da posição l do carro, e são dados por

$$S_{rs} = \begin{cases} \frac{M + M \cdot \sum_{i=1, i \neq r}^p \left[M \cdot \phi_i(l) \cdot \int_0^L \phi_i dx \right]}{1 + \sum_{i=1}^p \left[M \cdot \phi_i(l) \cdot \int_0^L \phi_i dx \right]}, & \text{para } r = s \\ \frac{-M^2 \cdot \int_0^L \phi_r dx \cdot \phi_s(l)}{1 + \sum_{i=1}^p \left[M \cdot \phi_i(l) \cdot \int_0^L \phi_i dx \right]}, & \text{para } r \neq s \end{cases} \quad (25)$$

Pode-se mostrar que a dinâmica do movimento angular do braço é descrita por

$$\begin{aligned} \frac{d^2\theta}{dt^2} = & \frac{1}{(J_B + J_C + M \cdot l^2)} \cdot \left\{ l \cdot \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(l) \cdot S_{rs} \cdot \omega_s^2 \cdot \eta_s \right] + \right. \\ & + 2 \cdot M \cdot l \cdot \frac{dl}{dt} \cdot \frac{d\theta}{dt} \cdot \left\{ \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(l) \cdot S_{rs} \cdot \int_0^L \phi_s dx \right] - 1 \right\} + \\ & \left. + \left\{ 1 - l \cdot \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(l) \cdot S_{rs} \cdot \frac{d\phi_s}{dx} \Big|_{x=0} \right] \right\} \cdot \tau \right\} \end{aligned} \quad (26)$$

As Eq. (13), (24) e (26) constituem o modelo por síntese de subestruturas, e descrevem, respectivamente, as dinâmicas de translação da massa acoplada, de oscilação transversal do braço devido à flexão e de rotação do mesmo.

4. Manobra pré-especificada da massa móvel

A primeira abordagem possível para o problema de controle empregando o movimento da massa sobre o braço consiste em tratar sua posição l e velocidade dl/dt como parâmetros dados. Nesse caso, torna-se desnecessária a descrição da dinâmica do movimento longitudinal da massa – Eq. (13) –, já que o posicionamento do carro é tratado de um ponto de vista exclusivamente cinemático. Das equações que descrevem os movimentos de oscilação e rotação da viga flexível, Eq. (24) e (26), decorre que, para um modelo considerando p autofunções,

$$\begin{cases} \frac{dx_{(2r-1)}}{dt} = x_{2r} \\ \frac{dx_{2r}}{dt} = \frac{1}{M} \cdot \sum_{s=1}^p \left[S_{rs} \cdot \left(-\omega_s^2 \cdot x_{(2s-1)} - 2 \cdot M \cdot \frac{dl}{dt} \cdot \int_0^L \phi_s dx \cdot x_{(2p+2)} + \frac{d\phi_s}{dx} \Big|_{x=0} \cdot u \right) \right] \\ \frac{dx_{(2p+1)}}{dt} = x_{(2p+2)} \\ \frac{dx_{(2p+2)}}{dt} = \frac{1}{J_B + J_C + M \cdot l^2} \cdot \left\{ l \cdot \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(l) \cdot S_{rs} \cdot \omega_s^2 \cdot x_{(2s-1)} \right] + \right. \\ \quad + 2 \cdot M \cdot l \cdot \frac{dl}{dt} \cdot \left\{ \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(l) \cdot S_{rs} \cdot \int_0^L \phi_s dx \right] - 1 \right\} \cdot x_{(2p+2)} + \\ \quad \left. + \left\{ 1 - l \cdot \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(l) \cdot S_{rs} \cdot \frac{d\phi_s}{dx} \Big|_{x=0} \right] \right\} \cdot u \right\} \end{cases} \quad (27)$$

onde as variáveis de estado e de controle do sistema são definidas por

$$\begin{aligned}
 x_1 &= \eta_1 & x_2 &= \frac{d\eta_1}{dt} & x_3 &= \eta_2 & x_4 &= \frac{d\eta_2}{dt} \\
 x_5 &= \eta_3 & x_6 &= \frac{d\eta_3}{dt} & x_{2p-1} &= \eta_p & x_{2p} &= \frac{d\eta_p}{dt} \\
 x_{2p+1} &= \theta & x_{2p+2} &= \frac{d\theta}{dt} \\
 u &= \tau
 \end{aligned} \tag{28}$$

A posição l do carro aparece implícita no modelo de estado do sistema, sendo τ , o torque aplicado ao cubo, a única variável de controle.

Friedland *et al* (1987) demonstram, através da Teoria de Liapunov, que uma solução sub-ótima do problema de controle LQ para sistemas variantes no tempo (*time varying systems*) pode ser determinada pela solução algébrica de Riccati para a lei de controle ótimo, resolvendo as equações do sistema para determinados instantes de tempo. Trata-se de uma discretização no tempo do problema de controle, que resulta sempre em respostas estáveis e com bom desempenho. Um controle desse tipo pode ser construído para o sistema descrito pelas equações lineares Eq. (27) e (28).

5. Controle ótimo do movimento da massa sobre o braço

Uma segunda abordagem do problema consiste em considerar no controlador o movimento da massa. Uma força F aplicada ao carro, conforme ilustra a Fig. (1), é usada como uma segunda variável de controle. A posição l e velocidade dl/dt são variáveis de estado. Deduz-se das equações que descrevem os movimentos de oscilação e rotação da viga flexível, Eq. (24) e (26), e da equação do movimento da massa, Eq. (13), que

$$\left\{ \begin{aligned}
 \frac{dx_{(2r-1)}}{dt} &= x_{2r} \\
 \frac{dx_{2r}}{dt} &= \frac{1}{M} \cdot \sum_{s=1}^p \left[S_{(2r-1)s} \cdot \left(-\omega_s^2 \cdot x_{(2s-1)} - 2 \cdot M \cdot \int_0^L \phi_s dx \cdot x_{(2p+2)} \cdot x_{(2p+4)} + \frac{d\phi_s}{dx} \Big|_{x=0} \cdot u_1 \right) \right] \\
 \frac{dx_{(2p+1)}}{dt} &= x_{(2p+2)} \\
 \frac{dx_{(2p+2)}}{dt} &= \frac{1}{J_B + J_C + M \cdot x_{(2p+3)}^2} \cdot \left\{ \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(x_{(2p+3)}) \cdot S_{rs} \cdot \omega_s^2 \cdot x_{(2s-1)} \right] \cdot x_{(2p+3)} + \right. \\
 &\quad \left. + 2 \cdot M \cdot \left\{ \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(x_{(2p+3)}) \cdot S_{rs} \cdot \int_0^L \phi_s dx \right] - 1 \right\} \cdot x_{(2p+2)} \cdot x_{(2p+3)} \cdot x_{(2p+4)} + \right. \\
 &\quad \left. + \left[1 - \sum_{r=1}^p \sum_{s=1}^p \left[\phi_r(x_{(2p+3)}) \cdot S_{rs} \cdot \frac{d\phi_s}{dx} \Big|_{x=0} \right] \cdot x_{(2p+3)} \right\} \cdot u_1 \right\} \\
 \frac{dx_{(2p+3)}}{dt} &= x_{(2p+4)} \\
 \frac{dx_{(2p+4)}}{dt} &= \frac{u_2}{M} + x_{(2p+2)}^2 \cdot x_{(2p+3)}
 \end{aligned} \right. \tag{29}$$

onde as variáveis de estado e de controle do sistema são definidas por

$$\begin{aligned}
 x_1 &= \eta_1 & x_2 &= \frac{d\eta_1}{dt} & x_3 &= \eta_2 & x_4 &= \frac{d\eta_2}{dt} \\
 x_5 &= \eta_3 & x_6 &= \frac{d\eta_3}{dt} & x_{2p-1} &= \eta_p & x_{2p} &= \frac{d\eta_p}{dt} \\
 x_{2p+1} &= \theta & x_{2p+2} &= \frac{d\theta}{dt} & x_{2p+3} &= l & x_{2p+4} &= \frac{dl}{dt} \\
 u_1 &= \tau & u_2 &= F
 \end{aligned} \tag{30}$$

O Problema do Controle Ótimo do sistema braço-massa móvel consiste na minimização da função objetivo

$$J = \int_0^{t_f} \left(\sum_{i=1}^{2p+4} Q_{ii} \cdot x_i^2 + \sum_{j=1}^2 R_{jj} \cdot u_j^2 \right) dt \quad (31)$$

sujeita aos vínculos dinâmicos expressos pela Eq. (29) e aos vínculos de trajetória

$$0 \leq l \leq L \quad (32)$$

6. Simulação computacional

Uma simulação computacional do sistema foi implementada, empregando o MATLAB e o RIOTS_95, um pacote para a resolução de problemas de controle ótimo, criado por Schwartz & Polak (1996). Foram simulados casos para as duas abordagens do problema de controle descritas nas Seções 4 e 5, todos considerando manobras do braço com giro de -45° a 0° , tempo de simulação de 4 segundos e 514 pontos de discretização. Os valores dos parâmetros físicos estão indicados na Tab. (1).

Tabela 1. Parâmetros físicos empregados na simulação.

Parâmetros físicos	Nomenclatura	Valor
Módulo de elasticidade do braço (Pa)	E	$7,1 \cdot 10^{10}$
Espessura do braço (m)	h	0,001
Altura do braço (m)	b	0,0254
Comprimento do braço (m)	L	0,7
Densidade volumétrica do braço (kg/m^3)	ρ_B	2710
Massa do braço (kg)	m	$\rho_B \cdot L \cdot b \cdot h$
Densidade linear do braço (kg/m^2)	ρ	m / L
Momento de inércia de área (m^4)	I_B	$b \cdot h^3 / 12$
Massa móvel (kg)	M	0,01

6.1 Manobra pré-especificada da massa móvel

A resposta do sistema foi descrita em termos de três autofunções, correspondentes às três primeiras frequências naturais do braço preso ao cubo rotativo, sem massa acoplada. Um controlador LQ de ganhos variáveis no tempo foi obtido através da resolução da equação algébrica de Riccati para os 514 pontos de discretização no tempo.

Alguns casos simulados, apresentados por Oliveira (2000), sugerem que o movimento da massa seja tal que a força de Coriolis aplicada ao braço pela massa móvel esteja em oposição de fase com as oscilações de η_l , coordenada generalizada associada à primeira frequência natural (Oliveira & Fleury, 1999; Oliveira & Fleury, 2001). A Fig. (2) ilustra a resposta do sistema para esse caso. Da esquerda para a direita e de cima para baixo, indicam-se: a posição e a velocidade da massa móvel, pré-especificadas; a posição e a velocidade angulares do braço resultantes; o deslocamento da ponta do braço; o torque τ aplicado. A massa se desloca num movimento periódico atenuado em torno da posição $x = 0,2552$ m (correspondente ao ponto de máximo deslocamento do primeiro modo de vibração do braço sem massa acoplada, excetuando-se a extremidade livre), procurando que as forças de Coriolis estejam em oposição de fase com η_l . Os pesos dos estados η_l e $d\eta_l/dt$ são 1000 vezes maiores que os pesos correspondentes às demais variáveis de estado e controle, a saber: θ , $d\theta/dt$ e τ .

Observa-se nesse caso uma forte atenuação das oscilações do braço. Certas manobras, entretanto, tais como o movimento senoidal da massa com uma frequência igual à primeira frequência natural da viga, resultaram em aumento das amplitudes de vibração.

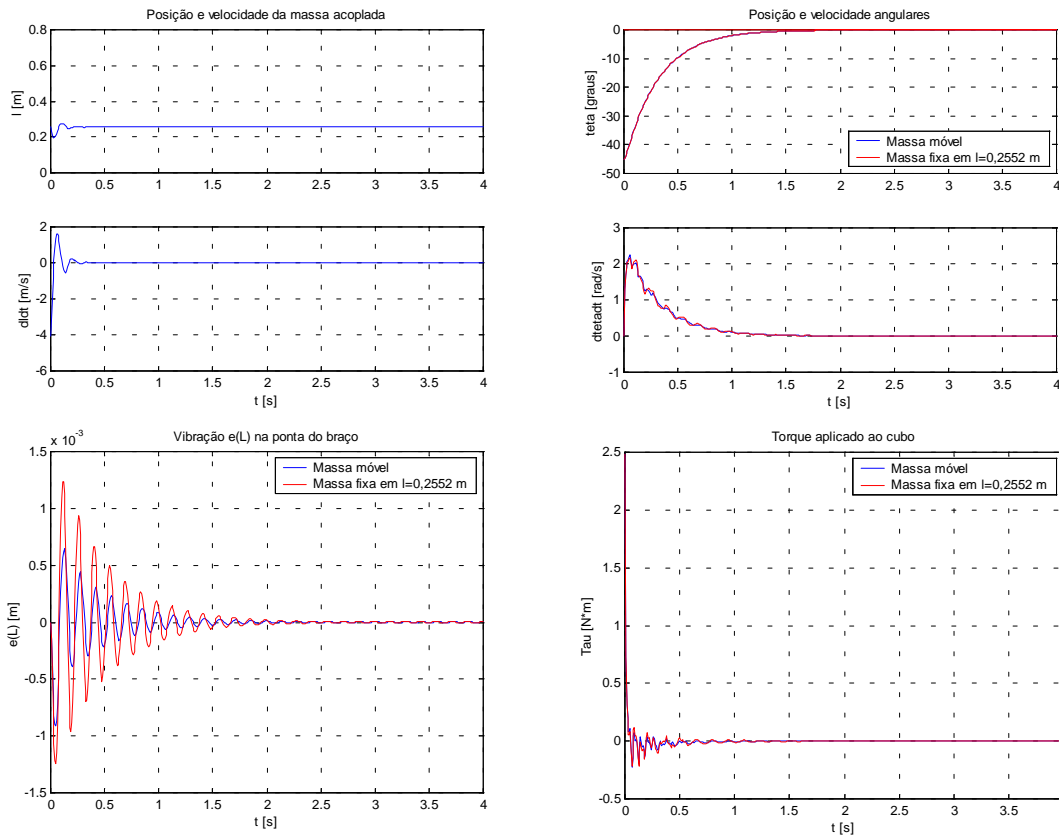


Figura 2. Rotação do braço flexível.

6.2 Controle ótimo empregando o movimento da massa

Nos casos simulados para a segunda abordagem, a resposta do sistema foi descrita em termos de uma única autofunção, correspondente à primeira frequência natural do braço sem massa. Empregando o RIOTS_95 (Schwartz et al, 1997), examinou-se a resposta do sistema para duas situações distintas: a massa deslocando-se sem vínculos de trajetória; a massa sujeitas aos vínculos indicados na Eq. (32).

A Fig. (3) ilustra o movimento da massa encontrado no caso de ausência de vínculos de trajetória. Indicam-se a posição e a velocidade da massa e a força F aplicada. O movimento da massa é suave, mas ultrapassa o cubo, assinalado pelo ponto $x = 0$. O controle inicial, a partir do qual o RIOTS_95 procura o controle ótimo, também está indicado. A Fig. (4) ilustra a vibração da extremidade livre do braço. Verifica-se uma forte atenuação das oscilações.

A Fig. (5) ilustra o movimento da massa quando são considerados os vínculos de trajetória apontados pela Eq. (32). A Fig. (6) indica a oscilação na ponta do braço e o torque aplicado ao cubo rotativo. Verifica-se novamente uma atenuação das oscilações. O movimento da massa nesse caso também é suave, respeitando os vínculos no cubo e na extremidade livre do braço.

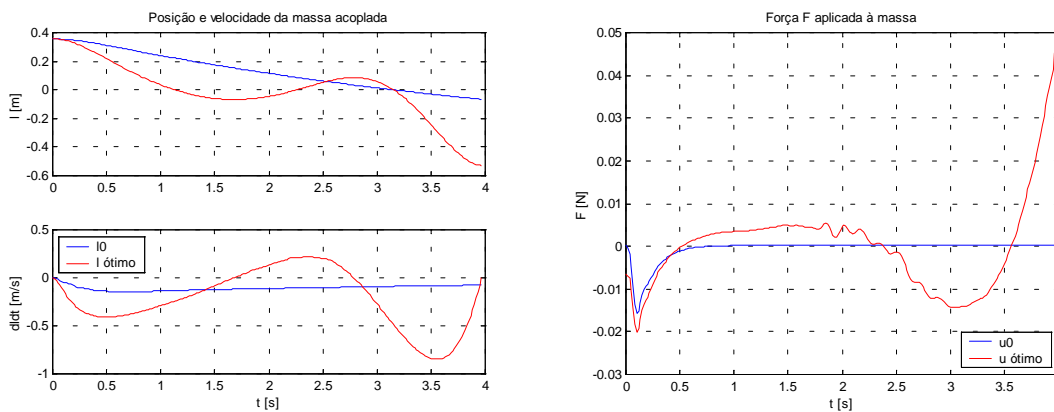


Figura 3. Posição da massa e força F aplicada na ausência de vínculos de trajetória.

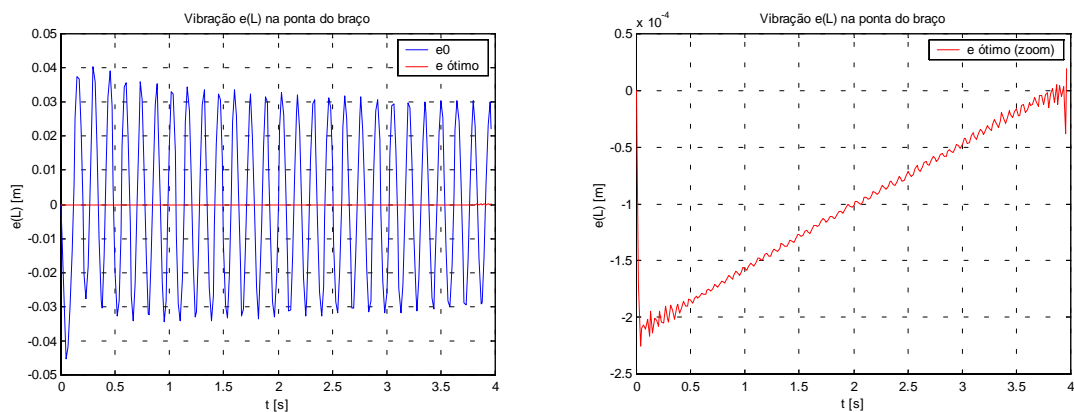


Figura 4. Vibração da ponta do braço na ausência de vínculos de trajetória.

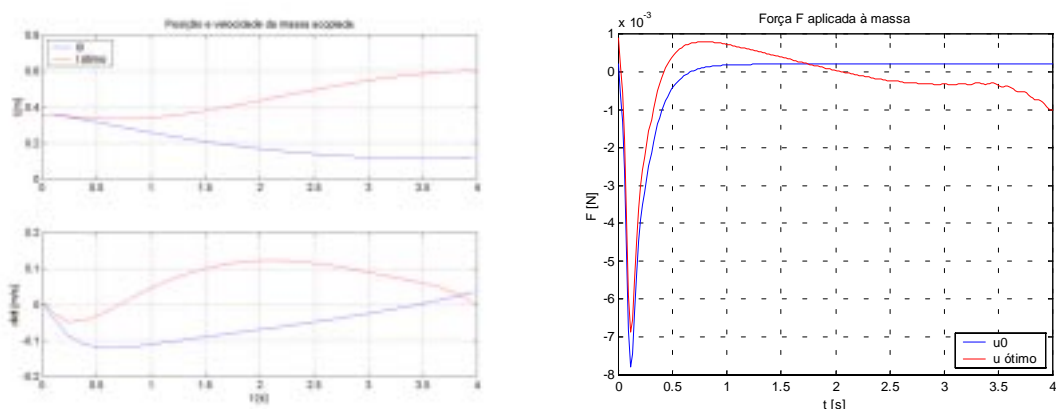


Figura 5. Posição da massa e força F aplicada na presença de vínculos de trajetória.

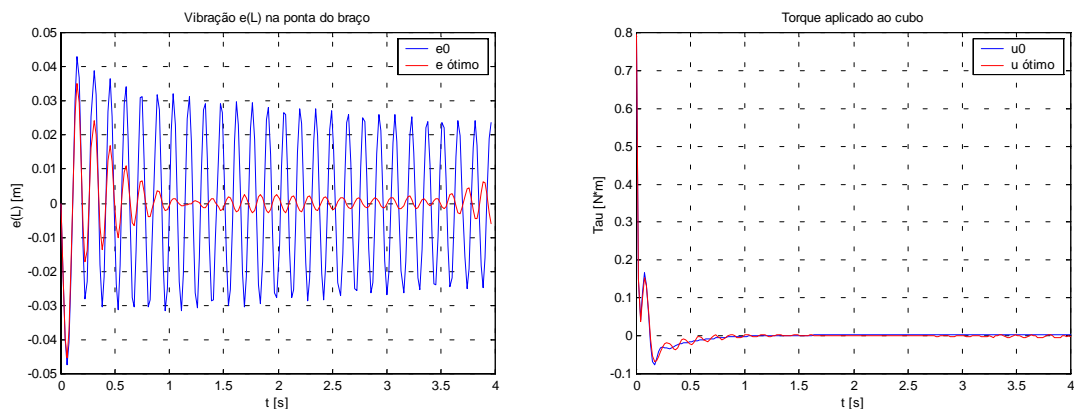


Figura 6. Vibração da ponta do braço e torque aplicado na presença de vínculos de trajetória.

7. Conclusões

O movimento de um carro móvel sobre um braço rotativo flexível foi empregado efetivamente no controle das vibrações induzidas. Esse resultado sugere novas perspectivas no estudo da atenuação de vibrações de estruturas flexíveis empregando a interação dinâmica entre as partes móveis do sistema. Os casos apresentados - juntamente com outros estudados, mas não exibidos - demonstram que a posição e a velocidade da massa podem causar tanto uma atenuação quanto um aumento das amplitudes de vibração do braço, fazendo-se necessária uma análise mais aprofundada e sistemática do Problema de Controle Ótimo e da dinâmica do sistema para inferir-se a trajetória ótima do carro, no sentido de atenuar vibrações. De qualquer modo, deduz-se que a interação entre as partes de um sistema pode ser empregada no amortecimento ativo de vibrações. Os resultados são suficientes para destacar a viabilidade e o interesse do estudo da Dinâmica e Controle de estruturas flexíveis, explorando-se o acoplamento de movimentos.

8. Agradecimentos

Os autores gostariam de expressar seus agradecimentos ao CNPq – Conselho Nacional de Pesquisa e Desenvolvimento Científico e Tecnológico, pelo apoio financeiro dado a este trabalho.

9. Referências

- Book, W.J., 1990, “Modeling, Design and Control of Flexible Manipulator Arms: a Tutorial Review”, Proceedings of the 29th Conference on Decision and Control, Honolulu, USA, pp. 500-506.
- Friedland, B., Richman, J. and Williams, D. E., 1987, “On the ‘Adiabatic Approximation’ for Design of Control Laws for Linear, Time-Varying Systems”, IEEE Transactions on Automatic Control, Vol. AC-32, No.1, pp. 62-63.
- Juang, J., Horta, L. G. and Robertshaw, H. H., 1986, “A Slewing Control Experiment for Flexible Structures”, Journal of Guidance and Control, Vol. 9, N. 5, pp. 599-607.
- Meirovitch, L., 1980, “Computational Methods in Structural Dynamics”, Alphen aan den Rijn Sijthoff, Noorthoof.
- Meirovitch, L., 1997, “Principles and Techniques of Vibrations”, Prentice-Hall.
- Oliveira, F. R. F. and Fleury, A. T., 1999, “Vibration Control of a Flexible Rotating Arm through the Movement of a Slider” (In Portuguese), Proceedings of the 15th Brazilian Congress of Mechanical Engineering, Águas de Lindóia, São Paulo, Brazil, paper AACFGB.
- Oliveira, F. R. F., 2000, “Vibration Control of a Flexible Rotating Arm through the Movement of a Sliding Mass over the Arm” (In Portuguese), MSc. Dissertation, University of São Paulo, São Paulo, Brazil.
- Oliveira, F. R. F. and Fleury, A. T., 2001, “Optimal Control of a Flexible Rotating Arm through the Movement of a Slider”, Proceedings of the 9th DINAME, Florianópolis, Santa Catarina, Brazil.
- Özguner, Ü. and Barbieri, E., 1988, “Unconstrained and Constrained Mode Expansions for a Flexible Slewing Link”, Journal of Dynamic Systems, Measurement and Control, Vol. 110, n. 4, pp. 416-421.
- Schwartz, A. and Polak, E., 1996, “Consistent Approximations for Optimal Control Problems Based on Runge-Kutta Integration”, SIAM Journal of Control Optimization, Vol. 34, 4.
- Schwartz, A., Polak, E. and Che, Y., 1997, “RIOTS_95: A MATLAB Toolbox for Solving Optimal Control Problems”, Stanford University.
- Warren, S., Voulgaris, P. and Bergman, L. 1995, “Robust Control of a Slewing Beam System”, Journal of Vibrations and Control, Vol. 1, pp. 251-257.

DYNAMICS AND CONTROL OF A FLEXIBLE ROTATING ARM THROUGH THE MOVEMENT OF A SLIDING MASS

Frederico Ricardo Ferreira de Oliveira

Mechanical Engineering Department; Escola Politécnica; University of São Paulo; P.O. Box 61548; Z.C. 05508-900; São Paulo, SP, Brazil; e-mail: fredrfo@usp.br

Agenor de Toledo Fleury

Control Systems Group; Mech&Elec Eng. Division; IPT; P.O. Box 0141; Z.C. 01064-970; São Paulo, SP, Brazil; e-mail: agfleury@ipt.br

Abstract. *A dynamical model for a flexible rotating arm carrying a sliding mass is deduced using the Extended Hamilton's Principle. This model is a coupled integro-differential system of equations, non-linear and time and space variant. In order to circumvent mathematical difficulties, substructure modeling was employed, considering the flexible beam and the moving mass separately. This second model is generic in the sense that one can include any number of autofunctions. Two distinct approaches have been considered for the mass movement. The first assumes that the mass trajectory over the arm is pre-specified during arm rotation. In a second approach, the mass positioning is also used for the control strategy, by considering one external force actuating over the mass as a second control variable, besides the torque. A LQ regulator with variable gain has been implemented for the first approach. An Optimal Control Problem is formulated for the second one. The two cases were simulated, and the results are presented and discussed. The results demonstrate that optimized coupled motions should be employed to very good vibration control of flexible structures.*

Keywords: *flexible structures, rotating arms, control, optimal control*

GERAÇÃO AUTOMÁTICA DE REGRAS DE CONTROLE PARA ALOCAÇÃO DE RECURSOS EM SISTEMAS PRODUTIVOS COM PROCESSOS CONCORRENTES

Francisco Yastami Nakamoto

Universidade de São Paulo – Escola Politécnica
francisco.nakamoto@poli.usp.br

Newton Maruyama

Universidade de São Paulo – Escola Politécnica
maruyama@usp.br

Paulo Eigi Miyagi

Universidade de São Paulo – Escola Politécnica
pemiyagi@usp.br

Diolino José dos Santos Filho

Universidade de São Paulo – Escola Politécnica
diolinos@usp.br

Resumo. *Sistemas produtivos flexíveis que executam múltiplos processos simultâneos compartilhando recursos podem apresentar indeterminismo em relação ao tempo e ao sequenciamento das atividades no contexto global. Tais sistemas podem ser classificados como Sistemas Produtivos Complexos e modelados através de Redes de Petri e suas derivadas quando abordados como sendo uma classe de Sistemas a Eventos Discretos. Neste contexto, se a alocação de recursos compartilhados não for gerenciada adequadamente, é possível ocorrer um auto-travamento ou 'deadlock' do sistema devido a falta de informação quanto ao sequenciamento do processo. Uma abordagem efetiva para este problema global realizado no sistema é o método 'deadlock avoidance' que consiste na detecção do estado pré-deadlock conjugada a uma reação que evita a evolução do sistema para o estado de travamento. Este método baseia-se no modelo GAR (Grafo de Alocação de Recursos). Este trabalho propõe uma sistematização deste método que possibilita a geração automática, através de ferramentas computacionais do modelo de controle de recursos.*

Palavras chave: *Deadlock avoidance, Redes de Petri, Controle de sistemas produtivos, Alocação de recursos.*

1. Introdução

Os sistemas produtivos flexíveis (SPF) caracterizam-se pela capacidade de produzirem uma variedade de produtos modificando suas configurações rapidamente de acordo com o planejamento da produção. Esta flexibilidade permite uma utilização mais adequada dos recursos, mas incrementa complexidade ao controle do sistema. Outro aspecto presente nos SPFs consiste no fato de evoluírem dinamicamente a partir da ocorrência de eventos e, portanto, pertencem a classe de Sistemas a Eventos Discretos (SED). Em um SED, pode ocorrer paralelismo e conflito de eventos, sendo possível inclusive haver indeterminismo quanto à ocorrência destes eventos em função do tempo (assincronismo).

A tecnologia fornece equipamentos cada vez mais eficientes, porém, esta eficiência restringe-se as atividades inerentes de cada processo. Aprimorar a eficiência dos processos de uma forma global requer um maior controle sobre a utilização dos equipamentos. O controle sobre um único processo é relativamente simples, mas em um sistema produtivo com múltiplos processos simultâneos e compartilhamento de um conjunto finito de recursos é possível a ocorrência de 'deadlock' ou travamento do sistema. Este fenômeno ocorre quando o fluxo dos processos são permanentemente impedidos e/ou quando as operações dos processos não podem ser executados. Basicamente existem três métodos para abordar o problema de 'deadlock': (i) Prevenir ('Prevent'), (ii) Evitar ('Avoid') e (iii) Detectar e Restabelecer ('Detect and Resolution'). Neste contexto, este trabalho baseia-se em um método de 'deadlock avoidance'.

No método de 'deadlock avoidance' proposto por Banaszak (1990) o algoritmo de restrições age sobre o sistema utilizando os buffer dos recursos. Viswanadham (1990) utiliza um algoritmo que age sobre o um grafo de atingibilidade do sistema. Tanto Banaszak (1990) quanto Viswanadham (1990) utilizam a Redes de Petri para modelar o sistema. Lawley (1999) propõe técnicas de monitoramento em tempo real sobre um GAR (Grafo de Alocação de Recursos) do sistema. No modelo proposto por Lawley é considerado a utilização de buffer nos recursos. Nestes trabalhos podemos verificar que a interação entre o controle sobre o processo e o controle sobre a utilização dos recursos é realizada em um mesmo plano, aumentando o grau de complexidade de acordo com o tamanho do sistema.

O método proposto por Santos Filho (2000a) para SPFs divide o sistema de controle dos processos e controle dos recursos. A estrutura hierárquica em dois níveis possibilita controlar os vários processos através de modelos de restrições (Santos Filho, 2000a) controlando como os recursos devem ser alocados durante a dinâmica dos

processos mediante a adição de regras de controle que provém um comportamento dinâmico que evita o ‘deadlock’ do sistema. O método utiliza ferramentas gráficas e matemáticas baseados em Redes de Petri para criar modelos do sistema de controle. As ferramentas utilizadas são: Enhanced Mark Flow Graph (E-MFG) (Miyagi, 2000; Santos Filho, 2000a) e o Grafo de Alocação de Recursos (GAR) (Santos Filho, 1998; Santos Filho, 2000a) associando-se regras de produção para representar o conjunto de restrições ao sistema de controle.

Portanto, o objetivo deste trabalho é apresentar um algoritmo que gere automaticamente as regras de controle, conforme o método proposto em Santos Filho (2000a), para ser implementada como parte integrante da arquitetura de uma ferramenta de auxílio ao projeto de sistemas de controle de SPFs.

2. ‘Deadlock’ em Sistemas Produtivos

Em um sistema produtivo flexível com compartilhamento de recursos finitos entre múltiplos processos, a competição de recursos pelos processos pode levar o sistema a um auto-travamento ou ‘deadlock’, conforme citado anteriormente.

O ‘deadlock’ pode ocorrer quando dois ou mais processos competem por um número limitado de recursos equivalente a situação em que processos aguardam a liberação de recursos utilizados por outros processos (Holliday, 1998). Esta descrição é aplicado em muitos sistemas onde são compartilhados recursos e informações. Por exemplo, um processo **A** aguarda a liberação do recurso alocado pelo processo **B** e o processo **B** aguarda a liberação do recurso alocado pelo processo **A**. Neste estado, só uma intervenção externa poderá tirar o sistema do estado de ‘deadlock’, conforme ilustrado na fig. (1).

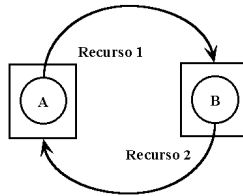


Figura 1 Exemplo de ‘deadlock’.

A interação entre processo e recurso é realizada em três etapas fundamentais: (i) Alocação de Recurso; (ii) Utilização de Recurso; (iii) Liberação do Recurso. Um processo que tenta alocar um recurso que não esteja disponível, causa um bloqueio no sistema. Existe uma diferença entre um sistema bloqueado e um sistema travado. Um sistema bloqueado aguarda até um determinado evento ocorrer, enquanto que no ‘deadlock’ o processo aguarda um evento que nunca irá ocorrer.

Em Cho (1993) são descritos os tipos de ‘deadlocks’ em sistemas produtivos como sendo:

- ‘Deadlock’ de Fluxo de Processos (‘Part Flow deadlock’): Corresponde à situação onde os processos estão em uma cadeia cíclica aguardando a liberação de recursos alocados por outros processos, exemplo fig. (2a);
- ‘Deadlock’ de Recursos (‘Processing Resource deadlock’): Corresponde à situação em que os processos estão travados devido à falta de recursos tais como: ferramentas, dispositivos de fixação, etc. (exemplo fig.(2b));
- ‘Deadlock’ de Materiais (‘Material Handler Deadlock’): É a situação onde os processos estão travados devido à falta de matéria-prima (exemplo fig. (2c)).

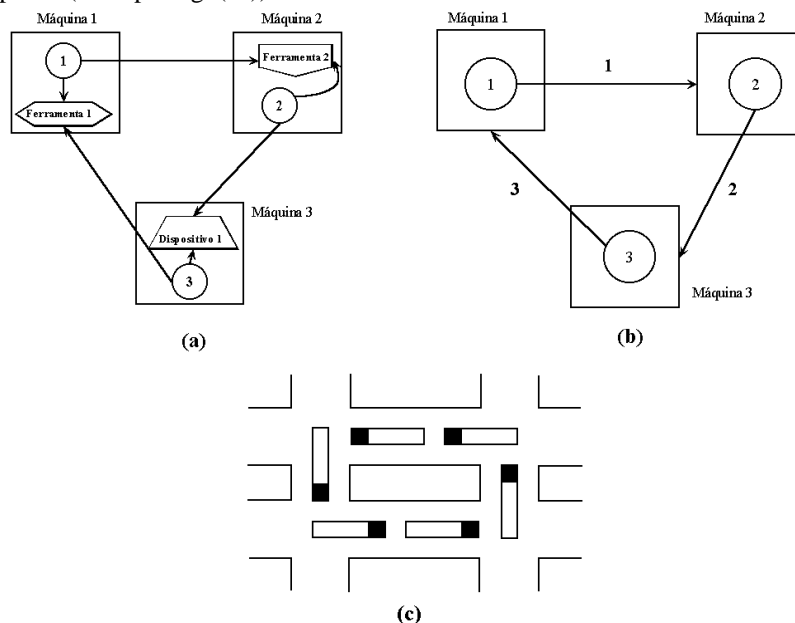


Figura 2 (a) ‘Deadlock’ de Recursos, (b) ‘Deadlock’ de Fluxo de Processo e (c) ‘Deadlock’ de Materiais.

Considerando o problema de ‘deadlock’ de fluxo de processos são necessários 4 (quatro) condições para a ocorrência de travamento onde há processos concorrentes (Santos Filho,2000a; Santos Filho,1998 e Banaszak, 1990):

- Mútua exclusão: Cada processo requisita o uso exclusivo de um recurso;
- Retenção enquanto espera: Enquanto aguarda a liberação de recursos de outros processos, um determinado processo não libera os recursos alocados por ele;
- Não há preempção: Um recurso só pode ser liberado pelo processo que o alocou primeiro;
- Espera circular: Existe uma cadeia cíclica fechada de processos aguardando a liberação dos recursos alocados por outros processos.

Neste contexto, para evitar o ‘deadlock’ em um sistema é suficiente garantir-se que pelo menos uma das condições acima nunca irá ocorrer. Em um SED as três primeiras condições são sempre verdadeiras, logo, resta apenas a condição de espera circular para ser controlada.

Basicamente existem três métodos para abordar o problema de ‘deadlock’ (Cho, 1993):

- Prevenir (Prevention): Na prevenção, um esquema estruturado do sistema garante que pelo menos uma das quatro condições para a ocorrência de ‘deadlock’ nunca ocorrerá. A esquematização é estática e portanto não podem ser alteradas em tempo real.
- Evitar (Avoidance): Consiste em uma técnica baseada em algoritmos que supervisiona os processos quanto à alocação de recursos e realiza decisões em tempo real para evitar o ‘deadlock’.
- Detectar e Restabelecer (Detection and Resolution): Consiste em uma técnica baseada em algoritmos que detectam o estado de ‘deadlock’ e realizam ações para sair deste estado. Para tirar o sistema do estado de ‘deadlock’, o método paralisa ou cancela um ou mais processos. Os processos travados podem perder parte ou todas as atividades que estavam sendo realizados durante a ocorrência do travamento.

Os dois primeiros métodos fornecem resultados satisfatórios, porém, ambos possuem um elevado custo de implementação. O último método apresenta o inconveniente de possibilitar a ocorrência de situações em que o tempo de recuperação do sistema, após a detecção de ‘deadlock’, é relativamente longo, comprometendo o desempenho global do sistema. O grau de dificuldade de implementação, utilizando-se o primeiro método, é proporcional à complexidade do sistema, assim, neste trabalho é adotada a abordagem de ‘deadlock avoidance’.

3. Utilização de Recursos e Sequenciação dos Processos em SPFs

Evitar o ‘deadlock’ é uma forma de controle essencial requerido em operações automatizadas nos sistemas de tecnologia avançada que envolvem um intenso compartilhamento de recursos. Na abordagem ‘deadlock avoidance’ deve-se determinar o estado anterior ao ‘deadlock’ e então tomar uma decisão em relação a ação a ser executada para que o sistema evolua para um estado que não determine o travamento.

3.1. Arquitetura Hierárquica de Controle

A estrutura hierárquica de controle proposta em Santos Filho (2000a) é uma divisão do controle em níveis semanticamente diferenciados (fig. (3)):

- Controle de processos: Tem por objetivo garantir a dinâmica do sequenciamento de atividades inerente a cada processo;
- Controle de recursos: Tem por objetivo ampliar o potencial de gerar diferentes estratégias de controle de utilização dos recursos considerando o aspecto de modelagem de sistemas complexos.

O controle orientado a processo atua diretamente sobre o objeto de controle. Por sua vez, em um nível superior situa-se o controle de recursos que atua e impõe restrições no controle de processos e pode monitorar o comportamento dinâmico do objeto de controle.

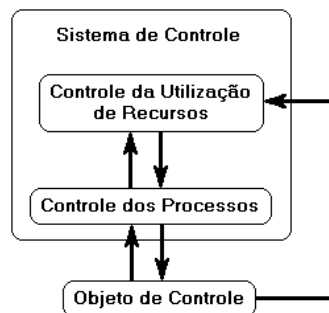


Figura 3 Arquitetura hierárquica de controle de SPFs.

A estrutura hierárquica proposta possibilita a aplicação do método de ‘deadlock avoidance’ no sistema. Uma vez que o SPFs, possuem um indeterminismo com relação ao tempo e a seqüência de utilização dos recursos, não é possível impor um determinado comportamento dinâmico global para o sistema. Para aplicar o método utiliza-se o modelo de restrições (Santos Filho, 2000), isto é, deve-se descobrir estados anteriores ao ‘deadlock’ e evitar estes estados.

3.2. Modelagem do Sistema de Controle

Os grafos possibilitam analisar e realizar simulações no modelo, fornecendo dados importantes sobre o comportamento dinâmico do sistema. Uma ferramenta eficaz na modelagem, análise e controle de SPFs são as Redes de Petri (Peterson, 1981; Holliday, 1998; Miyagi, 2000; Miyagi, 1996 e Viswanadham, 1990), um exemplo é apresentado na fig. (4). Atualmente uma série de variações de Redes de Petri original estão sendo propostos para diferentes propósitos (Holliday, 1998).

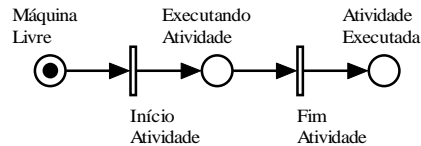


Figura 4 Exemplo de Redes de Petri.

Para este trabalho são utilizados as seguintes derivadas de Redes de Petri: (i) Enhanced Mark Flow Graph (E-MFG); (ii) Grafos de Alocação de Recursos (GAR), apresentados a seguir.

3.2.1. Enhanced Mark Flow Graph (E-MFG)

O E-MFG é uma extensão do MFG (Mark Flow Graph) (Miyagi, 2000; Santos Filho, 1998 e Miyagi, 1996). O MFG é uma rede interpretada derivada de Redes de Petri. Basicamente é composto pelos seguintes elementos estruturais: Transições, Boxes, Arcos orientados, Portas habilitadoras, inibidoras e arcos de sinal de saída (fig. (5)).

No MFG, um elemento externo pode habilitar ou inibir o disparo de uma transição de acordo com o estado do sistema real, ou ainda, enviar alguma informação para o sistema real.

No E-MFG, as marcas são individuais com atributos, capazes de controlar alterações de fluxo das marcas e seleção de tarefas associadas aos boxes. O controle das alterações de fluxo é realizado através de sinais de portas provenientes de elementos externos.

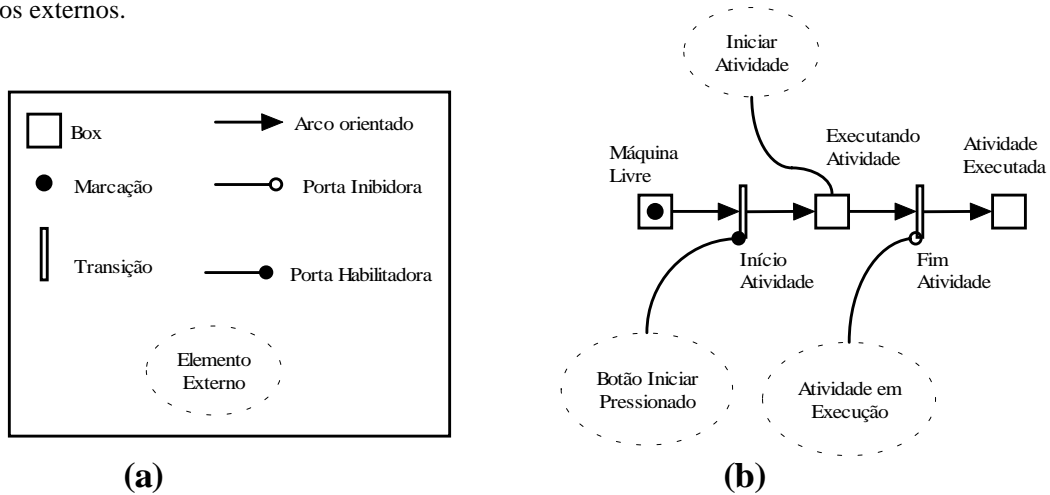


Figura 5 (a) Elementos Estruturais de MFG e (b) Exemplo de MFG.

3.2.2. Grafos de Alocação de Recurso (GAR)

O GAR é um grafo orientado onde pode ser modelado a seqüência de utilização de recursos pelos processos (Santos Filho, 1998; Santos Filho, 2000a e Santos Filho, 2000b). O GAR é constituído por nós e arcos. Um par contendo um nó e o respectivo arco de saída é denominado par de alocação e, o par contendo o nó e um arco de entrada é denominado par de requisição, como ilustra a fig. (6).

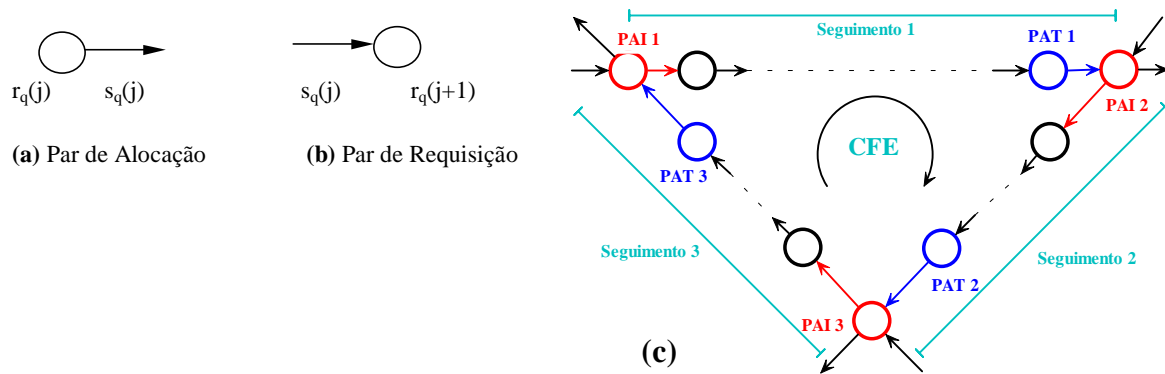


Figura 6 (a) Par de Alocação, (b) Requisição no GAR e (c) Exemplo de Modelo GAR com CFE.

No par de alocação, o nó representa um determinado recurso $r_q(j)$ e o arco de saída possui uma inscrição que representa a etapa do processo $s_q(j)$ que aloca $r_q(j)$ para ser executada; Por sua vez, em um par de requisição o nó também representa um determinado recurso $r_q(j+1)$ e o arco orientado de entrada possui uma inscrição que representa a etapa do processo $s_q(j)$ que requisita $r_q(j+1)$.

Os ciclos fechados em um GAR são denominados CFE (Ciclos Fechados de Espera) e representa compartilhamentos de recursos (exemplo fig. (6c)). Há dois tipos básicos de ciclos fechados:

- Quando a rota de um determinado processo cruza-se sobre si mesma antes de atingir a etapa final da seqüência de produção;
- Quando as rotas de processos diferentes cruzam-se antes de atingirem as etapas finais de cada seqüência de produção.

Num CFEs define-se:

- Intersecção: nó correspondente à intersecção é denominado nó de intersecção;
- Par de alocação de intersecção (PAI): Par de alocação que contém o nó;
- Par de alocação terminal (PAT): O Par de alocação que contém o arco que aponta para o nó de intersecção é denominado par de alocação terminal (PAT);
- Seguimento: conexão contínua de um determinado PAI até o próximo PAI.

4. Metodologia de Recursos Baseado no Método de ‘Deadlock Avoidance’

4.1 Método proposto

Dado o conjunto de recursos e a seqüência de utilização de recursos pelos processos, os passos do método proposto são:

- (i) Modelagem de cada processo através do GAR;
- (ii) Obtenção do modelo GAR e E-MFG global de utilização de recursos através da técnica de fusão de lugares (Santos Filho, 2000a e Santos Filho, 1998);
- (iii) Determinação dos ciclos fechados de espera (CFE) pelo modelo GAR global;
- (iv) Geração das regras de controle para alocação de recursos, baseado na abordagem ‘deadlock avoidance’ a partir dos ciclos fechados;
- (v) Tradução do GAR global para o E-MFG (Miyagi, 2000; Santos Filho, 1998), inserindo-se as regras de controle.

Os passos (i), (ii) e (iii) são apresentados em Santos Filho (2000a e 1998). Para a determinação dos ciclos fechados, é adotado o método apresentado em Santos Filho (2000a e 2000b), dentro deste contexto, a contribuição deste trabalho é focalizado no passo (iv) e (v). Cada um dos passos é detalhado acompanhado de um exemplo de aplicação.

A seguir, cada passo do método é apresentado detalhadamente, através de sua aplicação no projeto de controle de um SPF exemplo.

(i) Modelagem de cada processo através do GAR

O SPF exemplo possui um conjunto de recursos $r = \{r1, r2, r3, r4\}$, um conjunto de processos $p = \{p1, p2, p3, p4\}$ e a seqüência de utilização dos recursos de cada processo:

- $p1 = \{r1, r2, r4\}$;
- $p2 = \{r2, r3\}$;
- $p3 = \{r3, r2\}$;
- $p4 = \{r4, r2\}$.

As etapas do processo $p1$, que são $\{p1(1), p1(2), p1(3)\}$, utilizarão os recursos: $r1, r2$ e $r4$. Entre o nó de entrada (E) e o primeiro recurso (nó do recurso $r1$) constrói o arco $p1(0)$. Após o término da etapa $p1(1)$, o processo $p1$ requisita o recurso $r2$, assim, constrói-se o arco $p1(1)$ que liga o nó $r1$ com o $r2$ e assim por diante. A fig. (7) apresenta o GAR dos quatro processos do SPF exemplo.

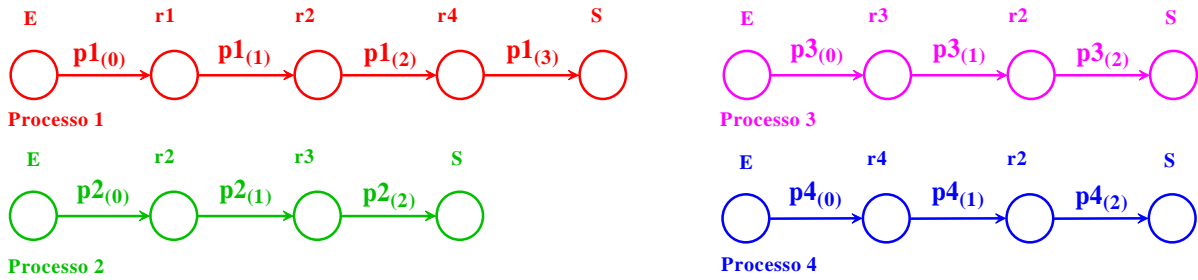


Figura 7 Modelos GAR de cada processo do SPF exemplo.

(ii) Obtenção do modelo GAR e E-MFG global

O próximo passo é a utilização da técnica de fusão de lugares (Santos Filho, 2000a e Santos Filho, 1998) para criar um modelo global da utilização de recursos, na fig. (8a) e fig. (8b) é apresenta o modelo GAR e E-MFG global, respectivamente. Do modelo GAR global, deduz o modelo E-MFG global de utilização de recursos.

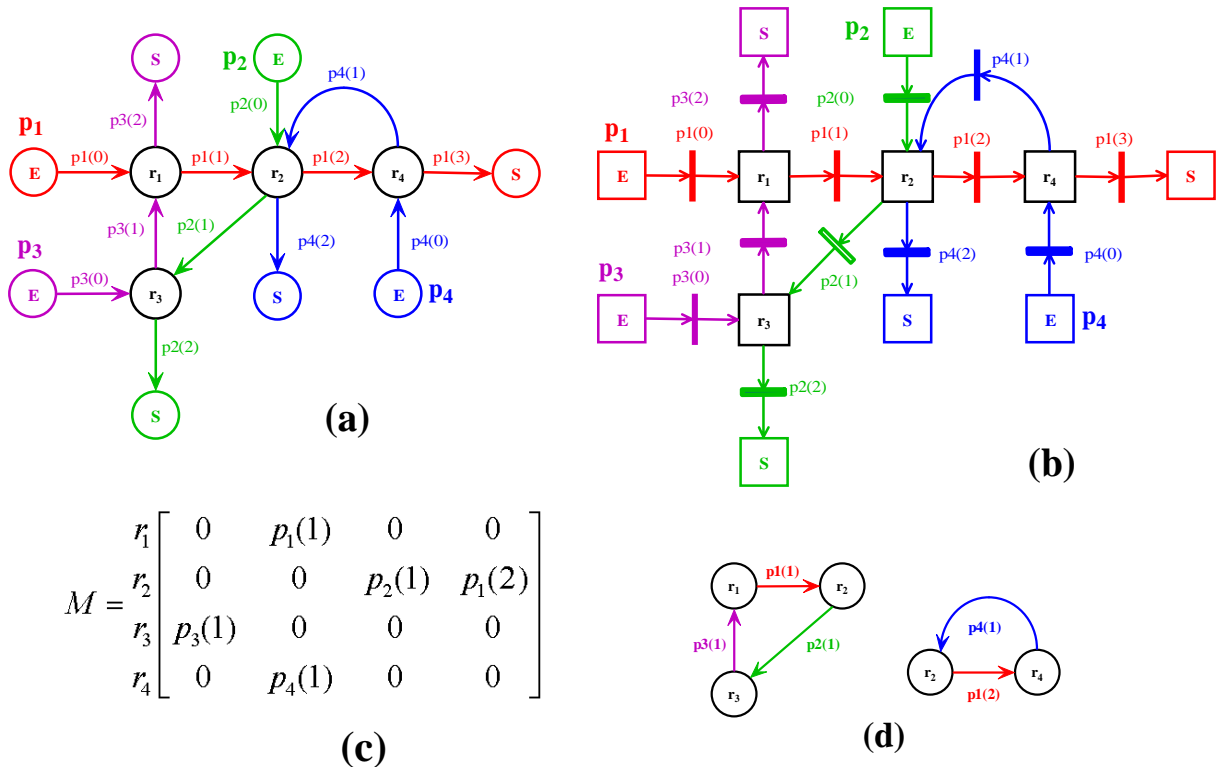


Figura 8 (a) Modelo GAR global, (b) E-MFG global, (c) Modelo GAR representado em forma de Matriz e (d) Os ciclos fechados.

(iii) Determinação dos ciclos fechados

Aplica-se o algoritmo para determinar os ciclos fechados conforme em (Santos Filho, 2000b), assim tem-se os seguintes CFE: $\{p1(2).p4(1)$ e $p1(1).p2(1).p3(1)\}$, ilustrado na fig. (8d). O algoritmo utilizado para determinar os ciclos fechados é apresentado na fig. (9a).

Este algoritmo baseia-se na potenciação da matriz M do grafo GAR. O elemento m_{ij} da matriz M representa todas as atividades que conectam R_i e R_j no modelo GAR. Quando um elemento não nulo é obtido na diagonal principal, a expressão contida neste elemento representa todos os caminhos possíveis que iniciam e retornam para R_i , indicando a existência de um ciclo fechado. Os elementos da diagonal principal devem ser eliminados para não serem recalculados novamente na determinação do próximo ciclo fechado. A potenciação é realizada até que a matriz M torne-se nula ou o

número de loops (p) seja igual a ordem da matriz (N). Não é necessário continuar com a potenciação além de N pois resultaria nos mesmos ciclos fechados já determinados.

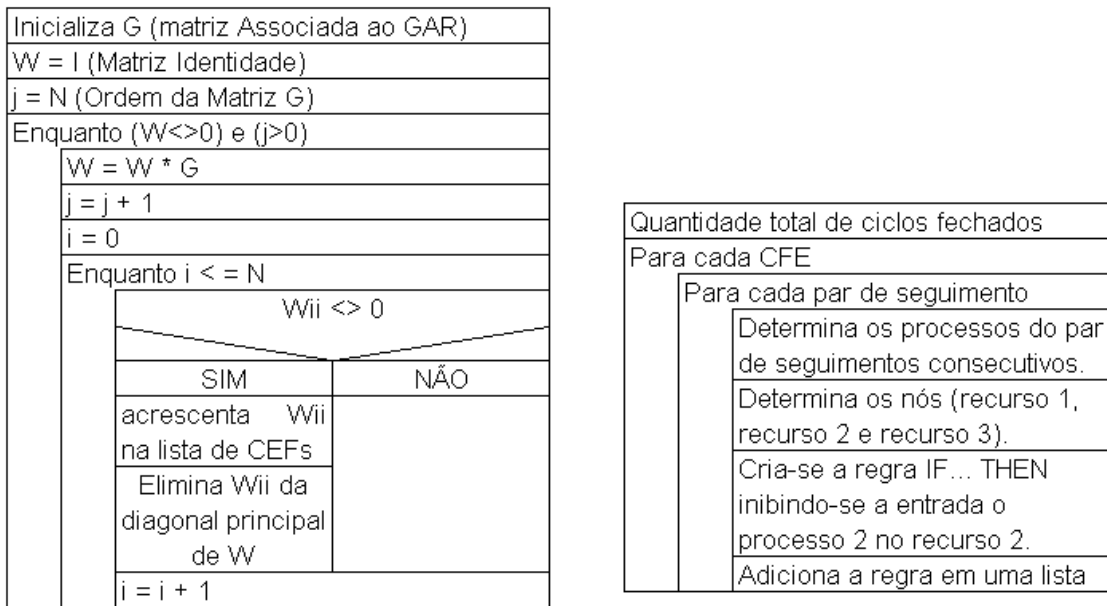


Figura 9 (a) Algoritmo para determinar os ciclos fechados. (b) Algoritmo para a criação das regras de controle.

(iv) Geração de regras de controle

O método de ‘deadlock avoidance’ verifica se existe uma situação pré ‘deadlock’, existindo esta condição, são aplicadas as regras de controle que impedem determinados processos de utilizar determinados recursos com o objetivo de evitar a quarta condição para ocorrer o ‘deadlock’.

O estado pré-deadlock é caracterizado quando do conjunto de recursos pertencentes ao CFE apenas um recurso está disponível, os demais estão sendo utilizados pelos processos que participam do ciclo. A entrada de mais um processo participante do ciclo irá causar o estado de “retém o recurso enquanto aguarda outro recurso” travando o sistema. Então, deve ser aplicada uma ação de controle inibindo a entrada desse novo processo no ciclo. As regras de controle são elaborados a partir da lista de ciclos fechados e do modelo E-MFG global de utilização de recursos aplicando-se a cada Ciclo Fechado de Espera o seguinte procedimento:

- (1) Seleciona-se dois seguimentos consecutivos. Cada seguimento possui a informação da atividade do processo (processo 1 e processo 2) e os nós de intersecção PAI e PAT (recurso 1, recurso 2 e recurso 3), estes nós representam os recursos;
- (2) Define-se a regra IF... THEN... da seguinte forma: Admitindo-se que o processo 1 esteja utilizando o recurso 1, inibi-se a entrada do processo 2 para o recurso 2. Portanto, para cada par consecutivo de seguimento é definido uma regra.
- (3) Adiciona-se cada regra obtida à lista de regras.

Este procedimento pode ser estruturado em um algoritmo. Propõe-se então, o algoritmo de geração de regras de controle de recursos baseado na abordagem ‘deadlock avoidance’, apresentado na fig. (9b).

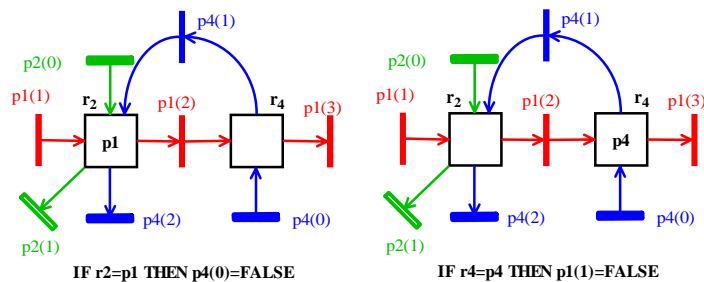


Figura 10 Exemplo do primeiro ciclo.

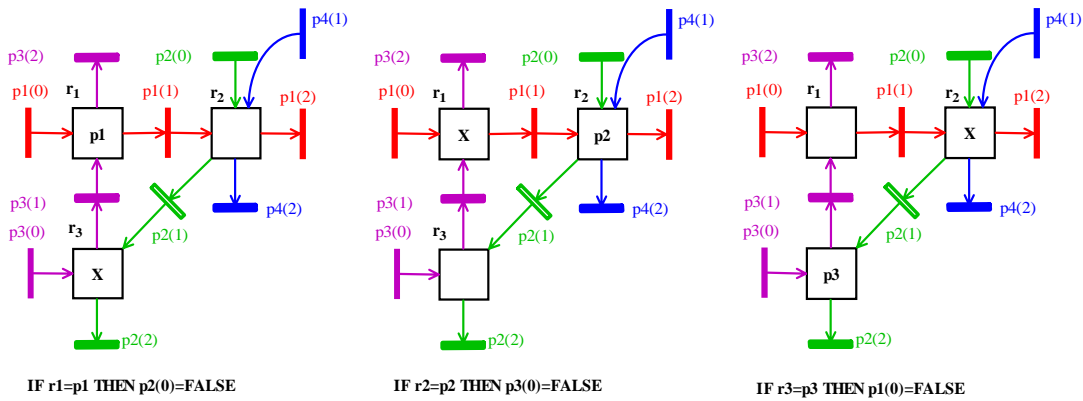


Figura 11 Exemplo do segundo ciclo.

As regras adicionais de controle determinadas em cada ciclo (fig. (10) e fig. (11)) estão apresentadas na tabela a seguir:

Ciclos Fechados de Espera	Intersecções	Estado anterior ao 'deadlock' (r1, r2, r3, r4)	Regras adicionais de controle	
1	p1(2).p4(1)	r1 e r4	(?, p1, ?, __)	IF (r2=p1) THEN p4(0)=FALSE
			(?, __, ?, p4)	IF (r4=p4) THEN p1(1)=FALSE
2	p1(1).p2(1).p3(1)	r1, r2 e r3	(p1, __, X, ?)	IF (r1=p1) THEN p2(0)=FALSE
			(X, p2, __, ?)	IF (r2=p2) THEN p3(0)=FALSE
			(__, X, p3, ?)	IF (r3=p3) THEN p1(0)=FALSE

Legenda: 'X' = Ocupado, '?' = Indiferente, '__' = Vazio

Tabela 1 Regras adicionais de controle do SPF exemplo.

4.2. Geração Automática de Regras de Controle

A geração automática de regras de controle é baseada em algoritmos utilizando o método proposto por Santos Filho (Santos Filho, 2000a). Os algoritmos serão utilizados em uma ferramenta de auxílio ao projeto de sistemas de controle. Esta ferramenta é composta por módulos, para o presente trabalho foi descrito apenas o módulo de geração das regras. Os dados de entrada da ferramenta são as informações sobre os recursos disponíveis e os processos que utilizarão estes recursos. O método é aplicado sobre as informações e através de simulação, a ferramenta fornecerá um conjunto de resultados para análise e avaliação do sistema em estudo.

A ferramenta proposta é baseada em módulos conforme apresentada na fig. (12).

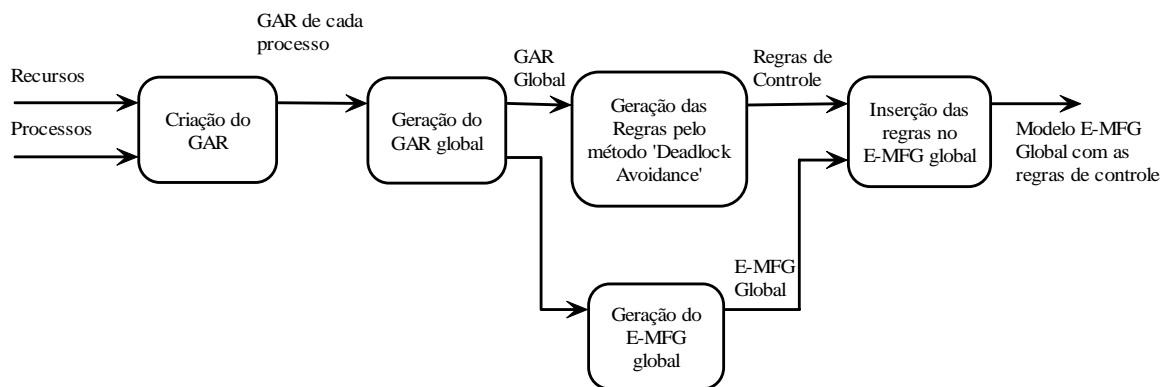


Figura 12 Diagrama de fluxo de dados da arquitetura da ferramenta.

5. Conclusão

A divisão hierárquica do sistema de controle possibilita monitorar o sistema global de forma relativamente simples. O controle sobre os processos estando caracterizados e definidos focaliza-se a alocação de recursos com restrições de utilização de acordo com as regras de controle.

Este trabalho apresenta um módulo da ferramenta de auxílio ao projeto de controle de SPFs com a sistematização do método proposto viabilizando a geração automática de regras de controle para alocação de recursos. Este protótipo

em desenvolvimento objetiva a geração dos modelos de controle e sua análise por meio da simulação. A sistematização do procedimento de modelagem da utilização dos recursos e do procedimento de obtenção de regras de controle de recursos é o primeiro passo na geração automática dos modelos de controle de SPFs.

Agradecimentos

Os autores agradecem a CAPES e CNPq pelo apoio a este trabalho.

Referências

- Cho, H., 1993, "An Intelligent Workstation Controller for Computer Integrated Manufacturing", Doctor Dissertation, Texas A&M University, pp.83-88.
- Banaszak, Z.A., Krogh, B.H., 1990, "Deadlock Avoidance in Flexible Manufacturing Systems with concurrently Competing Process Flows", IEEE Transactions on Robotics and Automation, vol.6, no.6.
- Holliday, J.L., Abbadi, A.E., 1998, "Distributed Deadlock Detection", Encyclopedia of Distributed Computing, Kluwer Academic Publishers.
- Lawley, M.A., 1999, "Deadlock Avoidance for Production Systems with Flexible Routing", IEEE Transactions on Robotics and Automation, Vol. 15, No. 3.
- Miyagi, P.E., 1996, "Controle Programável – Fundamentos do Controle de Sistemas a Eventos Discretos", Ed. Edgard Blücher Ltda, São Paulo, Brazil, pp. 1-5, 71-148.
- Miyagi, P.E., Maruyama, N., Santos Filho, D.J., 2000, "An Anthropocentric Approach for the Design of Production Systems Control", Proceeding of Mechatronics 2000. 7th Mechatronics Forum International Conference, Atlanta.
- Peterson, J.L., 1981, "Petri Net Theory and the Modeling of System", Englewood Cliffs, N.J., Prince-Hall, pp.1-78.
- Santos Filho, D.J., 2000 (a), "Aspectos do Projeto de Sistemas Produtivos", Tese (Livre Docência), Universidade de São Paulo – Escola Politécnica – Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos, pp.8-21, 61-95, 101-111.
- Santos Filho, D.J., Motohashi Matsusaki, C.T., Nelli Silva, E.C., Miyagi, P.E., 2000 (b), "Automatic Generation of Metamodels to Design Production Systems Control", Proceeding of Mechatronics 2000. 7th Mechatronics Forum International Conference, Atlanta.
- Santos Filho, D.J., 1998, "Controle de Sistemas Antropocêntricos de Produção Baseados em Redes de Petri Interpretadas", Tese (Doutorado), Universidade de São Paulo – Escola Politécnica – Departamento de Engenharia Mecatrônica e de Sistemas Mecânicos, pp. 84-119.
- Viswanadham, N., Narahari, Y., Johnson, T.L., 1990, "Deadlock Prevention and Deadlock Avoidance in Flexible Manufacturing Systems Using Petri Net Models", IEEE Transactions on Robotics and Automation, vol.6, No.6.

AUTOMATIC GENERATION OF CONTROL RULES FOR RESOURCE ALLOCATION IN PRODUCTION SYSTEMS WITH CONCURRENT PROCESSES

Francisco Yastami Nakamoto

University of São Paulo, Escola Politécnica
francisco.nakamoto@poli.usp.br

Newton Maruyama

University of São Paulo, Escola Politécnica
maruyama@usp.br

Paulo Eigi Miyagi

University of São Paulo, Escola Politécnica
pemiyagi@usp.br

Diolino José dos Santos Filho

University of São Paulo, Escola Politécnica
diolinos@usp.br

Abstract. *Flexible productive systems which execute multiple simultaneous processes using the same resources set is not possible to determine when which events occur regarding time and the sequence of the activities in the global context. Such systems can be classified as Complex Productive Systems and modeled through Petri's Net and its derivatives. In this context, if the allocation of shared resources is not managed adequately, is possible to occur deadlock of the system due to information lack regarding the sequence of the activities of the process. An effective approach for this global problem accomplished in the system is the method deadlock avoidance that consists in the detection of the previous state to deadlock conjugated to a reaction that avoids the evolution of the system for the state deadlock indeed. This method it bases on model RAG (Resource Allocation Graph). This work proposes an operations sequence of this method that enables the automatic generation, through computational tools of the resources control model.*

Keywords. *Deadlock avoidance, Petris Net, Production system control, resource allocation.*

CONTROLE PREDITIVO GENERALIZADO PARA SISTEMAS DE REFRIGERAÇÃO POR COMPRESSÃO DE VAPOR

José Maria Gálvez, Ph.D.

Departamento de Engenharia Mecânica, Universidade Federal de Minas Gerais, Brasil.
Av. Antônio Carlos 6627, Pampulha, 31.270-901 Belo Horizonte, MG, Brasil.
Fone: +55 31 3499-5236. Fax: +55 31 3443-3783.
E-mail: jmgm@dedalus.lcc.ufmg.br

Agostinho Gomes da Silva, M.Sc.

Departamento de Engenharia Mecânica, Universidade Federal de Minas Gerais, Brasil.
Av. Antônio Carlos 6627, Pampulha, 31.270-901 Belo Horizonte, MG, Brasil.
Fone: +55 31 3499-5073. Fax: +55 31 3443-3783.
E-mail: ags@demec.ufmg.br

Resumo. Máquinas de refrigeração são sistemas de múltiplas entradas e múltiplas saídas, variantes no tempo, não lineares e cujas entradas ou variáveis manipuladas apresentam saturação e restrições. Técnicas clássicas de controle tais como controladores liga-desliga e PID não são adequadas para lidar com o acoplamento cruzado existente entre as entradas e saídas do sistema. Neste contexto, novas técnicas de controle que permitam o controle independente da potência frigorífica e do superaquecimento estão sendo atualmente investigadas. Algoritmos de controle preditivo foram originalmente propostos para lidar com plantas não lineares complexas. Devido ao seu custo computacional elevado, a aplicação desses algoritmos ficou restrita a grandes plantas industriais. Atualmente, com a aparição de novos e poderosos computadores de pequeno porte, a implementação de algoritmos de controle preditivo tornou-se viável também para pequenas plantas. Este é o caso dos modernos sistemas de refrigeração. Este artigo apresenta um esquema de controle preditivo aplicado a sistemas de refrigeração baseados em compressão de vapor. A estrutura do controlador proposto foi projetada visando simplificar a sua posterior implementação em micro-controladores. O principal objetivo é manter condições de conforto térmico perante cargas variantes no tempo. O controlador proposto representa uma solução para o controle independente da potência frigorífica mantendo o grau de superaquecimento do evaporador próximo a seu valor ótimo inclusive durante os transitórios do sistema. Resultados de simulação são apresentados para ilustrar o desempenho do controlador.

Palavras Chave: Controle Preditivo, Controle Multivariável, Sistemas de Refrigeração.

1. Introdução

Com a crescente escassez das fontes de energia não renováveis e com o conseqüente aumento do custo da energia, o desenvolvimento de novos equipamentos que apresentem um melhor desempenho de consumo energético tem-se tornado uma prioridade para a comunidade técnico-científica. A maior eficiência energética dos atuais processos industriais deverá ser alcançada através da modernização dos atuais equipamentos e do desenvolvimento de novas técnicas de automação e controle. Um caso especialmente importante, do ponto de vista de consumo de energia, são os sistemas de aquecimento e refrigeração em instalações domésticas, comerciais e industriais os quais são responsáveis por aproximadamente 50% do consumo energético mundial (Imbabi, 1990).

Nos sistemas convencionais de controle de máquinas de refrigeração de pequeno porte a temperatura no interior da câmara frigorífica é monitorada por um termostato e comparada com a referência desejada; o sinal resultante desta comparação atua em um relê que liga ou desliga o compressor (Silva, 2000). Apesar da vantagem do baixo custo, os sistemas de controle liga-desliga, não são eficientes do ponto de vista de consumo energético (Rocha, 1995). Uma das causas principais do baixo rendimento energético deste tipo de controlador está relacionada com o seu funcionamento intermitente. Devido à característica liga-desliga da operação do compressor, a temperatura na câmara frigorífica oscila em torno de um valor médio o que provoca, de forma alternada, sobre-resfriamento e sobreaquecimento do ambiente alvo a cada ciclo de operação; acarretando desta forma um consumo desnecessário de energia.

No caso de sistemas de refrigeração de médio e grande porte, uma melhoria substancial da eficiência energética de tais sistemas tem sido obtida através da utilização de válvulas eletrônicas e compressores de velocidade variável. No entanto, a verdadeira causa do desperdício energético nestes sistemas não está na máquina frigorífica propriamente dita mas sim na forma como estes sistemas são operados na presença de cargas térmicas variantes.

De forma a obter um melhor aproveitamento da eficiência térmica dos sistemas de refrigeração, o sensor de temperatura do sistema deve estar apropriadamente localizado no interior do ambiente alvo (usualmente distante da fonte de potência frigorífica), isto causa o aparecimento de atrasos de tempo na malha de controle, os quais, em geral, não podem ser apropriadamente compensados através de técnicas convencionais.

Os esquemas tradicionais de controle de máquinas frigoríficas (tais como os controladores liga-desliga e os controladores PID), apesar de amplamente difundidos devido ao seu baixo custo, não são adequados para o controle de sistemas com atraso.

Do ponto de vista de sistemas de controle, máquinas de refrigeração são sistemas MIMO (multi-input - multi-output) com acoplamento cruzado entre as suas variáveis de entrada e saída e freqüentemente de fase não-mínima. São sistemas que apresentam restrições nas variáveis manipuladas e que geralmente possuem cargas com características variantes no tempo. Estas características dos sistemas de refrigeração colocam uma forte restrição nas tentativas de

melhorar o aproveitamento energético das máquinas frigoríficas através de controladores convencionais. Conseqüentemente, o desenvolvimento de novas estratégias de operação e controle de sistemas de refrigeração se faz necessário (Rocha, 1995). A obtenção de um sistema de refrigeração energeticamente eficiente requer portanto estratégias de controle capazes de lidar simultaneamente com tais características (Gálvez e Silva, 2000).

Controladores preditivos foram inicialmente aplicados a plantas industriais complexas (Richalet et ali., 1976 e 1978). O alto custo da implementação computacional dos algoritmos de controle preditivo restringiu as suas primeiras aplicações a plantas industriais de grande porte. Atualmente, com a diminuição do custo dos modernos sistemas computacionais, a aplicação de algoritmos de Controle Preditivo Baseado em Modelo (MBPC ou MPC) a plantas de pequeno porte tornou-se economicamente viável (Qin e Bagdwell, 1997). Este é o caso dos sistemas modernos de refrigeração, os quais são relativamente pequenos e apresentam dinâmica suficientemente complexa para justificar o uso de esquemas de controle preditivo.

Este trabalho apresenta uma estratégia de controle preditivo, utilizando o algoritmo GPC (Generalized Predictive Controller) na configuração SISO (single-input – single-output), aplicado ao controle de uma máquina de refrigeração baseada em compressão de vapor. Neste caso, os objetivos de controle foram especificados de maneira a:

- a) Controlar as condições de conforto no ambiente alvo na presença de cargas térmicas variantes no tempo,
- b) Manter a eficiência energética do sistema constante (máxima).

Desde que o grau de superaquecimento é uma medida da eficiência térmica da máquina de refrigeração, as especificações de controle poderão ser atingidas através do controle simultâneo da potência frigorífica e do superaquecimento do sistema. Adicionalmente, e de maneira a tornar economicamente viável a aplicação da técnica de controle proposta a sistemas de refrigeração de pequeno porte, o algoritmo de controle foi projetado visando a simplificação da sua posterior implementação computacional. Finalmente, será mostrado através de simulação que o esquema proposto representa uma solução computacional simples para o controle simultâneo e independente da velocidade do compressor e da abertura da válvula de expansão.

2. A Máquina de Refrigeração

A Figura (1) mostra um sistema típico de refrigeração por compressão de vapor. Neste caso, Q_1 corresponde ao calor retirado do ambiente alvo pela expansão do fluido refrigerante e Q_2 representa o calor gerado na compressão do fluido refrigerante.

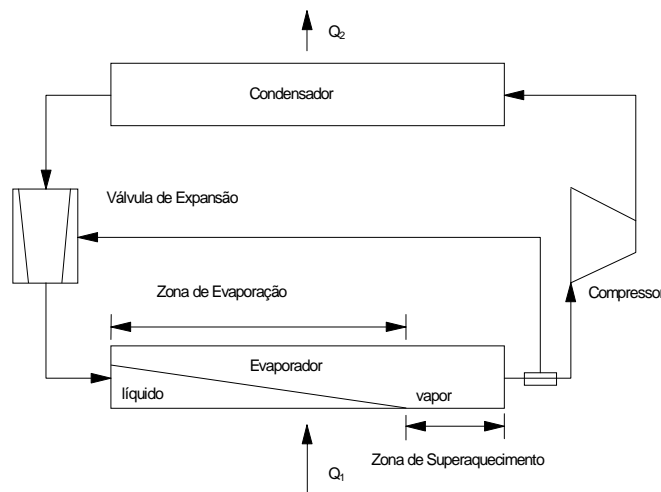


Figura 1. Sistema de Refrigeração por Compressão de Vapor.

A dinâmica do sistema é definida por uma função de transferência matricial da forma

$$[G(s)] = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \tag{1}$$

As variáveis de entrada do sistema são a vazão mássica, $MFR(s)$, e a vazão volumétrica, $VFR(s)$. As variáveis de saída são o superaquecimento, $\Delta T(s)$, e a potência frigorífica, $Q_1(s)$. A potência frigorífica, $Q_1(s)$, é ajustada de acordo com a demanda térmica. As relações entre as entradas e saídas do sistema são dadas por

$$\begin{bmatrix} \Delta T(s) \\ Q_1(s) \end{bmatrix} = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \begin{bmatrix} MFR(s) \\ VFR(s) \end{bmatrix} \tag{2}$$

Tanto o superaquecimento, $\Delta T(s)$, quanto a potência frigorífica, $Q_1(s)$, dependem da vazão do fluido refrigerante no interior do evaporador a qual pode ser controlada através da abertura ou fechamento da válvula de expansão e/ou do aumento ou diminuição da velocidade do compressor. As funções $G_{11}(s)$ e $G_{21}(s)$ determinam a influência da vazão mássica na entrada do evaporador sobre o grau de superaquecimento e sobre a potência frigorífica, respectivamente. De forma semelhante, as funções $G_{12}(s)$ e $G_{22}(s)$ determinam a influência da vazão volumétrica na saída do evaporador sobre o grau de superaquecimento e sobre a potência frigorífica, respectivamente. A Figura (2) apresenta um diagrama de blocos simplificado de uma máquina de refrigeração mostrando as variáveis de entrada e saída do evaporador.

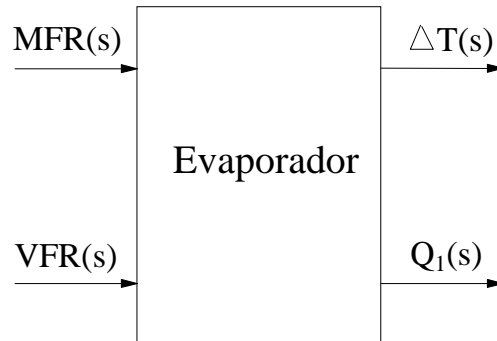


Figura 2. Variáveis de Entrada e Saída do Evaporador.

A Figura (3) mostra o acoplamento cruzado, representado pelas funções $G_{12}(s)$ e $G_{21}(s)$, entre as variáveis de entrada e saída do evaporador.

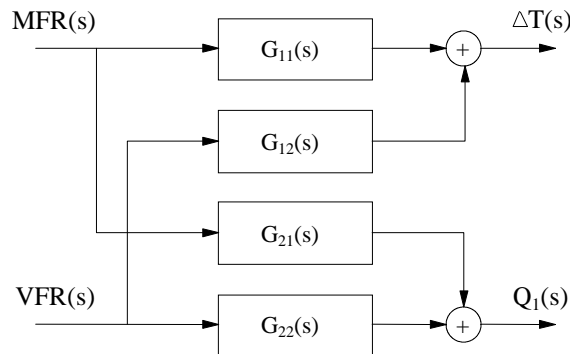


Figura 3. Acoplamento Cruzado entre Entradas e Saídas.

3. O Modelo da Máquina de Refrigeração

Diversos modelos computacionais, lineares e não-lineares de sistemas de refrigeração têm sido propostos pela comunidade técnico-científica; entre outros é relevante mencionar os trabalhos de Koury (1998), Machado (1996), Rocha (1995) e Outtagarts (1994). O modelo utilizado neste artigo é baseado no trabalho de Machado (1996). Neste caso, as funções de transferências correspondentes à Eq.(1) tem a forma:

$$G_{11}(s) = -5.62 \frac{1}{45s+1} \tag{3a}$$

$$G_{12}(s) = 2.49 \frac{(-70s+1)}{(59.52s+1)} \tag{3b}$$

$$G_{21}(s) = 33.89 \frac{(-36.37s+1)}{(25.65s+1)(67.79s+1)} \tag{3c}$$

$$G_{22}(s) = 22.20 \frac{(630s+1)}{(80s+1)(90s+1)} \tag{3d}$$

substituindo valores na Eq.(2) obtemos

$$\begin{bmatrix} \Delta T(s) \\ Q_1(s) \end{bmatrix} = \begin{bmatrix} \frac{-5.62}{(45s+1)} & \frac{2.49(-70s+1)}{(59.52s+1)} \\ \frac{33.89(-36.37s+1)}{(25.65s+1)(67.79s+1)} & \frac{22.20(630s+1)}{(80s+1)(90s+1)} \end{bmatrix} \begin{bmatrix} MFR(s) \\ VFR(s) \end{bmatrix} \quad (3e)$$

As funções de transferência $G_{12}(s)$ e $G_{21}(s)$ representam o acoplamento cruzado entre as entradas e saídas da planta frigorífica. Num caso hipotético (o sistema sem acoplamento cruzado), a abertura da válvula de expansão seria utilizada para regular o superaquecimento e a velocidade do compressor seria usada para regular a potência frigorífica. Nesse caso, as funções de transferência $G_{12}(s)$ e $G_{21}(s)$ seriam nulas. Entretanto, na prática, existe um forte acoplamento cruzado entre as entradas e saídas do sistema, de tal forma que os efeitos de $G_{12}(s)$ e $G_{21}(s)$ não podem ser desprezados. Este acoplamento cruzado é a principal dificuldade no projeto e implementação dos algoritmos convencionais de controle.

No regime permanente do sistema, os efeitos do acoplamento cruzado dependem dos valores relativos dos ganhos estáticos das funções de transferência $G_{11}(s)$, $G_{12}(s)$, $G_{21}(s)$ e $G_{22}(s)$. No regime transitório, os efeitos deste acoplamento dependem da posição relativa dos pólos e zeros das quatro funções de transferência do sistema. Em particular, a função de transferência $G_{12}(s)$ sendo própria e de fase não-mínima caracteriza uma forte influencia da vazão volumétrica sobre o superaquecimento o que dificulta, em última análise, a ação do algoritmo de controle e principalmente a estabilidade do sistema em malha fechada.

A Figura (4) apresenta a resposta do sistema a degraus unitários aplicados, não simultaneamente, às entradas da planta (abertura da válvula de expansão e aumento da velocidade do compressor). Neste caso, o acoplamento cruzado representado por $G_{12}(s)$ e $G_{21}(s)$ fica evidente nas respostas mostradas no primeiro e terceiro quadrante da Fig.(4). O primeiro quadrante da Fig.(4) mostra o efeito não desejado no superaquecimento do sistema devido à aplicação de um degrau na velocidade do compressor. O terceiro quadrante da Fig.(4) mostra a variação da potência frigorífica gerada pelo sistema provocada pela aplicação de um degrau na abertura da válvula de expansão.

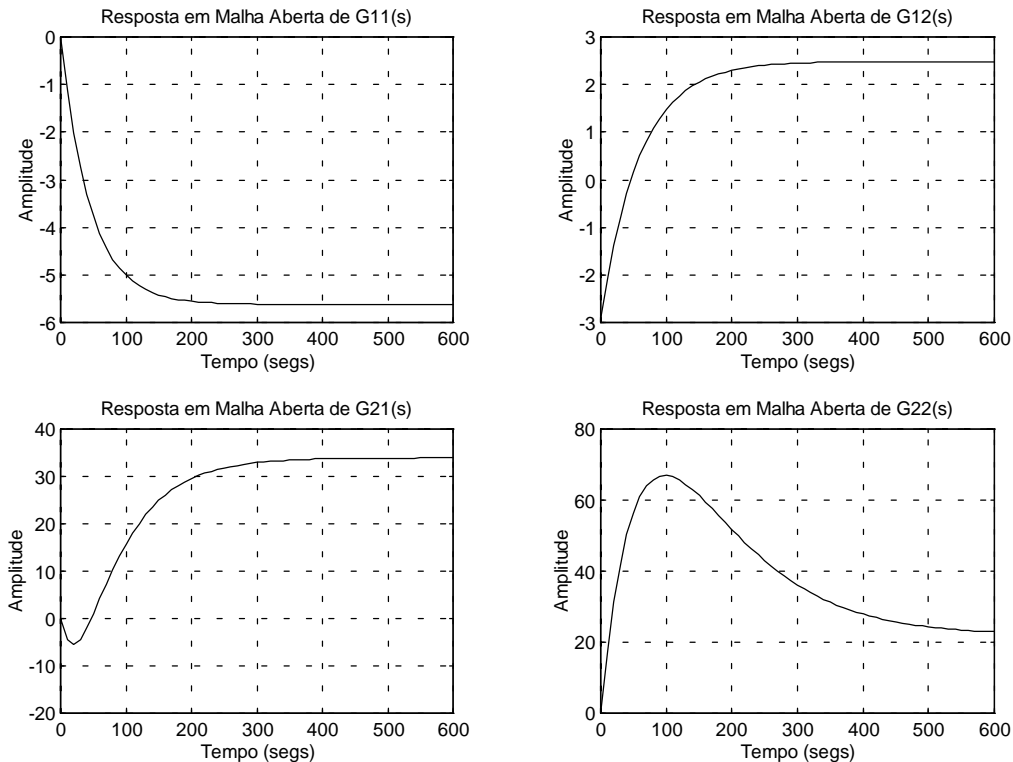


Figura 4. Resposta do Sistema em Malha Aberta a Entradas em Degrau Unitário.

A Figura (5) apresenta as respostas em frequência (magnitude) do sistema em malha aberta. Pode-se observar que todos os pólos e zeros da planta estão localizados em torno da frequência de 10^{-2} rd/s. Isto caracteriza uma forte interação cruzada entre as variáveis de entrada e saída da planta também durante a resposta transitória do sistema.

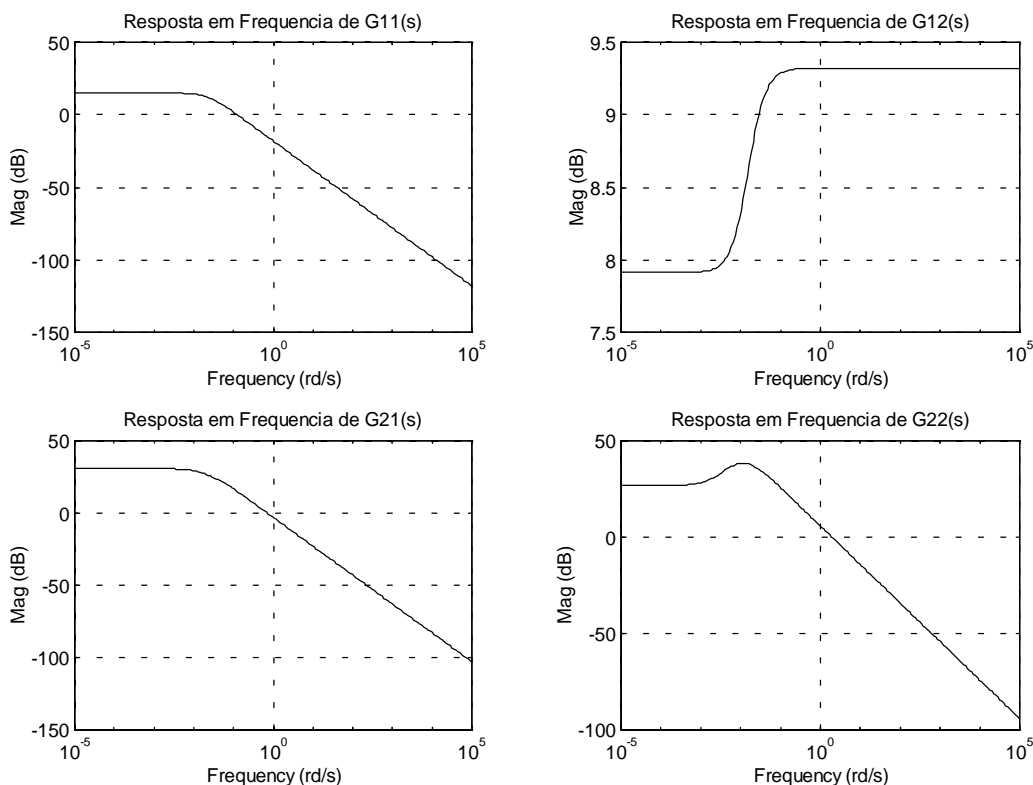


Figura 5. Resposta em Frequência do Sistema em Malha Aberta.

Deve também ser observado que as respostas no tempo, Fig.(4), e na frequência, Fig.(5), de $G_{12}(s)$ indicam que o superaquecimento (e não a potência frigorífica como seria desejado) responde instantaneamente a variações na velocidade do compressor. A consequência direta deste fato é mais uma dificuldade no projeto e implementação (fechamento) de uma segunda malha de controle para a potência frigorífica utilizando técnicas convencionais de controle.

Finalmente, cabe mencionar que no caso da planta utilizada, os números de condicionamento das matrizes de controlabilidade e observabilidade foram da ordem de 10^4 e 10^9 , respectivamente, indicando que num sistema de controle por realimentação de estados, o esforço de controle seria excessivamente elevado.

4. O Controlador Preditivo Generalizado (GPC)

A partir dos trabalhos de Clarke et al., 1987(a) e 1987(b), o Controlador Preditivo Generalizado (GPC) tornou-se uma das técnicas de controle preditivo mais populares, tanto no meio industrial quanto no meio acadêmico. A extensão do GPC para o caso multivariável foi inicialmente apresentada por Kinnaert (1989). O algoritmo GPC pode ainda ser implementado nas formas adaptativa e não adaptativa.

O GPC tem sido aplicado com sucesso em um grande número de aplicações industriais, apresentando bom desempenho e um certo grau de robustez. Seus proponentes defendem que o controlador GPC é a solução para uma grande variedade de problemas de controle, podendo ser aplicado a sistemas de fase não-mínima, a sistemas que incluem tempo de atraso variável ou desconhecido, a sistemas que possuem pólos fracamente amortecidos e mesmo a plantas instáveis em malha aberta.

O algoritmo GPC considera um modelo linearizado da planta operando em torno de um ponto de equilíbrio. O modelo, CARIMA (Controlled Auto-Regressive Integrated and Moving-Average), relaciona a saída da planta com as variações do sinal de controle.

A idéia básica do algoritmo GPC é o cálculo, baseado no modelo do processo, da seqüência futura do sinal de controle de forma a minimizar uma função de custo a qual é definida sobre um intervalo de tempo denominado horizonte de predição.

O índice a ser otimizado é o valor esperado de uma função de custo quadrática, a qual avalia a distância (no horizonte de predição) entre as saídas previstas do sistema e a seqüência de referência futura, somada a uma função quadrática que avalia o esforço de controle.

Por questão de espaço omitiremos neste trabalho uma descrição completa do desenvolvimento do algoritmo GPC, a qual encontra-se detalhada na literatura (Camacho e Bordons, 1999; Clarke et al. 1987(a) e 1987(b), etc.). A seguir, a formulação do algoritmo GPC para o caso monovariável.

Neste caso, o modelo CARIMA é definido por

$$A(q^{-1})y(t) = B(q^{-1})u(t-d-1) + C(q^{-1})\frac{\xi(t)}{\Delta(q^{-1})} \quad (4a)$$

ou

$$A(q^{-1})y(t) = B(q^{-1})q^{-d}u(t-1) + C(q^{-1})\frac{\xi(t)}{\Delta(q^{-1})} \quad (4b)$$

onde o termo q^{-d} corresponde ao tempo morto da planta; o termo $\xi(t)$ representa ruído branco de média zero; $\Delta(q^{-1})$ é o operador diferença, $\Delta(q^{-1}) = 1 - q^{-1}$; e finalmente, $A(q^{-1})$, $B(q^{-1})$ e $C(q^{-1})$ são polinômios no operador atraso unitário q^{-1} da forma:

$$A(q^{-1}) = 1 + \sum_{i=1}^{na} a_i q^{-i} = 1 + a_1 q^{-1} + \dots + a_{na} q^{-na} \quad (4c)$$

$$B(q^{-1}) = \sum_{i=0}^{nb} b_i q^{-i} = b_0 + b_1 q^{-1} + \dots + b_{nb} q^{-nb} \quad (4d)$$

$$C(q^{-1}) = 1 + \sum_{i=1}^{nc} c_i q^{-i} = 1 + c_1 q^{-1} + \dots + c_{nc} q^{-nc} \quad (4e)$$

O algoritmo GPC consiste em aplicar uma seqüência de controle que minimiza o valor esperado de uma função objetivo da forma:

$$J_{GPC} = E \left\{ \sum_{j=N_1}^{N_2} [\hat{y}(t+j) - w(t+j)]^2 + \sum_{j=1}^{N_u} \lambda(j) [\Delta u(t+j-1)]^2 \right\} \quad (5)$$

onde:

- $\hat{y}(t+j|t)$ é o valor no instante t da predição da saída do sistema para o instante $t+j$;
- $w(t+j)$ é a trajetória de referência futura;
- N_1 é o horizonte mínimo de custo;
- N_2 é o horizonte máximo de custo (horizonte de predição);
- N_u é o horizonte de controle;
- $\lambda(j)$ é a seqüência de ponderação do sinal de controle.

$\Delta u(t)$ corresponde aos incrementos futuros do sinal de controle dentro do horizonte de controle. É obtido através da minimização da função de custo definida pela Eq.(5). Deve ser observado que o sinal de controle enviado ao processo no instante t é dado por:

$$u(t) = u(t-1) + \Delta u(t) \quad (6)$$

isto é, o sinal de controle enviado ao sistema no instante atual e é dado pelo soma do sinal de controle enviado ao sistema no instante anterior, $u(t-1)$, mais o incremento do sinal de controle calculado no instante atual, $\Delta u(t)$.

5. A Estrutura do Controlador Proposto

A Figura (6) apresenta o diagrama de blocos do controlador preditivo proposto. A estrutura do controlador consiste em duas malhas independentes de controle, uma para o superaquecimento e outra para a potência frigorífica. O algoritmo de controle em cada malha é do tipo GPC SISO (single input – single output). Cada um dos controladores GPC controla o superaquecimento e a potência frigorífica de maneira independente.

Quando em malha fechada, o controlador GPC1 controla o superaquecimento (a malha da função de transferência $G_{11}(s)$) atuando na válvula de expansão e o controlador GPC2 controla a potência frigorífica (a malha da função de transferência $G_{22}(s)$) atuando na velocidade do compressor.

A principal vantagem desta configuração é que as operações computacionais do algoritmo de controle são todas escalares evitando-se assim a utilização de sub-rotinas matriciais em tempo real simplificando desta forma a sua implementação em micro-controladores.

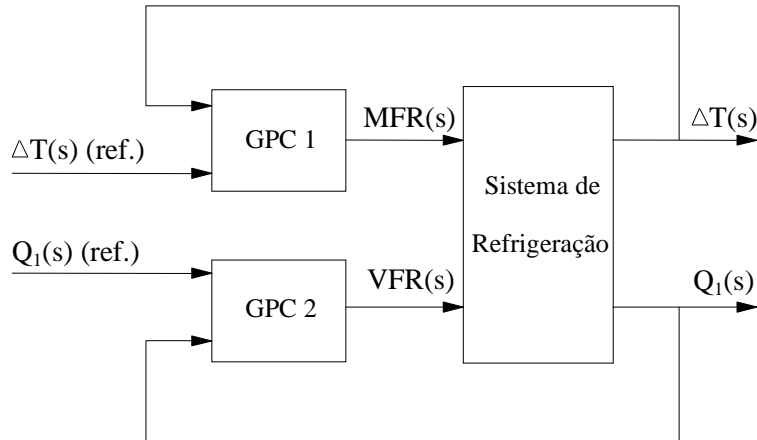


Figura 6. Esquema de Controle Preditivo para o Sistema de Refrigeração.

O desempenho do algoritmo de controle foi testado em simulação. Neste caso, um período de amostragem T_s igual a 1 segundo foi adotado para discretizar as quatro funções de transferências da planta e implementar o modelo CARIMA. O valor da taxa de amostragem foi escolhido de forma que a dinâmica mais rápida da planta fosse corretamente percebida pelo controlador GPC correspondente.

A Equação (7) apresenta a função de transferência matricial da planta utilizada na validação do controlador em simulação.

$$\begin{bmatrix} G_{11}(q^{-1}) & G_{12}(q^{-1}) \\ G_{21}(q^{-1}) & G_{22}(q^{-1}) \end{bmatrix} = \begin{bmatrix} \frac{-0.1235q^{-1}}{1-0.9780q^{-1}} & \frac{-2.9242+2.9656q^{-1}}{1-0.9833q^{-1}} \\ \frac{-0.6891q^{-1}+0.6910q^{-2}}{1-1.9471q^{-1}+0.9477q^{-2}} & \frac{1.9212q^{-1}-1.9182q^{-2}}{1-1.9765q^{-1}+0.9766q^{-2}} \end{bmatrix} \quad (7)$$

A Equação (8) apresenta a função de transferência matricial utilizada na implementação do modelo CARIMA dos controladores GPC1 e GPC2.

$$\begin{bmatrix} G_{11}(q^{-1}) & 0 \\ 0 & G_{22}(q^{-1}) \end{bmatrix} = \begin{bmatrix} \frac{-0.1235q^{-1}}{1-0.9780q^{-1}} & 0 \\ 0 & \frac{1.9212q^{-1}-1.9182q^{-2}}{1-1.9765q^{-1}+0.9766q^{-2}} \end{bmatrix} \quad (8)$$

A Tabela (1) apresenta os parâmetros de ajuste utilizados na sintonia do algoritmo de controle preditivo proposto.

Tabela 1. Parâmetros de Sintonia dos Controladores GPC1 e GPC2.

Parâmetros	GPC1	GPC2
N_2 – Horizonte de Predição	10	9
N_1 – Horizonte Mínimo de Custo	1	1
N_u – Horizonte de Controle	4	3
λ - Ponderação do Sinal de Controle	100	100
α - Ponderação da Referência	0.7	0.7
d - Atraso	1	1

6. Resultados Experimentais

A Figura (7) apresenta o desempenho do algoritmo proposto para uma entrada em degrau no superaquecimento. A Figura (8) apresenta o desempenho do controlador para uma entrada em degrau na potência frigorífica. Pode ser observado nas Figs.(7) e (8) que as saídas previstas pelo modelo de predição do algoritmo (em malha fechada) reproduzem os valores reais da planta, tanto para o superaquecimento quanto para a potência frigorífica, apresentando erros de predição quase desprezíveis a despeito do modelo CARIMA não incluir o acoplamento cruzado.

Na configuração proposta (dois controladores GPC independentes), os sinais de controle são produzidos pelos controladores com base apenas nas funções de transferência da diagonal principal, $G_{11}(q^{-1})$ e $G_{22}(q^{-1})$ sem levar em consideração (no modelo de predição) o acoplamento cruzado entre as entradas e saídas da planta. Isto é, o modelo interno do bloco GPC1 possui apenas a função de transferência $G_{11}(q^{-1})$ e o modelo do bloco GPC2 possui, por sua vez, apenas a função de transferência $G_{22}(q^{-1})$. Nesta configuração os dois controladores GPC mantêm o sistema estável e com bom desempenho como mostram as Figs. (7) e (8).

7. Comentários Finais

Este trabalho apresentou uma estrutura de controle preditivo, baseada no algoritmo GPC, aplicada ao controle de uma máquina refrigeração por compressão de vapor.

A estrutura de controle proposta demonstrou excelente desempenho a despeito do elevado grau de acoplamento cruzado existente no sistema. O esquema teve excelente desempenho lidando com uma planta MIMO de fase não-mínima, elevado acoplamento cruzado entre suas entradas e saídas e com número de condicionamento da ordem de 10^4 e 10^9 para as matrizes de controlabilidade e de observabilidade, respectivamente. Ficando assim mostrado a viabilidade da utilização de controladores preditivos no controle da operação de sistemas de refrigeração de pequeno porte por compressão de vapor.

Os resultados obtidos sugerem o esquema de controle proposto como uma excelente alternativa para a substituição de controladores convencionais por técnicas de controle preditivo capazes de lidar com sistemas que incluem tempo morto. Finalmente, o algoritmo de controle proposto, consistindo em dois controladores GPC SISO independentes, se apresentou como uma solução econômica, de baixo custo computacional e de fácil implementação em micro-controladores comerciais.

8. Referências

- Camacho, E.F. and Bordons, C., 1999, "Model Predictive Control", Ed. Springer-Verlag.
- Clarke, D.W., Mohtadi, C. and Tuffs, P.S., 1987(a), "Generalized Predictive Control – Part I. The Basic Algorithm", *Automatica*, Vol.23, No.2, pp.137-148.
- Clarke, D.W., Mohtadi, C. and Tuffs, P.S., 1987(b), "Generalized Predictive Control – Part II. Extensions and Interpretations", *Automatica*, Vol.23, No.2, pp.149-160.
- Gálvez, J.M. and Silva, A.G. 2000, "Multivariable Control for Multi Input – Multi Output Cooling Machines Based on Vapor Compression", Proceedings of the 1st National Conference on Mechanical Engineering, Natal, Brasil.
- Imbabi, M.S., 1990, "Computer Validation of Scale Model Test for Building Energy Simulation", *International Journal of Energy Research*, Vol.14, pp.727-736.
- Kinnaert, M., 1989, "Adaptive Generalized Predictive Controller for MIMO Systems", *International Journal of Control*, Vol.50, No.1, pp.161-172.
- Machado, L., 1996, "Modele de Simulation et Etude Experimentale d'un Evaporateur de Machine Frigorifique en Regime Transitoire", Thèse de Doctorat, L'Institute National des Sciences Appliquees de Lyon, França.
- Machado, L., et ali., 1995, "Estudo Experimental do Comportamento em Regime Transiente de um Evaporador de uma Máquina Frigorífica", Anais do XIII Congresso Brasileiro de Engenharia Mecânica, Belo Horizonte, Brasil.
- Outtagarts, A., 1994, "Comportement Dynamique d'un Evaporateur de Machine Frigorifique Alimente par un Detendeur Electronique", Thèse de Doctorat, L'Institut National des Sciences Appliquees de Lyon, França.
- Qin, S.J. and Bagdwell, T.A., 1997, "An Overview of Industrial Model Predictive Control Technology, Chemical Process Control – Assessments and New Directions for Research", *AIChE Symposium Series*, Vol.93, No.316, pp.232-256.
- Richalet, J., Rault, A., Testud, J.L. and Papon, J., 1976, "Algorithmic Control of Industrial Processes", Proceedings of the 4th IFAC Symposium on Systems Identification and Parameter Estimation, pp.1119-1167.
- Richalet, J., Rault, A., Testud, J.L. and Papon, J., 1978, "Model Predictive Heuristic Control: Applications to Industrial Processes", *Automatica*, Vol.14, pp.413-428.
- Rocha, R., 1995, "Contribuição ao Estudo de um Sistema de Refrigeração com Compressor Operando a Velocidade Variável", Dissertação de Mestrado, Programa de Pós-graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Silva, A.G., 2000, "Controle Preditivo Monovariável Aplicado a Máquinas Frigoríficas", Dissertação de Mestrado, Programa de Pós-graduação em Engenharia Mecânica, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.

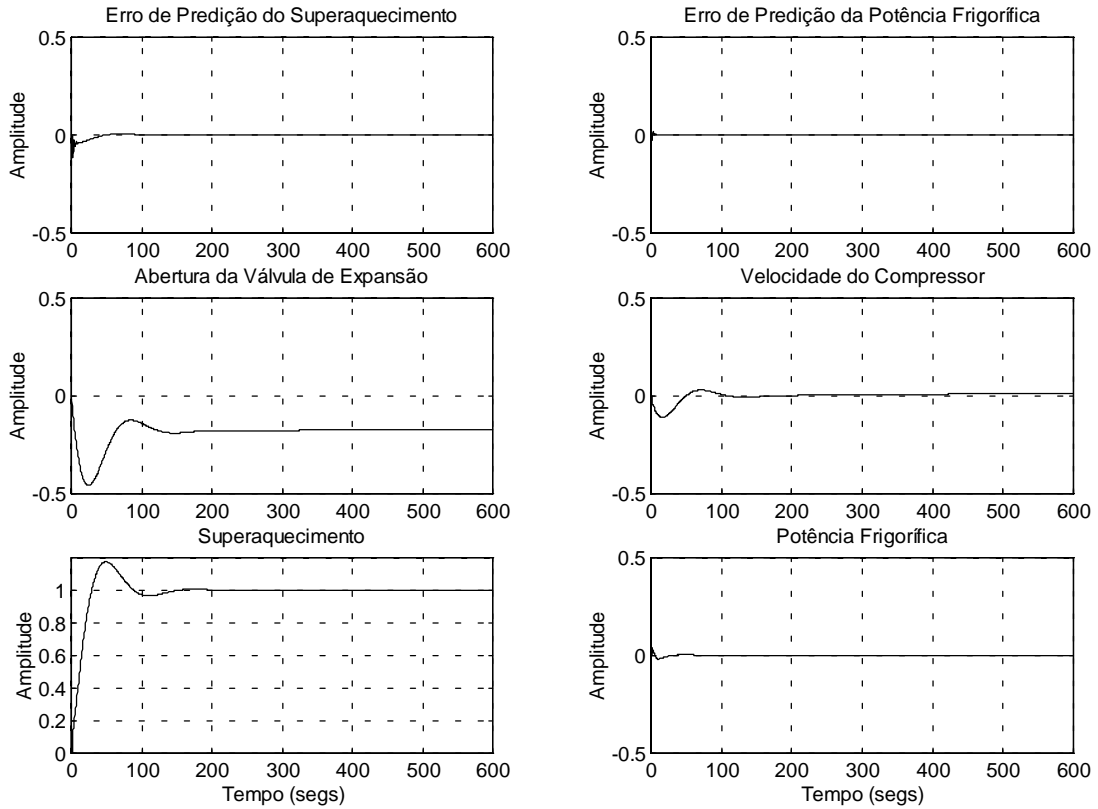


Figura 7. Desempenho do Controlador Preditivo para uma Entrada em Degrau no Superaquecimento.

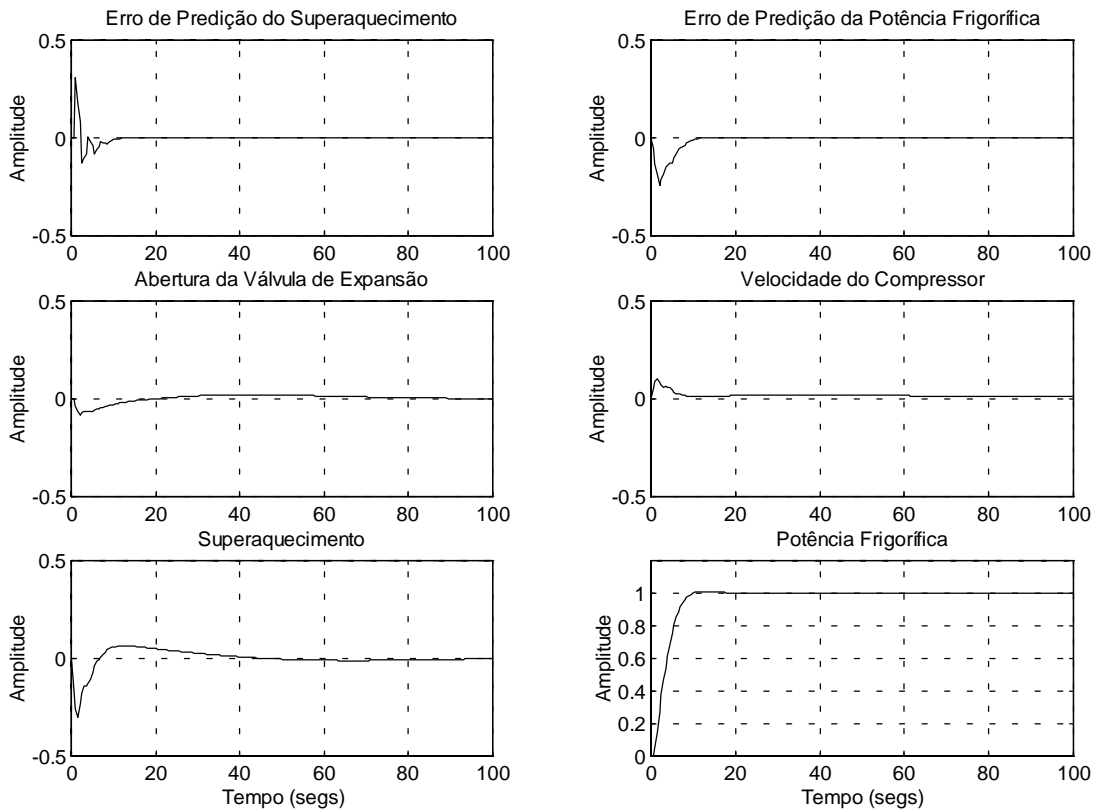


Figura 8. Desempenho do Controlador Preditivo para uma Entrada em Degrau na Potência Frigorífica.

GENERALIZED PREDICTIVE CONTROL FOR COOLING SYSTEMS BASED IN VAPOR COMPRESSION

José Maria Gálvez, Ph.D.

Department of Mechanical Engineering, Federal University of Minas Gerais, Brazil.
Av. Antônio Carlos 6627, Pampulha, 31.270-901 Belo Horizonte, MG, Brazil.
Phone: +55 31 3499-5236. Fax: +55 31 3443-3783.
E-mail: jmgm@dedalus.lcc.ufmg.br.

Agostinho Gomes da Silva, M.Sc.

Department of Mechanical Engineering, Federal University of Minas Gerais, Brazil.
Av. Antônio Carlos 6627, Pampulha, 31.270-901 Belo Horizonte, MG, Brazil.
Phone: +55 31 3499-5236. Fax: +55 31 3443-3783.
E-mail: ags@demec.ufmg.br.

Abstract. *Cooling machines are multi-input multi-output, cross-coupled, time-varying, nonlinear systems whose inputs (manipulated variables) present saturation and rate constraints. Classical control techniques such as PID controllers are not usually adequate to deal with the existing cross coupling among inputs and outputs. Under this scenario the independent control of superheating and freezing-power is a challenging task and so new control schemes are currently under research. Predictive control algorithms were originally proposed to deal with complex nonlinear industrial plants. Due to their computationally costly algorithms, the application of these controllers became initially restricted to large industrial plants. Currently, with the appearance of new and powerful small computers, the implementation of complex algorithms became economically feasible for small plants. This is the case of modern cooling systems, which are relatively small plants with sufficiently complex dynamics to justify the use of model predictive control (MPC) techniques. This paper presents a predictive control scheme for a cooling system based on vapor compression. The structure of the proposed control algorithm was designed looking forward the simplification of its on-line implementation in micro-controllers. The main objective is to keep comfortable conditions under time varying thermal loads. The proposed scheme provides a solution for the independent control of the freezing power keeping the superheating close to its optimal value even during the system transients. Simulation results are presented to illustrate the controller performance.*

Keywords. *Predictive Control, Multivariable Control, Cooling Systems.*

ARQUITETURA DE CONTROLE BASEADA EM MODELOS ESTRUTURADOS APLICADA A SISTEMAS PRODUTIVOS

Diolino José dos Santos Filho

Escola Politécnica da Universidade de São Paulo
diolinos@usp.br

Cristina Toshie Motohashi Matsusaki

Escola Politécnica da Universidade de São Paulo
cristm@usp.br

Newton Maruyama

Escola Politécnica da Universidade de São Paulo
maruyama@usp.br

Paulo Eigi Miyagi

Escola Politécnica da Universidade de São Paulo
pemiagi@usp.br

Resumo. *Sistemas produtivos (SPs) com múltiplos processos simultâneos e com recursos intensamente compartilhados tendem a apresentar indeterminismo em relação ao instante de ocorrência dos eventos e indeterminismo em relação à seqüência de realização das atividades no nível global. Tais sistemas são classificados neste trabalho como sistemas produtivos complexos. As atuais técnicas de modelagem, análise e controle baseadas em redes de Petri, adequadas a processos com lógica simples de alocação dos recursos, precisam ser adaptadas para permitir controlar principalmente o uso dos recursos compartilhados. Este trabalho propõe uma sistemática de estruturação no processo de modelagem que culmina com a introdução de uma arquitetura de sistema de controle que distingue nitidamente os algoritmos de controle em camadas hierárquicas segundo suas naturezas e objetivos: na camada inferior situam-se os controles estruturados de cada um dos múltiplos processos, responsáveis pelo sequenciamento e realização das ações de cada processo; na camada superior situa-se o algoritmo de controle dos recursos, responsável pela supervisão e alocação dinâmica dos recursos do sistema como um todo. Com esta abordagem, abre-se um horizonte para a aplicação de diversas estratégias de controle visando o funcionamento efetivo do SP e quando possível, visando uma melhora no desempenho do sistema independente da complexidade que o comportamento dinâmico do sistema em questão possa vir a apresentar.*

Palavras chave: *Arquitetura de sistemas de controle, Redes de Petri, Sistemas Produtivos, Modelagem estruturada, Modelo baseado em restrições.*

1. Introdução

Os Sistemas Produtivos (SPs) podem ser entendidos como uma classe de sistemas cuja finalidade é a produção, isto é, a transformação de itens ou prestação de serviços, com a peculiaridade do seu comportamento dinâmico ser dirigido a partir da ocorrência de eventos, classificando-os como Sistemas a Eventos Discretos (SED) (Ho 1989).

SPs que executam apenas um processo bem definido, isto é, com uma seqüência pré-estabelecida de etapas que constituem o processo, apresentando um conjunto dedicado e exclusivo de recursos, tem sido exaustivamente explorados através de abordagens top-down para auxiliar no processo de modelagem e síntese do seu sistema de controle.

As necessidades atuais de competitividade, onde o ciclo de vida dos produtos tende a ser cada vez mais curto, exigem que os SPs comportem a produção de uma variedade cada vez maior de produtos e conseqüentemente é importante que suportem a execução de múltiplos processos envolvendo o compartilhamento dos recursos disponíveis. Neste contexto, os componentes básicos do SP, máquinas, manipuladores e transportadores, possuem agora autonomia e flexibilidade funcional para aumentar a eficiência em termos de sua capacidade de utilização. O problema consiste então em não somente automatizar os diferentes processos de transformação, mas também supervisionar a execução dos processos definindo a seqüência adequada de alocação dos recursos compartilhados e, quando possível, aprimorando o desempenho do SP segundo algum critério pré-estabelecido.

Apesar da comprovada eficácia das redes de Petri (Reisig 85) na modelagem de sistemas de controle de SED, a abordagem top-down para modelagem dos processos e bottom-up para inserção dos recursos no modelo de controle, até então apresentadas em (Hasegawa et al 1984), (Miyagi 1988), (Zhou 1993), (Miyagi 1996), (Santos Filho 1998), (Miyagi2000), (Villani 2000) não são eficazes no desenvolvimento de sistemas de controle para SPs com múltiplos processos simultâneos e forte compartilhamento de recursos. Isto deve-se ao fato da lógica de utilização dos recursos não ser suficientemente simples e tão pouco visível quando aplicada a abordagem orientada a processos.

A dificuldade em gerenciar ou até mesmo representar a lógica de alocação de recursos compartilhados por diversos processos deve-se principalmente à sua característica de complexidade, que consiste na inerente propriedade de indeterminismo em relação ao tempo e indeterminismo em relação ao seqüenciamento do ponto de vista global do sistema.

Considerando-se o nível de complexidade presente em SPs, em que há o problema do controle da utilização dos recursos compartilhados por múltiplos processos simultâneos, este trabalho propõe a utilização de um modelo de controle baseado em restrições. Esta abordagem torna-se efetiva dentro de uma arquitetura semanticamente híbrida do sistema de controle, onde coexistem controladores de processos que desempenham a função de um processador de tarefas pré-programadas e o controladores de recursos que, dependendo da complexidade do sistema podem agregar modelos de restrições, cada um desempenhando as suas funções apropriadas.

A seção 2 deste trabalho apresenta as características fundamentais dos SPs que os classificam como SED e a característica de complexidade dos processos em SPs justificando a necessidade de uma nova abordagem na modelagem do controle. Na seção 3 é proposta a arquitetura híbrida de sistemas de controle e seus níveis. Na seção 4 é apresentada a proposta de estruturação do modelo de controle de processos e a seção 5 apresenta o modelo do controle dos recursos compartilhados em SPs.

2. Complexidade de Processos em Sistemas Produtivos

O conceito de SP considerado neste trabalho está intrinsecamente relacionado ao conceito de Sistemas a Eventos Discretos - SED (Ho 1989), (Santos Filho, Miyagi 1991). Refere-se a uma classe de sistemas que são estruturalmente compostos por um conjunto de elementos independentes e que, funcionalmente, são essencialmente dirigidos por eventos que, por sua vez, representam ações que ocorrem nesses sistemas durante um intervalo de tempo finito, provocando uma transição de estado discreta.

Durante a realização dos processos em um SP dessa natureza, poderá ocorrer então paralelismo de eventos, conflito de eventos, sendo possível inclusive haver indeterminismo quanto à ocorrência destes eventos em função do tempo (assincronismo).

Outro aspecto fundamental considerado neste trabalho a cerca dos SPs é a execução de múltiplos processos de transformação. Quando estes múltiplos processos compartilham recursos, define-se então um processo global que representa a seqüência em que os recursos são requisitados pelos diversos processos de transformação. O processo global do SP, que define a dinâmica do SP como um todo, pode possuir a característica complexidade.

O termo complexidade de acordo com Bruce (1999) não é usualmente atribuído ao mundo real, mas sim ao modelo que se constrói a partir da observação do mundo real. Bruce (1999) define complexidade como a propriedade que um modelo pode possuir e que dificulta uma formulação mais completa de seu comportamento.

O processo de abstração, ou modelagem, envolve dois aspectos:

- λ [Sintético: Baseia-se em uma abordagem “bottom-up” a partir da compreensão individual dos elementos que constituem o sistema em estudo;
- λ [Analítico: Baseia-se em uma abordagem “top-down” a partir da compreensão do comportamento geral do sistema.

Portanto, a complexidade de um modelo está na dificuldade em gerar-se a parte analítica a partir de sua parte sintética (Bruce 1997). Isto significa que a definição das propriedades de todas as partes que constituem um sistema não implica na determinação das propriedades do sistema como um todo; ou seja, apesar do todo ser constituído por partes bem definidas, o todo é maior que a somatória destas partes. Este é o motivo pelo qual é possível haver dificuldades em prever ou descrever o comportamento de uma organização, apesar de existir uma vasta quantidade de informações pertinentes às partes e elementos constituintes da mesma. Portanto, complexidade não deve ser confundida com a situação em que há uma falta de conhecimento a respeito do comportamento das partes, ou seja, esta desinformação não é sinônimo de complexidade.

Portanto, ainda que não se considere a questão de falhas inesperadas a que estão sujeitos os recursos responsáveis pela realização dos processos em um SP, ou então, uma possível alteração da programação da produção previamente estabelecida por alguma razão especial, há outras características que podem estar presentes no comportamento dinâmico desses sistemas e que dificultam as atividades relacionadas a modelagem e controle do mesmo. A situação em que se executa simultaneamente vários processos em um determinado SP constitui um cenário em que é suposto que cada um dos processos é bem definido, isto é, cada processo pode ser decomposto em um conjunto de ações com um seqüenciamento pré-determinado, o que permitiria uma adequada alocação dos recursos necessários para a execução de cada processo.

Entretanto, apesar do comportamento dinâmico de cada processo individual ser bem conhecido, e ainda que sejam modelados todos os compartilhamentos de recursos que se fizerem necessários, quase sempre não é possível pré-definir o comportamento dinâmico do sistema como um todo em relação a seqüência de ações do processo global. Assim, do ponto de vista global do sistema como um todo não é possível pré-definir por exemplo, se uma determinada ação de um processo irá ocorrer antes ou depois de uma ação pertinente ao outro processo. Disto conclui-se que o sistema em questão possui comportamento complexo, isto é, o processo global do sistema é complexo.

Se for considerado o fato de cada processo ser dirigido por eventos, não será necessariamente possível determinar em que instante irá ocorrer cada uma das ações que constituem cada um dos processos. Portanto, passam a coexistir dois graus de indeterminismo a nível de processo global:

- λ[Indeterminismo em relação ao tempo e;
- λ[Indeterminismo em relação à seqüência de ocorrência de eventos.

A partir dos fatos expostos é possível concluir-se que os processos em SPs podem ser complexos, necessitando de um abordagem adequada para a sua representação, análise e controle.

3. Arquitetura Híbrida para Sistemas de Controle

A consideração das características comportamentais dos SPs revela dois fluxos de informações de naturezas distintas. O primeiro, pertinente ao controle de cada um dos processos, destina-se à efetiva realização do seqüenciamento lógico das etapas de transformação de um determinado item. O segundo, pertinente ao controle do uso dos recursos, destina-se à alocação dos recursos para as atividades solicitantes, segundo um objetivo global do sistema como um todo.

O controle dos processos de transformação, tendo seqüências pré-definidas, pode ser representado adequadamente por modelos que ditam a seqüência desejada de transformação. A execução das ações conforme a seqüência imposta pelo controlador resultará na produção/realização do item/serviço. Um modelo de controle desta natureza é denominado controle sequencial (Miyagi 1996).

O controle do uso dos recursos, ou controle de recursos traz inerente o indeterminismo da seqüência de ocorrência das ações que irão requisitar os recursos. Isto, em outras palavras, indica a existência de inúmeras possibilidades de seqüências para o processo global que envolve os recursos do SP e, dependendo do número de elementos (recursos) presentes, pode ser inviável a definição da seqüência desejada para o processo global. Portanto uma abordagem de controle que pode ser considerada adequada é, ao invés de impor a seqüência de evolução de estados desejada, inibir as ações que levam o sistema a estados indesejados. Um modelo de controle desta natureza pode ser chamado de controle baseado em restrições e constitui-se na alternativa adotada neste trabalho para tratar a complexidade dos SPs.

Esta distinção das naturezas dos controles leva a definição de uma arquitetura híbrida de sistema de controle onde os controles de naturezas diferentes situam-se em níveis semanticamente diferentes. É proposto então a Arquitetura Híbrida de Controle, esquematizada na Figura 3.1, comporta pelos seguintes níveis hierarquicamente organizados:

- λ[Nível Superior – Controle de Recursos, responsável pela alocação adequada dos recursos;
- λ[Nível Inferior – Controle de Processos., responsável pela seqüenciação do processo de transformação.

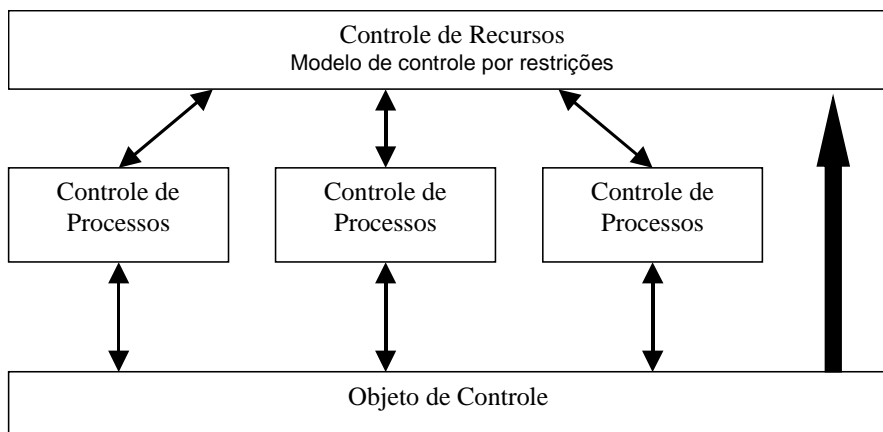


Figura 3.1: Arquitetura híbrida de controle de sistemas produtivos.

Nesta arquitetura, o controle de recursos situa-se no nível superior no sentido de tratar-se de um controle global que visualiza todos os recursos do sistema e todas as requisições de acesso a esses recursos. E o controle de processos situa-se no nível inferior no sentido de ser um controle localizado para um determinado processo de transformação, sem a visão do todo. Ressalta-se aqui que a hierarquia de níveis desta arquitetura nada tem em comum com níveis de abstração funcional.

Conforme elucidado anteriormente, em um SP com múltiplos processos, para cada processo existe um controle do processo cujo principal objetivo é assegurar a efetiva realização da seqüência de ações de transformação do item correspondente. A medida que o processo evolui, ou a cada etapa do processo, o controle do processo requisita ao nível superior de controle (controle de recursos) o acesso ao recurso (necessário para a realização da próxima etapa). Utilizando o recurso alocado, o controle do processo opera a planta do SP, que é o objeto de controle, enviando-lhe sinais de atuação e recebendo sinais de detecção.

O controle de recursos tem como principal função decidir as designações dos recursos durante a evolução dinâmica dos processos de transformação, de modo a garantir o comportamento desejado para o SP como um todo, seguindo uma determinada estratégia de gerenciamento dos recursos.

O modelo de controle dos processos pode ser obtido através de uma abordagem estruturada orientada a processos; e o modelo de controle dos recursos pode ser obtido através da representação da utilização dos recursos numa abordagem orientada aos recursos, como apresentado nos itens 4 e 5 respectivamente.

4. Modelo Estruturado do Processo – Controle de Processos

Há várias propostas no sentido de sistematizar o procedimento de projeto do sistemas de controle de SPs, considerando inclusive as situações em que este SP englobe uma abordagem antropocêntrica em seu comportamento dinâmico. Em Santos Filho (1998), a proposta de sistematização baseia-se na metodologia Enhanced Mark Flow Graph/ Production Flow Schema (E-MFG/PFS) (Santos Filho; Miyagi 1995), (Miyagi et al. 1998).

Ainda que os resultados obtidos tenham sido efetivos, há um outro aspecto que deve ser observado e que não foi considerado em tal trabalho. Trata-se do problema de estruturação do processo de modelagem.

Mesmo que seja possível aplicar-se os conceitos de refinamentos sucessivos para a modelagem dos processos em PFS e, posteriormente, em E-MFG, o projetista ainda está sujeita ao grau de dificuldade correspondente à representação do compartilhamento de recursos e, dependendo da flexibilidade operacional do sistema, torna-se necessário também projetar compensadores para disciplinar a alocação destes recursos para que não haja “deadlock” no sistema, devido a um fluxo incorreto de materiais. Tanto esta alocação de recursos quanto a inserção destes compensadores comprometem a estruturação no processo de modelagem.

Portanto, ainda que as metodologias existentes auxiliem no processo de modelagem de SPs, pretende-se contribuir de uma forma mais efetiva, objetivando-se estruturar o processo de modelagem do controle de SPs para aprimorar o projeto de sistemas de controle desses sistemas, a exemplo do que ocorreu com o processo de desenvolvimento de programas a partir do momento em que os conceitos de programação estruturada foram colocados em prática.

A subseção 4.1 apresenta os conceitos fundamentais da programação estruturada e a aplicação na síntese de algoritmos de controle de SPs. Na subseção 4.2 é proposta a metodologia de modelagem estruturada E-MFG/PFS.

4.1. Sistemas Produtivos e Estruturação

Uma das grandes vantagens de implementar soluções estruturadas está em gerar-se um código mais simples de ser interpretado, facilitando qualquer procedimento de manutenção, atualização e permitindo a execução de testes mais eficiente.

A programação estruturada fundamenta-se em dois aspectos (Ghezzi, Jazayeri, 87): o conceito de refinamentos sucessivos, que envolve projeto top-down, hierarquia de abstração e desenvolvimento modular; e o uso racional de desvios incondicionais (GOTO).

Para definir-se uma sistematização do processo de desenvolvimento de programas, (Bohn e Jacopini 1996) formularam o teorema da estruturação garantindo que o desenvolvimento de um código estruturado envolve a construção de unidade de programa utilizando apenas as seguintes declarações do tipo seqüência, desvio condicional e repetição iterativa.

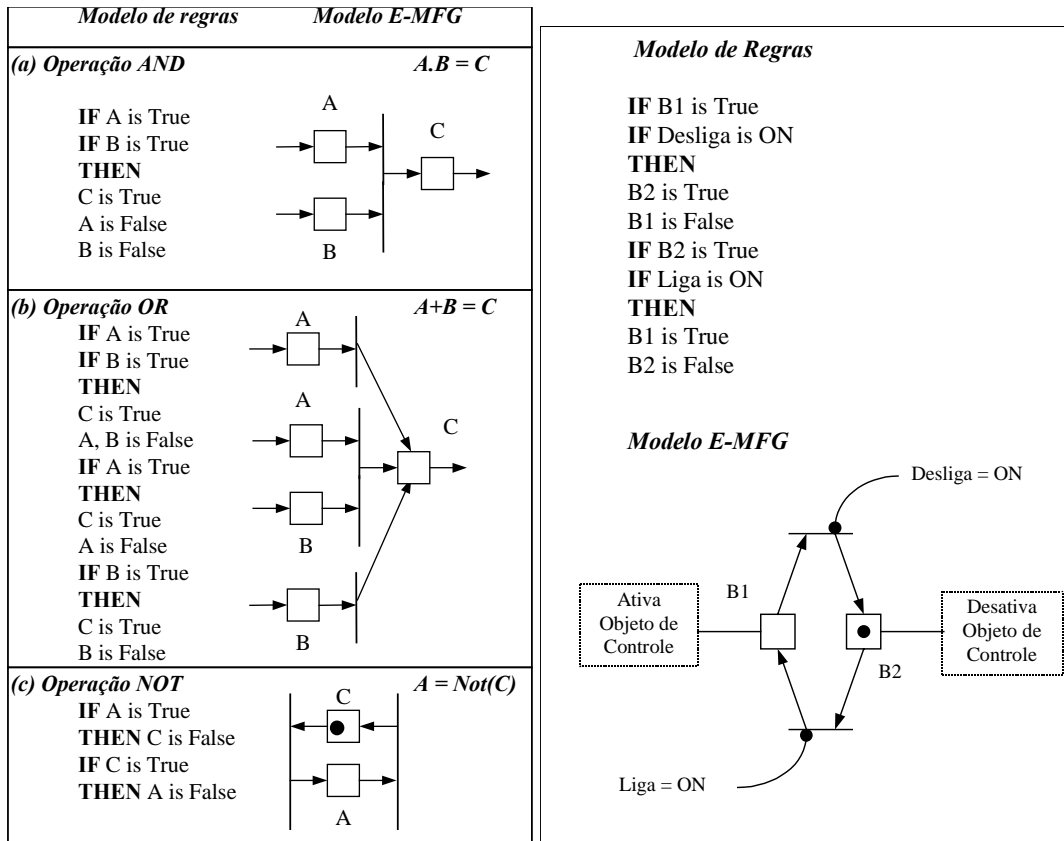


Figura 4.1: Representação das operações lógicas e da retenção de estados através de regras de produção e através de MFG.

Estabelece-se que qualquer programa computacional pode ser construído utilizando-se estruturas do tipo IF/THEN/ELSE, WHILE/DO e operações seqüenciais adequadas do tipo atribuições, entradas, saídas entre outras.

Por sua vez, o controle qualitativo usado no controle de SED pode ser interpretado como controle lógico que manipula decisões orientadas a atributos (correspondentes ao estados das variáveis lógicas), associando-se operadores de lógica de primeira ordem – mais especificamente da Álgebra Booleana – e elementos de seqüenciamento do tipo temporizadores e memórias (Asfahl 1985). Observa-se que o comportamento dinâmico desses sistemas pode ser expresso por regras de produção do tipo 'IF ... THEN ...'. (Figura 4.1).

O refinamento sucessivo é possível aplicando-se o PFS em conjunto com o E-MFG, num exercício de abstração das atividades do sistema. O PFS descreve funcionalmente o sistema, partindo de um modelo conceitual com elevado nível de abstração. Numa abordagem top-down, os detalhes são inseridos progressivamente a cada nível de refinamento. A modularização é mantida eliminando-se a possibilidade de fluxos secundários, que não correspondem a entradas e saídas principais, das atividades.

O uso racional do GOTO é alcançado através das estruturas padrões para modelagem (Figura 4.2) que são equivalentes às estruturas de seqüenciamento, desvio condicional e repetição iterativa da programação estruturada (Santos Filho 98).

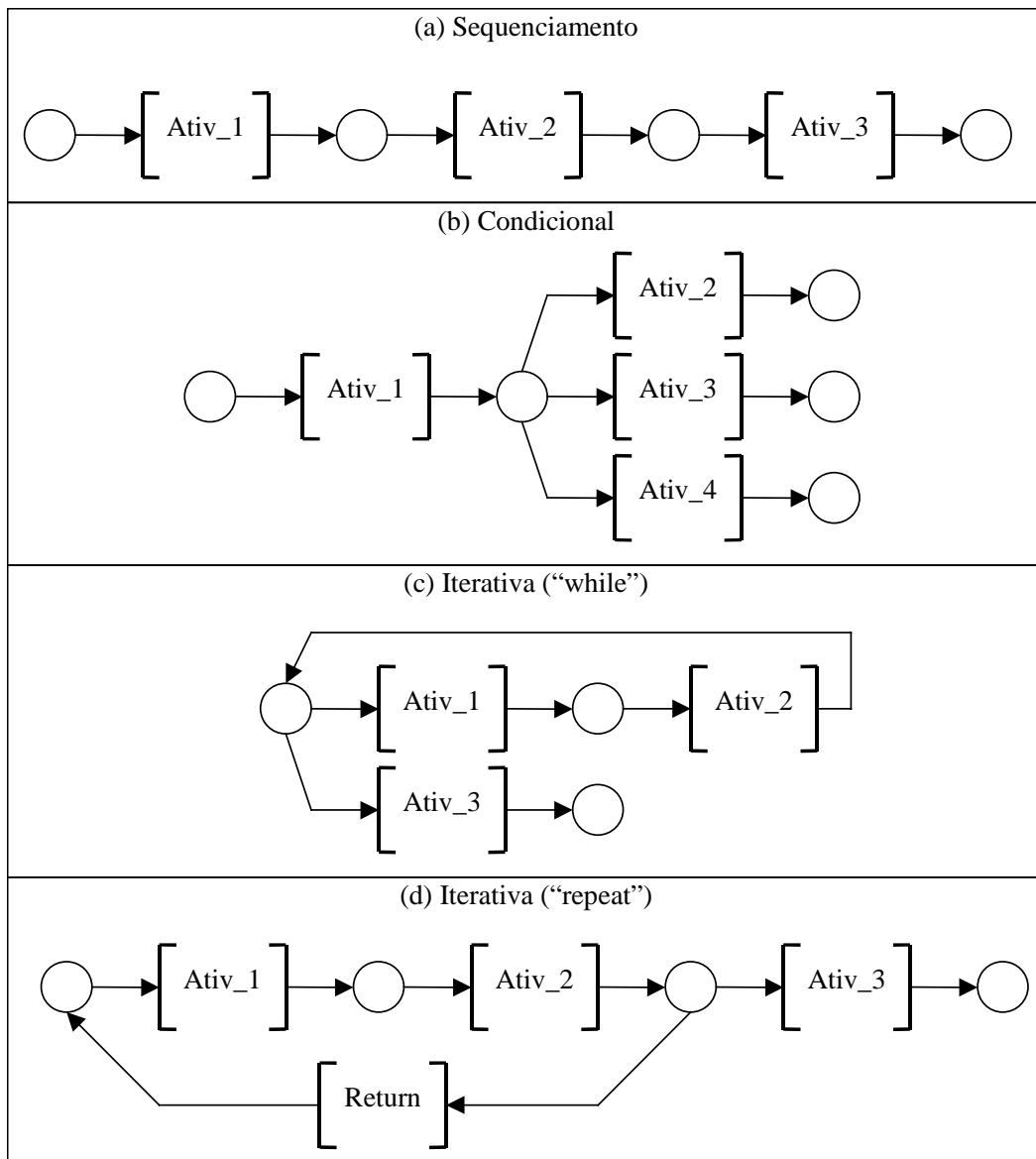


Figura 4.2: Estruturas padrões em PFS estruturado.

É importante observar que todos os aspectos de estruturação discutidos para o PFS aplicam-se também para o E-MFG.

4.2 Metodologia de Modelagem Estruturada PFS/E-MFG

Aplicando-se os conceitos de estruturação discutidos no item anterior, é proposto uma metodologia de modelagem de processos dos SPs. A metodologia proposta tem como base a metodologia PFS/MFG apresentada em (Miyagi 96), modificando-a para que os procedimentos que resultam na quebra da estruturação sejam eliminados. As principais modificações são: a eliminação dos fluxos secundários no PFS, que possibilitariam entradas e saídas incondicionais; o uso racional dos desvios incondicionais de fluxos; o acréscimo de uma etapa de definição dos atributos das marcas individuais, para uma extensão do MFG para o E-MFG; e a eliminação da etapa de representação e controle da alocação dos recursos.

Propõe-se então a Metodologia de Modelagem Estrutura PFS/E-MFG, que engloba as seguintes etapas:

- I. Representação de cada processo por um modelo PFS conceitual de alto nível de abstração.
- II. Detalhamento de cada processo em atividades, definindo a lógica de seqüenciamento destas atividades.
- III. Definição do conjunto de atributos que deve ser associado às marcas do grafo E-MFG.
- IV. Detalhamento das atividades dos processos através de modelos E-MFG.
- V. Mapeamento dos sinais de controle indicando os eventos gerados pelo controlador a partir de arcos de sinal de saída e os eventos ocorridos na planta a partir de portas externas.

Esta metodologia é especialmente adequada para aplicar-se aos processos de transformação situados no nível inferior da arquitetura híbrida de controle, proporcionando vantagens similares ao do projeto estruturado de software, isto é, facilidade de compreensão, manutenção, implementação e execução.

5. Modelo da Utilização de Recursos– Controle baseado em Restrições

Observa-se que a modelagem do compartilhamento dos recursos não é mais pertinente metodologia descrita na seção anterior.

O paradigma adotado nesta arquitetura baseia-se no fato da modelagem do nível de controle de recursos utilizar como referencial os próprios recursos no contexto de cada processo para iniciar o processo de modelagem em questão.

Nesta sessão, inicialmente apresenta-se a modelagem da seqüência de utilização dos recursos através do grafo tipo RAG (Resource Allocation Graph). Em seguida, explana-se a interpretação deste modelo de uso dos recursos e apresenta-se a possibilidade de se atuar sobre a alocação dos recursos.

5.1. Modelo da Utilização de Recursos

5.1.1 Grafo de Alocação de Recursos

Grafo de Alocação de Recursos (GAR) (Santos Filho 1993) é um grafo orientado não marcado e com inscrições nos arcos, capazes de representar os processos produtivos através dos recursos. O RAG consiste de nós, que indicam os recursos r_q e são representados graficamente por círculos, e arcos, que indicam as etapas dos processos, representados por arcos orientados. A matriz M de adjacência de um GAR tem como seus elementos a identificação das etapas que interligam um dado nó-recurso e outro. Um nó e o seu arco de saída são chamados de par de clocação e indicam o recurso necessário para a etapa referente ao arco. Um nó e seu arco de entrada são chamados de par de requisição, e indicam que quando a etapa referente ao arco terminar, o recurso será requisitado para a execução da próxima etapa.

A Figura 5.1 é o esquema do SP que será utilizado para ilustrar os conceitos desta seção. O SP exemplo é um sistema de manufatura onde o produto A é transformado pelas máquinas R1, R2, R3 e R4, nesta ordem. O modelo RAG e a respectiva matriz M do processo de transformação do produto A, ou processo p_A é mostrado na Figura 5.1.

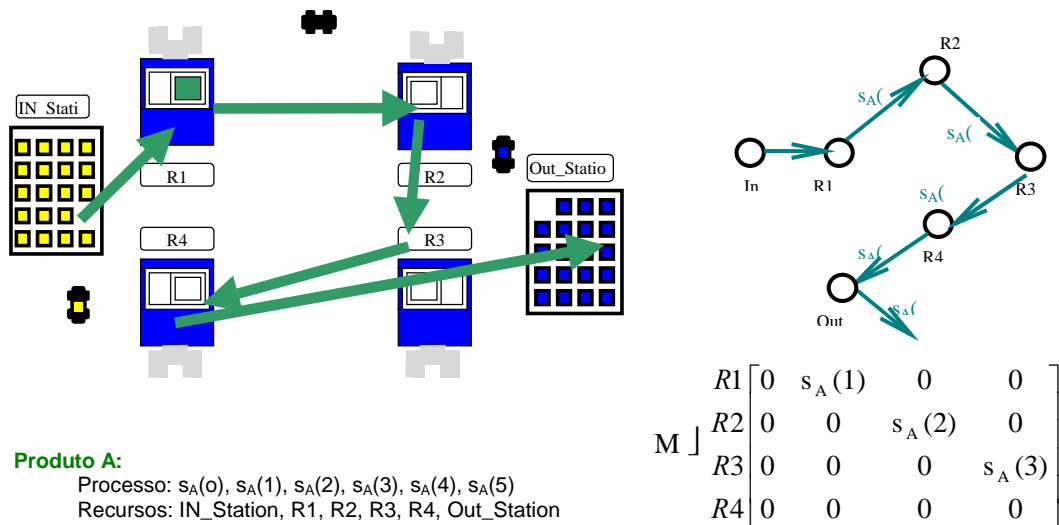


Figura 5.1: SP exemplo e o processo A; modelo GAR e matriz M do processo A.

5.1.2 Tradução do Grafo de Alocação de Recursos para Mark Flow Graph Estendido

Uma vez que o GAR é um grafo transição que representa um autômato finito (Reisig 1985), é possível mapeá-lo em uma rede de Petri (Cassandras 1993).

Definição 5.1. Considere um autômato de estados finitos dado pela quintupla (E, X, f, x_0, F) onde (Cassandras 1993):

- E é um alfabeto finito;
- X é um conjunto de estados finito;
- f é uma função de transição de estados $f: X \times E \rightarrow X$
- x_0 é o estado inicial tal que $x_0 \in X$;
- F é um conjunto de estados tal que $F \subseteq X$.

Definição 5.2. Considere o grafo Mark Flow Graph bipartido, representado por uma sextupla (B, T, A, G_i, G_e, S) onde (Miyagi (1996)):

- $\mathbf{B} = \{B_1, \dots, B_i\}$ com $(i \geq 1)$ é um conjunto de boxes;
- $\mathbf{T} = \{T_1, \dots, T_j\}$ com $(j \geq 1)$ é um conjunto de transições;
- \mathbf{A} é o conjunto de arcos orientados;
- $\mathbf{G}_I = \{G_{I1}, \dots, G_{Im}\}$ com $(m \geq 0)$ é um conjunto de portas internas;
- $\mathbf{G}_E = \{G_{E1}, \dots, G_{En}\}$ com $(n \geq 0)$ é um conjunto de portas externas;
- $\mathbf{S} = \{S_1, \dots, S_p\}$ com $(p \geq 0)$ é um conjunto de arcos de sinais de saída;

De acordo com Cassandras (1993) dado um autômato de estados finito (E, X, f, x_0, F) , é possível construir-se uma rede de Petri elementar (Reisig (1985)) uma vez que se pode associar a cada estado global representado pelo autômato uma determinada marcação na rede de Petri correspondente. Uma vez que o MFG baseia-se em redes de Petri com a ressalva de ser um grafo interpretado que possui um mapeamento dos sinais externos através de arcos de sinal de saída e portas externas, baseado em Cassandras (1993) é possível inferir que para construir-se um modelo MFG a partir de um GAR implica no seguinte procedimento:

Cada estado em \mathbf{X} é definido como um box em \mathbf{B} ;

A cada par de estados (x, x') , onde $x' = f(x, e)$ para $e \in E$, define-se uma transição em \mathbf{T} tal que $T = \{(x, x') : x = X, x' = f(x, e) \text{ para } qq. e \in E\}$

A seguir, define-se um arco a partir de cada par (x, t) e (t, x') com $x = X$ e $x' = X$ e $t = T$ tal que $A = \{(x, t) : x = X, t = T\} \cup \{(t, x') : x' = X, t = T\}$

A partir deste procedimento obtemos o modelo MFG correspondente ao GAR.

Uma vez que é possível manipular-se marcas individuais em uma rede para representar situações que envolvem o compartilhamento de um recurso para o processamento de diferentes elementos associados a processos distintos, é possível estender-se o mesmo procedimento para o caso em que as marcas do grafo possuem identidade.

Considerações:

- λ Na situação em que o SP envolve múltiplos processos, para cada um dos processos, modela-se um GAR equivalente que, por sua vez, é traduzido em um modelo E-MFG equivalente. Através do método de fusão de lugares (Perkusich 1998) obtem-se o modelo E-MFG global de utilização dos recursos desejado.
- λ A utilização de cada recurso $r_q(j)$ é modelado por um box capacidade $Cr_q(j)$ para permitir a representação do número de unidades deste recurso disponíveis no sistema.
- λ A marcação inicial é tal que não considera nenhuma etapa de processo em andamento no estado inicial, correspondendo a uma situação em que o sistema encontra-se completamente ocioso.

A figura 5.2 apresenta um exemplo de modelo E-MFG global de utilização de recursos de um SP com três processos simultâneos s_A, s_B, s_C , compartilhando três recursos EST_1, EST_2, EST_3 e EST_4.

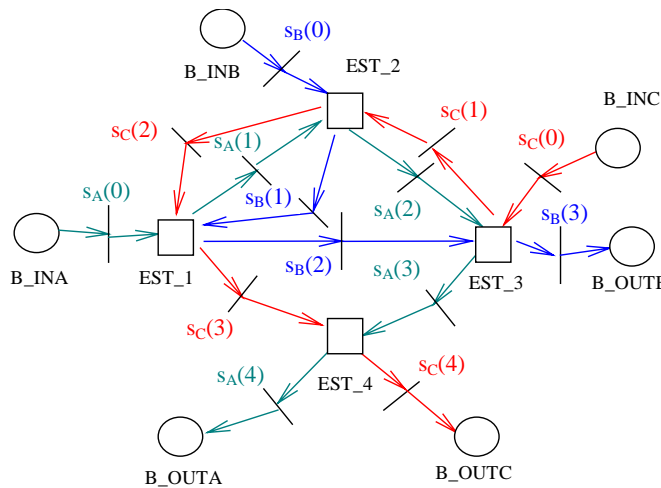


Figura 5.2: Exemplo de modelo E-MFG global de utilização dos recursos de um SP com processos simultâneos que compartilham recursos.

5.2. Modelo de Controle Baseado em Restrições

O grafo E-MFG construído na seção anterior está desempenhando a função de apenas representar a utilização de recursos em cada processo e, caso haja fusão de lugares, nestes pontos há com certeza um compartilhamento para a utilização dos correspondentes recursos envolvendo processos diferentes.

Entretanto, um modelo que representa o compartilhamento de vários recursos e estabelece como estratégia de controle deste compartilhamento apenas a fusão de lugares, associada a uma estratégia de arbitragem dos conflitos que

possam vir a ocorrer, baseado em estabelecer critérios, por exemplo, de prioridade representa uma estrutura de controle inadequada considerando o caráter complexo do problema.

Portanto, o desafio está em como proceder para controlar efetivamente o comportamento deste sistema global. Vários autores questionam o problema de que para obter um bom controlador de um sistema é necessário que este controlador reflita o modelo do sistema em si. Discutem-se também os efeitos das incertezas da planta sobre a realimentação e propõe métodos de redução da sensibilidade através de aproximadas inversas.

No caso dos SPs, realiza-se Controle Qualitativo, envolvendo lógica de primeira ordem. Neste trabalho, está sendo proposto um modelo para o controle dos recursos que apresenta um caráter dual em relação ao modelo tradicionalmente adotado de representação de alocação de recursos (Hasegawa et. al. 1984), (Miyagi 1988), (Miyagi 1996), (Santos Filho 1998), (Miyagi M 2000), (Santos Filho 2000), (Villani 2000).

Por exemplo, enquanto no estado inicial em que todos os recursos estão ociosos tem-se no modelo tradicional de alocação de recursos uma marcação contendo o número total de recursos disponíveis, no caso do modelo de utilização de recursos, a marcação inicial é zero ou nula.

De uma forma genérica, controlar um sistema pode ser interpretado como impôr um determinado comportamento desejado para o mesmo. No caso dos sistemas complexos em que o controle que atua no mesmo é do tipo Controle Quantitativo, correspondendo portanto a um SP complexo, torna-se difícil impor um comportamento dinâmico desejado uma vez que não existe este modelo de referência pré-definido. Portanto, é adequado adotar-se o controle baseado em restrições, que consiste na aplicação de regras de controle no modelo de utilização de recursos, de modo a inibir a evolução do sistema para estados não desejados. Estas regras de controle podem ser baseadas em uma gama de estratégias que objetivam a efetiva operação do SP e/ou aprimoramento do desempenho do sistema através do eficiente uso dos recursos. Uma destas possíveis estratégias de controle, fundamental para o funcionamento do SP, é o método deadlock-avoidance através de flow-in-suppression (Hasegawa et al 1996).

6. Observações Finais

A estruturação da modelagem proposta neste trabalho contribui efetivamente em dois aspectos no contexto de projeto de sistemas de controle de SPs:

- λ] Obtenção de modelos de controle estruturados para o nível de controle de processos de transformação, incorporando todas as vantagens das técnicas de projeto e programação estruturada no que concerne à reutilização e manutenção de algoritmos de controle;
- λ] Separação do fluxo de utilização de recursos, que antes confundia-se com o fluxo de realização dos processos, permitindo a sua visualização, análise e aplicação de estratégias de controle de recursos adequadas ao problema, independente do nível de complexidade do mesmo.

Portanto, este trabalho pode ser interpretado como uma forte motivação no sentido de alavancar futuras pesquisas na área de controle de sistemas produtivos e que estavam estagnadas devido a complexidade do problema. Graças ao processo de estruturação proposto, torna-se exequível a automação da síntese de controle de SPs (Nakamoto et al 2001).

Agradecimentos

Os autores agradecem a CAPES e CNPq pelo apoio a este trabalho.

Referências

- Asfhal, C.R., 1985, "Robots and Manufacturing Automation", John Wiley & Sons, Inc, USA.
- Bohn, C. and Jacopini, G., 1996, "Flow Diagrams, Turing Machines and Languages with Only Two Formation Rules", Comm. ACM, Vol. 9 No 5.
- Bruce E., 1999, "What is Complexity? - The philosophy of complexity per se with application to some examples in evolution", F. Heylighen and D. Aerts (Eds.), The Evolution of Complexity, Kluwer, Dordrecht.
- Cassandras, C.G., 1993, "Discrete Event Systems", IRWIN.
- Ghezzi, C.; Jazayeri, M., 1987, "Programming Language Concepts", John Wiley & Sons.
- Hasegawa, K. et.al., 1984, "Proposal of Mark Flow Graph for Discrete System Control", Trans. of SICE, Vol.20, No.2, pp.122-129.
- Hasegawa, K. et al., 1996, "Deadlock avoidance policy based on request/allocation graph", Proceedings of '96 Joint Symposium on System and Information, SICE, Matsuyama, Japan, pp.219-24.
- Ho, Y.C., 1989, "Special Issue on Dynamics of Discrete Event Systems", Proceedings of IEEE, Vol.77, pp.3-6.
- Miyagi, P.E., 1988, "Control System Design, Programming and Implementation for Discrete Event Production Systems by using Mark Flow Graph", Tese de Doutorado, Tokyo Institute of Technology, Japan.
- Miyagi, P.E., 1996, "Controle Programável - Fundamentos do controle de sistemas a eventos discretos", Ed. Edgard Blücher, São Paulo, Brazil.

- Miyagi, P.E. et al., 1998, “Modelagem e Análise de Sistemas Complexos com Recursos Compartilhados Através do PFS (Production Flow Schema)”, Proceedings of 12th CBA – Congresso Brasileiro de Automática, SBA, Uberlândia, Brazil.
- Miyagi, M.; Kisil, M.; Santos Filho, D.J.; Maruyama, N.; Miyagi, P.E., 2000, “Modelagem de Serviço de Saúde Utilizando Rede de Petri”, Proceedings of 4th. SBAI – Simpósio Brasileiro de Automação Inteligente., SBA, São Paulo, Brazil.
- Nakamoto et al., 2001, “Geração Automática de Regras de Controle para Alocação de Recursos em Sistemas Produtivos com Porcessos Concorrentes”, Proceedings of COBEM2001 – Congresso Brasileiro de Engenharia Mecânica, Uberlândia, Brazil.
- Perkusich, A. and Barros, T.C., 1998, “Redes de Petri: Um Procedimento de Modelagem Aplicado aos Sistemas de Produção em Lotes”, Proceedings of XII Brazilian Automatic Control Conference, Vol.4, pp1377-1382, Uberlândia, Brazil.
- Reisig, W., 1985 “Petri Nets: An Introduction”, Springer-Verlag, Berlin Heidelberg.
- Santos Filho, D. J.; Miyagi, P. E., "Sistemas de Eventos Discretos e seu Controle", Proceedings of 1st COBISA/CINISA, ISA International, São Paulo, SP, pp.2.1-12.
- Santos Filho, D.J., 1993, “Proposta do Mark Flow Graph Estendido para a Modelagem e Controle de Sistemas Integrados de Manufatura”, Dissertação de Mestrado, Escola Politécnica, Universidade de S. Paulo, São Paulo, Brazil.
- Santos Filho, D.J. and Miyagi, P.E., 1995, “Enhanced Mark Flow Graph to Control Flexible Manufacturing Systems”, Journal of The Brazilian Society of Mechanical Sciences, ABCM, Rio de Janeiro, Vol.17, No.2, pp.232-248.
- Santos Filho, D.J., 1998, “Controle de Sistemas Antropocêntricos de Produção Baseado em Redes de Petri Interpretadas”, Tese de Doutorado, Escola Politécnica da Universidade de S. Paulo, São Paulo, Brazil.
- Villani, E.; Miyagi, P.E.; Santos Filho, D.J.; Maruyama, N; 2000, “Metodologia para Modelagem de Sistemas de Ar-condicionado em Edifícios Inteligentes”, Controle & Instrumentação, Ano 5 No. 46, SUCESU-SP, São Paulo, Brazil.
- Zhou, M.C. and Cesare, F., 1993, “Petri Net Synthesis for Discrete Event Control of Manufacturing Systems”, Kluwer Academic Publishers, Massachusetts.

STRUCTURED MODEL BASED CONTROL ARCHITECTURE FOR PRODUCTION SYSTEMS

Diolino José dos Santos Filho

Escola Politécnica da Universidade de São Paulo
diolinos@usp.br

Cristina Toshie Motohashi Matsusaki

Escola Politécnica da Universidade de São Paulo
cristm@usp.br

Newton Maruyama

Escola Politécnica da Universidade de São Paulo
maruyama@usp.br

Paulo Eigi Miyagi

Escola Politécnica da Universidade de São Paulo
pemiyagi@usp.br

Abstract. Production systems (PS) with concurrent processes and shared resources tend to present uncertainty related to time and activity sequence in the global level. These systems are classified as complex production systems and can be approached by Petri nets while modelled as Dynamic Event Driven Systems. The modelling, analysis and control techniques, suitable for process control including simple resource allocation logic, have to be adequate to support shared resource control. This work proposes the structured modelling methodology for PS, which organises the control system in a hybrid architecture. This hybrid architecture separates the control algorithms in semantically distinguished levels: the lower level contains the structured process controllers, which are responsible for the production process sequence execution; the upper level contains the resource controller, responsible for the supervision and dynamical allocation of overall system resources.

Keywords. *Structured modeling, Control systems architecture, Constraint based control models, Petri nets, Production systems.*

MICRO-ROBOT SOCCER TEAM - MECHANICAL AND HARDWARE IMPLEMENTATION

Frederico Carvalho Vieira

DCA – CT – Universidade Federal do Rio Grande do Norte, Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN
fred@leca.ufrn.br

Pablo Javier Alsina

DCA – CT – Universidade Federal do Rio Grande do Norte, Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN
pablo@leca.ufrn.br

Adelardo Adelino Dantas de Medeiros

DCA – CT – Universidade Federal do Rio Grande do Norte, Campus Universitário – Lagoa Nova – 59072-970 – Natal - RN
adelardo@leca.ufrn.br

Abstract: *This paper describes the hardware implementation of a micro-robot soccer team. The robotic systems were developed to meet the FIRA specifications. A modular development methodology was adopted. Thus, each robot system was divided into three main subsystems. The mechanical sub-system includes all mechanical characteristics related to the robot, such as the body and wheels dimensions, motors specifications and motion characteristics. The robot body and motors arrangement were designed using Auto-CAD. A differential drive system was adopted, allowing the robot to execute straight forward/backward trajectories or circular ones and also to rotate around itself. The motors were chosen to reach a 2 m/s maximum velocity, with high acceleration and torque. The communication sub-system involves the hardware implementation and protocol specification used for data communication between a host computer and the micro-robots. The digital processing of the data protocol was implemented using PIC microcontrollers. The power subsystem was designed considering the motors specifications, such as the peak current and motor torque. This sub-system was implemented using a L293 IC, which is capable to drive two bi-directional DC motors. The proposed system was designed using components acquired in the national market, resulting on a low cost, robust and compact team.*

Keywords: *Mobile Robot, Micro-Robot, Robot Soccer.*

1. Introduction

The Robot-Soccer Championship, as proposed by FIRA (Federation of International Robot-Soccer Association), is a competition created to promote significant developments on robotics, sensor fusion, intelligent control, data communication, vision, mechatronics, computation, artificial life and related areas of engineering and technology (Veloso et al, 1998; Wyeth and Brown, 2000; Wyeth et al, 2000).

Several Educational and Research Institutions around the world have developed efficient robot soccer teams (Veloso et al, 1998; Wyeth et al, 2000), although, such implementations involves top technologies and, consequently, high costs. In Brazil, a few Research Institutions have proposed their own teams (Tavares, 1997; Costa et al, 1999). Due to limited resources, such Institutions were forced to look for low cost solutions.

According to the MiroSot-FIRA rules (FIRA, 1999), each team is constituted by three robots, whose dimensions can not be greater than $7.5 \times 7.5 \times 7.5 \text{ cm}^3$. Most of the information processing can be done in a host PC and commands can be transmitted, through radio link, to the robots, by means of unique carrier frequency. These hard constraints constitutes a technological challenging problem. In order to achieve such objectives, a robot soccer team that allies reduced dimensions, robustness and extremely low-cost was developed and its implementation details are presented in this paper.

The robots were developed in a modular fashion, composed by three integrated subsystems, namely: the mechanical subsystem, the communication subsystem and the power subsystem. In this paper, Section 2 describes the Global System focusing on the functional integration of these subsystems. In Section 3, implementation details of the mechanical subsystem are presented. Considerations about the power supply and driving motors designs are presented in Section 4. Aspects of the communication between the host PC and the robot team, part of the communication subsystem, are described in Section 5. In Section 6 is presented the communication and driving subsystems integration through a microcontroller. Finally, the conclusions and future works are presented in Section 6.

2. Architecture Overview

The proposed system follows the general configuration established by the MiroSot-FIRA rules: A vision system, an information processing system, a radio link and the robotic system, as shown in Fig.(1). The vision system consists of a top CCD camera plus a frame grabber connected to the host PC, which processes the captured image of both robot-soccer teams and the ball in order to compute their positions. The information processing system is a software module, running also in the host PC, that computes the global team strategy and the robot motion commands from the positions given by the vision system. The radio link provides the communication between the host PC and the robot team by

means of a radio transmitter module connected to the host PC and receivers modules embedded in each robot. The robotic system consists of the three robots of the team, each one including the mentioned receiver module, a microcontroller, and a motor driving system, allowing the robots to execute the commands sent by the host PC.

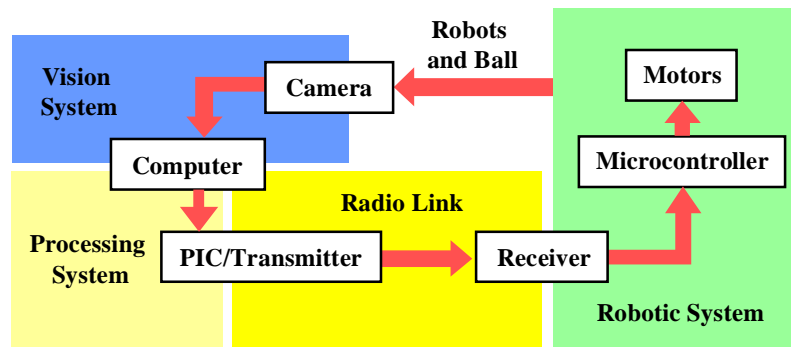


Figure 1. General System Architecture (PIC is a microcontroller).

3. The Mechanical Subsystem

The mechanical sub-system includes all mechanical characteristics related to the robot, such as the body and wheel dimensions, motors specifications and motion characteristics. The robot body and motors arrangements were designed using Auto-CAD. A differential drive system was adopted, allowing the robot to execute straight forward/backward trajectories or circular ones and also to rotate around its own body. The motors were chosen to reach a 2 m/s maximum velocity, with high acceleration and torque. No mechanical reductions were needed, since the motors have shown a good torque response.

The locomotion system was designed to allow the robots to execute fast and simple movements, like circular trajectories. The locomotion system presents two wheels, each one coupled to one motor. The locomotion system and robot body are shown in fig.(2).

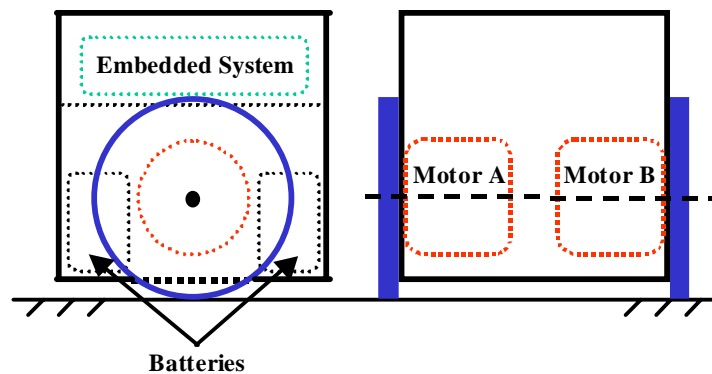


Figure 2. The robot locomotion system.

The robot body was built in acrylic material, with 2mm of thickness, which is very resistant for this purpose. The motors (screwed to the robot body) are localized between the tow batteries, and the embedded system is above an acrylic platform, to avoid direct contact to the motors. The wheels were made using a CAD-CAM system, in order to avoid different wheel sizes. Some mechanical characteristics, like the wheel size, body momentum and weight distribution, have a heavy contribution to the system dynamics and must be considered by the control algorithm. Another problem to face is that the system is time variant, due to the batteries discharge. This characteristic requires a continuous dynamic model adaptation.

4. The Driving Subsystem

On reduced scale mobile robot design, the most common actuators are DC motors. This kind of motor presents several advantages, like: easy to drive, easy to control, and mainly, low cost. On this project, DC motors were used to actuate on the robot wheels, allowing the robot to produce movements. DC Motors are “easy” to drive, because when you apply a positive DC component to it terminals, it produces a direct way rotation. The angular velocity of this rotation is proportional to the DC component applied, that is, as greater it is the DC component, greater will be the angular velocity. The reverse way rotation is analog to the direct way rotation.

There are several ways to drive a DC motor, however, some of them offer a greater loss of the power delivered to the motor. To implement a driving system with low loss of power, it is common to use a well-known driver circuit, the chopper or “H-Bridge”. The H-Bridge is constituted of four power transistors, which are switched from on to off accordingly to an input signal. This specific kind of signal is called PWM (Pulse Width Modulation). The PWM is a logic control signal that provides the information about how the transistors must be switched to produce a desired output. This output applied to the motor will rotate the motor on direct or reverse way, and with different velocity levels. The PWM is a rectangular wave, with constant frequency (constant period) and variable duty-cycle. The duty-cycle determines how much time the signal must hold on high level during one period. This value is specified by period percentages, that is, if the signal holds on high level during half-period, the duty-cycle is said to be of 50%. The input signal is a switched signal that switches between two states, on and off (or logic levels, ‘1’ and ‘0’). Such input signal will produce a similar output switched signal, however, the output signal can assume positive or negative values and it delivers the power needed to drive the motor. Figure (3) shows an example of a H-Bridge output signal, generated by a PWM input signal.

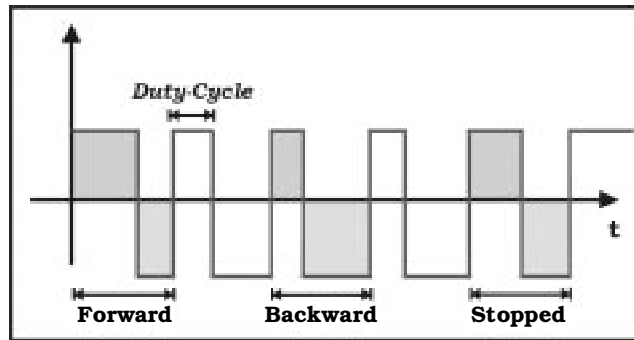


Figure 3. H-Bridge output signal.

Looking at fig.(3) it can be observed that when the green area is greater than the gray one (i.e., the duty-cycle is greater than 50%), the output signal produces a positive DC component. If a positive DC component is applied to a DC motor it will cause the motor to rotate on direct way (i.e., the motor will produce a forward movement). The backward movement is analog to the forward movement. When the green area equals to the gray area, there is no DC component, so the motor stops.

The driving subsystem was developed based on integrated circuits that implements tow H-Bridge. This IC was the L293. The L293 is capable of delivering output currents to 1A per channel. Since the motor peak current is about 700mA, the L293 fits on this requirement. The use of the L293 IC reduced the intern space that would be occupied by a discrete implementation of this circuit. Thus, the implementation of the driving subsystem hardware was simplified by the use of the L293, which requires only the addition of some free-wheeling diodes to protect the motors.

To generate the PWM control signals used to drive the motors, there was used a microcontroller PIC16F84. For each motor a different control signal is needed, in order to allow the independent driving. As the microcontroller is shared with the communication subsystem, the integration of both subsystems will be described in section 7. Figure(4) shows the driving system circuit scheme.

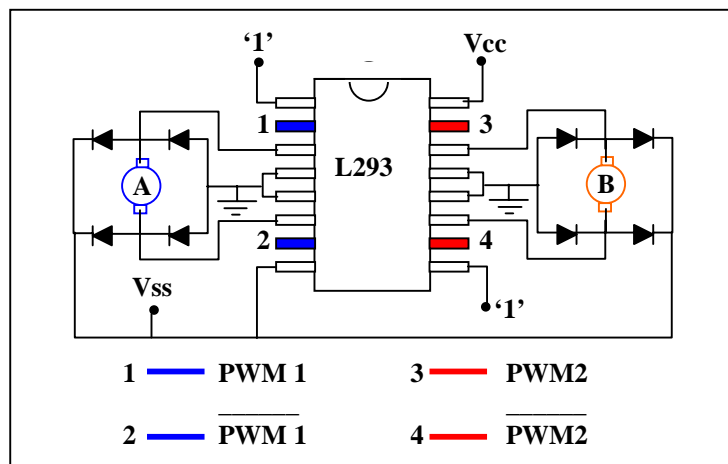


Figure 4. Driving circuit scheme.

On fig.(4) there are two DC motors ‘A’ and ‘B’, the L293 IC, the protection diodes, and it shows how the components are linked. The colored pins indicate the PWM and its complement input signals for each motor. The Vcc

symbol indicates the L293 supply voltage. The **V_{ss}** indicates the maximum voltage that can be delivered to the motors. One 9Volts battery is used as power supply to the driving subsystem and another one to the others devices on the embedded system.

5. The Communication Subsystem

In order to allow the host computer to remotely control the robot team, it is necessary a way of communication between them. There are many ways to achieve this goal, however, depending on the transmission rates and data length to be transmitted, some simple protocols can be adopted. After defining the communication protocol, it is necessary to define a protocol data unit, that is, a structure that defines how the data must be allocated to be transmitted and interpreted on the right way. After all, there must be defined the hardware necessary to achieve the communication. All those aspects will be considered in this section.

5.1 The RS-232C Protocol

The transmission rates requirement is not the main obstacle on the design of this subsystem, since the image capture rate is not so fast (30 frames per second). Thus, we can opt for a simple communication protocol, like the RS-232C, that allows communication rates up to 115.000 BPS. The RS-232C is the most common protocol used for digital communication through the personal computers serial interface. The computers and microcontrollers that implement this protocol have a built-in chip denominated UART (Universal Asynchronous Receiver Transmitter), that manages all information and configurations related to the protocol. The UART defines the maximum transmission rates that can be implemented.

The RS-232C is capable to realize serial bytes transfers, where each byte is preceded by a start bit, that indicates the transmission beginning, and succeeded by a stop bit, that indicates the end of the transmission. This kind of transmission is denominated "8N1". There are others transmission configurations that allow, for example, to define a parity bit, or even two stop bits.

5.2. The Protocol Data Unit (PDU)

Now that the communication protocol is defined, a protocol data unit (PDU) has to be specified. Since the RS-232C can transfer only one byte per transmission, the definition of a PDU whose length is greater than one byte would require more processing time, and techniques to recompose the PDU. The solution was to define a one-byte length PDU, containing all the information needed to interpret and execute a determinate command. The defined PDU is shown on fig.(5).

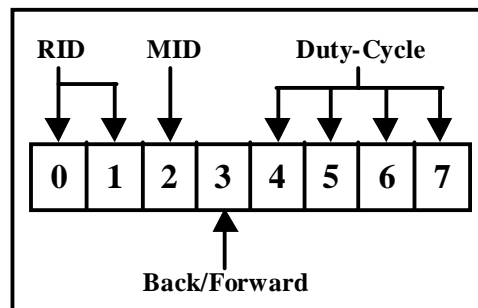


Figure 5. PDU specification.

The fig.(5) shows how the information needed to be transmitted is allocated in fields inside the PDU. The **RID** field indicates the bits used to identify each robot, that is, as each robot uses the same transmission frequency, it is necessary a way to distinguish if a data must be interpreted by a determinate robot. Since the teams have three players (robots), two identification bits are sufficient to distinguish each robot. The **MID** field is the bit used to identify which motor should be actuated. The **Back/Forward** field indicates if the specified motor should rotates on the direct way or on the reverse one. This operation requires just one bit. The reminding bits, 4 to 7, are assigned to motor velocity variation, given by the duty-cycle (as shown in section 4). Using four bits to specify the duty-cycle is equivalent to have sixteen velocity levels for each motor rotation way. This information should be read and interpreted by the embedded communication system (in each robot of the team). After the PDU definition it is possible to know how the programs for transmission and reception should be done, in order to interpret correctly the information transmitted.

5.3 Hardware Implementation Considerations

On the development of the hardware for transmission and reception, there were used specific communication modules. These modules are radio transmitters and receivers that allow the radio link between the host PC and the robot

team. Due to the limited robot dimension ($7.5 \times 7.5 \times 7.5 \text{ cm}^3$), the receiver module dimension should be very reduced. Another requirement was the disposability of equivalent modules with different work frequency, since there are two teams and they must work on different frequencies. After analyzing several modules, the communication modules adopted were the RT-4 (Transmitter) and RR-3 (Receiver) from Telecontrolli. Two sets of different frequencies (315 MHz and 433 MHz) were used.

The next decision was the choice of how the transfer should be done from the computer to the transmitter module. Two possibilities were considered: through the parallel interface, or through the serial interface. Each one presents advantages and disadvantages. Only the first method was implemented and tests are still being taken to verify the adequability of this method. Future works will include the serial interface implementation for comparison ends. To implement the parallel interface method, it was necessary a microcontroller in order to serialize the parallel data sent through the parallel interface.

The transmission software at the host PC was written in C and is responsible for the PDU construction and transmission through the parallel interface. A microcontroller reads that data and serialize it as the RS-232C would do (start-bit, data-bits, stop-bit), and send it to the radio transmitter module. This procedure is shown on fig.(6).

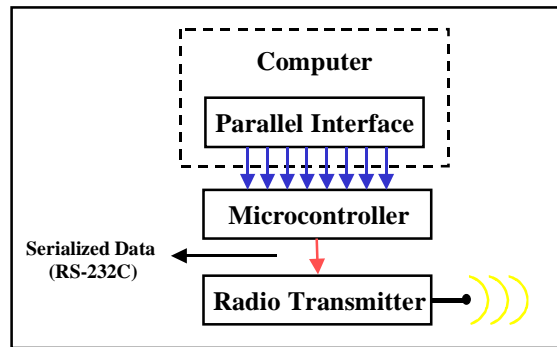


Figure 6. Transmission scheme.

In order to receive the transmitted data, the embedded communication system (in the robots) detects the start-bit, and starts to read the transmitted data. This embedded communication system is constituted of a radio receiver module and a microcontroller. There is another embedded system, the driving system, which shares the same microcontroller used by the communication system. After receiving the data transmitted, a program in the microcontroller verifies (through RID PDU field verification) if the received data should be interpreted by that robot, if so, it interprets the data and store them, for further use by the driving system. If RID PDU field verification fails, the system discards the received data. Figure (7) shows the transmitter hardware implementation.

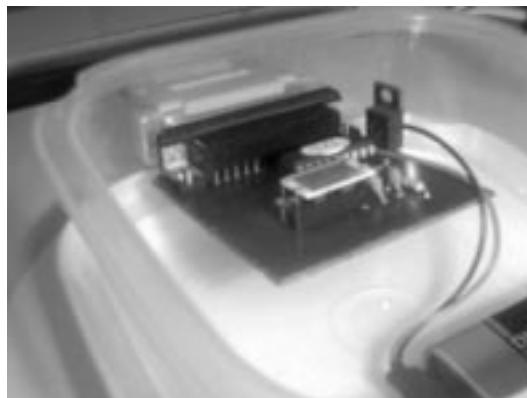


Figure 7. Transmitter hardware implementation.

6. Integrating Subsystems Using a Microcontroller

As mentioned in earlier sections, the embedded systems on each the robot is an integration of part of the communication subsystem and the driving subsystem. This integration is possible through a shared microcontroller that gets information from one subsystem and passes it to the other. The microcontroller tasks are: reception of data, interpretation of the received data and execution of the commands. These tasks can be classified into two groups:

1. Driving Task;
2. Communication Task.

These tasks will be detailed in the subsections that follow.

6.1 Driving Task

The driving task generates the PWM control signal to drive the motors 'A' and 'B', on an independent way. The algorithm that describes this task is not so complex. The algorithm just needs information about the duty-cycle, that is, the velocity level, for each motor, and the desired PWM work frequency. The frequency must be constant (a 500Hz frequency was adopted). There were defined variables to store these values and four output pins (two for each motor). The communication task passes these values and indicates which motor should actuate and the duty-cycle to be used. This task makes use of the microcontroller timer, and here is the point where the program gets complex. Some microcontrollers have a built-in PWM generator, which makes the implementation much simpler. For cost reasons, the microcontroller adopted doesn't offer those features, requiring the ability to program these functions.

6.2 Communication Task

The main programming obstacle takes place here, where the communication task also requires the use of the microcontroller timer. The communication task must receive correctly a RS-232C transmission and interpret the data received. This task makes use of the microcontroller interrupts. The interrupts allow a program to stop whenever they are called and to jump to an interrupt treatment routine. This feature was used to implement the reception task. When a start-bit is detected, it generates an interrupt, causing the system to jump to the interrupt routine that configures the program to sample the receiver pin in specific times, in order to correctly read the data transmitted. After the complete reception, the program starts the data verification, and signals that the reception task is over, and the system goes back to the normal driving task. When a data is being received, the driving task doesn't stop, the program schedules the tasks in order to execute both of them without system faults or lost of data. Due to the microcontroller limited resources (just one timer), the scheduler programming was a little complex, but it works perfectly.

7. Conclusions

The developed robot-soccer team was implemented with a minimum set of integrated circuits, resulting on a very compact onboard system (only three IC's in an unique printed circuit board of 50 x 40 mm²). This compactness allied to the absence of any mechanical reduction, leads to a mechanical robustness of the system. Furthermore, it guarantees sufficient free space for future expansions, such as more batteries, embedded sensors or a kick device (solenoid). Figure (8) shows a picture of the developed robot teams.



Figure 8. Picture of the developed robot-soccer teams.

The mechanical and driving characteristics of the developed robot system have achieved the Mirosoft-FIRA requirements and, besides, resulting in a good torque-velocity relation, which is very important from the competition view point.

The communication protocol makes use of one byte frame size, leading to a simple implementation, with low transmission delays, which is fundamental for this real-time application. In spite of this simplicity, this arrangement can be easily modified to include a limited amount of additional information.

All the mechanical and electronic parts were acquired in the national market (Brazil). The search for modular integrated components results on a extremely low cost implementation (less than US\$ 50,00 for each robot). This feature is one of the most important results of this project and is attractive for public research institutions, with limited financial resources.

8. References

- Costa, A. H. R., Pegoraro, R., Stolfi, G., Sichman, J. S., Pait, F. M. and Ferasoli, F. H., 1999. "GUARANÁ Robot-Soccer Team: Some Architectural Issues", Proceedings of IV Simpósio Brasileiro de Automação Inteligente, São Paulo, Brazil, pp. 457-462.
- FIRA, 1999. Fira Mirosoft Game Rules. Approved by the FIRA Executive Committee and disponible at <http://www.fira.net>.
- Kim, J. H. and Vadakkepat, P., 2000. "Multi-Agent Systems: A Survey from the Robot-Soccer Perspective". Intelligent Automation and Soft Computing, 6(1).
- Tavares, R. F., 1997. "Um Breve Histórico dos Robôs no País do Futebol". Disponible at <http://www.ia.cti.br/~leopard>.
- Veloso, M., Stone, P. and Han, K., 1998. "The CMUnited-97 Robot-Soccer Team: Perception and Multi-Agent Control", Proceedings of the Second International Conference on Autonomous Agents.
- Wyeth, G. and Brown, B., 2000. "Robust Adaptive Vision for Robot-Soccer". Mechatronics and Machine Vision in Practice.
- Wyeth, G., Brown, B. and Tews, A., 2000. "UQ RoboRoos: Preliminary Design of a Robot Soccer Team". Lectures Notes in AI: RoboCup'98.

SPECIFIC PROBLEMS OF THE CAM MECHANISMS KINETO-ELASTODYNAMICS

Oprisan Cezar

Technical University „Gh. Asachi” Iasi
Bdul D. Mangeron nr. 59 – 61, Iasi – 6600, Romania
e-mail: coprisan@diac.mt.tuiasi.ro

Leohchi Dumitru

Technical University „Gh. Asachi” Iasi
Bdul D. Mangeron nr. 59 – 61, Iasi – 6600, Romania
e-mail: dleohchi@diac.mt.tuiasi.ro

Abstract. The paper presents a method of dynamic analysis of the cam and flexible follower mechanisms, with motion laws with constant acceleration and trapezoidal acceleration. The influence of the ratio between the period of the motion law and of the free vibration, on the dynamic loading factor is studied, defined as the ratio between the maximum module of the follower acceleration considering the flexibility of the follower and the maximum module of the follower acceleration without considering the follower's flexibility. With the help of the numeric examples, ways of modifying the motion laws with acceleration which presents finished (finite) discontinuities by the motion law with continuous acceleration (the trapezoidal acceleration) and their implications on the dynamic behaviour of the mechanism, are studied.

Keywords. Dynamic, cam, flexible follower.

1. Introduction

The investigation in the dynamic analysis of the cam mechanisms, considering the flexible linkage, led to the developing of some physical and mathematical models, which characterize the behavior of the real mechanism. The physical model keeps the basic characteristics of the real system and it is easier to study.

The mathematical model, which describes the behavior of the physical model, is made of motion equations, which can be differential.

The number of freedom degree of the physical model gives the complexity of the mathematical model the number of the generalized coordinates. The solution of the mathematical model representing the motion law of the system is named the system response. The cam mechanism may be conceived (by approximation) as an oscillating system with a freedom degree. The physical and mathematical models are developed for two representative work hypothesis:

flexible follower and the rigid cam shaft (Rothbart, 1956; Shigley, 1961; Levitskii, 1988);

flexible camshaft and rigid follower (Ziya, 1989).

Of course, the technical literature in the domain of the dynamic analysis of the cam mechanisms, considering the flexibility of the linkage, assures a diversity of problems, which are raised by the working of these mechanisms. The influence of the clearances from the kinematic joints (Dubowsky, 1987; Yu Wang et al, 1996; Osman et al, 1987) and the investigation of the dynamic behavior of the cam mechanisms which work at high speed, considering the precision of the cam profile (Grewal and Newcombe, 1988; Mahyuddin and Midha, 1994), represent only some of the investigated problems.

This paper presents a way of dynamic analysis of flexible cam mechanisms, which work at high speed, considering the dynamic loading factor. This factor can be defined as a performance index, which can be used by the designer in the choice of the motion law. Motion laws with constant and trapezoidal acceleration are analyzed.

2. Dinamic model of the cam mechanism

The dynamic model of the cam and flexible follower mechanism is presented in Fig. (1). The parameters, which define this model, are:

m – the reduced mass of the follower;

k – the resulted rigidity, corresponding to the series (or series – parallel) system;

c_1 – the damping coefficient, specific to the external damping;

c_2 – the damping coefficient, specific to the internal damping of the follower;

F_0 – the preloading force of the compression spring;

F_t – technological force which drives on the mass m .

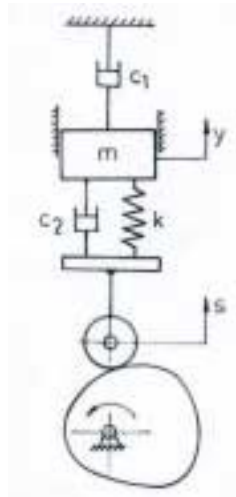


Figure 1. Dynamic model of the cam mechanism

The displacement of the flexible follower is noted with y and the displacement due to the cam profile is s Fig. (1).

2.1. Equation of motion

The equation determined by applying the dynamic equilibrium to the mass m , has the following form:

$$m \cdot \ddot{y} + (c_1 + c_2) \cdot \dot{y} + k \cdot y = c_2 \cdot \dot{s} + k \cdot s - F(t) , \tag{1}$$

where $F(t) = F_o + F_t$.

Using the generalized coordinates Lagrange, $q = y - s$, $\dot{q} = \dot{y} - \dot{s}$, $\ddot{q} = \ddot{y} - \ddot{s}$, the relation (1) becomes:

$$\ddot{q} + 2 \cdot (n_1 + n_2) \cdot \dot{q} + \omega_n^2 \cdot q = -\ddot{s} - 2 \cdot n_1 \cdot \dot{s} - \frac{F(t)}{m} , \tag{2}$$

where

$$\omega_n = \sqrt{\frac{k}{m}} \text{ - natural frequency of the linkage, radians / sec;}$$

$$n_1 = \frac{c_1}{2 \cdot m} \text{ - external damping factor;}$$

$$n_2 = \frac{c_2}{2 \cdot m} \text{ - internal damping factor.}$$

In the hypothesis of neglecting the friction forces and considering $F(t) = 0$, the equation (2) becomes:

$$\ddot{q} + \omega_n^2 \cdot q = -\ddot{s} . \tag{3}$$

For the rotation model specific to the mechanisms with flexible shaft, the motion equation is similar to the relation (3).

2.2 The dynamic loading factor

This factor is noted with k_a and represents the ratio between the maximum module of the output linkage acceleration, considering the flexibility of the mechanism linkages and the maximum module of the acceleration of the same linkage not considering the flexibility (Levitskii, 1988):

$$k_a = \frac{|\ddot{y}|_{\max}}{|\ddot{s}|_{\max}} \tag{4}$$

The k_a factor can also be considered as the force module, developed by the driven linkage, considering the flexibility and the maximum module of the inertia force without considering the flexibility of the linkages.

3. The dynamic analysis

The dynamic behavior is determined by the law of the follower acceleration variation and by the value of the dynamic loading factor. The most unfavorable dynamic phenomenon is the hard shock, which is recorded when the speed is discontinuous, and the acceleration tends to infinite. This phenomenon is similar to a knock and is followed by a very big overtaking of the mechanism.

When the acceleration has finite discontinuities and the over acceleration (pulse) tends to infinite, soft shocks are produced. This constitutes an important vibrations source, and their unfavorable effect can be evaluated with the help of the k_a dynamic loading factor.

In the continuation, the dynamic response of the follower and the value of the dynamic loading factor for the motion laws with constant and trapezoidal acceleration.

3.1 The motion law with constant acceleration

The real motion of the follower is defined by the differential equation (3). The motion law with constant acceleration Fig. (2) is given by the following equations:

for $0 \leq t \leq 0.5 \cdot t_1$:

$$s = \frac{a_1 \cdot t^2}{2} ;$$

$$\dot{s} = a_1 \cdot t ;$$

$$\ddot{s} = a_1 ,$$

where

$$a_1 = \frac{4 \cdot h}{t_1^2} ;$$

h – the follower course;

t_1 – the time of the rising phase;

for $0.5 \leq t \leq t_1$:

$$s = -\frac{a_1 \cdot t^2}{2} + a_1 \cdot t_1 \cdot t - h ;$$

$$\dot{s} = a_1 (t_1 - t) ;$$

$$\ddot{s} = -a_1 .$$

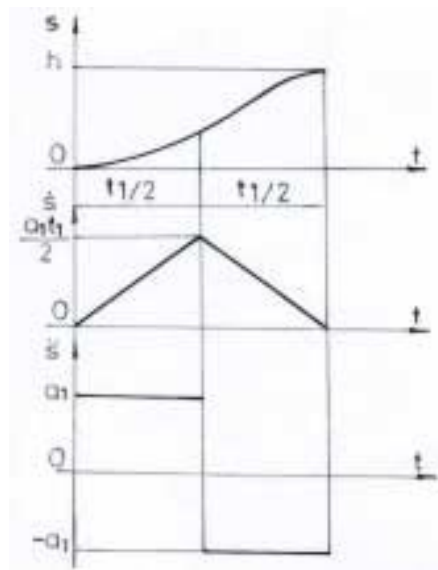


Figure 2. The equations of the motion law with constant acceleration

In the positive acceleration interval ($0 \leq t \leq 0.5 \cdot t_1$), the differential equation of the motion has the form:

$$\ddot{q} + \omega_n^2 \cdot q = -a_1 \tag{5}$$

The general solution of the equation (5) is:

$$q = C_1 \cdot \cos(\omega_n t) + C_2 \cdot \sin(\omega_n t) - \frac{a_1}{\omega_n^2} .$$

Using the initial conditions at $t = 0, q = 0, \dot{q} = 0$, the constant C_1 and C_2 become:

$$C_1 = \frac{a_1}{\omega_n^2}, C_2 = 0 .$$

The solution of the equation (5) can be presented:

$$\begin{aligned} q &= \frac{a_1}{\omega_n^2} (\cos(\omega_n t) - 1) ; \\ \dot{q} &= -\frac{a_1}{\omega_n} \sin(\omega_n t) ; \\ \ddot{q} &= -a_1 \cos(\omega_n t) . \end{aligned} \tag{6}$$

The real motion of the follower results from the replacing of the relation (6) in the relation defined by the generalized coordinates:

$$y = s + q ; \quad \dot{y} = \dot{s} + \dot{q} ; \quad \ddot{y} = \ddot{s} + \ddot{q} ,$$

resulting:

$$\begin{aligned} y &= \left(\frac{t^2}{2} + \frac{\cos(\omega_n t) - 1}{\omega_n^2} \right) \cdot a_1 ; \\ \dot{y} &= \left(t - \frac{\sin(\omega_n t)}{\omega_n} \right) \cdot a_1 ; \\ \ddot{y} &= (1 - \cos(\omega_n t)) \cdot a_1 . \end{aligned} \tag{7}$$

The dynamic loading factor k_a , for the positive acceleration period, has the following form:

$$k_a = \frac{|\ddot{y}|_{\max}}{a_1} = \frac{a_1 |1 - \cos(\omega_n t)|_{\max}}{a_1} = 2 .$$

The differential equation, for the negative acceleration period (Fig. 2), is obtained by replacing a_1 with $-a_1$, resulting:

$$\ddot{q} + \omega_n^2 \cdot q = a_1 . \tag{8}$$

The solution of the equation (8) is:

$$q = C_1 \cdot \cos(\omega_n t) + C_2 \cdot \sin(\omega_n t) + \frac{a_1}{\omega_n^2} .$$

For the initial conditions $t = 0, q = q_0, \dot{q} = \dot{q}_0$, the constant C_1 and C_2 become: $C_1 = q_0 - \frac{a_1}{\omega_n^2}, C_2 = \frac{\dot{q}_0}{\omega_n}$.

In this case the solution of the equation (8) has the following form:

$$\begin{aligned} q &= q_0 \cos(\omega_n (t - 0,5 t_1)) + \frac{\dot{q}_0}{\omega_n} \sin(\omega_n (t - 0,5 t_1)) + \frac{a_1}{\omega_n^2} (1 - \cos(\omega_n (t - 0,5 t_1))) ; \\ \dot{q} &= -\omega_n q_0 \sin(\omega_n (t - 0,5 t_1)) + \dot{q}_0 \cos(\omega_n (t - 0,5 t_1)) + \frac{a_1}{\omega_n} \sin(\omega_n (t - 0,5 t_1)) ; \\ \ddot{q} &= (a_1 - \omega_n^2 q_0) \cos(\omega_n (t - 0,5 t_1)) - \omega_n \dot{q}_0 \sin(\omega_n (t - 0,5 t_1)) . \end{aligned} \tag{9}$$

The relation (9) define the real motion of the follower for the period $0,5 \leq t \leq t_1$.

In the hypothesis of neglecting the damping, at the moment $t = 0,5 t_1$, can be considered $0,5 t_1 = p T + \alpha T$, where p is an integer of period T , of the vibration with its own natural frequency ω_n in radian / sec, and $\alpha \in (0,1)$.

In this case the period T is calculated with the relation:

$$T = \frac{t_1}{2(p + \alpha)}$$

or

$$\omega_n = \frac{4\pi}{t_1}(p + \alpha) .$$

Replacing these relations in (6), for $t = 0,5 t_1$ results

$$q_0 = \frac{a_1}{\omega_n^2}(\cos(2\pi\alpha) - 1);$$

$$\dot{q}_0 = -\frac{a_1}{\omega_n} \sin(2\pi\alpha) .$$

In table 1, values for q_0 and \dot{q}_0 in report with α are presented.

Table 1. Initial values calculated related with α

α	0	0.15	0.25	0.5	0.65	0.75
q_0	0	$-0.412 \frac{a_1}{\omega_n^2}$	$-\frac{a_1}{\omega_n^2}$	$-\frac{2a_1}{\omega_n^2}$	$-1.587 \frac{a_1}{\omega_n^2}$	$-\frac{a_1}{\omega_n^2}$
\dot{q}_0	0	$-0.809 \frac{a_1}{\omega_n}$	$-\frac{a_1}{\omega_n}$	0	$0.809 \frac{a_1}{\omega_n}$	$\frac{a_1}{\omega_n}$

For $\alpha = 0.5$ the acceleration of the \ddot{y} follower becomes:

$$\ddot{y} = -a_1 + 3a_1 \cos(\omega_n(t - 0.5t_1)),$$

and the dynamic loading factor k_a , has the value:

$$k_a = \frac{|\ddot{y}|_{\max}}{|\ddot{s}|_{\max}} = 4 .$$

To conclude, what is important for the dynamic behavior of the cam mechanism is the ratio between the time of the positive acceleration phase ($t = 0,5 t_1$) and the period of the free oscillation T. If it is an integer, then $k_a = 2$ for both phases (positive and negative accelerations) if it is not an integer, then it depends on the value of α , that is of the initial conditions of the phases with negative acceleration.

Generalizing, we can say that the „soft shocks”, which interfere when the motion law presents finite discontinuities at the accelerations, are corresponding with finite or bigger than 2 values of the dynamic loading factor.

3.2 The motion law with trapezoidal acceleration

The motion law with trapezoidal acceleration is obtained from the motion law with constant acceleration Fig. (2), on which linear parts, which avoid acceleration discontinuities, are added Fig. (3).

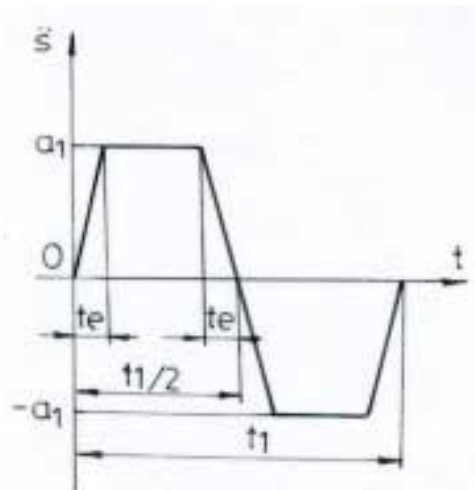


Figure 3. The motion law with trapezoidal acceleration

In Fig. (3) it is noticed that the acceleration is positive and equal with a_1 , on the constant portion positive and linear related to the motion law on the linear portion is:

$$s = \frac{a_1 \cdot t^3}{6 \cdot t_e} ; \quad \dot{s} = \frac{a_1 \cdot t^2}{2 \cdot t_e} ; \quad \ddot{s} = \frac{a_1 \cdot t}{t_e} .$$

The motion differential Eq. (3) for this phase, has the form:

$$\ddot{q} + \omega_n^2 \cdot q = -\frac{a_1}{t_e} \cdot t . \tag{10}$$

Applying the Laplace transform to the Eq. (10) an algebraic equation is obtained in connection to the Lq imaginary parts:

$$s_0^2 \cdot Lq + \omega_n^2 \cdot Lq = -\frac{a_1}{t_e} \cdot \frac{1}{s_0^2}$$

where s_0 is a complex variable, and the Lq imaginary part becomes:

$$Lq = -\frac{a_1}{t_e} \cdot \frac{1}{s_0^2 \cdot (s_0^2 + \omega_n^2)} .$$

Using the inverse Laplace transformation, the solution of the Eq. (10) is determined

$$q = -\frac{a_1}{\omega_n^3 \cdot t_e} \cdot (\omega_n \cdot t - \sin(\omega_n \cdot t)) .$$

Deriving the above equation in relation to the time it is obtained:

$$\dot{q} = -\frac{a_1}{\omega_n^2 \cdot t_e} \cdot (1 - \cos(\omega_n \cdot t)) ;$$

$$\ddot{q} = -\frac{a_1}{\omega_n \cdot t_e} \cdot \sin(\omega_n \cdot t) .$$

For the second phase in which the acceleration is constant and positive, the motion law is defined by the following relations:

$$s = \frac{a_1 \cdot t^2}{2} - \frac{a_1 \cdot t_e}{2} \cdot t + \frac{a_1 \cdot t_e^2}{6} ; \quad \dot{s} = a_1 \cdot t - \frac{a_1 \cdot t_e}{2} ; \quad \ddot{s} = a_1 .$$

In the case, the differential motion equation is given by the Eq. (5). The general solution of the Eq. (5) has the following form:

$$q = C \cdot \sin(\omega_n \cdot t + \theta) - \frac{a_1}{\omega_n^2} .$$

The constant C and the phase angle θ are calculated with the help of the initial conditions: $t = 0, \dot{q} = \dot{q}_0, \ddot{q} = \ddot{q}_0,$

$$q_0 = C \cdot \sin(\theta) - \frac{a_1}{\omega_n^2} ;$$

$$\dot{q}_0 = \omega_n \cdot C \cdot \cos(\theta) ,$$

resulting

$$C = \frac{a_1}{\omega_n^2} \sqrt{\left(q_0 + \frac{a_1}{\omega_n^2} \right)^2 + \frac{\dot{q}_0^2}{\omega_n^2}} ;$$

$$\tan \theta = \left(q_0 + \frac{a_1}{\omega_n^2} \right) \cdot \frac{\omega_n}{\dot{q}_0} .$$

Noting with $\xi = \frac{t_e}{T}$, and t_e - the time of the linear acceleration phase, and T the free vibration period, the constant C and angle θ are calculated with the following relations:

$$C = \frac{a_1 \cdot \sin(\pi \cdot \xi)}{\omega_n^2 \cdot \pi \cdot \xi} ; \quad \tan \theta = \frac{\sin(2 \cdot \pi \cdot \xi)}{\cos(2 \cdot \pi \cdot \xi) - 1} .$$

The general solution of the Eq. (5) becomes:

$$q = \frac{a_1}{\omega_n^2} \cdot \left(\frac{\sin(\pi \xi)}{\pi \xi} \sin(\omega_n t + \theta) - 1 \right). \quad (11)$$

Deriving relation (11), in relation to time, it results:

$$\dot{q} = \frac{a_1 \sin(\pi \xi)}{\omega_n \pi \xi} \cdot \cos(\omega_n \cdot t + \theta);$$

$$\dot{q} = -\frac{a_1 \sin(\pi \xi)}{\omega_n \pi \xi} \cdot \cos(\omega_n \cdot t + \theta) .$$

In this case, the real motion law of the flexible follower is:

$$y = s + q ;$$

$$\dot{y} = \dot{s} + \dot{q} ;$$

$$\ddot{y} = \ddot{s} + \ddot{q} .$$

We are interested in the influence of the phase of the linear acceleration on the dynamic loading factor k_a . Applying the relation (4), results:

$$k_a = 1 + \frac{|\sin(\pi \xi)|}{\pi \xi} ,$$

for

$$\ddot{y} = a_1 - \frac{a_1 \cdot \sin(\pi \xi)}{\pi \xi} \cdot \sin(\omega_n t + \theta) .$$

It is noticed that in the absence of the linear acceleration, the dynamic loading factor k_a is

$$k_a = \lim_{\xi \rightarrow 0} \left(1 + \frac{|\sin(\pi \xi)|}{\pi \xi} \right) = 2 .$$

and corresponds to the positive constant acceleration motion law.

4. Numerical results

For the numerical calculation the following data have been used: $t_1 = 0.022$ [s]; $t_e = 0.002$ [s]; $h = 0.025$ [m] and $\omega_n = 1570.7963$ [radians/sec].

In Fig. (4) the diagrams with the variation of the constant acceleration motion law Fig. (4 a, b, c) and of the vibration motion parameters which overlap the motion imposed by the cam profile Fig. (4, d, e, f) are presented. It is noticed that when $\alpha = 0.5$ a rising of the value of the vibration parameters is produced for the negative acceleration phase, in report with the parameters corresponding to the positive acceleration phase. For distinct representation of the diagrams $s(t)$ and $y(t)$ from Fig. (4a) and Fig. (5a) it used following relation $y = s + 5 \cdot q$.

This raising is analyzed by using the value of the dynamic loading factor, for the negative acceleration phase (k_a) in relation to the positive acceleration phase ($k_a = 2$).

The comparative analysis of the diagrams from Fig. (4) and (5) permit to remark the following conclusions.

The displacements determined by vibration motion of the flexible follower with the trapezoidal acceleration motion law, Fig. (5 d) in relation with the flexible follower displacement with constant acceleration motion, Fig. (4 d) are decreased with approximate 25%.

The speed and acceleration of the vibration motion of the flexible follower with trapezoidal acceleration motion law Fig. (5 e, f) are decreased also in relation with other motion law Fig. (4 e, f) with over 35%.

In conclusion, for follower with trapezoidal acceleration motion law it is noticed a bettering of the dynamic characteristics of the cam mechanism made evident by the variation of the dynamic load factor k_a , Fig. (6).

For $\xi = 0.5$, the dynamic loading factor has the value $k_a = 1.6366$.

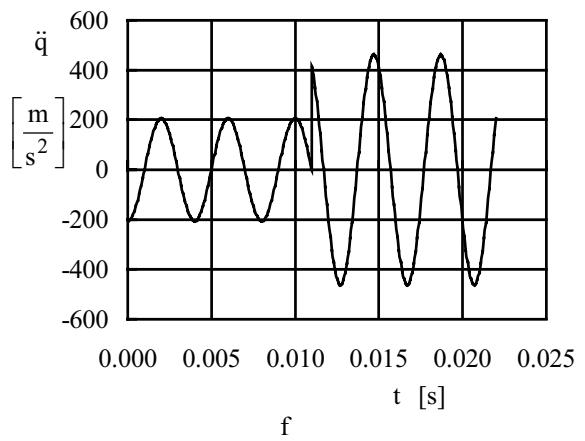
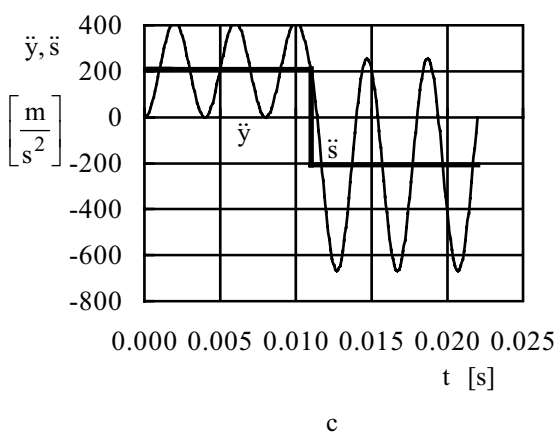
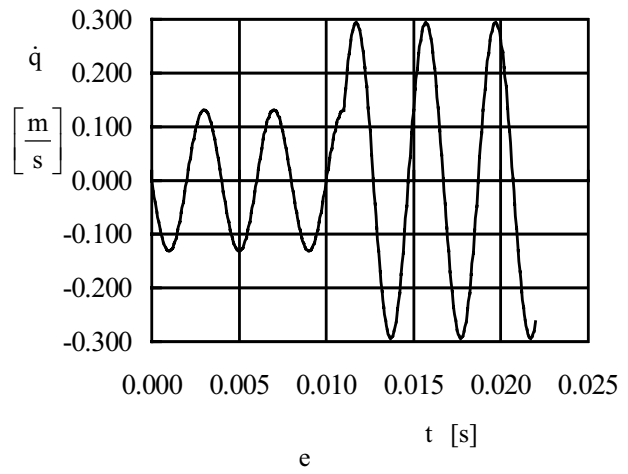
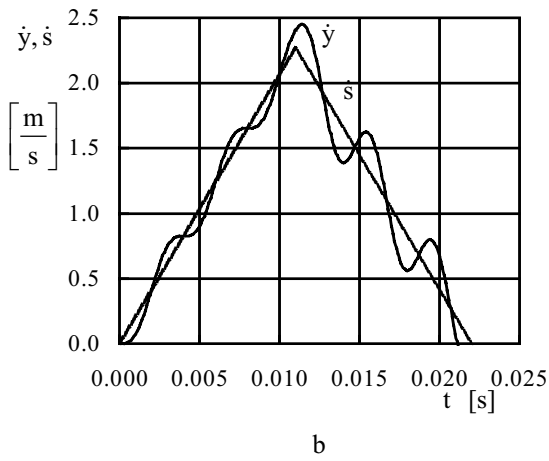
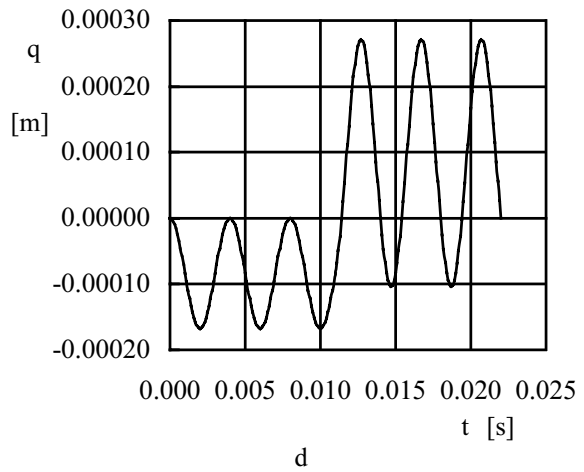
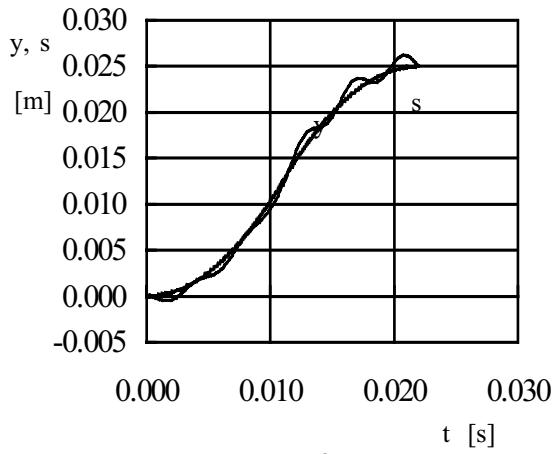


Figure 4. The variation of the parameters of the constant acceleration motion law

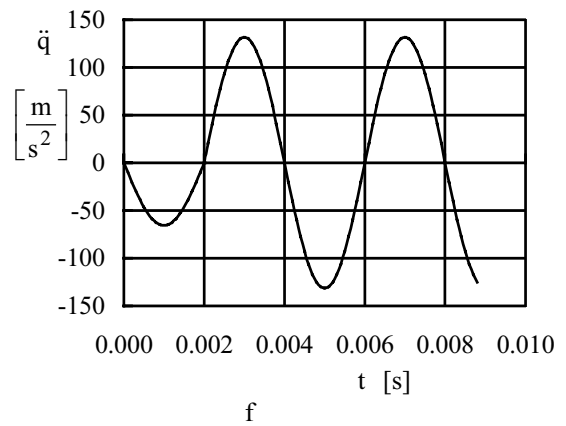
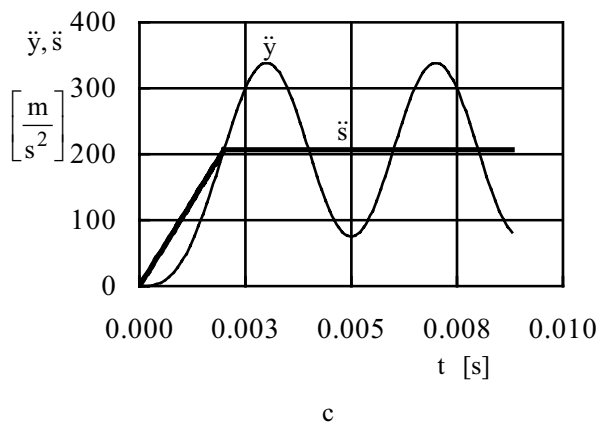
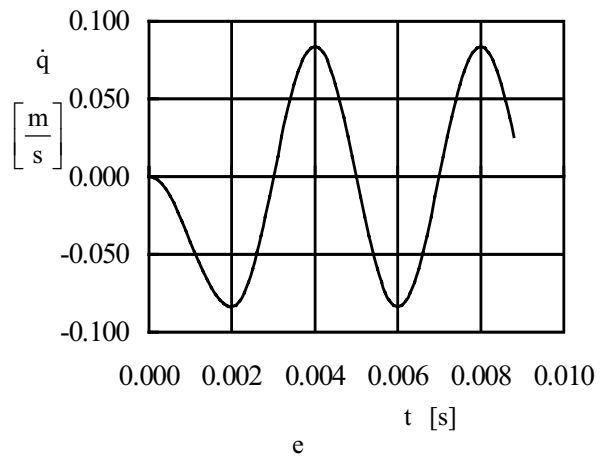
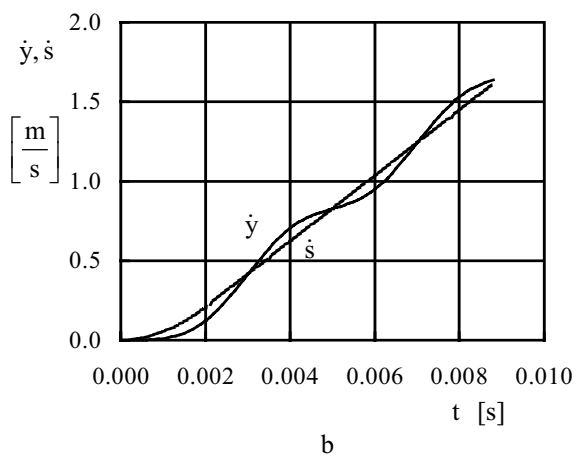
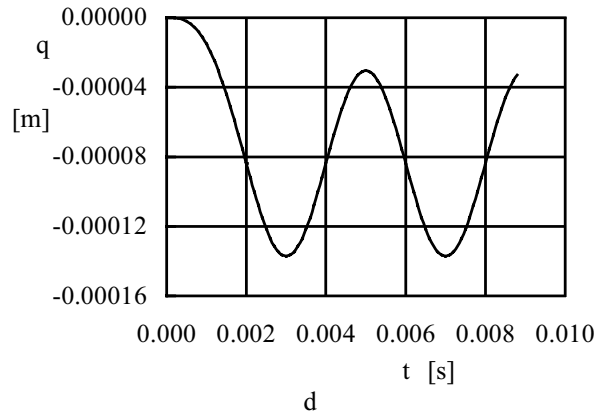
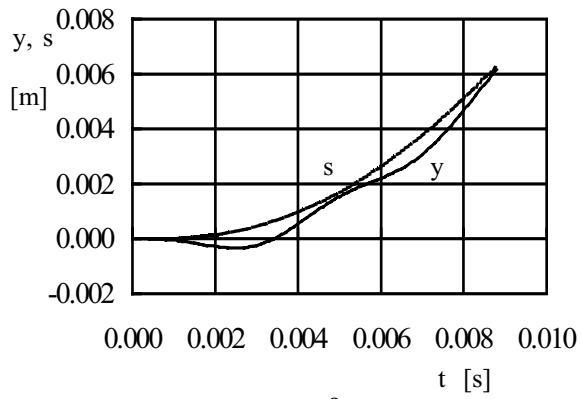


Figure 5. The variation of the parameters of the trapezoidal acceleration motion law

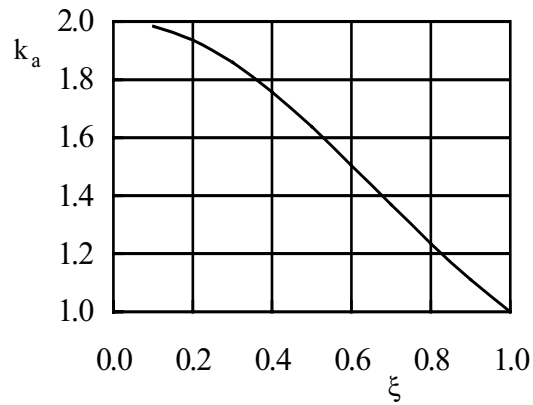


Figure 6. The variation of the dynamic loading factor for the trapezoidal acceleration motion law

5. Conclusions

The dynamic loading factor is performance index, which can be characterized from the dynamic, pointing of view the working of the cam mechanism.

For the motion laws, which present acceleration discontinuities, an important influence on the dynamic loading factor, is given by the ratio between the period of time of the motion law and the period of the free vibration of the follower.

If the motion law which presents acceleration discontinuities, is turned into continue acceleration law, by applying some linear portions (trapezoidal acceleration motion law) then the dynamic behavior of the cam mechanism is bettering, this fact being made evident by the value of the dynamic loading factor, which is smaller than the calculated value for the initial motion law (constant acceleration).

6. References

- Benson, H.T., 1996, „Principles of Vibration”, Oxford University Press.
- Dubowsky, S., Deck, J.F. and Costello, H., 1987, „The dynamic modeling of flexible spatial machine systems with clearance connections”, ASME – Transmissions and Automation in Design, vol. 109, pp. 87-94.
- Duca, C., Popovici, A., Oprisan, C., 1999, „The basis of the cam mechanisms design (in Romanian), Gh. Asachi, Iasi.
- Grewall, P.S., Newcombe, W.R., 1988, „A comparative study of cam motions for high – speed semi – rigid follower cam systems”, Transaction of the ASME, vol. 12, No. 3, pp. 121 – 128.
- Levitskii, N.I., 1988, „The vibrations of the mechanisms” (in Russian), Nauka, Moskva.
- Mahyuddin, A.I., Midha, A., 1994, „Influence of varying cam profile and follower motion event types on parametric vibration and stability of flexible cam follower systems”, Transaction of the ASME, vol. 116, pp. 298 – 305.
- Rothbart, H.A., 1956, „Cams”, John Wiley & Sons, Inc., New York.
- Sigley, H.A., 1961, „Dynamic Analysis of Machines”, McGraw – Hill Book Company Inc.
- Yu Wang, Zaiping Wang, 1996, „Dynamic analysis of flexible mechanisms with clearances”, Transactions of the ASME, vol. 118, pp. 592 – 594.

REDES NEURAI APLICADAS AO RECONHECIMENTO DE PADRÕES

Luiz Eduardo Nicolini do Patrocínio Nunes

Universidade de Taubaté – Dep. de Engenharia Mecânica
Rua Daniel Danelli, s/n – Taubaté – SP – CEP: 12060-440
lgpnunes@uol.com.br

Prof. Dr. Pedro Paulo Leite do Prado

Universidade de Taubaté – Dep. de Engenharia Elétrica
Rua Daniel Danelli, s/n – Taubaté – SP – CEP: 12060-440
pplprado@uol.com.br

Resumo. *Esse trabalho implementou um simulador de sistema de visão computacional, utilizando redes neurais artificiais para a classificação de objetos de formas geométricas (quadrado, círculo, triângulo e retângulo) contidos em imagens bidimensionais. As imagens para os testes foram adquiridas por dois equipamentos de captura: um scanner e uma câmera CCD. A tarefa de reconhecimento de imagens foi dividida em duas etapas: pré-processamento e classificação através de uma rede neural. Na etapa de pré-processamento, as imagens foram processadas por um algoritmo de detecção de bordas, utilizando o método de Sobel, que elimina o fundo contra o qual a imagem se encontra, deixando apenas o contorno do objeto a ser reconhecido. As imagens adquiridas pela câmera CCD sofreram binarização antes de serem processadas pelo algoritmo de detecção de bordas. As imagens binárias resultantes desse algoritmo foram, então, processadas pela Transformada de Fourier Log-polar, para tornar o sistema invariante a efeitos de translação, rotação e escala. Os resultados do algoritmo log-polar foram as entradas para a rede neural. Na etapa de classificação foi utilizada uma rede neural probabilística, por meio do software Matlab. Das imagens pré-processadas, duas amostras foram utilizadas para treinar a rede, e as restantes, para os testes de classificação. A rede classificou corretamente todas as imagens, sendo necessário, no caso das imagens adquiridas pela câmera, aumentar o número de amostras no vetor de treinamento. Os resultados do trabalho demonstraram que as redes neurais podem ser empregadas como ferramentas eficientes na tarefa de reconhecimento de formas geométricas.*

Palavras-chave: *redes neurais, reconhecimento de padrões, transformada log-polar, método de Sobel, formas geométricas.*

1. Introdução

As redes neurais artificiais têm evoluído intensamente nos últimos anos em decorrência da sua constante aplicação nas mais diversas áreas e atividades. Como exemplo, as redes neurais artificiais têm sido utilizadas em sistemas de controle e otimização, análise de aplicações financeiras, reconhecimento de voz, controle de manipuladores robóticos, classificação, visão artificial, reconhecimento de padrões em imagens, entre outros (Demuth e Beale, 1996).

Redes neurais artificiais são técnicas computacionais que apresentam um modelo matemático inspirado na estrutura neural de organismos inteligentes, e que adquirem conhecimento através da experiência. São compostas por diversas unidades computacionais (neurônios artificiais) operando em paralelo, interconectadas total ou parcialmente. Cada neurônio artificial executa um certo número de operações simples e transmite seus resultados aos neurônios vizinhos, com os quais possui conexão (Perelmuter et al., 1995).

2. Revisão da Literatura

Diversas pesquisas estão direcionadas à solução do problema de reconhecimento e classificação de objetos contidos numa imagem bidimensional.

He e Kundu (1991) usaram uma técnica de classificação baseada no modelo de Cadeias de Markov. Essa técnica não modela todo o padrão ou forma como um simples vetor de características, mas explora a relação entre segmentos consecutivos de um padrão a ser classificado.

Zwicke e Kiss (1983) e Perelmuter et al. (1995) usaram a Transformada de Mellin para tornar o sistema invariante a transformações de rotação, translação e escalonamento.

Moreira e Costa (1996) apresentaram um método para segmentação de imagens em cores que usa a classificação para agrupar pixels em regiões. A cromaticidade é usada como fonte de dados para o método, porque é normalizada e considera somente matiz e saturação, excluindo a componente luminância. A classificação é feita por uma rede neural do tipo Self-Organizing Map (SOM), que é empregada para obter as cromaticidades principais apresentadas na imagem. Cada pixel é classificado então, de acordo com as classes identificadas. A priori, o número de classes é desconhecido e a rede neural artificial é usada para determinar as classes principais.

Huang e Liu (1997) propuseram um sistema de reconhecimento de padrão híbrido, baseado em um Algoritmo Genético (AG) com uma rede neural modelo Hopfield (HP), que pode reconhecer padrões modificados pela transformação causada pela rotação, escala ou translação. O método proposto se apoia na técnica de aproximação poligonal, que extrai os vetores de características apropriados de dimensões especificadas, caracterizando uma forma.

Eom (1998) apresentou a classificação de formas planas usando características espectrais obtidas por modelamento autoregressivo de uma função de contorno, representada por funções periódicas definidas pelo comprimento e ângulo de um vetor radial da centróide para um ponto de contorno movendo-se em uma velocidade constante.

Nos trabalhos de Loesch (1998) e Davoli et al. (1999) foi proposto um sistema neural híbrido projetado para o reconhecimento de objetos singulares em imagens em tons de cinza com a utilização de uma transformada discreta com fundamentação matemática na transformação log-polar, que possui propriedades invariantes em relação à translação, rotação e escalonamento.

3. Metodologia

A visão computacional é efetuada através de um conjunto de transformações, que podem ser em paralelo, permitindo a extração dos aspectos invariantes das imagens, independentemente do ambiente em que estejam situadas. Essas invariâncias possibilitam o reconhecimento da imagem, permitindo a interação do sistema com as mesmas. A visão computacional procura extrair informações contidas numa imagem com o objetivo de classificá-las, caracterizá-las e/ou reconstruí-las (Perelmuter et al., 1995). O simulador do sistema de visão computacional por redes neurais foi implementado com a utilização do software Matlab, seu Toolbox de Processamento de Imagens, empregado no pré-processamento das imagens, e seu Toolbox de Redes Neurais, composto por uma coleção de funções para projetar e simular redes neurais.

3.1. Sistema de aquisição de imagens

O simulador do sistema de visão computacional tem o objetivo de reconhecer, através de uma rede neural artificial, objetos de formas geométricas diversas, independentemente de suas variações em relação à rotação, translação e escalonamento. As classes de objetos utilizadas nesse trabalho foram o quadrado, o círculo, o triângulo e o retângulo. Para a realização de testes foram criadas duas redes neurais, sendo que, para cada uma, as imagens foram obtidas utilizando-se dois equipamentos distintos de captura: uma câmera CCD e um *scanner*. Tanto as imagens da câmera CCD, seis de cada classe, quanto as imagens do *scanner*, quarenta de cada classe, foram adquiridas com resolução de 256 x 256 *pixels* e em 256 níveis de cinza. O motivo da criação de redes neurais distintas para a realização dos testes deve-se ao fato das imagens adquiridas pela câmera apresentarem distorções em relação às imagens adquiridas pelo *scanner*, conforme ilustra a Fig. (1).

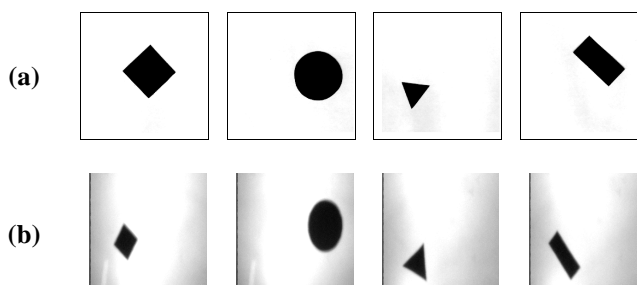


Figura 1. Variação entre as imagens adquiridas: (a) *scanner* e (b) câmera CCD.

3.2. Etapas da tarefa de reconhecimento

A tarefa de reconhecimento de objetos em imagens consistiu em duas etapas, conforme ilustra a Fig. (2).

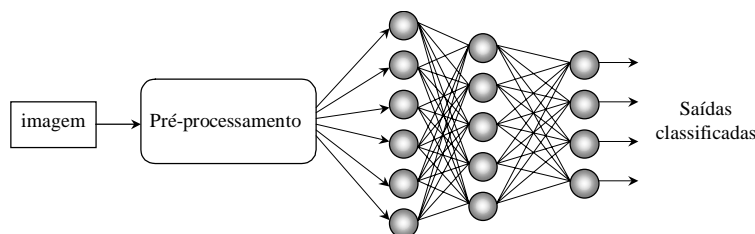


Figura 2. Etapas do reconhecimento.

3.2.1. Pré-processamento

A etapa de pré-processamento foi de grande importância nesse trabalho, pois executou uma série de transformações na imagem de modo a extrair suas características identificadoras, além de reduzir as dimensões dos dados de entrada da rede neural. Tais transformações foram obtidas com a utilização do método de Sobel e da Transformada Log-polar. O método de detecção de bordas (método de Sobel) foi aplicado como uma das fases da etapa do pré-processamento, visto

que interessa ao sistema somente a geometria do objeto a ser reconhecido. Informações desnecessárias como, por exemplo, o fundo contra o qual a imagem se encontra, são descartadas. Um sistema de visão computacional busca a classificação correta do objeto independentemente de seu tamanho, posição e rotação. Por essa razão, a transformada Log-polar também foi aplicada como uma das fases da etapa de pré-processamento. A transformada Log-polar torna o sistema invariante a efeitos de translação, rotação e escalonamento. O esquema ilustrado na Fig. (3) descreve as fases da etapa de pré-processamento.

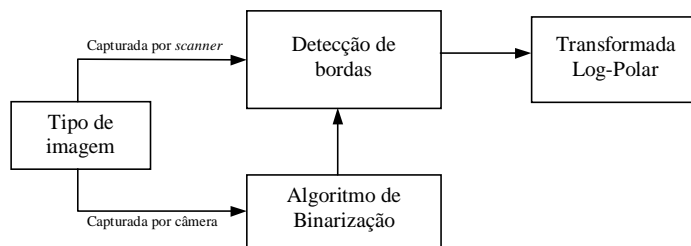


Figura 3. Fases do pré-processamento.

A etapa de pré-processamento foi iniciada com as imagens brutas de 256 x 256 pixels de resolução e com 256 níveis de cinza.

Para eliminar o fundo contra o qual a imagem se encontra, aplicou-se em cada imagem, o algoritmo de detecção de bordas.

No caso das imagens capturadas pela câmera CCD, as imagens foram inicialmente processadas por um algoritmo de binarização para transformá-las em imagens binárias (branco e preto), para posterior aplicação do algoritmo de Sobel.

O algoritmo de binarização calcula um valor de limiar da imagem e compara esse valor com os valores dos pixels da imagem, transformando em branco os valores maiores que o limiar e, em preto, os valores menores.

Esse procedimento foi adotado devido aos resultados não-satisfatórios obtidos com a aplicação direta do algoritmo de detecção de bordas sobre as imagens.

Para terminar a etapa de pré-processamento foi aplicado a todas as imagens o algoritmo que executa a Transformada Log-polar (Loesch, 1998), para tornar o sistema invariante.

A entrada para o algoritmo Log-polar é a imagem binária resultante do processo de detecção de bordas. Antes de processar a transformada da imagem, o algoritmo procura por um ponto inicial que garanta a invariância às diferenças de escala.

Nesse algoritmo, inicialmente são tomados apenas os valores não-nulos (cores não-pretas), colocados num vetor, para serem processados, a fim de evitar o processamento desnecessário com zeros. Nesse caso, o ganho de tempo foi grande, visto que, como estão sendo processados apenas os contornos da imagem, a maior parte da imagem não carrega informação significativa, e aparece em preto.

A invariância em relação à rotação foi obtida pela aplicação da transformada de Fourier nas colunas da matriz resultante da operação log-polar. A imagem resultante da operação log-polar tinha dimensões de 64 linhas por 15 colunas, o que reduziu o tempo de processamento.

3.2.2. Classificação

Na tarefa de reconhecimento das imagens foi utilizada uma rede neural probabilística, que é um tipo de rede empregado em problemas de classificação e aproximação de funções.

Essa rede possui apenas três camadas: camada de entrada, camada escondida e camada de saída, conforme ilustra a Fig. (4). A vantagem desse tipo de rede é ter um tempo de treinamento bastante reduzido, além de prescindir da determinação do número de neurônios na camada escondida.

O número de neurônios na camada escondida é definido de acordo com o número de vetores utilizados para o treinamento. Na camada de saída, a quantidade de neurônios é a mesma das classes a serem reconhecidas.

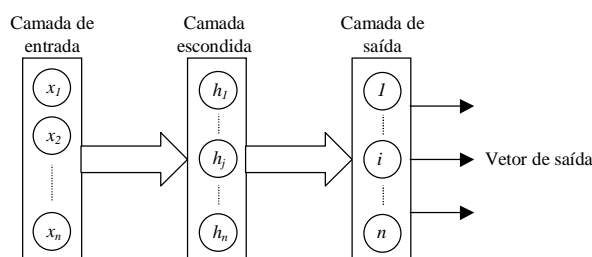


Figura 4. Camadas da rede neural probabilística.

Após a realização do pré-processamento de todas as imagens, os resultados da operação Log-polar foram utilizados como as entradas da rede neural. Inicialmente, esses resultados foram importados para o Matlab e transformados em vetores coluna.

Para cada classe de objetos (quadrado, circunferência, triângulo e retângulo) foram utilizadas duas amostras para o treinamento da rede e, as outras, para a tarefa de classificação propriamente dita.

A composição do conjunto de treinamento teve como base a observação dos gráficos gerados pela transformada Log-polar: para cada classe foram escolhidas as duas amostras que apresentaram maior variação entre si.

Na fase de aprendizado da rede, foi fornecida a saída desejada (alvo), de acordo com o procedimento de aprendizado supervisionado.

A Tabela (1) mostra as classes de objetos com os respectivos vetores de saída desejada. Na fase de operação, somente as entradas são fornecidas para a rede, que deverá retornar uma classificação coerente.

Tabela 1. Vetores de saída desejada.

Classe	Vetor de saída desejada
Quadrado	[1 0 0 0]
Circunferência	[0 1 0 0]
Triângulo	[0 0 1 0]
Retângulo	[0 0 0 1]

4. Resultados e Discussão

4.1. Resultados obtidos para as imagens adquiridas por scanner

Devido à grande quantidade de imagens adquiridas, 160 no total, foi selecionada apenas uma amostra de cada classe para a apresentação dos resultados obtidos na etapa de pré-processamento das imagens.

4.1.1. Pré-processamento

A Figura (5) apresenta as imagens capturadas por scanner e os resultados obtidos na etapa de pré-processamento, que inclui a detecção de bordas e a transformada log-polar.

Na fase de detecção das bordas, os resultados foram satisfatórios, dispensando a aplicação prévia de um algoritmo de binarização.

As imagens binárias geradas pelo algoritmo de detecção de bordas foram, então, processadas pela Transformada de Fourier Log-polar, para tornar o sistema invariante.

Como pode ser observado nas Fig. (5), as imagens geradas pelo algoritmo log-polar são praticamente idênticas para cada classe de objeto.

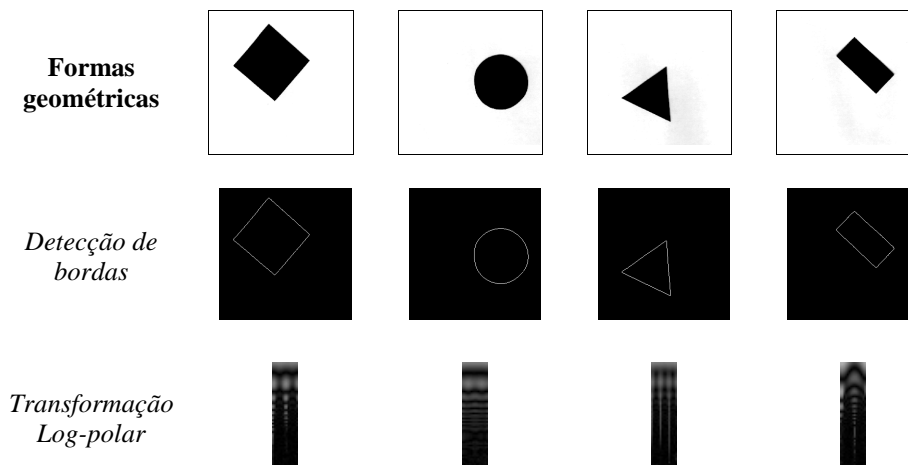


Figura 5. Imagens adquiridas por scanner e resultados do pré-processamento.

4.1.2. Classificação

Na etapa de classificação foram utilizadas duas amostras de cada classe para treinar a rede neural e as 38 restantes para os testes. A rede apresentou bons resultados no reconhecimento dos objetos nas imagens adquiridas por scanner,

classificando corretamente todos os objetos. A taxa de erro da rede foi estabelecida de acordo com o número ou porcentagem de objetos classificados errados. Os quadrados, circunferências, triângulos e retângulos, todos foram corretamente classificados, com porcentagem de acerto de 100%. Devido a esses resultados, não houve a necessidade de aumentar o número de amostras no vetor de treinamento da rede. A Tab. (2) apresenta a porcentagem de acerto da rede na classificação dos objetos nas imagens.

Tabela 2. Porcentagem de acerto da rede.

Classe	Porcentagem de acerto
Quadrado	100%
Circunferência	100%
Triângulo	100%
Retângulo	100%

4.2. Resultados obtidos para as imagens adquiridas pela câmera CCD

Esse trabalho propunha-se inicialmente a realizar testes com figuras capturadas por *scanner*, mas também foram realizados testes com algumas imagens adquiridas por uma câmera CCD, para se verificar o comportamento da rede no processo de reconhecimento.

4.2.1. Pré-processamento

A Figura (6) apresenta as imagens capturadas pela câmera CCD e os resultados obtidos na etapa de pré-processamento. Pode-se observar que as bordas detectadas aparecem bem definidas, o que justificou a utilização do algoritmo de binarização antes da detecção de bordas. As imagens geradas pelo algoritmo log-polar apresentam variações de maior ou menor intensidade conforme a classe de objetos, sendo o caso mais crítico a dos quadrados, que apresentaram três tipos de gráficos distintos entre si. Tais variações se devem ao fato das imagens adquiridas pela câmera apresentarem distorções na geometria dos objetos, o mesmo não ocorrendo nas imagens adquiridas por *scanner*.

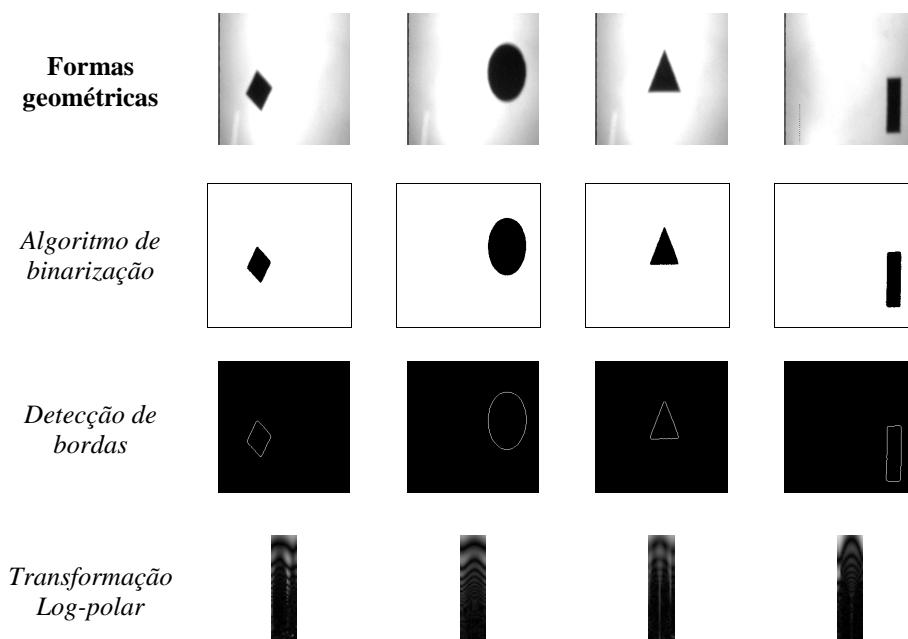


Figura 7. Imagens adquiridas pela câmera CCD e resultados do pré-processamento.

4.2.2. Classificação

Na etapa de classificação, foram utilizadas duas amostras de cada classe para treinar a rede neural, e as quatro restantes para os testes. A rede apresentou resultados razoáveis no reconhecimento dos objetos nas imagens adquiridas pela câmera.

Os círculos e os triângulos foram classificados com 100% de acerto, três quadrados foram classificados corretamente e, apenas dois triângulos obtiveram classificação correta. Outras duas amostras de cada classe no vetor de treinamento foram utilizadas, a fim de se verificar uma possível melhora no desempenho da rede.

A Tabela (3) apresenta os diversos vetores utilizados no treinamento da rede e as respectivas porcentagens de acerto na classificação dos objetos nas imagens.

Tabela 3. Vetores de treinamento e porcentagem de acerto.

Vetor de Treinamento	Porcentagem de acerto			
	Quadrado	Círculo	Triângulo	Retângulo
1	75%	100%	100%	50%
2	100%	100%	100%	75%
3	100%	100%	75%	25%
4	75%	100%	100%	100%
5	50%	100%	100%	50%

De acordo com a Tab. (3), em todos os vetores de treinamento utilizados, os círculos foram corretamente classificados e os triângulos também obtiveram boa classificação em todos os vetores. Para as imagens utilizadas nesse teste, pode-se observar que a rede treinada com os vetores 2 e 4, apresentou os melhores resultados na tarefa de classificação.

Procurando melhorar o desempenho da rede na tarefa de reconhecimento, ela também foi treinada com três amostras de cada classe no vetor de treinamento. As três imagens restantes foram utilizadas na tarefa de classificação. Foram escolhidas para o vetor de treinamento as imagens geradas pelo algoritmo log-polar que apresentaram maiores diferenças entre si. Os resultados dos testes estão apresentados na Tab. (4).

Tabela 4. Resultados da rede treinada com três amostras de cada classe.

Classe	Porcentagem de acerto
Quadrado	100%
Circunferência	100%
Triângulo	100%
Retângulo	100%

De acordo com o observado na Tab. (4), o acréscimo do número de amostras no vetor de treinamento, possibilitou à rede reconhecer todos os objetos, melhorando, assim, o seu desempenho para 100% de acerto.

5. Conclusões

O objetivo proposto de implementação do simulador de um sistema de visão computacional em Matlab foi atingido, pois a rede neural utilizada classificou corretamente os objetos contidos nas imagens. O trabalho mostrou a viabilidade da utilização das redes neurais artificiais na tarefa de classificação de objetos em imagens. A importância desse trabalho se deve ao fato de ter apresentado um sistema de visão que pode ser implementado e utilizado em aplicações reais.

O Método de Sobel para a detecção das bordas dos objetos e a Transformada Log-Polar para extrair as variações de rotação, posição e escala foram considerados suficientes e eficientes como ferramentas na etapa de pré-processamento das imagens.

As imagens adquiridas por scanner se mostraram apropriadas para o teste dos algoritmos: tiveram um pré-processamento mais rápido e a escolha das matrizes resultantes da transformação log-polar para treinar a rede neural foi praticamente aleatória, visto que, as variações entre as mesmas eram quase imperceptíveis. Essas imagens apresentaram ótimos resultados tanto na etapa de pré-processamento, quanto na etapa de classificação.

Para as imagens adquiridas pela câmera, foi necessário o emprego, na etapa de pré-processamento, da binarização, devido à distorção ocasionada pela câmera utilizada. As matrizes da transformação log-polar tiveram que ser escolhidas para compor o vetor de treinamento, visando a atingir melhores resultados da rede na tarefa de classificação. Na etapa de classificação, a rede teve também que ser treinada com um número maior de amostras no vetor de treinamento. Com o acréscimo de amostras de cada classe de objeto no vetor de treinamento, a rede foi capaz de reconhecer todos os objetos, validando perfeitamente o sistema de visão computacional desenvolvido.

6. Referências

- Davoli, R., Tamburini, F., 1993, "Data Algorithm: A Numerical Method to Extract Shape Information from Gray Scale Images", Technical Report UBLCS-93-15, University of Bologna, Italy.
- Demuth, H., Beale, M., 1998, "Neural Network Toolbox User's Guide", The Mathworks Inc.
- Eom, K. B., 1998, "Shape Recognition Using Spectral Features", Pattern Recognition Letters, No. 19, pp. 189-195.

- He, Y., Kundu, A., 1991, "2-D Shape Classification using Hidden Markov Model", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 13, No. 11.
- Huang, J. S., Liu, H. C., 1997, "Object Recognition using Genetic Algorithms with a Hopfield's Neural Network", Expert Systems With Applications, Vol. 13, No. 3, pp. 191-199.
- Loesch, C., 1998, "Reconhecimento de Objetos em Imagens Bidimensionais", Rev. Ciências Exatas, Taubaté, Vol. 4, No. 1/2, pp. 31-38.
- Moreira, J., Costa, L. F., 1996, "Neural-based Color Image Segmentation and Classification using Self-organizing Maps", Anais do IX SIBGRAPI, pp. 47-54.
- Perelmuter, G., Carrera, E. V., Vellasco, M., Pacheco, A., 1995, "Reconhecimento de Imagens Bidimensionais utilizando Redes Neurais Artificiais", Anais do VIII SIBGRAPI, pp. 197-203.
- Zwicke, P. E., Kiss, I. Jr., 1983, "A New Implementation of the Mellin Transform and its Application to Radar Classification of Ships", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-5, No. 2.

NEURAL NETWORKS APLLIED TO THE PATTERN RECOGNITION

Luiz Eduardo Nicolini do Patrocínio Nunes

University of Taubaté – Department of Mechanical Engineering
Rua Daniel Danelli, s/n – Taubaté – SP – CEP: 12060-440
lgpnunes@uol.com.br

Prof. Dr. Pedro Paulo Leite do Prado

University of Taubaté – Department of Electrical Engineering
Rua Daniel Danelli, s/n – Taubaté – SP – CEP: 12060-440
pplprado@uol.com.br

***Abstract.** This work developed a computational vision system simulator, using artificial neural networks for the geometric shapes classification (square, circle, triangle and rectangle) contained in bidimensional images. The images for the tests were acquired from two capture devices: a scanner and a CCD camera. The images recognition task was divided in two phases: pre-processing and classification through a neural network. In the pre-processing stage, the images were processed by a border detection algorithm, using the Sobel method, that eliminated the image background, just leaving the contour of the object to be recognized. The images acquired by the CCD camera suffered binarization before the processing by the border detection algorithm. The resulting binary images were, then, processed by the Fourier Log-polar transform, to make the system invariant to translation, rotation and scale effects. The results of the log-polar algorithm were the inputs for the neural network. In the classification phase, a probabilistic neural network was adopted, using the Matlab software. Two samples of the processed images were used to train the network, and the remaining ones, for the classification tests. The network classified correctly all the images. In the case of the images acquired by the camera, the number of samples in the training vector had to be increased. The results of this work demonstrated that the neural networks can be used as efficient tools in the geometric shapes recognition task.*

***Key-words:** neural networks, pattern recognition, log-polar transform, Sobel method, geometric shapes.*

CINEMÁTICA DE EQUILÍBRIO PARA UM ROBÔ QUADRÚPEDE

Antônio Bento Filho

Depto. Eng. Mecânica
CTC/UFES
abento@ele.ufes.br

Paulo Faria Santos Amaral

Depto. Eng. Elétrica
CTC/UFES
paulo@ele.ufes.br

Benedito Geraldo Miglio Pinto

Depto. Eng. Mecânica
CTC/UFES
automatica@automatica.com.br

Resumo. O presente trabalho descreve a implementação de um modelo cinemático para a determinação dos ângulos de rolagem, mergulho e guinada e do plano definido pelas coordenadas de apoio das patas de um robô móvel. Estas informações serão utilizadas em tempo real, para a supervisão de equilíbrio de um robô quadrúpede. Utiliza as coordenadas de 3 patas em apoio para obter o referencial de bordo do robô, que representa a rotação da plataforma em relação ao plano de apoio. A bordo do robô, sensores de inclinação em relação à horizontal permitirão formar a matriz de rotação inercial da plataforma. Com o modelo, obtém-se também a equação do plano de apoio, que é utilizada determinar a coordenada vertical de pouso de uma pata no ar e as coordenadas da projeção do CDG da plataforma no plano de apoio. O modelo foi desenvolvido para robôs que caminham em equilíbrio estaticamente estável com 3 patas sempre em apoio, porém, pode ser estendido para robôs de 6 ou mais patas. Com auxílio do MatLab, o modelo foi testado para algumas configurações de inclinação da plataforma, tendo apresentado resultados satisfatórios para sua futura implementação na supervisão do equilíbrio do robô.

Palavras chave: Legged robot, attitude, stability margin, quadruped

1. Introdução

Robôs quadrúpedes tem um número mínimo de pernas que permitem que eles caminhem em equilíbrio estaticamente estável. Mesmo quando uma das pernas estiver em movimento aéreo para novo posicionamento, é possível manter o robô em equilíbrio apoiado isostaticamente nas três outras pernas que estão em apoio. Robôs com maior número de pernas podem apresentar melhor estabilidade, pois é possível ter mais de três pernas em apoio durante o caminhar, porém, necessitam maior número de pernas, sensores e atuadores, o que piora sua relação peso-potência e aumenta seu custo. Além disso, a distribuição de forças de apoio em robôs que caminham com mais de 3 patas em apoio é hiperestática, ou seja, existem mais equações de equilíbrio que reações a determinar. Diversos robôs quadrúpedes importantes foram construídos recentemente tais como Ishihara 1998; Hirose & Kato, 1998; Kitano & all, 1998; dentre outros, e também diversas pesquisas em locomoção e controle de robôs quadrúpedes vem sendo realizadas tais como Martins Filho, 1998; Schmiedeler & Waldron, 1999; Smith & Berkemeyer, 1997; e Sukhatme & all, 1997; dentre outros.

O modelo aqui apresentado foi desenvolvido para ser utilizado no controle de inclinação da plataforma de um robô quadrúpede que caminha em equilíbrio estaticamente estável. O robô poderá caminhar no plano, em rampa ascendente ou descendente, subir ou descer um degrau e virar curvas caminhando.

Para efetuar estes movimentos e comandar as patas em vôo para novo posicionamento ou durante a fase de apoio, é necessário que o sistema de controle disponha de um modelo capaz de determinar a inclinação relativa entre a plataforma e o piso. Conhecidos os ângulos das juntas de cada perna em tempo real, torna-se possível determinar as coordenadas das patas em relação ao referencial de bordo do robô. Com estas coordenadas, o modelo proposto permite determinar os ângulos de rolagem, mergulho e guinada e a matriz de rotação da plataforma em relação a um referencial fixo na partida. O movimento do robô pressupõe o acionamento das pernas segundo uma trajetória determinada no espaço cartesiano, mapeada no espaço das articulações pela cinemática inversa da perna.

O modelo cinemático é construído com o robô apoiado em um plano horizontal. A orientação da plataforma e a posição do CDG são corrigidos em tempo real, em função da inclinação detectada pelos sensores a bordo do robô. Através do modelo, robô poderá ser acionado para ajustar a plataforma à orientação desejada, o que permitirá, determinar a posição de apoio da quarta pata que está no ar.

O presente trabalho utiliza conhecimento e ferramentas de geometria analítica e análise vetorial para construção de um modelo matemático capaz de permitir que o sistema de controle de um robô móvel a pernas possa reagir ou mesmo antecipar-se à mudanças no piso. Sua abrangência compreende, principalmente, as aplicações de robôs móveis a pernas, podendo seu princípio ser adaptado para o estudo de manipulação de grandes objetos por mais de um robô manipulador.

2. Orientação da plataforma do robô

Para efeito da análise cinemática, a plataforma e as pernas do robô são consideradas um sistema de corpos rígidos. Os pontos de apoio das patas, que são fornecidos pelo módulo de comando das pernas, formam sempre um plano que será paralelo a superfície de apoio.

Na partida, o robô posiciona a plataforma a uma altura h do solo, comandando ângulos iguais para juntas equivalentes de cada perna, de acordo com a cinemática inversa. Se o robô estiver apoiado em um plano horizontal, detectado pelos sensores a bordo, a matriz de rotação do referencial robô em relação ao referencial fixo, neste instante, será a matriz identidade de ordem 3, pois os ângulos de rolagem, mergulho e guinada serão nulos. Se o robô não estiver em um plano horizontal, os sensores de bordo indicarão os ângulos de rolagem, mergulho e guinada não nulos, que permitirão obter a matriz de rotação do referencial robô em relação ao referencial fixo, neste instante.

Então, na partida do repouso, definimos o referencial ${}^0x_0{}^0y_0{}^0z_0$, fazendo coincidirem os eixos 0z_0 do referencial fixo e ${}^r z_0$ do referencial robô e mantendo-se paralelos, e com mesmo sentido, os eixos ${}^0x_0/{}^f x_0$ e ${}^0y_0/{}^f y_0$ do referencial fixo e do referencial robô, respectivamente, conforme Fig. (1).

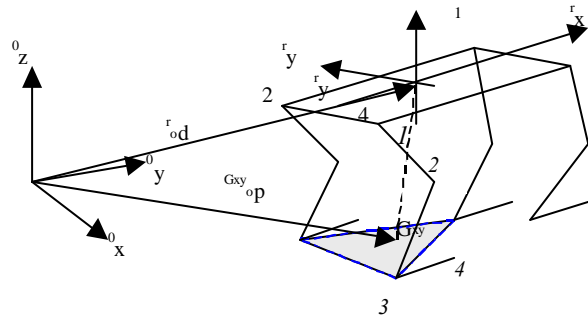


Figura 1: Referenciais utilizados e variáveis do robô

Em relação ao referencial robô, os vetores posição dos pontos de apoio das patas em relação ao ponto de fixação da perna na plataforma são definidos por:

$$\begin{aligned} {}^r p_1 &= \begin{bmatrix} \frac{1}{4}x_1 + \frac{L}{2} & \frac{1}{4}y_1 + \frac{W}{2} & \frac{1}{4}z_1 \end{bmatrix}^t \\ {}^r p_2 &= \begin{bmatrix} \frac{2}{4}x_2 - \frac{L}{2} & \frac{2}{4}y_2 + \frac{W}{2} & \frac{2}{4}z_2 \end{bmatrix}^t \\ {}^r p_3 &= \begin{bmatrix} \frac{3}{4}x_3 + \frac{L}{2} & \frac{3}{4}y_3 - \frac{W}{2} & \frac{3}{4}z_3 \end{bmatrix}^t \\ {}^r p_4 &= \begin{bmatrix} \frac{4}{4}x_4 - \frac{L}{2} & \frac{4}{4}y_4 - \frac{W}{2} & \frac{4}{4}z_4 \end{bmatrix}^t \end{aligned} \tag{3.a, b, c, d}$$

Existem 4 combinações possíveis de 3 patas no solo: 123, 124, 134 e 234, mostradas na figura 3.2 a seguir.

Os vetores posição unem os pontos de apoio e definem ao plano de apoio das patas. Efetuando o produto vetorial dos vetores posição nas configurações da Figura (2), na ordem dos vetores paralelos para os perpendiculares ao eixo longitudinal da plataforma, de acordo com a regra da mão direita, obtemos o vetor normal ao plano da plataforma, em cada configuração de apoio, na forma:

$$\begin{aligned} {}^{123}r p &= {}^r p_{21} \times {}^r p_{31} = [S({}^r p_{21})] [{}^r p_{31}] \\ {}^{124}r p &= {}^r p_{21} \times {}^r p_{42} = [S({}^r p_{21})] [{}^r p_{42}] \\ {}^{134}r p &= {}^r p_{43} \times {}^r p_{31} = [S({}^r p_{43})] [{}^r p_{31}] \\ {}^{234}r p &= {}^r p_{43} \times {}^r p_{42} = [S({}^r p_{43})] [{}^r p_{42}] \end{aligned} \tag{4.a, b, c, d}$$

onde o super-índice esquerdo e o sub-índice direito do vetor ${}^r p_{nm}$ identificam o referencial e o sentido do vetor posição, respectivamente, e $S({}^r p_{nm})$ é a matriz do operador produto vetorial.

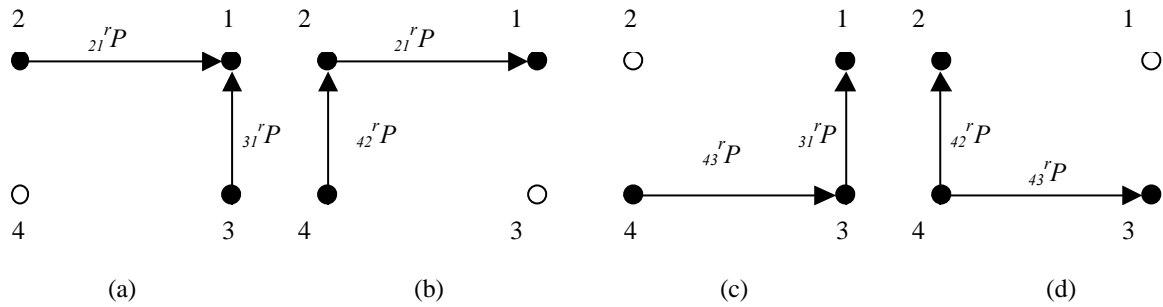


Figura 2: Configurações para 3 patas no solo: (a)patas 1,2 e 3; (b)patas 1,2 e 4; (c)patas 1, 3 e 4 e (d)patas 2, 3 e 4. Apesar de dispostos de forma retangular os pontos de apoio podem ocupar qualquer coordenada no espaço de alcance das pernas no plano de apoio.

O vetores unitários são então:

$${}_{mnp}^r n = \frac{{}_{mnp}^r P}{\sqrt{{}_{mnp}^r P \cdot {}_{mnp}^r P}} = \begin{bmatrix} {}_{mnp}^r n^x & {}_{mnp}^r n^y & {}_{mnp}^r n^z \end{bmatrix}^t = {}_{mnp}^r z \quad (5)$$

onde $mnp = 123, 124, 134$ ou 234 ., ${}_{mnp}^r n$ é o vetor unitário na direção ${}_{mnp}^r z$ e ${}_{mnp}^r n^x, {}_{mnp}^r n^y$ e ${}_{mnp}^r n^z$ são os cosenos diretores do referencial robô. O super-índice à direita em ${}_{mnp}^r n^x$ identifica a componente x, y ou z do vetor ${}_{mnp}^r n$ que é o próprio vetor unitário ${}_{mnp}^r z$ do referencial robô.

Definimos os vetores unitários na direção ${}_{mnp}^r x$ a partir dos vetores posição que unem os pontos de apoio de 2 patas de um mesmo lado do robô em cada configuração, conforme a seguir:

$$\begin{aligned} {}_{123}^r x &= \frac{{}_{21}^r P}{\sqrt{{}_{21}^r P \cdot {}_{21}^r P}} = \begin{bmatrix} {}_{123}^r x^x & {}_{123}^r x^y & {}_{123}^r x^z \end{bmatrix}^t \\ {}_{124}^r x &= {}_{123}^r x; \\ {}_{134}^r x &= \frac{{}_{43}^r P}{\sqrt{{}_{43}^r P \cdot {}_{43}^r P}} = \begin{bmatrix} {}_{134}^r x^x & {}_{134}^r x^y & {}_{134}^r x^z \end{bmatrix}^t \\ {}_{234}^r x &= {}_{134}^r x \end{aligned} \quad (6.a, b, c, d)$$

O vetor unitário ${}_{mnp}^r y$, que completa o referencial, é obtido pelo produto vetorial do vetor unitário normal ${}_{mnp}^r n$ da Eq. (5) pelo vetor unitário ${}_{mnp}^r x$ na direção x , da Eq. (6), conforme a seguir:

$${}_{mnp}^r y = S({}_{mnp}^r n) \wedge ({}_{mnp}^r x) = \begin{bmatrix} {}_{mnp}^r y^x & {}_{mnp}^r y^y & {}_{mnp}^r y^z \end{bmatrix}^t \quad (7.a, b, c)$$

sendo que :

$$\begin{aligned} {}_{134}^r y &= {}_{123}^r y \\ {}_{234}^r y &= {}_{124}^r y \end{aligned} \quad (8.a, b)$$

A matriz de rotação, cujas colunas são as componentes dos vetores unitários nas direções x, y e z do referencial fixo à plataforma, é dada por:

$$[{}_{i-1} R_{mnp}^i] = \begin{bmatrix} {}_{mnp}^r x^x & {}_{mnp}^r y^x & {}_{mnp}^r n^x \\ {}_{mnp}^r x^y & {}_{mnp}^r y^y & {}_{mnp}^r n^y \\ {}_{mnp}^r x^z & {}_{mnp}^r y^z & {}_{mnp}^r n^z \end{bmatrix} \quad (9)$$

onde os sub índices à direita e à esquerda de ${}_{i-1}R^i_{mnp}$ identificam dois instantes consecutivos de comando de posição das patas. A matriz de rotação Eq. (9) foi obtida a partir das coordenadas dos pontos de apoio das patas em relação ao referencial robô.

Na partida do robô, os referenciais fixo e robô são paralelos. Então, entre os instantes 0 da partida, com o robô ainda em repouso, e 1 , imediatamente após o primeiro comando de movimento das patas, a matriz (9) representa também a rotação da plataforma em relação ao referencial fixo. Após o segundo comando, a matriz de rotação obtida será a rotação do referencial robô em relação a ele próprio na posição anterior, após o primeiro comando. Assim, sucessivamente, após o comando $i-1$, a matriz de rotação obtida será a rotação do referencial robô em relação a ele próprio, ocorrida entre os comandos $i-1$ e i .

Então, a matriz de rotação da plataforma após o comando i em relação ao referencial fixo será dada por:

$$[{}^0R^i_{mnp}] = [{}^sR^0_{mnp}] [{}^hR^0_{mnp}] [{}^0R^1_{mnp}] [{}^1R^2_{mnp}] \dots [{}^{i-1}R^i_{mnp}] \quad (10)$$

onde a matriz ${}^sR^0_{mnp}$ é a matriz de rotação da plataforma em relação à horizontal no instante 0 . Esta matriz é matriz identidade de ordem 3, se o robô está apoiado em um plano horizontal neste instante

Em termos dos ângulos de rolagem ${}^r\mathbf{f}$, de mergulho ${}^r\mathbf{q}$ e de guinada ${}^r\mathbf{y}$, define-se a matriz de rotação (Spong & Vidyasagar, 1992), conforme a seguir:

$$[{}^0R] = [R_{z,\mathbf{f}}][R_{y,\mathbf{q}}][R_{x,\mathbf{y}}] = \begin{bmatrix} c^0\mathbf{f}c^0\mathbf{q} & -s^0\mathbf{f}c^0\mathbf{y} + c^0\mathbf{f}s^0\mathbf{q}s^0\mathbf{y} & s^0\mathbf{f}s^0\mathbf{y} + c^0\mathbf{f}s^0\mathbf{q}c^0\mathbf{y} \\ s^0\mathbf{f}c^0\mathbf{q} & c^0\mathbf{f}c^0\mathbf{y} + s^0\mathbf{f}s^0\mathbf{q}s^0\mathbf{y} & -c^0\mathbf{f}s^0\mathbf{y} + s^0\mathbf{f}s^0\mathbf{q}c^0\mathbf{y} \\ -s^0\mathbf{q} & c^0\mathbf{q}s^0\mathbf{y} & c^0\mathbf{q}c^0\mathbf{y} \end{bmatrix} \quad (11.)$$

Igualando os termos das Eq. (10) e (11), sabendo que os termos da diagonal da matriz $[{}^0R^i_{mnp}]$ da Eq. (10) não são nulos, poderemos obter os ângulos de rolagem, mergulho e guinada, que definem a orientação da plataforma do robô, conforme a seguir:

$$\begin{aligned} {}^0_{mnp}\mathbf{y} &= \text{atan} \left(\frac{{}^0r^i_{32}}{{}^0r^i_{33}} \right)_{mnp} \\ {}^0_{mnp}\mathbf{q} &= -\text{asen} \left({}^0r^i_{31} \right)_{mnp} \\ {}^0_{mnp}\mathbf{f} &= \text{asen} \left(\frac{{}^0r^i_{21}}{{}^0r^i_{11}} \right)_{mnp} \end{aligned} \quad (12.a, b, c)$$

onde $({}^0r^i_{ij})_{mnp}$ são os elementos da matriz $[{}^0R^i_{mnp}]$, Eq. (10).

3. Plano de apoio das patas

Para obtermos a equação do plano de apoio das patas, basta efetuar o produto escalar de qualquer vetor do plano com a sua normal e igualá-lo a zero (Efimov, 1987). Vamos tomar o vetor que une o ponto de apoio de uma das patas e um ponto qualquer no plano de apoio, conforme a seguir:

$${}^r_{mnp}n \cdot {}^r_{mnp}p = {}^r_{mnp}n^x ({}^r x - {}^r_{mnp}x^x) + {}^r_{mnp}n^y ({}^r y - {}^r_{mnp}y^x) + {}^r_{mnp}n^z ({}^r z - {}^r_{mnp}z^z) = 0 \quad (13)$$

onde ${}^r_{mnp}p$, é o vetor posição do ponto, $({}^r_{mnp}x, {}^r_{mnp}y, {}^r_{mnp}z)$ e $({}^r x, {}^r y, {}^r z)$ são um ponto de apoio e um ponto qualquer no plano de apoio, respectivamente, e ${}^r_{mnp}n$ é a normal ao plano de apoio.

Reduzindo-se a Eq. (13) à forma geral da equação do plano, tem-se:

$${}^x_{mnp}n^x {}^r x + {}^x_{mnp}n^y {}^r y + {}^x_{mnp}n^z {}^r z - ({}^x_{mnp}n^x {}^r_{mnp}x^x + {}^x_{mnp}n^y {}^r_{mnp}y^x + {}^x_{mnp}n^z {}^r_{mnp}z^z) = 0 \quad (14)$$

onde:

$${}^r_{mnp}p = ({}^x_{mnp}n^x {}^r_{mnp}x^x + {}^x_{mnp}n^y {}^r_{mnp}y^x + {}^x_{mnp}n^z {}^r_{mnp}z^z) \quad (15)$$

O valor numérico de ${}^r_{mnp}p$ é obtido substituindo-se em (3.18) as coordenadas de um dos pontos do plano de apoio.

4. Posição da projeção do CDG no plano de apoio

A projeção do CDG no plano de apoio é a interseção da reta normal ao plano, que passa pelo CDG, com o plano de apoio. As equações canônicas desta reta, em cada uma das configurações de apoio das patas são as seguintes (Efimov, 1987):

$$\frac{{}^r x}{mnpn^x} = \frac{{}^r y}{mnpn^y} = \frac{{}^r z}{mnpn^z} = mnp^t \quad (16.)$$

Tomando-se os valores de rx, ry e rz de cada uma das Eq.(16) e substituindo na Eq. (14) do plano, obteremos os valores de mnp^t da Eq. (15) conforme a seguir:

$$mnp^t = \frac{\left(mnpn^x \cdot mnpn^x + mnpn^y \cdot mnpn^y + mnpn^z \cdot mnpn^z \right)}{mnpn^x \cdot mnpn^x + mnpn^y \cdot mnpn^y + mnpn^z \cdot mnpn^z} = mnp^p \quad (17.)$$

Voltando com os valores de mnp^t dados pela Eq. (17) na Eq. (16), obteremos as coordenadas para a projeção do CDG no plano de apoio.

5. Distância da projeção vertical do CDG aos lados do polígono de apoio

Uma rotação de 90° de qualquer vetor em torno da normal n ao plano de apoio é definida conforme a seguir: (Spong & Vidyasagar, 1992)

$${}^r R_{n,q} = \begin{bmatrix} \left(mnpn^x \right)^2 & mnpn^x mnpn^y - mnpn^z & mnpn^x mnpn^z + mnpn^y \\ mnpn^x mnpn^y + mnpn^z & \left(mnpn^y \right)^2 & mnpn^y mnpn^z - mnpn^x \\ mnpn^x mnpn^z + mnpn^y & mnpn^y mnpn^z + mnpn^x & \left(mnpn^z \right)^2 \end{bmatrix} \quad (18.)$$

Se mnp^u o vetor unitário da Eq. (3) e mnp^u^x , mnp^u^y e mnp^u^z as suas componentes, os vetores diretores das normais aos vetores posição, são então obtidos por:

$$mnp^d = \left[mnp^u \right] \begin{bmatrix} mnp^u^x & mnp^u^y & mnp^u^z \end{bmatrix} \quad (19.)$$

onde mnp^d e mnp^d são os vetores diretores nas direções perpendiculares aos vetores posição.

As equações canônicas das retas normais aos vetores posição e que passam pela projeção do CDG, em cada uma das configurações de apoio das patas, são então as seguintes:

$$\frac{{}^r x - mnp^x G}{mnp^d^x} = \frac{{}^r y - mnp^y G}{mnp^d^y} = \frac{{}^r z - mnp^z G}{mnp^d^z} = mnp^t \quad (20.)$$

onde mnp^d^x , mnp^d^y e mnp^d^z são os cossenos diretores dos vetores da Eq. (19).

Tomando-se os valores de mnp^x , mnp^y e mnp^z de cada uma das Eq. (20) e substituindo na equação do plano referente à configuração, dada de forma geral, Eq. (14), obteremos os valores de mnp^t conforme a seguir:

$$mnp^t = \frac{mnp^x G \cdot mnp^d^x + mnp^y G \cdot mnp^d^y + mnp^z G \cdot mnp^d^z - mnp^p}{mnp^d^x \cdot mnp^d^x + mnp^d^y \cdot mnp^d^y + mnp^d^z \cdot mnp^d^z} \quad (21.)$$

As coordenadas dos pontos de interseção das normais com os respectivos vetores posição são obtidas da Eq. (20) com o valor de mnp^t obtido da Eq. (21) e as distâncias entre estes pontos e a projeção do CDG no plano de apoio são:

$$mnp^d = \sqrt{\left(mnp^x - mnp^x G \right)^2 + \left(mnp^y - mnp^y G \right)^2 + \left(mnp^z - mnp^z G \right)^2} \quad (22.)$$

A determinação das distâncias do CDG aos lados do polígono de apoio, não permite concluir se o mesmo está dentro ou fora do polígono. Devemos assegurar que os ângulos formados pelos segmentos que unem a projeção do CDG a cada um dos vértices do polígono de apoio, serão sempre menores que 180° para que o CDG esteja dentro do polígono de apoio. Os ângulos α , β e γ podem ser obtidos conforme a seguir:

$$\begin{aligned} {}_{mn}^r \mathbf{a} &= \text{acos} \left(\frac{{}_m^r p \cdot {}_n^r p - \left[\left({}_m^r p - {}_{mn}^r x_G \right) \cdot \left({}_m^r p - {}_{mn}^r x_G \right) + \left({}_n^r p - {}_{mn}^r x_G \right) \cdot \left({}_n^r p - {}_{mn}^r x_G \right) \right]}{2 \cdot {}_m^r p \cdot {}_n^r p} \right) \\ {}_{mn}^r \mathbf{b} &= \text{acos} \left(\frac{{}_m^r p \cdot {}_p^r p - \left[\left({}_m^r p - {}_{mn}^r x_G \right) \cdot \left({}_m^r p - {}_{mn}^r x_G \right) + \left({}_p^r p - {}_{mn}^r x_G \right) \cdot \left({}_p^r p - {}_{mn}^r x_G \right) \right]}{2 \cdot {}_m^r p \cdot {}_p^r p} \right) \\ {}_{mn}^r \mathbf{g} &= \text{acos} \left(\frac{{}_p^r p \cdot {}_n^r p - \left[\left({}_p^r p - {}_{mn}^r x_G \right) \cdot \left({}_p^r p - {}_{mn}^r x_G \right) + \left({}_n^r p - {}_{mn}^r x_G \right) \cdot \left({}_n^r p - {}_{mn}^r x_G \right) \right]}{2 \cdot {}_p^r p \cdot {}_n^r p} \right) \end{aligned} \quad (23.)$$

onde ${}_{mn}^r p = \left({}_m^r p - {}_n^r p \right)$, sendo ${}_m^r p$ e ${}_n^r p$ dados pela Eq. (3).

6. Resultados

Para verificar os resultados do modelo proposto, foram utilizadas coordenadas precisas das patas referentes a três posições conhecidas da plataforma, obtidas a partir de um desenho tridimensional em IntelCAD. Os valores de coordenadas foram introduzidos em um programa desenvolvido em MatLab, cujos resultados estão na Tab. (1) a seguir.

A simulação consistiu em fornecer os valores das coordenadas das patas e determinar os cossenos diretores da normal à plataforma, para as situações conhecidas.

Na primeira linha da Tab. (1)(a) a plataforma é submetida a um mergulho de 30°, correspondente à uma rotação em torno do eixo y do referencial de bordo. Os resultados dos cossenos diretores da normal e da projeção do centro de gravidade no plano de apoio estão na primeira linha da Tab. (1)(b).

Na segunda linha da Tab. (1)(a), a plataforma foi submetida a uma inclinação de 30° em relação à diagonal que une os pontos de fixação das patas 1 e 4, e na terceira a uma situação em que a normal forma ângulos de 45° com os eixos do referencial robô. Os resultados estão nas linhas correspondentes da Tab. (1)(b).

Tabela 1. Resultados de simulação para uma plataforma de um robô quadrúpede com comprimento $L = 400 [mm]$ e largura $W = 300mm$; referencial de bordo do robô conforme fig. 1

(a) valores de coordenadas utilizados na simulação;

Simulação Número	Pata 1 [mm]			Pata 2 [mm]			Pata 3 [mm]			Pata 4 [mm]		
	x1	y1	z1	x2	y2	z2	x3	y3	z3	x4	y4	z4
1	200	150	-330.94	-200	150	-100	200	150	-330.94	-200	-150	-100
2	100	150	-100	-200	150	-130,94	200	-150	-330.94	-200	-150	-100
3	200	150	-950	-200	150	-550	200	-150	-650	-200	-150	-250

(b) resultados obtidos;

No.	Cossenos diretores da normal à plataforma			Coordenadas da projeção do CDG no plano de apoio[mm]		
	cos(a)	cos(b)	cos(g)	x	y	z
1	0.5	0	0.866	-93.301	0	-161.603
2	0	-0.5	0.866	6.601	-10.145	-98.365
3	0.577	0.577	0.577	-200	-200	-200

7. Conclusões

Por se tratar de um modelo analítico, os valores obtidos não apresentam erro até a terceira casa decimal, indicando que o mesmo foi construído de maneira consistente.

A utilização dos conceitos da geometria analítica para definição de planos e retas na forma canônica mostra-se bastante eficiente oferecendo uma solução analítica simples e adequada para aplicações em tempo real do controle e supervisão de equilíbrio de robôs a pernas.

Os cosenos diretores da normal permitem que seja obtida a equação do plano de apoio, que é utilizada na determinação da coordenada vertical para aterrissagem da pata em movimento para novo posicionamento. O seu conhecimento permite ainda a implementação de procedimentos para detectar e identificar obstáculos existentes nos pontos de apoio das patas.

As coordenadas da projeção do CDG juntamente com a Eq. (19) fornecerão as distâncias da projeção do CDG aos lados do triângulo de apoio, permitindo monitorar continuamente a margem de estabilidade instantânea do robô, e, finalmente, através da Eq. (23), controlar o robô para que a projeção do CDG nunca ultrapasse os limites definidos pelo triângulo de apoio das patas.

Este trabalho está sendo continuado no sentido de ampliar o modelo até aqui desenvolvido para obtenção da odometria instantânea do robô, a partir do conhecimento de seu ponto de partida e das coordenadas sucessivas de apoio das patas. Em seguida, estas informações serão utilizadas na formulação do equilíbrio do robô para as condições operacionais previstas.

8. Referências

- (Efimov, 1987) Efimov, N., "Elementos de Geometria Analítica", Trad. David Jardim Jr., Liv. Cultura Brasileira, Belo Horizonte, Brasil, 1972;
- (Fu & all, 1987) Fu, K., Gonzales, R., Lee, C., "Robotics: Control, Sensing, Vision and Intelligence", CAD/CAM, Robotics and Computer Vision, McGraw Hill, 1987;
- (Hirose & Kato, 1998) Hirose, S., Kato, K., "Development of a Quadruped Walking Robot with the Mission of Mine Detection and Removal- Proposal of Shape - Feedback Master-Slave Arm", Proc. Of the 1998 IEEE International Conference on Robotics & Automation, Leuven, Belgium, May 1998, pp. 1713-1718;
- (Ishihara, 1998) Hishihara, Wilson M. , "Desenvolvimento e construção de um robô de 4 pernas", Dissertação de Mestrado submetida ao Programa de Pós Graduação do Departamento de Engenharia Elétrica da UFES, julho, 1998;
- (Kitano & all, 1998) Kitano, H., Fujita, M., Zrehen, S., Kageyama, K., "Sony Legged Robot for RoboCup Challenge", Proc. Of the 1998 IEEE International Conference on Robotics & Automation, Leuven, Belgium, May 1998, pp. 2605-2612;
- (Martins Filho 1998) Martins Filho, L., 1998, "Une Approche de Supervision de Marche pour Robots Quadrupèdes Intégrant Commande et Décision", Thèse de Doctorat, Université Paul Sabatier, Toulouse, France;
- (Petkov & all, 1991) Petkov, P.H., Christov, N.D., Konstantinov, M.M., "Computational Methods for Linear Control Systems", Prentice Hall International Series in Systems and Control Engineering, UK, 1991;
- (Santos & all, 1997) Santos, P. G., Armada, M. A., Gimenez, M. A., "An Industrial Walking Machine for Naval Construction", Proc. of the IEEE Int. Conf. on Robotics and Automation, pp. 28-33, April 1997. Albuquerque, New Mexico;
- (Schmiedeler & Waldron, 1999) Schmiedeler, J. P., Waldron, K. J., "The Mechanics of Quadrupedal Galloping and the Future of Legged Vehicles", Int. Journal Robotic Research, vol. 18, no. 12, pp. 1224-1234, December, 1999;
- (Smith & Berkemeyer, 1997) Smith, A. C., Berkemeyer, M. D. "Passive Dynamic Quadruped Walking", Proc. of the IEEE Int. Conf. on Robotics and Automation, pp. 34-39, April, 1997, Albuquerque, New Mexico;
- (Spong & Vidyasagar, 1992) Spong, M.W., Vidyasagar, M., "Robot Dynamics and Control", John Wiley & Sons, New York, 1992;
- (Sukhatme & all, 1997) Sukhatme, G. S.; Brizius, S.; Cozzy, S., Bekey, G. A. A., "Strategy for Quadruped Walking on Uneven Terrain", Proc. of the ICAR'97, pp. 291-296, July 7-9, 1997, Monterey, CA;
- (Yoneda & all, 1996) Yoneda, K., Iiyama, H., Hirose, S., "Intermittent Trot Gait of a Quadruped Walking Machine: Dynamic Stability Control of a Omnidirectional Walk", Proc. of the International Conference on Robotics and Automation, pp. 3002-3007, Minneapolis, April, 1996;

KINEMATIC MODEL FOR A QUADRUPED ROBOT

Antônio Bento Filho

Depto. Eng. Mecânica
CTC/UFES
abento@ele.ufes.br

Paulo Faria Santos Amaral

Depto. Eng. Elétrica
CTC/UFES
paulo@ele.ufes.br

Benedito Geraldo Miglio Pinto

Depto. Eng. Mecânica
CTC/UFES
automatica@automatica.com.br

Abstract. The present work describes the implementation of a kinematic model for the determination of the pitch, roll and yaw angles and the plane containing the footprint coordinates. These information will be used in real time, for the supervision of balance of a quadruped robot. It uses the coordinates of 3 supporting legs to get the robot coordinate system, which represents the rotation of the platform in relation to the footprints plan. Inclometers on board, pick up the robot attitude and rotation matrix of the platform. With the model, we got also the equation of the footprints plane, which is used to determine the vertical coordinate of a landing leg, and the coordinates of the projection of the CDG of the platform in the footprints plane. Being developed for robots that walk in steady static balance with 3 supporting legs always, however, it can be extended for robots of 6 or more legs. With aid of the MatLab, the model was tested for some patterns of inclination of the platform, having presented resulted satisfactory for its future implementation in the supervision of the balance of the robot.

Keywords *quadruped, legged, mobile, robot, attitude*