



ESTRATÉGIAS COMPUTACIONAIS PARA O ESTUDO DA DINÂMICA DE ROBÔS MANIPULADORES: FERRAMENTAS DE PROJETO DE SISTEMAS MULTICORPOS

Alencar Soares Bravo

Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia
Av. João Naves de Ávila, 2160 - Campus Santa Mônica - Bloco 1M - Uberlândia/MG
asbravo@mecanica.ufu.br

Rogério Rodrigues dos Santos

Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia
rrsantos@mecanica.ufu.br

Valder Steffen, Jr.

Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia
vsteffen@mecanica.ufu.br

Resumo: Diante do crescente uso dos métodos computacionais para a análise e projeto de sistemas mecatrônicos, neste trabalho consideram-se diferentes metodologias para a construção de modelos matemáticos que forneçam resultados utilizados na simulação de robôs manipuladores seriais. É apresentado um estudo sobre o uso do algoritmo de Newton-Euler Recursivo (RNE – Recursive Newton-Euler) para o cálculo de informações dinâmicas durante o movimento do robô. O resultado obtido é comparado com aquele fornecido por softwares comerciais. A concordância dos resultados viabiliza a implementação e uso do método, em diversas linguagens de programação, para as mais diferentes aplicações de simulação robótica. Os resultados comprovaram que algumas simplificações na descrição da geometria do manipulador não comprometem a análise realizada pela estratégia RNE escolhida. Assim, constata-se que o algoritmo provê informações confiáveis e representativas.

Palavras-chave: Dinâmica de sistemas articulados, simulação de robôs manipuladores, robótica.

1. INTRODUÇÃO

A simulação computacional é uma ferramenta bastante utilizada na engenharia moderna, pois a simulação através de modelos possibilita um grande corte dos custos associados a um projeto. Cada vez mais esta ferramenta tem sido aprimorada de forma a fornecer resultados que represente de forma cada vez mais precisa o comportamento do sistema físico real.

No campo da robótica, os algoritmos de simulação têm uma grande importância, não só pela possibilidade de dimensionamento de projeto, como também para a análise do controle e funcionamento durante sua vida útil. Há uma tendência de que os novos robôs se tornem cada vez mais autônomos, afim de que a intervenção humana durante seu trabalho seja a mínima possível. Isto requer algoritmos cada vez mais elaborados, que antes não tinham sua implementação efetiva devido a limitações das tecnologias computacionais envolvidas.

Neste trabalho considera-se o uso do Método de Newton-Euler Recursivo (RNE – Recursive Newton-Euler) (Fu *et al.*, 1987) para a obtenção de informações sobre a dinâmica de robôs seriais. Realiza-se uma comparação entre o comportamento dinâmico obtido a partir do *software* visualNastran, e de uma implementação de um método para o cálculo da dinâmica em linguagem de programação. Espera-se estabelecer uma metodologia para obtenção da informação, além da

comparação entre os resultados obtidos pelos diferentes *softwares*. A concordância entre os resultados obtidos pelos diferentes procedimentos valida os resultados, bem como a metodologia estabelecida para o uso do algoritmo.

2. DINÂMICA DE RÔBOS MANIPULADORES

O comportamento dinâmico de um manipulador robótico está relacionado ao estudo do seu movimento. Através da representação de Denavit-Hartenberg (*D-H*) (Denavit *et al.*, 1955) é possível determinar informações sobre a cinemática direta.

Outro aspecto importante é a forma de controlar as forças e torques que devem ser aplicados a cada articulação em cada instante de tempo, para que o robô execute o movimento conforme as especificações cinemáticas desejadas. Além disto, deve-se prever a existência de uma carga adicional do objeto a ser manipulado, a qual influenciará os valores de torque a serem aplicados a todas as articulações durante a realização da tarefa.

Espera-se de um método eficiente para o cálculo dos efeitos dinâmicos a determinação das informações sem grandes distúrbios em relação aos valores reais, e que ao mesmo tempo, exija um custo computacional razoável. Estratégias rápidas e precisas são desejadas para o estabelecimento de controle em tempo real.

2.1 Modelos Dinâmicos

Existem diversas formulações de dinâmica de robôs que podem ser base para a construção de um algoritmo computacional. O método de Newton-Lagrange (*NL*) (Paul, 1981) estabelece um conjunto de equações não-lineares diferenciais de segunda ordem que descrevem o comportamento dinâmico de um robô manipulador a partir do modelo de corpo rígido. A derivação da dinâmica de um modelo de um manipulador baseado na formulação de *NL* é simples e sistemática. Porém é sempre necessário calcular alguns coeficientes dinâmicos, cujo estabelecimento requer diversas operações aritméticas.

Existem propostas de simplificação deste método, as quais ignoram componentes de forças, tais como as forças de Coriolis e Centrifuga. Isto aumenta o desempenho computacional. Contudo, estas componentes de força têm grande importância em algumas situações, como no caso de movimento rápido, não podendo, portanto, ser meramente ignorada.

Pode-se obter uma formulação mais eficiente ao se utilizar informações sobre vetores de posições relativas e matrizes de rotação, de forma a obter informações de energia cinética e potencial de cada *link* para a determinação da função Lagrangeana, utilizando a formulação de Lagrange-Euler para obtenção de equações de movimento. A formulação de d'Alembert generalizada utiliza este enfoque, o que melhora seu desempenho computacional.

Contudo o método que representa melhor desempenho computacional é o método de Newton-Euler (Fu *et al.*, 1987), conforme apresentado na Tabela 1, onde *n* indica o número de graus de liberdade.

Tabela 1: Comparação de operações entre métodos

	Lagrange-Euler	Newton-Euler	D'Alembert Generalizado
Multiplicações	$128/3 n^4 + 512/3 n^3 + 739/3 n^2 + 160/3 n$	$132n$	$13/6 n^3 + 105/2 n^2 + 268/3 n + 69$
Adições	$98/3 n^4 + 781/6 n^3 + 559/3 n^2 + 245/6 n$	$111n - 4$	$4/3 n^3 + 44n^2 + 146/3 n + 45$

2.2 Método de Newton-Euler Recursivo

A formulação recursiva de Newton-Euler parte do princípio que toda a informação cinemática de cada *link* é conhecida o, isto é possível utilizando a representação *D-H* e aplicando transformações de sistemas de coordenadas necessárias. A partir da informação cinemática conhecida é então aplicado o princípio de d'Alembert para descrever o movimento em cada *link*. Aplica-se então as condições de equilíbrio estático aos problemas de dinâmicas em cada instante de tempo.

Isto é possível pois o princípio de d'Alembert é uma forma da segunda lei de Newton que pode ser aplicada de forma conveniente em certos casos, como este. O enunciado do princípio é:

Para qualquer corpo, a somatória algébrica de todas as forças externas aplicadas e todas as forças restritivas ao movimento em certa direção é zero.

Assim, levam-se em conta as forças externas aplicadas que devem guiar o movimento, além das forças restritivas ao movimento. Este método permite a obtenção resultados iguais àqueles obtidos pelo método de Lagrange, apesar de utilizar um princípio diferente em sua formulação (Spong *et al.*, 1989)

2.3 Matlab Toolbox

A *toolbox* de robótica desenvolvida por (Corke, 1996), para uso no programa Matlab, oferece recursos e funções úteis para estudo em robótica, como cinemática, dinâmica e geração de trajetórias.

Nela está contida uma implementação do método de Newton-Euler Recursivo, de forma que este pode ser utilizado como ferramenta de análises do comportamento dinâmico de robôs manipuladores.

O sistema utiliza a representação de *D-H* para a representação dos robôs, o que facilita a inserção de novos modelos, permitindo também a representação gráfica do manipulador através de uma função apropriada.

Os dados referentes à dinâmica do manipulador, tais como tensores de inércia, devem ser informados explicitamente.

Isto pode significar uma tarefa trabalhosa quando se deseja considerar toda a complexidade da geometria de robôs reais e seus efeitos de inércia. Por este motivo a obtenção destas propriedades deve ser obtida através de uma metodologia alternativa.

2.4 visualNastran

O visualNastran é um *software* comercial de simulação de propósito geral, composto de vários módulos. O programa oferece vários recursos de simulação do comportamento cinemático e dinâmico de corpos rígidos que são bastante úteis para a simulação computacional de robôs.

O sistema conta com um algoritmo para a simulação do comportamento físico de corpos rígidos. Os resultados sobre a cinemática e dinâmica do manipulador são computados através de métodos numéricos, não havendo a necessidade de especificações como os tensores de inércia. Estas informações são geradas pelo programa ao se inserir qualquer geometria no mesmo.

Para a construção de estruturas robóticas, as restrições de movimentação dos *links* são estabelecidas durante a construção do manipulador através de uma interface visual.

Através da definição de um conjunto de corpos rígidos, relacionados por restrições que representam o comportamento de cada articulação, é possível construir o modelo de robôs seriais sem grandes dificuldades. As restrições que atuam como articulações são chamadas de "*joints*", e estas podem ser escolhidas de forma a representar articulações rotacionais, prismáticas, ou outras,

possibilitando a aplicação de torques e/ou forças controladas. Assim, é possível observar os efeitos cinemáticos e dinâmicos do manipulador. Pode-se alternativamente estabelecer a trajetória, e obter como resposta do sistema os torques/forças necessários para a sua realização.

3. EXPERIMENTOS NÚMERICOS

3.1 O Robô Manipulador *Elbow*

3.1.1 Modelagem

Para a avaliação do método, é considerado um robô manipulador serial do tipo “*Elbow*”. O modelo computacional utilizado é um robô fictício que possui os três *links* iguais em forma e dimensão, conforme apresentado na Figura 1.

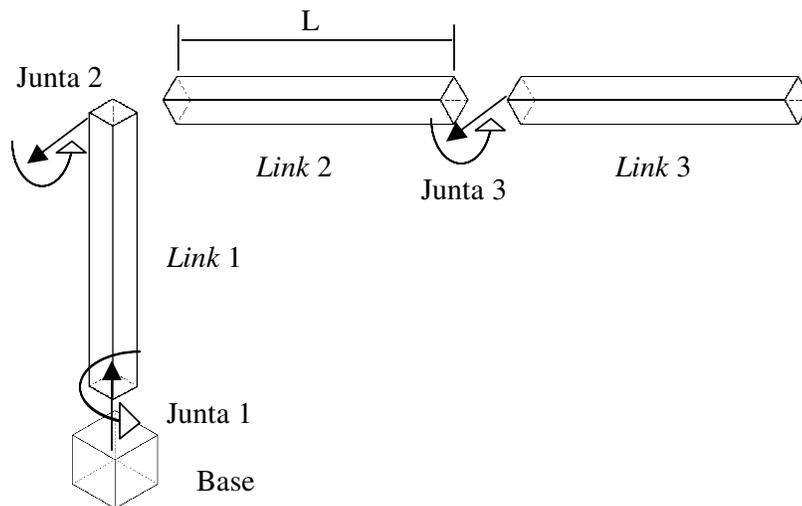


Figura 1: Representação esquemática de um robô do tipo “*Elbow*”.

Neste modelo foi utilizada uma forma de prisma regular, como na Figura 2, com seção transversal quadrada para a representação de cada *link*, conforme a Figura 1.

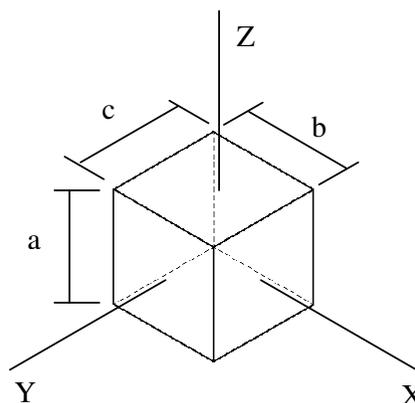


Figura 2: Momentos de inércia para uma geometria regular.

Desta forma, é descrito um robô que possui, em cada *link*, 1 m de comprimento (parâmetro “L” representado na Figura 1), com seção transversal quadrada de 0,01 m² e massa de 48,5 kg.

Para os valores dos momentos de inércia, são obtidos por meio de cálculo utilizando as expressões $I_x = m(a^2 + b^2)/12$, $I_y = m(a^2 + c^2)/12$ e $I_z = m(b^2 + c^2)/12$.

A descrição cinemática deste modelo através da representação de $D-H$ é dada pela Tabela 2 a seguir.

Tabela 2: Tabela $D-H$ para robô *Elbow* (* variável de junta).

<i>Link</i>	α (rad)	a (m)	θ (rad)	d (m)
1	$\pi/2$	0	0*	L
2	0	L	0*	0
3	0	L	0*	0

Assim, são especificados todos os dados físicos necessários para a determinação do modelo matemático.

3.1.2 Resultados Numéricos

Para a avaliação de resultados obtidos segundo este modelo de robô, é escolhido que as juntas executem um movimento simultâneo de $\pi/2$ rad, em um intervalo de 20 s sendo o teste realizado com precisão de integração no tempo de 0,05 s, totalizando 401 pontos no intervalo para avaliação.

O formato de movimento é especificado pela interpolação através de um polinômio de terceiro grau, considerando-se as restrições de velocidade inicial e final iguais a zero. Os valores iniciais das coordenadas de junta são $\theta_1 = \theta_2 = \theta_3 = 0$ rad, e os valores finais são $\theta_1 = 1,5708$ rad, $\theta_2 = 1,5708$ rad e $\theta_3 = -1,5708$ rad. Com este tipo de movimento foram obtidos os resultados para cada junta apresentados na Figura 3.

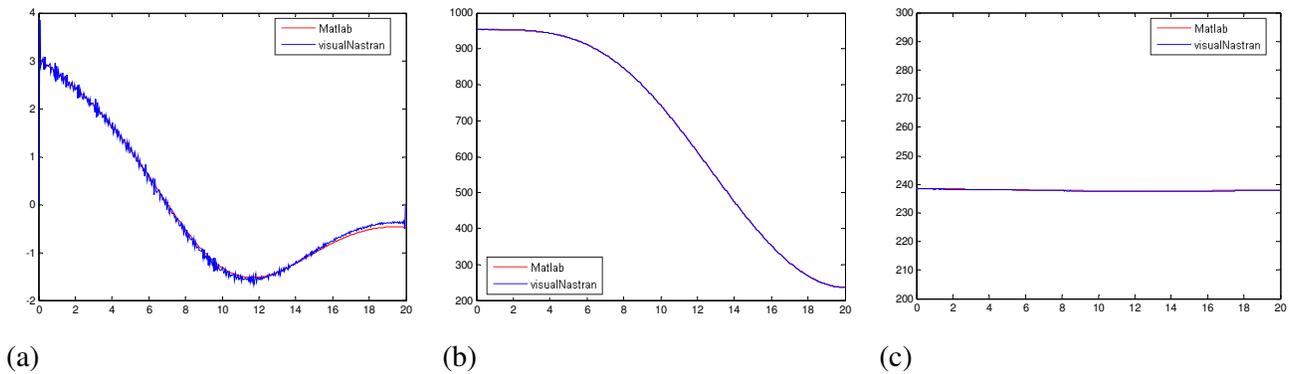


Figura 3: Torque aplicado a cada junta.

Para verificar o desvio dos dados em cada junta, é efetuado o cálculo dos erros absolutos em relação aos dados do visualNastran. O valor do erro em cada ponto, representado pela Figura 3, é analisado e resumido na Tabela 3 a seguir:

Tabela 3: Erro máximo, mínimo e médio para cada junta.

Junta	1 (N.m)	2 (N.m)	3 (N.m)
Valor máximo	0,822801885	0,603867687	0,118601347
Valor mínimo	0,000140932	0,031476302	0,047498046
Média	0,062361539	0,219901501	0,081289433
Desvio padrão	0,057251465	0,069360684	0,011406532

Verifica-se que a média dos erros não é superior a 1 *Nm* em todos os instantes. Constatou-se que em algumas situações os primeiros pontos e os últimos apresentam picos de discordâncias, ou seja, erros muito superiores aos demais. Isto se deve ao método de formulação das derivadas por diferenças finitas nas extremidades onde faltam pontos adjacentes, resultando nestes erros numéricos. Na atual análise os pontos extremos (primeiro e último) não são considerados.

Durante todo o movimento o valor do erro permanece estável e apresenta valores pequenos.

A média dos erros é pequena em todas as juntas. Além disso, o desvio padrão também é pequeno e o valor máximo do erro é próximo aquele observado para o erro médio.

Em todas as análises realizadas, considerou-se a ausência de atrito nas juntas. Espera-se que ao se considerar o efeito do atrito em cada junta, os resultados sejam igualmente compatíveis em haver igual incremento de torque.

3.2 Robô Rixan RV-2AJ-S11

O objetivo desta análise é representar a configuração de um robô industrial. A geometria utilizada é aquela estabelecida pelo projeto do fabricante. A aquisição das informações físicas dos componentes robóticos é obtida de diferentes formas: a) aproximação através da geometria de um prisma, e b) descrição mais precisa que considera detalhes da geometria do mesmo, usando recursos de CAD.

O robô tem as especificações de formas e características cinemáticas descritas na Tabela 4 a seguir.

Tabela 4: Tabela *D-H* para o robô Rixan (* *variável de junta*).

<i>Link</i>	α (rad)	<i>a</i> (m)	θ (rad)	<i>d</i> (m)
1	$-\pi/2$	0	0*	0,135
2	0	0,250	0*	0
3	0	0,160	0*	0
4	0	0,076	0*	0

Considera-se a movimentação igual àquela realizada no exemplo anterior. A última junta (quarta) não sofrerá movimento, mas apenas acompanhará o movimento do *link* anterior. Sendo assim, os valores de torque informados nesta junta são necessários apenas para a manutenção da posição relativa em relação ao *link* anterior. As propriedades físicas de material (densidade média) são as mesmas do robô apresentado anteriormente.

Pode-se construir um modelo numérico com poucas ou sem simplificações, adquirindo dados físicos por um programa de CAD. A modelagem da forma dos *links* em escala real a partir de um programa de CAD facilita a obtenção de informações ou parâmetros físicos decorrentes, como posição do centro de massa e momentos de inércia.

3.2.1 Modelagem CAD

Após a obtenção da especificação física do manipulador (Rixan, 2007) e da obtenção das informações necessárias para o cálculo da dinâmica do sistema, o procedimento realizado através do Matlab e do visualNastran resulta nas informações apresentadas na Figura 4.

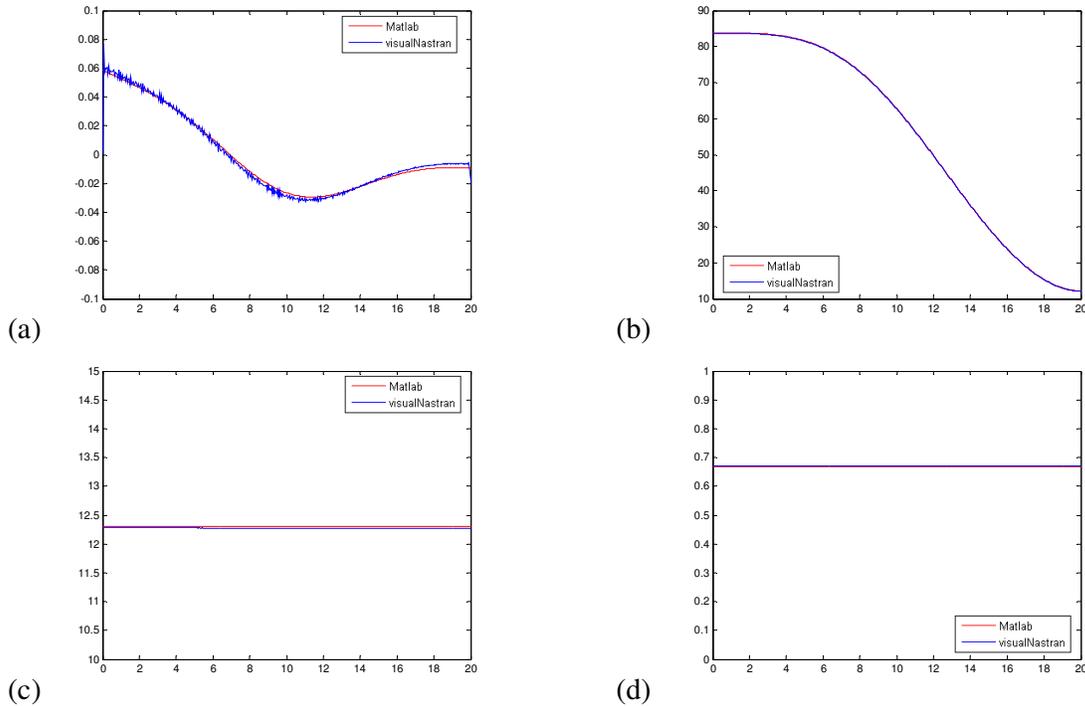


Figura 4. Torque aplicado a cada uma das juntas.

Tabela 5: Erro máximo, mínimo e médio para cada junta.

Junta	1 (N.m)	2 (N.m)	3 (N.m)	4 (N.m)
Valor máximo	0,018520918	0,143923164	0,023957507	0,002293160
Valor mínimo	0,000005099	0,094207014	0,019140677	0,002159049
Média	0,001780932	0,125831104	0,022814069	0,002183981
Desvio padrão	0,001386051	0,013772301	0,001042932	0,000009918

Os resultados obtidos, conforme a análise de erro da Tabela 5, mostram erros bastante baixos. Verifica-se que maior erro ocorreu na junta 2, sendo de um valor inferior a $0,145 \text{ Nm}$, com valor médio $0,095 \text{ Nm}$. As juntas 1 e 3 tiveram erros ainda menores. Consequentemente, pode-se verificar que o modelo *RNE* leva a resultados satisfatórios, tendo, portanto, grande representatividade para modelar o comportamento de um robô real.

3.2.2 Modelagem Aproximada

Alternativamente, pode-se optar pela substituição do uso das formas reais dos *links* robóticos, por formas aproximadas de geometrias regulares (prismas regulares ou cilindros regulares) onde estas informações físicas podem ser facilmente obtidas por cálculos simplificados.

Na Tabela 6 a seguir estão descritas as características físicas do robô serial utilizado, onde as informações dos *links* têm a representação simplificada através da geometria de prismas regulares, com as respectivas dimensões e momentos de inércia.

Tabela 6: Representação física dos *links* simplificada a partir de prismas regulares.

	<i>Link 1</i>	<i>Link 2</i>	<i>Link 3</i>	<i>Link 4</i>
Momentos de inércia (Kg m²)	$I_1 = 0,0707$	$I_1 = 0,1350$	$I_1 = 0,0074$	$I_1 = 0,0011$
	$I_2 = 0,0457$	$I_2 = 0,2689$	$I_2 = 0,0335$	$I_2 = 0,0021$
	$I_3 = 0,0905$	$I_3 = 0,3536$	$I_3 = 0,0408$	$I_3 = 0,0016$
Dimensões (m)	$a = 0,090$	$a = 0,090$	$a = 0,008$	$a = 0,070$
	$b = 0,190$	$b = 0,188$	$b = 0,090$	$b = 0,040$
	$c = 0,143$	$c = 0,280$	$c = 0,192$	$c = 0,089$

Ao se realizar a simulação numérica para obtenção das informações dinâmicas com os dados da Tabela 6, são obtidos os resultados apresentados na Figura 5.

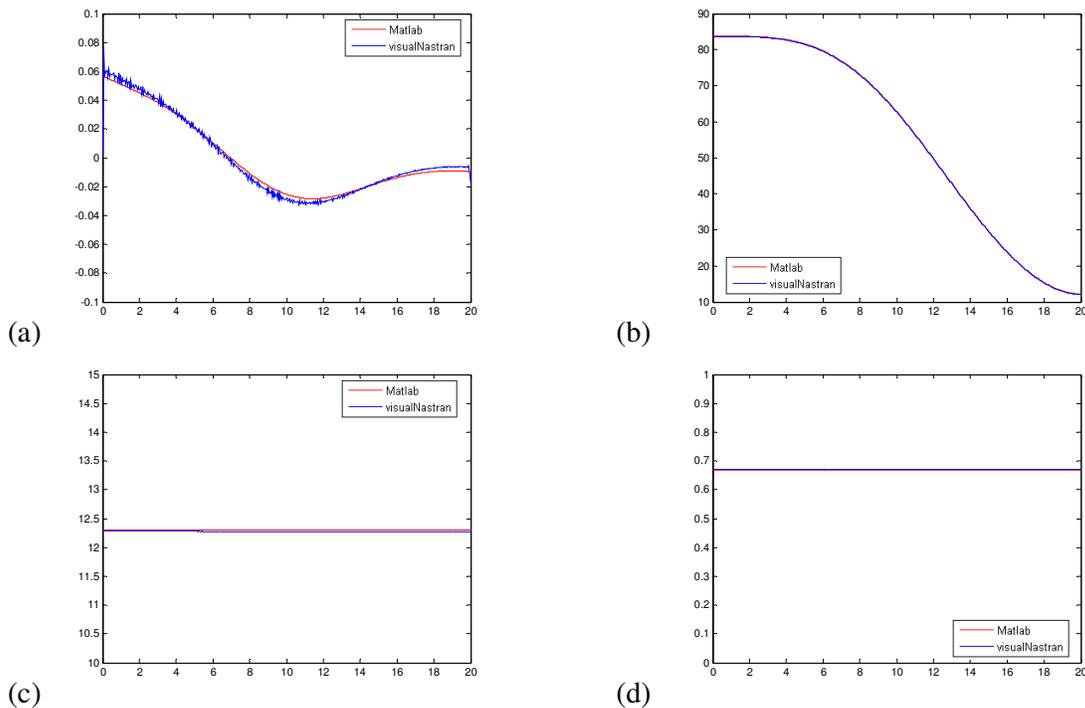


Figura 5: Torque aplicado a cada junta.

A simplificação da geometria deve ter o cuidado de apresentar parâmetros que resultem no mesmo volume total, pois a massa de cada *link* deve ser a mesma, independente da representação escolhida. Deve-se considerar também a forma original, para que as dimensões tenham proporções semelhantes.

A seguir, apresenta-se na Tabela 7 o cálculo do erro, de acordo com o critério adotado anteriormente.

Tabela 7: Erro máximo, mínimo e médio para cada junta.

Junta	1 (N.m)	2 (N.m)	3 (N.m)	4 (N.m)
Valor máximo	0,020295918	0,143246164	0,023957507	0,002293160
Valor mínimo	0,000011049	0,096517014	0,019140677	0,002159049
Média	0,002271797	0,125610029	0,022814069	0,002183981
Desvio padrão	0,001649428	0,012778887	0,001042932	0,000009918

O perfil observado para os torques das diferentes juntas têm grande concordância, e seus valores numéricos são próximos. Segundo a análise realizada, apenas a junta 2 apresentou erro que está numa ordem de grandeza na primeira casa decimal, sendo que as demais juntas apresentam erros somente a partir da segunda casa decimal.

Os valores máximos dos erros estão próximos àqueles obtidos para os erros médios. Os resultados comprovam que o método usado para modelagem de um robô real através de aproximações geométricas é válido, mesmo considerando alguma incerteza na escolha das dimensões dos *links*. Assim como no exemplo numérico anterior, neste experimento também se constatou um erro maior na junta 2.

3.2.3 Comparação dos Resultados Obtidos

A Tabela 8 a seguir apresenta a diferença entre os resultados obtidos a partir dos dois experimentos numéricos discutidos acima. Estes dados foram obtidos a partir do módulo da diferença dos dados apresentados nas Tabelas 5 e 7.

Tabela 8: Erro máximo, mínimo e médio para cada junta.

Junta	1 (N.m)	2 (N.m)	3 (N.m)	4 (N.m)
Valor máximo	0,001775000	0,000677000	0,000000000	0,000000000
Valor mínimo	0,000005949	0,002310000	0,000000000	0,000000000
Média	0,000490865	0,000221075	0,000000000	0,000000000
Desvio padrão	0,000263377	0,000993414	0,000000000	0,000000000

Ao se analisar os dados mostrados na Tabela 8, constata-se que a discordância de resultados obtidos empregando as diferentes técnicas não é muito representativa. Mesmo utilizando um grande número de casas decimais para os cálculos (nove, no presente estudo), não foram detectadas diferenças entre os resultados apresentados para os dois métodos, ao se considerar as juntas 3 e 4.

Na medida em que se afasta do efetuador do robô indo em direção à base, os cálculos tornam-se mais sensíveis a erros, pois o comportamento e a qualidade da modelagem dos *links* à frente, em direção ao efetuador, passam a influir nos resultados devido ao acúmulo de erros individuais dos *links* considerados.

Ainda sim, as diferenças têm em geral ordem de grandeza na quarta casa decimal, sendo, portanto, desprezíveis para as aplicações consideradas. O desvio padrão se manteve com ordem de grandeza na quarta casa decimal. Considera-se que estes resultados são satisfatórios para a solução de problemas de engenharia.

4. CONCLUSÃO

O algoritmo *RNE* utilizado neste trabalho consiste em um método numérico adequado para a obtenção de informações sobre a dinâmica de robôs seriais, fornecendo resultados confiáveis quando comparado com um tradicional *software* utilizado pelo mercado.

Assim, constatou-se a possibilidade de implementação do algoritmo em uma linguagem de programação. Este estudo abre perspectivas de utilização dos resultados nas mais diversas aplicações. Entretanto, em alguns casos o grau de confiabilidade depende do nível de detalhamento das especificações utilizadas.

Para aplicações onde os dados são de robôs fictícios de formas geométricas regulares para os *links*, a aplicação do método provê confiabilidade para obtenção de resultados. Ao utilizar *software* de CAD para a obtenção de informações físicas mais aproximadas dos *links*, a concordância de resultados é plena.

Constatou-se que quando é desejado simular um robô real, onde não se têm as informações exatas sobre a geometria dos *links*, bem como inércia e massa, pode se estimar os valores simplificando a forma elaborada dos *links* através de prismas regulares de mesma massa e forma semelhante.

Tem-se assim mais uma perspectiva para a simulação, pois os resultados deste algoritmo podem ser utilizados em variadas aplicações. Uma vez que eles foram validados, e uma metodologia para simulação foi estabelecida, tem-se a disponibilidade de uma importante ferramenta, que serve como base para o desenvolvimento de aplicações em robótica.

5. REFERÊNCIAS

- Corke, P. I., 1996, "A Robotics Toolbox for MATLAB", IEEE Robotics and Automation Magazine, Volume 3(1), pp. 24-32.
- Denavit, J. and Hartenberg, R. S., 1955, "A kinematic notation for lower-pair mechanisms based on matrices", ASME Journal of Applied Mechanisms 215-221.
- Fu, K. S., Gonzales, R. C. and Lee, C. S. G., 1987, "Robotics: Control, Sensing, Vision, and Intelligence.", McGraw-Hill Publishing Company.
- Paul, R. P., 1981, "Robot manipulators: Mathematics, Programming, and Control.", Cambridge, MA: MIT Press.
- RIXAN web site: "<http://www.rixan.com/new/rixan/content.asp?contentid=2016348006>", acessado em 03 de abril de 2007.
- Spong, M. W. and Vidyasagar, M., 1989, "Robot Dynamics and Control.", John Wiley and Sons.

COMPUTATIONAL STRATEGIES FOR THE STUDY OF ROBOT MANIPULATORS: TOOLS FOR MULTIBODY PROJECT

Alencar Soares Bravo

School of Mechanical Engineering, Federal University of Uberlândia
Av. João Naves de Ávila, 2160 - Campus Santa Mônica - Bloco 1M - Uberlândia/MG
asbravo@mecanica.ufu.br

Rogério Rodrigues dos Santos

School of Mechanical Engineering, Federal University of Uberlândia
rrsantos@mecanica.ufu.br

Valder Steffen, Jr.

School of Mechanical Engineering, Federal University of Uberlândia
vsteffen@mecanica.ufu.br

Abstract: *Due to the increasing use of computational methods for the analysis and design of mechatronic systems, different methodologies for constructing numerical models that provides useful results for the robotic simulations, are considered. A study concerning the use of the Recursive Newton-Euler (RNE) algorithm to compute dynamic informations during the robot movement is presented. The computed result and those obtained by using commercial software are compared. The agreement between the results enables the implementation and use of the studied methodology, in several programming languages, to perform a number of tasks in the field of robotic simulation. The obtained results ensure that some simplifications during the description of the manipulator geometry do not compromise the performance of the RNE strategy. Therefore, it is confirmed that the algorithm provides very accurate solutions.*

Keywords: *Multibody dynamics, robot manipulator simulation, robotics.*