

EOM2SS: Uma rotina para geração automática de equações de estado de sistemas biomecânicos

Olavo Luppi Silva, Departamento de Engenharia Mecânica e de Materiais, Instituto Militar de Engenharia, e-mail: olavo@ime.eb.br

Luciano Luperini Menegaldo, Departamento de Engenharia Mecânica e de Materiais, Instituto Militar de Engenharia, e-mail: lmeneg@ime.eb.br, home-page: <http://lmeneg.tripod.com>

Cláudio Leandro Alves Moura, Divisão de Engenharia, Arsenal de Guerra de São Paulo, e-mail: imemoura@yahoo.com.br

Introdução

A análise dinâmica direta de sistemas multicorpos é um dos problemas recorrentes da biomecânica músculo-esquelética. No mercado comercial, existem diversos pacotes que realizam esta tarefa. Entretanto, tais sistemas são usualmente do tipo caixa-preta. Com isso, quem modela o sistema mecânico e faz a simulação, não tem acesso aos algoritmos nem às equações de movimento do sistema. Em muitas aplicações, porém, é necessário dispor das expressões analíticas do sistema, especialmente no projeto de sistemas de controle. No entanto, até o presente momento, não temos conhecimento de rotinas computacionais de domínio público que facilitem a dedução das equações de movimento, para um sistema biomecânico.

Outra dificuldade é o fato de que os integradores numéricos usuais não operam com as equações de movimento na forma direta. Em geral, sistemas biomecânicos são modelados por sistemas de equações diferenciais de 2ª ordem, que devem ser colocadas na forma de equações de estado para poderem ser integradas.

Este trabalho apresenta o desenvolvimento de uma *function* para o Matlab denominada EOF2SS (*Equation of Motion to State Space*), que se propõe a deduzir as equações de movimento de um sistema mecânico, usando os recursos de manipulação de variáveis simbólicas do Matlab e em seguida reescrevê-las na forma de equações de estado. Dessa forma seria possível utilizar integradores numéricos de 1º ordem, tais como o ODE45, para realizar a simulação.

Utiliza-se aqui o programa aplicado à simulação de um pêndulo triplo submetido a condições iniciais (Figura 1) e compara-se os resultados com o mesmo sistema modelado e simulado no software comercial ADAMS. O pêndulo triplo foi escolhido por descrever

diversos modelos biomecânicos, tais como os utilizados para estudo de postura (Menegaldo, 2003) e de dedo (Massa, 2002).

Equação de Lagrange

Uma das formas mais sistemáticas e diretas de se deduzir equações de movimento é através do uso das equações de Lagrange. No caso do pêndulo triplo, considerado neste trabalho, uma vez que se determine a origem do sistema fixo de coordenadas e se identifique um número de coordenadas absolutas e independentes entre si, é conceitualmente simples a dedução de tais equações. Basta que se escreva o vetor posição dos centros de massa de cada corpo rígido, em função das coordenadas independentes. Desta forma é possível deduzir as equações de movimento apenas com derivações (totais e parciais) e operações aritméticas simples.

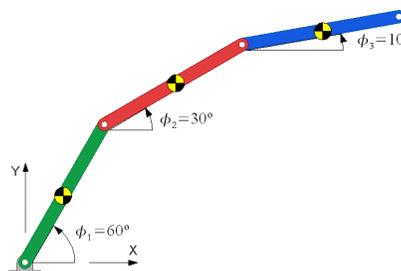


Figura 1: Condição inicial utilizada na simulação.

Diferenciação

Uma das dificuldades encontradas na elaboração da *function* EOM2SS foi o cálculo das derivadas totais (d/dt) que aparece tanto no cálculo das velocidades dos centros de massa quanto na 1ª parcela da equação de Lagrange:

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{q}} \quad (1)$$

O comando `diff` disponível no Matlab não é capaz de calcular a derivada de uma expressão com múltiplas variáveis, em relação a todas as estas variáveis. Este problema foi resolvido usando-se a *function* `fulldiff`¹.

Agrupamento dos termos

Outro problema encontrado foi reescrever a equação de movimento automaticamente na forma de equações de estado. Um sistema de equações diferenciais de 2ª ordem, que representa o comportamento de um sistema mecânico, pode ser expresso, genericamente através da seguinte equação matricial:

$$[A]\{\ddot{q}\} + [B]\{\dot{q}^2\} + [C]\{\dot{q}\} + [D]\{q\} + [E] = [F]\{u\} \quad (2)$$

Para sistemas simples, é possível identificar os elementos das matrizes [A], [B], [C], [D], [E] e [F] manualmente. Fazer isso automaticamente para grandes equações é normalmente um desafio. A estratégia adotada foi reagrupar cada uma das variáveis (de ordem 0, 1 e 2), recursivamente com o comando `collect`. Em seguida, com o auxílio da *function* `SymPoly2Sym`¹, cada um dos elementos que apareciam multiplicados por uma das variáveis eram colocados no elemento correspondente da respectiva matriz. Uma vez que as matrizes eram calculadas, o vetor de acelerações poderia ser isolado da seguinte forma:

$$\{\ddot{q}\} = [A]^{-1} \cdot ([F]\{u\} - [B]\{\dot{q}^2\} - [C]\{\dot{q}\} - [D]\{q\} - [E]) \quad (3)$$

Variáveis de estado

Com o vetor $\{\dot{q}\}$ calculado, basta apenas fazer a substituição das variáveis utilizadas no problema em variáveis de estado e em seguida montar o vetor da derivada das variáveis de estado, que deve estar presente na *function* a ser fornecida como argumento do comando que realiza a integração.

$$\begin{aligned} x_1 &= \phi_1 \xrightarrow{d/dt} \dot{x}_1 = x_4 = \dot{\phi}_1 \\ x_2 &= \phi_2 \xrightarrow{d/dt} \dot{x}_2 = x_5 = \dot{\phi}_2 \\ x_3 &= \phi_3 \xrightarrow{d/dt} \dot{x}_3 = x_6 = \dot{\phi}_3 \\ x_4 &= \dot{\phi}_1 \xrightarrow{d/dt} \dot{x}_4 = \ddot{q}(1) = \ddot{\phi}_1 \\ x_5 &= \dot{\phi}_2 \xrightarrow{d/dt} \dot{x}_5 = \ddot{q}(2) = \ddot{\phi}_2 \\ x_6 &= \dot{\phi}_3 \xrightarrow{d/dt} \dot{x}_6 = \ddot{q}(3) = \ddot{\phi}_3 \end{aligned} \quad (4)$$

Na Equação (5), $\ddot{q}(1), \ddot{q}(2), \ddot{q}(3)$ representam o 1º, 2º e 3º elementos do vetor mostrado na Equação (3), respectivamente.

$$\{\dot{x}\} = \{x_4, x_5, x_6, \ddot{q}(1), \ddot{q}(2), \ddot{q}(3)\}^T \quad (5)$$

Resultados e discussão

Pela análise da Figura (2) pode-se verificar que as curvas obtidas através de simulação utilizando um programa comercial coincidem com as que foram obtidas a partir das equações de movimento geradas automaticamente com o EOM2SS. O código deste programa é de uso livre para fins não comerciais e pode ser solicitado diretamente ao autor.

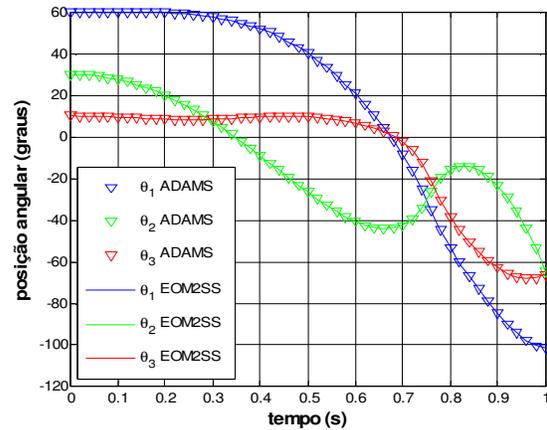


Figura 2: Comparação entre a simulação feita no ADAMS e integração feita com a equação de movimento gerada pelo EOM2SS.

Referências bibliográficas

- Menegaldo, L.L. Fleury, A.T., Weber, H.I., Biomechanical modeling and optimal control of human posture. J. Biomech, v.36, p. 1701-1712, 2003.
- Massa, B., Roccella, S., Carozza, M.C., Dario, P., Design and development of an underactuated prosthetic hand, Proc. IEEE Int. Conf. Robotics & Automation, 2002.

Agradecimentos

Este trabalho é financiado com uma bolsa de mestrado da CAPES e utiliza recursos que foram adquiridos através de Auxílio à Pesquisa FAPERJ.

¹ Disponível em www.mathworks.com/matlabcentral