Universidade Federal da Paraíba

Álgebra Linear Computacional

Título

Problema massa-mola composto por 3 molas

Resumo

O presente trabalho tem como objetivo apresentar um problema de sistema massa-mola onde é utilizada a implementação de sistemas lineares, na qual será mostrado no decorrer, a realização do código feito em octave para o cálculo do sistema utilizando o algoritmo de bisseção, técnica essa apresentada no curso de Álgebra linear computacional.

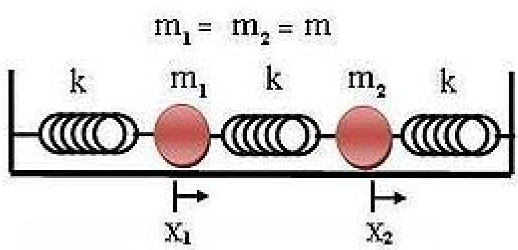
Autores

Nathan Carlos de Macena Gomes
Denilson Pedro Coutinho da silva

Introdução

O sistema massa-mola é composto por dois tipos: **simples** e **composto**. o sistema massa-mola **simples** é constituído por um corpo de **massa m** acoplado a uma mola com fator restaurador k (constante de deformação), enquanto a outra extremidade está ligada a um ponto fixo. Se tal sistema encontra-se em equilíbrio, a posição da massa é denotada por O (x = 0) e toda vez que a massa é deslocada em relação a ponto, surge uma força restauradora **F = - kx**, que tenta trazê-la de volta à situação inicial. As posições -xM e xM representam, respectivamente, a mola comprimida e a mola estendida. Quando o bloco de massa m é deslocado em relação à posição inicial e solto em seguida, o sistema passa a oscilar em torno da posição de equilíbrio.

Por outro lado, o sistema massa-mola **composto** é quando existe mais de uma massa considerada no sistema ou seja,



Onde m_1, m_2 respectivamente são, de acordo com a segunda lei de Newton $F = m.a$

$$m_1 x_1 = -kx_1 - k(x_1 - x_2),\ m_2 x_2 = -kx_2 - k(x_2 - x_1)$$

Analogamente, para um sistema de três molas:

mola 1: $m_1 x_1 = -k_1 x_1 - k_2 x_1 + k_2 x_2$

mola 2: $m_2 x_2 = k_2 x_1 - k_2 x_2 + k_3 x_3 - k_3 x_2$

mola 3: $m_3 x_3 = -k_3 x_3 + k_3 x_2$

- Pegando-se a equação da mola 1 temos:
    - *Colocando em evidência x_1 e x_2*

$$m_1 x_1 = x_1(-k_1 - k_2) + k_2 x_2$$

-
    - *Logo, o vetor x é :*

$$\begin{bmatrix} -k_1 - k_2 \\ k_2 \\ 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

- Pegando-se a equação da mola 2 temos:
    - Colocando em evidência x_1, x_2 e x_3

$$m_2 x_2 = k_2 x_1 + x_2 (-k_2 - k_3) + k_3 x_3$$

-
    - Logo, o vetor x é

$$\begin{bmatrix} k_2 \\ -k_2 - k_3 \\ k_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

- Pegando-se a equação da mola 3 temos:
    - Colocando em evidência x_1, x_2 e x_3

$$m_3 x_3 = -k_3 x_3 + k_3 x_2$$

- Logo, o vetor x é

$$\begin{bmatrix} 0 \\ k_3 \\ -k_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

Então a matriz m é

$$\begin{bmatrix} m_1 & & \\ & m_2 & \\ & & m_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \text{ é igual a } - \begin{bmatrix} k_1 + k_2 & -k_2 & 0 \\ -k_2 & k_1 + k_3 & -k_3 \\ 0 & -k_3 & k_3 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix}$$

ou $m.x(t) + k.x(t) = 0$

onde k *é a matriz*

$$- \begin{bmatrix} k_1 + k_2 & -k_2 & 0 \\ -k_2 & k_1 + k_3 & -k_3 \\ 0 & -k_3 & k_3 \end{bmatrix}$$

e $x(t)$ *é o vetor de acelerações que queremos encontrar. com* $t \varepsilon R$

Podemos notar que a matriz k é tridiagonal, portanto, para encontrarmos a aceleração (vetor x) podemos calcular a bisseção e obter os lambdas, que serão o resultado da aceleração de cada mola.

Então a condição inicial do problema foi 3 massas de três molas (10 kg, 20kg e 30kg) e as constantes elásticas dos blocos (0.2, 0.3 e 0.4) para então calcular a força (massa vezes a aceleração) de cada bloco para obter o x em questão.

Algorítmos

De início foi definida as condições iniciais do problema seguindo a lógica em que cada força depende da outra. Seja o vetor x a aceleração que queremos encontrar, é organizado os vetores x_1, x_2 e x_3, para então formar a matriz m que será multiplicada pelo vetor x. Feito isso, podemos calcular os autovalores da mesma. Como se trata de uma matriz tridiagonal, foi utilizado o algoritmo de bisseção e o fornecido pela função do octave, obtendo assim as acelerações de cada bloco.

O algoritmo de bisseção consiste em buscar raízes que bissecta repetidamente um intervalo e então seleciona um subintervalo contendo a raiz para processamento adicional, compreende vários outros algoritmos: são eles o de Sturm, Gershgorin e o msinal para encontrar os números de raízes.

O algoritmo de Sturm traz a possibilidade de determinar os limites sucessivos inferiores e superiores das raízes reais de uma equação polinomial.

O de Gershgorin tem sua maior importância no cômputo dos autovalores de uma matriz. A aplicação mais refinada do Teorema de Gershgorin possibilita a obtenção de conhecimento mais preciso acerca da localização dos autovalores da matriz calculando o maior intervalo entre os encontrados.

Primeiramente é necessário calcular com o algoritmo de Gershgorin no qual iremos encontrar a localização dos autovalores da matriz (intervalos), em seguida usaremos o algoritmo de sturm para determinar os limites superiores e inferiores das raízes reais, logo após o algoritmo de Sturm usamos o msinal para encontrar a variação de sinais no polinômio,e por último não menos importante usamos a bisseção para calcular os autovalores da matriz.

Código:

problema_massa_mola.m

```
%considerando um sistema de 3 masssas com 3 molas:

m = [20 30 40]; %Massas
k = [0.2 0.3 0.4]; %O quanto a mola expande

A = [ k(1) + k(2) -k(2) 0
      -k(2) k(2)+k(3) -k(3)
      0 -k(3) k(3)

]

b = zeros(size(m));

x = eig(A)

n = size(A);
d = diag(A);
for i = 1:n-1
  e(i) = A(i,i+1);
 endfor

 x = lambda_sturm(d, e)

```

Conclusões e resultados

Concluímos que é possível calcular em um problema real com sistemas lineares. Segue o resultado das acelerações obtidas no problema:

```
x =

   0.047392
   0.466097
   1.086512

x =

   0.043750
   0.481250
   1.093750
```

O primeiro resultado em x foi feito pela função do octave, já o segundo feito com bisseção.

Bibliografia

- MARRION, J.B.; THORNTON, S.T. Classical Dynamics of Particles & Systems, 1995.
- COHEN-TANNOUDJI, C.; DIU, B.; LALOË, F. Quantum Mechanics, 1ª edição. Wiley, Vol. 2, p.1442-1446, 1977.