

Segundo EE Vibrações Mecânicas

2016.1

Questão 1

As matrizes de massa, rigidez e a matriz de flexibilidade são, respectivamente:

```
import numpy as np
import numpy.linalg as la

m = np.diag([2.5, 2, 4, 2.5])
k = np.matrix([[12, -2.5, 0.0, -2.5],
               [-2.5, 10.0, -3.0, -4.5],
               [0.0, -3.0, 21.0, -18.0],
               [-2.5, -4.5, -18, 32]])
a = np.matrix([[ 0.1032, 0.0562, 0.0419, 0.0395],
               [ 0.0562, 0.1831, 0.1004, 0.0866],
               [ 0.0419, 0.1004, 0.1484, 0.1009],
               [ 0.0395, 0.0866, 0.1009, 0.1032]])

show(m, k, a)
```

```
[[ 2.5  0.   0.   0. ]  [[ 12.  -2.5  0.  -2.5]  [[ 0.1032  0.0562  0.0419  0.0395]
 [ 0.   2.   0.   0. ]  [ -2.5  10.  -3.  -4.5]  [ 0.0562  0.1831  0.1004  0.0866]
 [ 0.   0.   4.   0. ]  [  0.  -3.  21. -18. ]  [ 0.0419  0.1004  0.1484  0.1009]
 [ 0.   0.   0.   2.5]] [ -2.5 -4.5 -18.  32. ]] [ 0.0395  0.0866  0.1009  0.1032]]
```

Só para verificar que eu digitei tudo certo

```
show(a*k)
```

```
[[ 9.99150000e-01  5.50000000e-04  3.00000000e-04 -1.10000000e-01]
 [ 1.50000000e-04  9.99600000e-01  3.00000000e-04 -4.50000000e-01]
 [-4.50000000e-04 -5.55111512e-17  9.99000000e-01  1.05000000e-01]
 [-5.00000000e-04  1.50000000e-04  1.50000000e-03  9.97000000e-01]]
```

Como nós estamos usando apenas 4 casas decimais na matrix de flexibilidade, a inversa não é exata, os termos da ordem de 10^{-4} devem ser considerados com zero.

A matriz dinâmica é dada por $D = K^{-1}m = am$, **mas não é necessário calculá-la**, pois é necessária apenas para o cálculo dos seus autovalores e autovetores, que **já são dados!**

```
D=la.inv(k)*m
show(D)
v,X = la.eig(D)
show(v)
show(X)
```

```
[[ 0.25821955  0.11242443  0.16778751  0.09892331]
 [ 0.14053053  0.36634739  0.40173901  0.21661232]
 [ 0.1048672   0.20086951  0.59384553  0.25227566]
 [ 0.09892331  0.17328986  0.40364106  0.25821955]]

[[ 1.03350843  0.22331741  0.15752536  0.06228082]

 [[-0.27508373 -0.83306376  0.48719524 -0.07004014]
 [-0.5763069  -0.39678346 -0.84070492 -0.12800445]
 [-0.60314358  0.36340147  0.21996516 -0.37272556]
 [-0.47793004  0.12847967  0.08643699  0.91639778]]
```

Tudo o que está acima é dado no enunciado, a resolução da prova começa efetivamente aqui.

Parte a)

Se X_i e X_j são modos normais, temos que $X_i^t k X_j = 0$, para $i \neq j$ e $X_i^t k X_j = K_{ii}$. A mesma coisa vale para a matriz de massa. K_i and M_{ii} são os coeficientes de rigidez e massa generalizados.

Vamos pegar o modo 1, e multiplicá-lo como acima contra o modo 2, com as duas matrizes.

```
x1 = X[:, [0]]
x2 = X[:, [1]]
show(x1, x2)
show(x1.T*k*x2)
show(x1.T*m*x2)
```

```

[[-0.27508373]  [[-0.83306376]
 [-0.5763069 ]  [-0.39678346]
 [-0.60314358]  [ 0.36340147]
 [-0.47793004]] [ 0.12847967]]

[[ -2.77555756e-15]]
[[ 2.22044605e-16]]

```

Isto é tão próximo de zero quanto dá para ser numericamente. Claramente os dois modos escolhidos são ortogonais em relação às matrizes de massa e rigidez.

Parte b)

Conforme presente no formulário e explicado em sala de aula, as frequências naturais estão relacionadas com os autovalores da matriz dinâmica por $\lambda_i = 1/\omega_i^2$. Podemos calcular então todas as frequências naturais facilmente.

```

omega = 1.0/np.sqrt(v)
show(omega)

```

```

__main__:3: VisibleDeprecationWarning: using a non-integer number
instead of an integer will result in an error in the future

[ 0.98365542  2.1161123  2.5195603  4.00703215]

```

Claramente então a frequência fundamental é 0.9836 rad/s

Parte c)

Conforme explicado na parte a)

```

K11 = x1.T*k*x1
M11 = x1.T*m*x1
show(K11, M11)

```

```

[[ 2.78624573]] [[ 2.87960845]]

```

Parte d)

Para normalizar o modo, basta dividi-lo pela raiz quadrada do coeficiente de massa generalizado. (Na célula anterior os resultados terminaram como matrizes 1×1 , daí vou extrair o valor daqui a pouco).

```
m11 = M11[0,0]
x1n = x1/sqrt(m11)
show(x1n)
```

```
[[ -0.16210566]
 [ -0.33961518]
 [ -0.35542993]
 [ -0.28164212]]
```

Vamos verificar os coeficientes de massa e rigidez generalizados.

```
M11n = x1n.T*m*x1n
K11n = x1n.T*k*x1n
show(M11n, np.sqrt(K11n))
```

```
[[ 1.]] [[ 0.98365542]]
```

Claramente, o coeficiente de massa generalizado é a primeira frequência natural ao quadrado.

Parte e)

O primeiro modo e o quarto modo são:

```
x1 = X[:, [0]]
x4 = X[:, [3]]
show(x1, x4)
```

```
[[ -0.27508373]  [[ -0.07004014]
 [ -0.5763069 ]  [[ -0.12800445]
 [ -0.60314358]  [[ -0.37272556]
 [ -0.47793004]]  [[ 0.91639778]]
```

Para fazer com que a maior amplitude seja unitária, basta dividir pelo maior valor que aparece em cada vetor.

```
x1n = x1/x1[2]
x4n = x4/x4[3]
show(x1n, x4n)
```

```
[[ 0.45608332]  [[-0.07642985]
 [ 0.95550531]  [-0.1396822 ]
 [ 1.          ]  [-0.40672901]
 [ 0.79239845]] [ 1.          ]]
```

A energia potencial elástica é dada por $\frac{1}{2} x^t k x$, então,

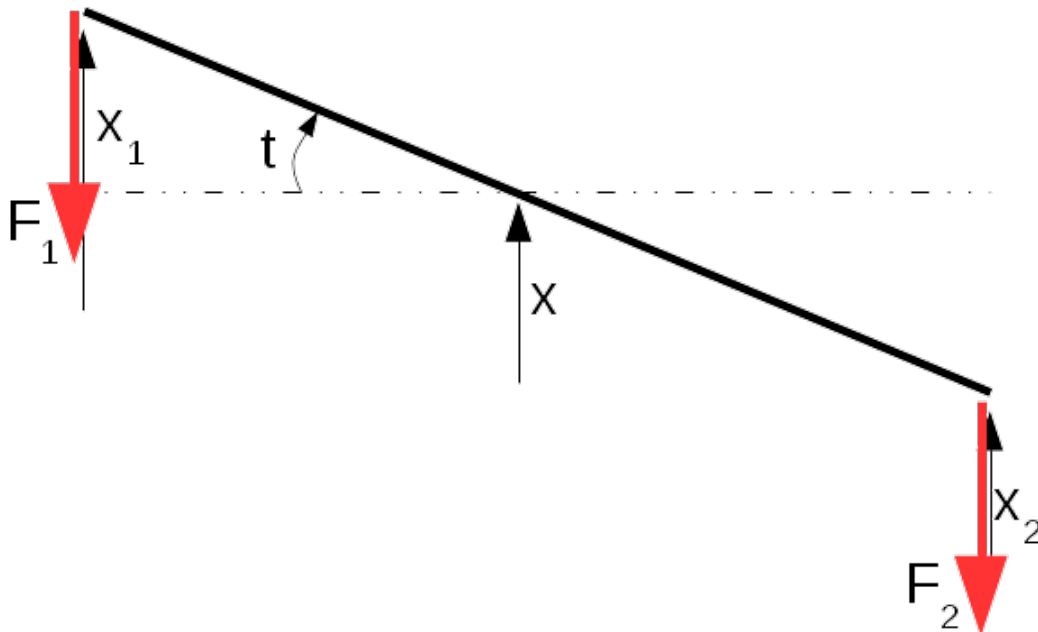
```
U1 = x1n.T*k*x1n
U4 = x4n.T*k*x4n
show(U1, U4)
```

```
[[ 7.6591045]] [[ 51.62648453]]
```

Percebam como a energia potencial elástica total para o quarto modo é muito maior do que aquela para o primeiro modo, mesmo com as amplitudes máximas iguais.

Questão 2

Na figura abaixo está mostrado um DCL da barra. Para facilitar a minha vida, representei a rotação θ pela letra t . Para ter que escrever menos o divisor "2", vou tomar o comprimento total da barra com $2l$, leve isto em consideração quando for comparar com a sua solução.



Temos as óbvias restrições geométricas dadas pela rigidez da barra:

$$x = \frac{x_1 + x_2}{2} \quad \text{e} \quad \theta = \frac{x_1 - x_2}{2l}$$

Do somatório de forças na direção vertical temos,

$$m\ddot{x} = -2kx_1 - kx_2, \quad \text{ou} \quad m\ddot{x} + 2kx_1 + kx_2 = 0$$

Do somatório de momentos, temos

$$J_0\ddot{\theta} = -2klx_1 + klx_2 \quad \text{ou} \quad J_0\ddot{\theta} + 2klx_1 - klx_2 = 0$$

Temos então o seguinte sistema de equações,

$$\begin{cases} m\ddot{x} + 2kx_1 + kx_2 = 0 \\ J_0\ddot{\theta} + 2klx_1 - klx_2 = 0 \end{cases}$$

Obviamente temos que substituir θ e x pelos valores acima,

$$\begin{cases} m \frac{\ddot{x}_1 + \ddot{x}_2}{2} + 2kx_1 + kx_2 = 0 \\ J_0 \frac{\ddot{x}_1 - \ddot{x}_2}{2l} + 2klx_1 - klx_2 = 0 \end{cases}$$

Podemos reescrever o sistema como

$$\begin{cases} m\ddot{x}_1 + m\ddot{x}_2 + 4kx_1 + 2kx_2 = 0 \\ \frac{J_0}{l^2}\ddot{x}_1 - \frac{J_0}{l^2}\ddot{x}_2 + 4kx_1 - 2kx_2 = 0 \end{cases}$$

O sistema pode ser escrito na forma matricial como:

%hide

$$\begin{bmatrix} m & m \\ \frac{J_0}{l^2} & -\frac{J_0}{l^2} \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} 4k & 2k \\ 4k & -2k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Tanto a matriz de massa quanto a matriz de rigidez apresentam dois graves problemas:

- Não são simétricas!
- Apresentam termos negativos na diagonal principal

A primeira viola o teorema da reciprocidade e a segunda implica que a aceleração desviada uma força é na direção oposta à força, o que é meio ruim de engolir.

Na verdade, não há problema algum. As matrizes estão deste jeito por um acidente de formulação. Se as matrizes de massa e rigidez estão calculadas corretamente, elas sempre podem ser reescritas como matrizes simétricas. Lembres-se que estas matrizes são apenas representações dos coeficientes do sistema de equações diferenciais ordinárias acima. Um sistema de equações não se altera quando fazemos combinações lineares entre suas linhas. Esta é, inclusive, uma das ferramentas mais usadas para resolver um sistema de equações.

Neste caso particular, vamos substituir a primeira equação pela soma das duas equações.

%hide

$$\begin{bmatrix} m + \frac{J_0}{l^2} & m - \frac{J_0}{l^2} \\ \frac{J_0}{l^2} & -\frac{J_0}{l^2} \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} 8k & 0 \\ 4k & -2k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Fazemos a mesma coisa, substituindo a segunda equação pela diferença entre a primeira e a segunda equações originais.

%hide

$$\begin{bmatrix} m + J_0/l^2 & m - J_0/l^2 \\ m - J_0/l^2 & m + J_0/l^2 \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} 8k & 0 \\ 0 & 4k \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Que é lindamente simétrica e que não tem termos negativos na diagonal principal.

Importante: realizar uma combinação linear entre linhas e colunas da matriz não muda seu determinante, portanto calcular as frequências naturais com o sistema original ou com este transformado deve dar o mesmo resultado.

Curiosidade: percebam que neste sistema, com o sistema de coordenadas generalizadas x_1 e x_2 , o sistema tem acoplamento dinâmico, mas não tem acoplamento estático.

Frequências Naturais

Vamos supor que x_1 e x_2 são harmônicas, isto é $x_i = X_i \cos(\omega t - \phi)$, e substituímos no sistema de equações diferenciais ordinárias acima, com estamos cansados de fazer. Obtemos

%hide

$$-\omega^2 \begin{bmatrix} m + J_0/l^2 & m - J_0/l^2 \\ m - J_0/l^2 & m + J_0/l^2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \cos(\omega t - \phi) + \begin{bmatrix} 8k & 0 \\ 0 & 4k \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} \cos(\omega t - \phi)$$

Como isto tem que ser nulo para qualquer tempo, podemos eliminar o cos, e, rearrumando

%hide

$$\begin{bmatrix} 8k - \omega^2(m + J_0/l^2) & -\omega^2(m - J_0/l^2) \\ -\omega^2(m - J_0/l^2) & 4k - \omega^2(m + J_0/l^2) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Antes de seguir em frente, é útil lembrar que (o que eu deveria ter feito antes, mas vai dar muito trabalho e eu não vou refazer agora) $J_0 = ml^2/12$, e portanto,

$$J_0/l^2 = ml^2/(12l^2) = m/12$$

%hide

$$\begin{bmatrix} 8k - \omega^2(m + m/12) & -\omega^2(m - m/12) \\ -\omega^2(m - m/12) & 4k - \omega^2(m + m/12) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

ou

%hide

$$\begin{bmatrix} 8k - \omega^2(13m/12) & -\omega^2 11m/12 \\ -\omega^2 11m/12 & 4k - \omega^2(13m/12) \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

Como o sistema é homogêneo, só tem solução quando o discriminante é nulo. Vou fazer simbolicamente agora, para facilitar a minha vida. Vou também já cancelar o $1/12$ que aparece em todos os termos.

```
var('k, m, omega')
R = matrix([[ 8*k - 13*m*omega^2, -11*m*omega^2], [-11*m*omega^2,
4*k-13*m*omega^2]])
show(R)
```

$$\begin{pmatrix} -13m\omega^2 + 8k & -11m\omega^2 \\ -11m\omega^2 & -13m\omega^2 + 4k \end{pmatrix}$$

A equação característica é o determinante desta matriz.

```
eq = R.det().full_simplify()
show(eq)
```

$$48m^2\omega^4 - 156km\omega^2 + 32k^2$$

Para simplificar, vou fazer $\omega^2 = y$.

```
var('y')
eq2=eq.substitute(omega^2 == y, omega^4 == y^2)
show(eq2)
```

$$48m^2y^2 - 156kmy + 32k^2$$

Resolvendo para as raízes (lembrando que estes são os quadrados das frequências naturais)

```
roots = eq2.solve(y)
w12, w22 = [ root.rhs() for root in roots]
show([w12, w22])
```

$$\left[-\frac{k(\sqrt{1137} - 39)}{24m}, \frac{k(\sqrt{1137} + 39)}{24m} \right]$$

Eu gostaria destes números como números de ponto flutuante, para poder comparar com as soluções das provas, e por que eu sou engenheiro. Este truque é meio bobo...

```
w12n = N(w12(k=1.0, m=1.0))*k/m
w22n = N(w22(k=1.0, m=1.0))*k/m
show([w12n, w22n])
```

$$\left[\frac{0.220023724992719 k}{m}, \frac{3.02997627500728 k}{m} \right]$$

As frequências naturais são então

```
w1 = sqrt(w12n)
w2 = sqrt(w22n)
show([w1, w2])
```

$$\left[0.469066866227747 \sqrt{\frac{k}{m}}, 1.74068270371348 \sqrt{\frac{k}{m}} \right]$$

Modos Normais

Basta usar qualquer uma das linhas do sistema homogêneo, com as duas frequências.

```
l1=R[0]
show(l1)
```

$$(-13 m\omega^2 + 8 k, -11 m\omega^2)$$

```
l1w = l1(omega=w1)
show(l1w)
```

$$(5.13969157509465 k, -2.42026097491991 k)$$

Dividindo a amplitude do lado esquerdo pela amplitude do lado direito

```
r1 = l1w[1]/l1w[0]
show(r1)
```

$$-0.470896150003971$$

Fazendo a mesma coisa para a segunda frequência natural

```
l2w = l1(omega=w2)
show(l2w)
```

$$(-31.3896915750947 k, -33.3297390250801 k)$$

```
r2 = l2w[1]/l2w[0]
show(r2)
```

$$1.06180524091306$$

Obviamente, os modos normais, com a máxima amplitude normalizada são

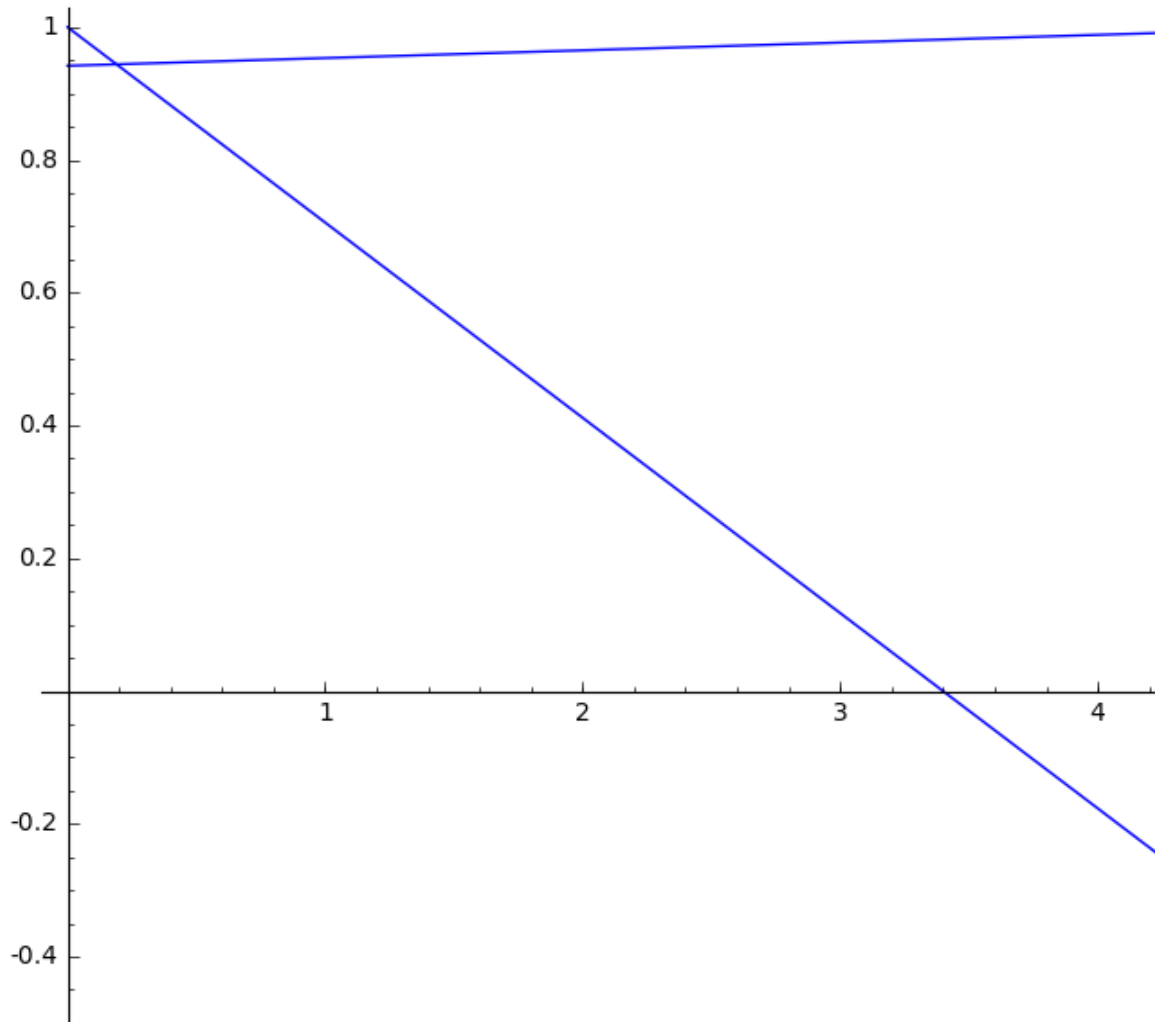
```
X1=(1.0, r1)
X2=(1/r2, 1.0)
show(X1)
show(X2)
```

$$(1.00000000000000, -0.470896150003971)$$

$$(0.941792300007943, 1.00000000000000)$$

Para fazer uma figura rápida, , esquemática, vamos supor que o comprimento original é igual a 5 unidades.

```
l=5.0  
v1 = [(0.0, X1[0]), (l, X1[1])]   
v2 = [(0.0, X2[0]), (l, X2[1])]   
line(v1)+line(v2)
```



É curioso que no segundo modo a barra fica quase horizontal. No primeiro modo, a amplitude é muito maior no lado no qual a rigidez é menor, como era de se esperar. Como os dois lados tem quase a mesma amplitude, a energia potencial elástica total é maior neste caso do que no primeiro.