PME3380 – Modelagem de Sistemas Dinâmicos
LISTA E

José Felipe Felix Rafael
10333139
22/10/2020

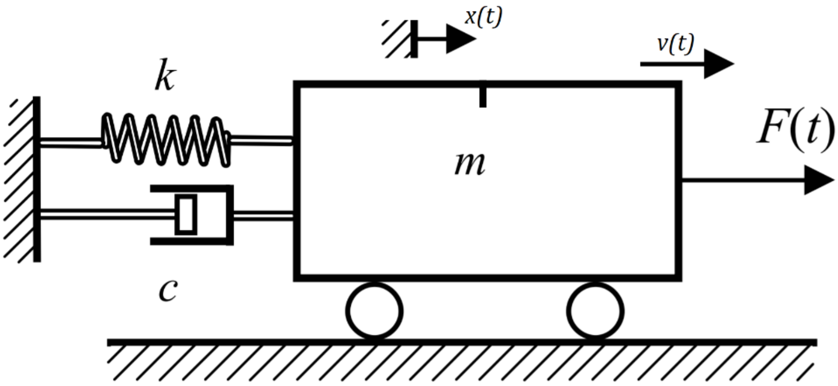


Figura 1 - Modelo físico

Equação diferencial do movimento
Do Teorema do Movimento do Baricentro:

$$m\ddot{x} = -kx - cv + F(t)$$

Variáveis de estado
Descrevem onde o sistema está e aonde ele vai.

$$\begin{cases} x_1 = x(t) \\ x_2 = v(t) \end{cases}$$

Equações de estado

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\dfrac{c}{m}x_2 - \dfrac{k}{m}x_1 + \dfrac{1}{m}F \end{cases}$$

Entrada
Forçante:

$$F(t) = u$$

Saída
Deslocamento da massa/distensão da mola:

$$x(t) = y$$

Matricialmente

$$\begin{bmatrix}\dot{x}_1\\ \dot{x}_2\end{bmatrix} = \begin{bmatrix}0 & 1\\ -\frac{k}{m} & -\frac{c}{m}\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{m}\end{bmatrix}F(t)$$
$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu}$$

$$y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}x_1\\ x_2\end{bmatrix} + [0]F(t)$$
$$\boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du}$$

Ponto de equilíbrio
Derivadas no tempo das variáveis de estado são nulas. Definindo $F_e = 0$.

$$\dot{x} = v = 0$$
$$\dot{v} = 0$$

$$\therefore \begin{cases} x_e = 0 \\ v_e = 0 \end{cases}$$

Função de transferência
Aplicando as transformadas de Laplace às equações de estado

$$\boldsymbol{\mathcal{L}}[f(t)] = F(s)$$
$$\boldsymbol{\mathcal{L}}\left[\dot{f}(t)\right] = sF(s) - f(0)$$

Obtemos o sistema

$$\begin{cases} sX_1 - x_1(0) = X_2 \\ sX_2 - x_2(0) = -\frac{c}{m}X_2 - \frac{k}{m}X_1 + \frac{1}{m}U \end{cases}$$

Impondo $x_1(0) = x(0) = 0$ (sistema parte da posição de equilíbrio) e $x_2(0) = v(0) = 0$ (sistema parte do repouso)

$$\begin{cases} sX_1 = X_2 \\ sX_2 = -\frac{c}{m}X_2 - \frac{k}{m}X_1 + \frac{1}{m}U \end{cases}$$

Temos um sistema de equações algébricas (não mais diferenciais). Ou seja, podemos manipulá-lo de maneira a obter

$$X_1 = \frac{1}{ms^2 + cs + k}U$$

Lembrando que $y = x = x_1$, definimos a função de transferência $G(s)$

$$G(s) = \frac{Y}{U} = \frac{1}{ms^2 + cs + k} = \frac{N(s)}{D(s)}$$

Notar que a ordem do denominador $D(s)$ é superior à do numerador $N(s)$.

Solução analítica

Este problema possui solução analítica, que será apresentada para fins de comparação.

Definindo

$$\omega_0 \triangleq \sqrt{k/_m}$$

$$\gamma \triangleq \frac{c}{m}$$

Vale ressaltar que ω_0 é a frequência natural do sistema.

Fazendo

$$F(t) = F_0$$

A EDO do movimento é reescrita como

$$\ddot{x} + \gamma \dot{x} + {\omega_0}^2 x = \frac{F_0}{m}$$

Sendo as condições iniciais dadas por

$$\begin{cases} x_0 = x(t = 0) \\ \dot{x}_0 = \dot{x}(t = 0) \end{cases}$$

Existem três situações distintas:

1. Amortecimento subcrítico ($\omega_0 > \gamma/2$)

$$x(t) = Ae^{-\frac{\gamma}{2}t} cos(\omega t + \varphi) + F_0/k$$

onde

$$\omega \triangleq \sqrt{{\omega_0}^2 - \frac{\gamma^2}{4}}$$

com

$$\varphi = \tan^{-1} \frac{(x_0 - F_0/k)\,\gamma/2 - \dot{x}_0}{\omega(x_0 - F_0/k)}$$

$$A = \frac{-1}{\omega} \sqrt{\left(x_0 - \frac{F_0}{k}\right)^2 {\omega_0}^2 - \dot{x}_0 \gamma \left(x_0 - \frac{F_0}{k}\right) + {\dot{x}_0}^2}$$

2. Amortecimento crítico ($\omega_0 = \gamma/2$)

$$x(t) = e^{-\frac{\gamma}{2}t}(At + B) + F_0/k$$

com

$$A = \dot{x}_0 + \frac{\gamma}{2}\left(x_0 - \frac{F_0}{k}\right)$$

$$B = x_0 - \frac{F_0}{k}$$

3. Amortecimento supercrítico ($\omega_0 < \gamma/2$)

$$x(t) = Ae^{\left(-\frac{\gamma}{2}+\beta\right)t} + Be^{\left(-\frac{\gamma}{2}-\beta\right)t} + F_0/k$$

onde

$$\beta \triangleq \sqrt{\frac{\gamma^2}{4} - {\omega_0}^2}$$

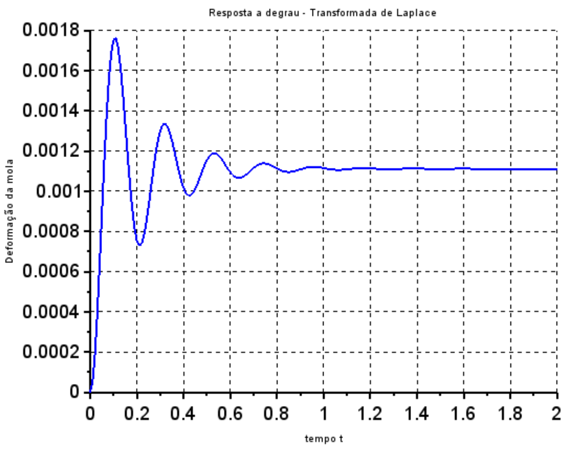
com

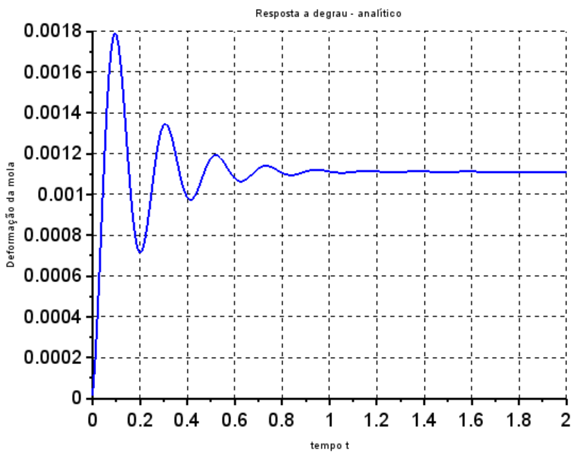
$$A = \frac{\dot{x}_0 - (x_0 - F_0/k)\left(-\frac{\gamma}{2} - \beta\right)}{2\beta}$$

$$B = \frac{\dot{x}_0 - (x_0 - F_0/k)\left(-\frac{\gamma}{2} + \beta\right)}{-2\beta}$$

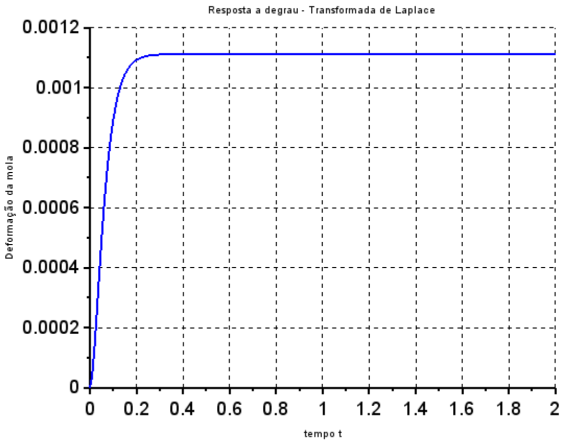
Resultados das simulações

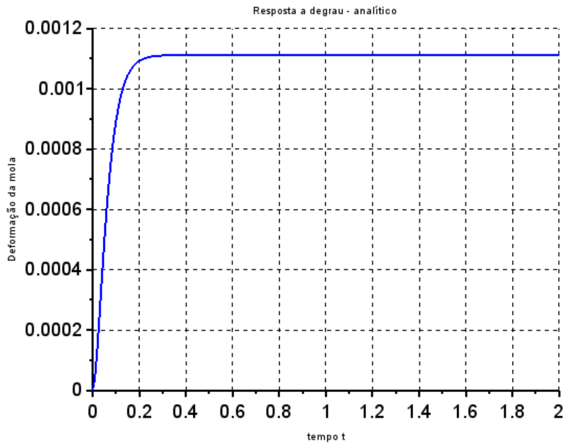
a) $m = 1\ kg$, $k = 900\ N/m$, $c = 10\ Ns/m$: amortecimento subcrítico.



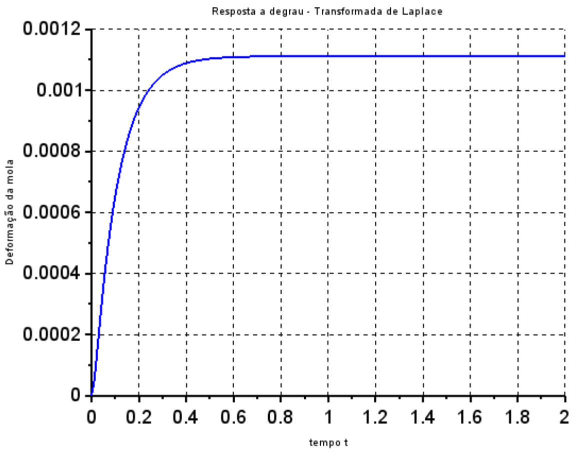


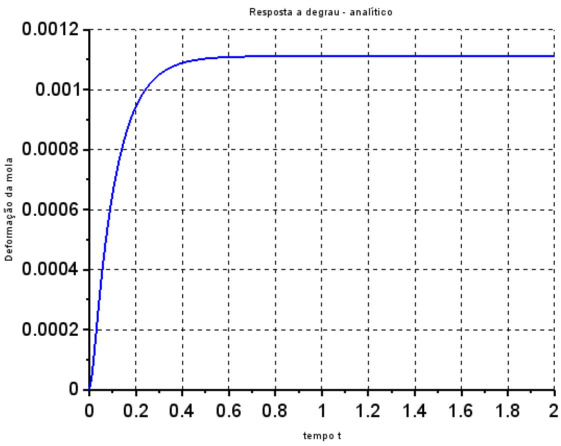
b) $m = 1\,kg$, $k = 900\,N/m$, $c = 60\,Ns/m$: amortecimento crítico.





c) $m = 1\,kg$, $k = 900\,N/m$, $c = 100\,Ns/m$: amortecimento supercrítico.





Análise

Observa-se que os resultados obtidos pela função de transferência são praticamente iguais aqueles obtidos analiticamente nos casos "crítico" e "supercrítico", mas no caso "subcrítico" apresentam uma pequena dessemelhança aos resultados analíticos.

QUESTÃO 1

Os autovalores λ de **A** são tais que

$$\det(\boldsymbol{A} - \lambda \boldsymbol{I}) = 0$$

em que ***I*** denota a matriz identidade.

$$\begin{vmatrix} -\lambda & 1 \\ -\dfrac{k}{m} & -\dfrac{c}{m} - \lambda \end{vmatrix} = 0 \rightarrow \lambda = -\frac{c}{2m} \pm \sqrt{\frac{c^2}{4m^2} - \frac{k}{m}}$$

Observa-se que os autovalores de **A** são iguais às raízes do polinômio $D(s)$

$$D(s) = ms^2 + cs + k \rightarrow s = -\frac{c}{2m} \pm \sqrt{\frac{c^2}{4m^2} - \frac{k}{m}}$$

Definindo

$$\zeta \triangleq c/2\sqrt{km}$$

no caso em que temos

$$\zeta < 1 \rightarrow \frac{c^2}{4m^2} < \frac{k}{m} \rightarrow \lambda \in \mathbb{C}$$

podemos então escrever

$$\lambda = -\frac{c}{2m} \pm i\sqrt{\frac{k}{m} - \frac{c^2}{4m^2}}$$

em que $i = \sqrt{-1}$ denota a unidade imaginária.

Neste caso, verifica-se que

$$|\lambda| = \omega_0$$

$$\begin{aligned} |\lambda| &= \sqrt{\lambda \cdot \lambda^*} \\ &= \sqrt{\left(-\frac{c}{2m} + i\sqrt{\frac{k}{m} - \frac{c^2}{4m^2}}\right) \cdot \left(-\frac{c}{2m} - i\sqrt{\frac{k}{m} - \frac{c^2}{4m^2}}\right)} \\ &= \sqrt{\frac{c^2}{4m^2} + \frac{k}{m} - \frac{c^2}{4m^2}} = \sqrt{\frac{k}{m}} \triangleq \omega_0 \end{aligned}$$

em que $*$ denota o conjugado do número complexo e ω_0 é a frequência natural do sistema.

Observa-se também que

$$Im\{\lambda\} = \sqrt{\frac{k}{m} - \frac{c^2}{4m^2}} \triangleq \omega$$

em que ω é a frequência do sistema.

Ainda, demonstra-se que

$$\frac{|Re\{\lambda\}|}{|\lambda|} = \zeta$$

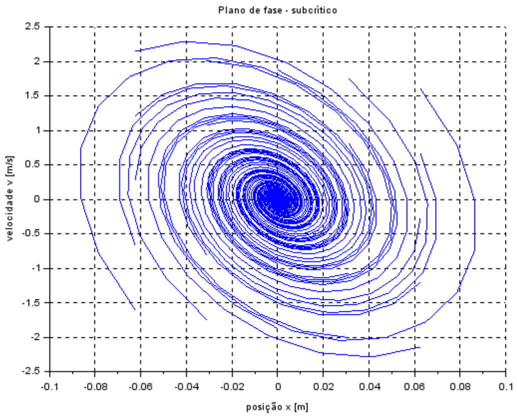
$$\frac{|Re\{\lambda\}|}{|\lambda|} = \frac{c/2m}{\sqrt{k/m}} = \frac{c}{2\sqrt{km}} \triangleq \zeta$$

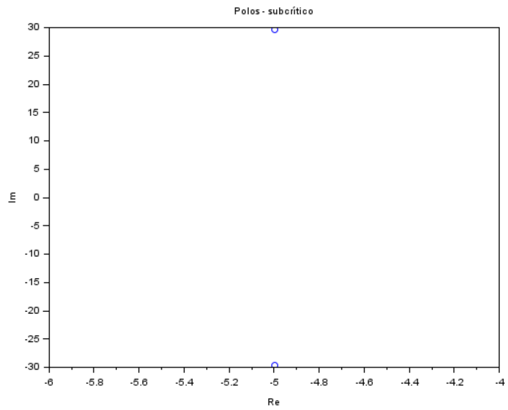
QUESTÃO 2

Para a construção do plano de fase, foram consideradas como condições iniciais

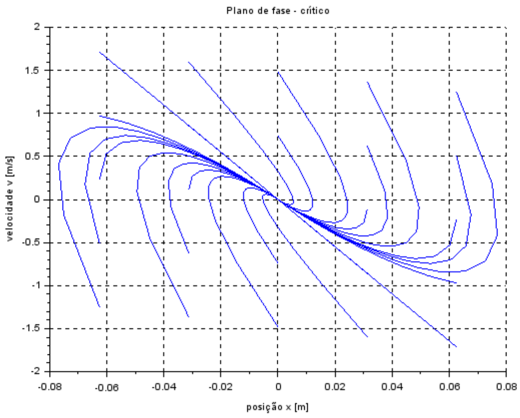
$$x_0 \in \{-2, -1, 0, 1, 2\}\ [m] \text{ e } \dot{x}_0 \in \{-2, -1, 0, 1, 2\}\ [m/s].$$

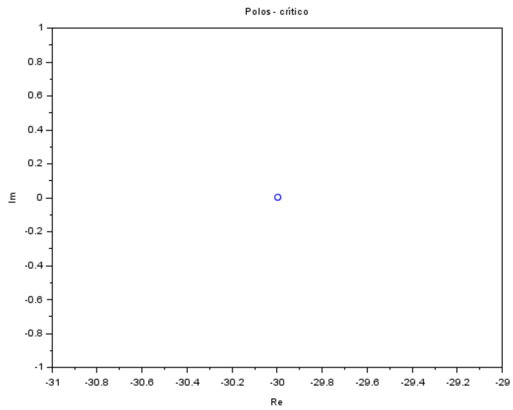
a) $m = 1\ kg$, $k = 900\ N/m$, $c = 10\ Ns/m$: amortecimento subcrítico.



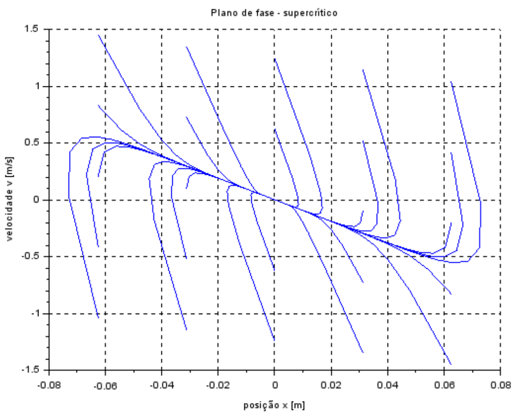


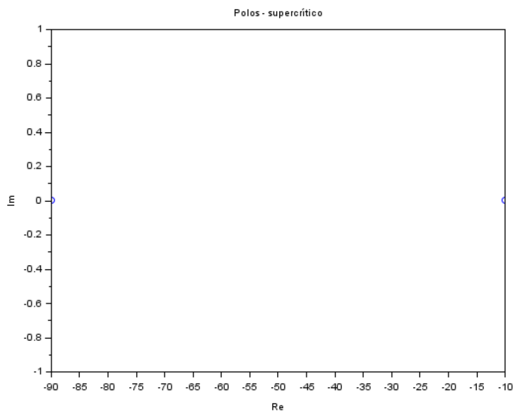
b) $m = 1\ kg$, $k = 900\ N/m$, $c = 60\ Ns/m$: amortecimento crítico.





c) $m = 1\ kg$, $k = 900\ N/m$, $c = 100\ Ns/m$: amortecimento supercrítico.





Código do Scilab

Solução por função de transferência

```
//TRANSFORMADA DE LAPLACE//
//Parâmetros do sistema:
m=1 //massa [kg];
k=900 //constante elástica [N/m];
c=10 //constante de amortecimento [Ns/m].

// Definindo os polinômios da função de transferência
// Numerador:
n=poly([1],'s','coeff')
// Denominador:
d=poly([k c m],'s','coeff') //observe a ordem contrária dos coeficientes
// Montando a função de transferência G, onde o parâmetro 'c' indica sistema de
//tempo contínuo. Se for um sistema de tempo discreto, use o parâmetro 'd'.
G=syslin('c',n/d)
// Simulando o sistema para uma entrada degrau (F=0 para t<0 e F=1 para t>0)
// Definindo o vetor tempo:
t=0:0.01:2
// Definindo a entrada:
u=ones(t)
// Definindo o vetor de condições iniciais:
// O sistema é de segunda ordem, logo são duas condições iniciais.
// Não definindo as condições iniciais o programa assume como sendo nulas.
x0=[0;0]; // x(0) = 0 e a derivada de x(t) no instante inicial tambem é nula.
// Realizando a simulação com o comando csim:
[y]=csim(u,t,G,x0);
// Abrindo uma nova janela de gráficos:
xset('window',1)
// Mostrando o resultado da simulação:
xset('thickness',2)
xset('font size',4)
plot2d(t,y,2)
xtitle('Resposta a degrau - Transformada de Laplace','tempo t','Deformação da mola')
xgrid(0,1)
```

Solução analítica

```
//ANALÍTICO

m = 1 //massa [kg];
k = 900 //constante elástica [N/m];
c = 10 //constante de amortecimento [Ns/m];
F0 = 1 //módulo da forçante [N].

//condições iniciais:
x0 = 0 //posição inicial [m];
xp0 = 0 //veocidade inicial [m/s].

gama = c/m
omega0 = sqrt(k/m)

//vetor tempo:
t=0:0.01:2

if omega0 > gama/2 then
  //caso sub-crítico

  omega = sqrt(omega0^2-gama^2/4)
  phi = atan((gama/2*(x0-F0/k)-xp0)/(omega*(x0-F0/k)))
  A = sqrt((x0-F0/k)^2*omega^2-xp0*gama*(x0-F0/k)+xp0^2)/omega
```

```
   funcprot(0)
   function x=deslocamento(t)
      x = -A*exp(-gama/2*t).*cos(omega*t+phi)+F0/k
   endfunction

elseif omega0 == gama/2 then
   //caso crítico

   A = xp0+gama/2*(x0-F0/k)
   B = x0-F0/k

   function x=deslocamento(t)
      x = exp(-gama/2*t).*(A*t+B)+F0/k
   endfunction

elseif omega0 < gama/2 then
   //caso super-crítico

   betaa = sqrt(gama^2/4-omega0^2)
   A = (xp0-(x0-F0/k)*(-gama/2-betaa))/(2*betaa)
   B = (xp0-(x0-F0/k)*(-gama/2+betaa))/(-2*betaa)

   function x=deslocamento(t)
      x = A*exp((-gama/2+betaa)*t)+B*exp((-gama/2-betaa)*t)+F0/k
   endfunction

end

y=deslocamento(t)

xset('window',2)
//plotando
xset('thickness',2)
xset('font size',4)
plot2d(t,y,2)
xtitle('Resposta a degrau - analítico','tempo t','Deformação da mola')
xgrid(0,1)
```

Plano de fase

```
//PLANO DE FASE
//Parâmetros do sistema:
m=1 //massa [kg];
k=900 //constante elástica [N/m];
c=100 //constante de amortecimento [Ns/m].

// Definindo os polinômios da função de transferência
// Numerador:
n=poly([1],'s','coeff')
// Denominador:
d=poly([k c m],'s','coeff') //observe a ordem contrária dos coeficientes
// Montando a função de transferência G, onde o parâmetro 'c' indica sistema de
//tempo contínuo. Se for um sistema de tempo discreto, use o parâmetro 'd'.
G=syslin('c',n/d)
// Simulando o sistema para uma entrada degrau (F=0 para t<0 e F=1 para t>0)
// Definindo o vetor tempo:
t=0:0.01:2
// Definindo a entrada:
u=zeros(t) //forçante externa nula

//varre alguns conjuntos de condições iniciais
for i = 0:4
   for j = 0:4
      x0=[i-2;j-2]; //condições iniciais
```

```
    [y]=csim(u,t,G,x0); //posição
    [v]=diff(y)/(0.01); //velocidade
    plot2d(y(1:$-1),v,2) //plota v(y)
  end
end

xtitle('Plano de fase - supercrítico','posição x [m]','velocidade v [m/s]')
xgrid(0,1)

//polos
lambda1=-c/(2*m)+sqrt(c^2/(4*m^2)-k/m)
lambda2=-c/(2*m)-sqrt(c^2/(4*m^2)-k/m)

//plotando os polos no plano complexo
xset('window',2)
plot(real(lambda1),imag(lambda1),'o')
plot(real(lambda2),imag(lambda2),'o')

xtitle('Polos - supercrítico','Re','Im')
```