Escola Politécnica da Universidade de São Paulo
Departamento de Engenharia Mecânica

PME3380 - Modelagem de Sistemas Dinâmicos
Lista E

Kevin Chu
N°USP: 10705908

São Paulo
2020

Exercício 1:

Obtendo as equações de estado e a função de transferência do sistema apresentado na figura 1 a seguir:

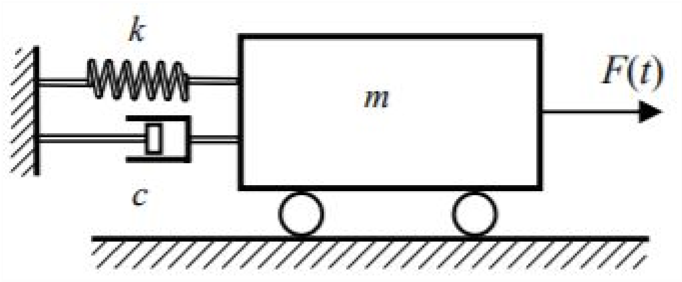


Figura 1 - Esquema do sistema massa, mola, amortecedor

Do esquema, é possível aplicar o TMB, obtendo a seguinte equação:

$$m\ddot{x} + c\dot{x} + kx = F(t) \Rightarrow \ddot{x} = -\frac{c}{m}\dot{x} - \frac{k}{m}x + F(t)$$

Representando a equação diferencial na forma matricial:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} F(t)$$

Calculando os autovalores da matriz A:

$$matriz\ A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix}$$

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

Calculando a função de transferência:

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

Raízes do denominador da função de transferência:

$$ms^2 + cs + k = 0 \Rightarrow s = -\frac{c \pm \sqrt{c^2 - 4mk}}{2m}$$

Percebe-se que os autovalores da matriz A equivalem às raízes do denominador da função de transferência.

Considerando os valores de m=1kg, k=900 N/m e ζ =0.1 < 1:
Verifica-se que o módulo do número complexo é igual à frequência natural do sistema:

$$\lambda = -\frac{6 \pm \sqrt{6^2 - 4 * 1 * 900}}{2 * 1} \Rightarrow |\lambda| = \sqrt{3^2} + (9 * \sqrt{11})^2 = 30$$

$$w = \sqrt{\frac{k}{m}} = \sqrt{\frac{900}{1}} = 30rad/s$$

Verificando que a divisão do módulo da parte real pelo módulo do número complexo vale o coeficiente de amortecimento:

$$\zeta = 0.1$$

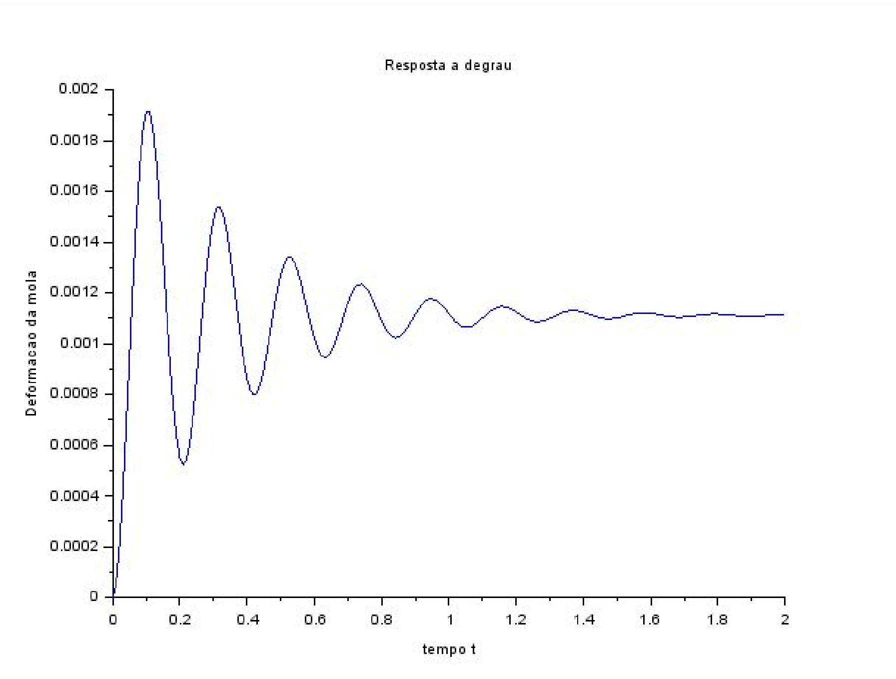
$$\frac{|Re|}{|\lambda|} = \frac{3}{30} = 0.1$$

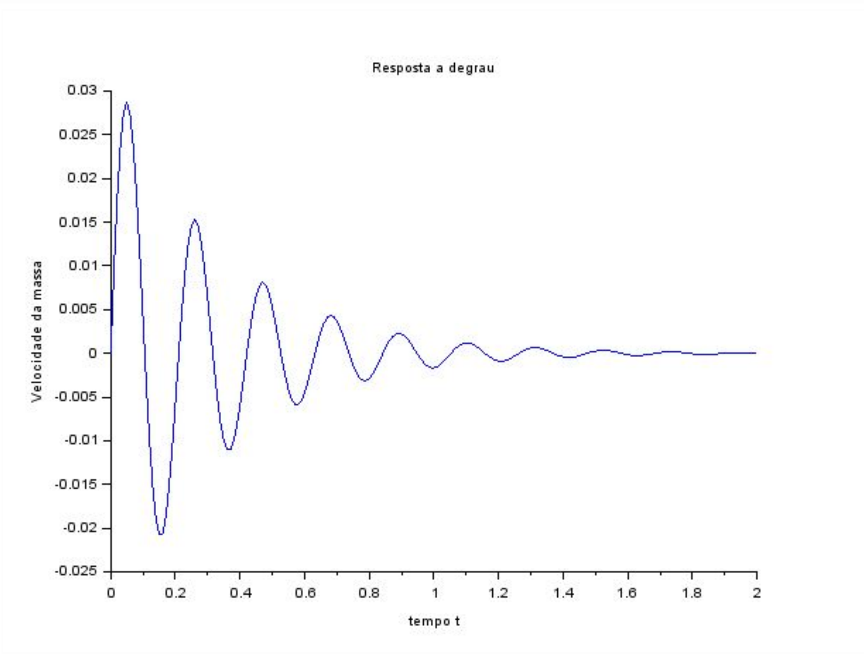
Verificando que a frequência de oscilação vale o módulo da parte imaginária:

$$w * \sqrt{1-\zeta^2} = 30 * \sqrt{1-0.1^2} = 29,85 rad/s$$

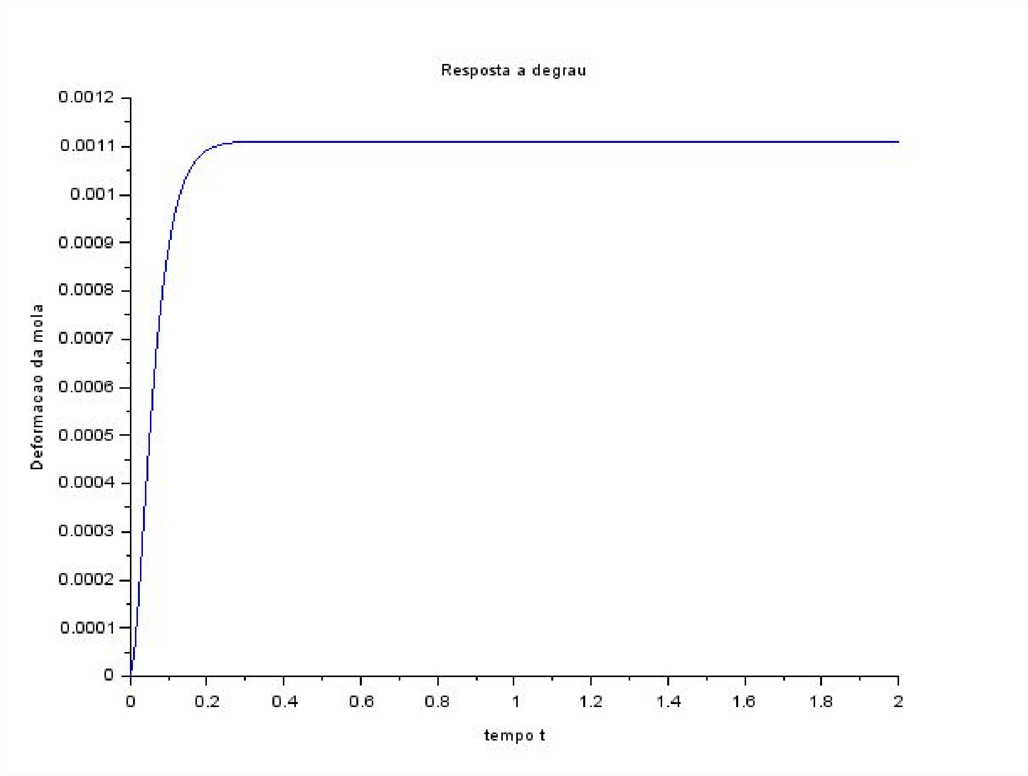
$$|Im| = 9 * \sqrt{11} = 29,85$$

Simulando o sistema para diferentes valores de coeficiente de amortecimento:
Para ζ =0.1:

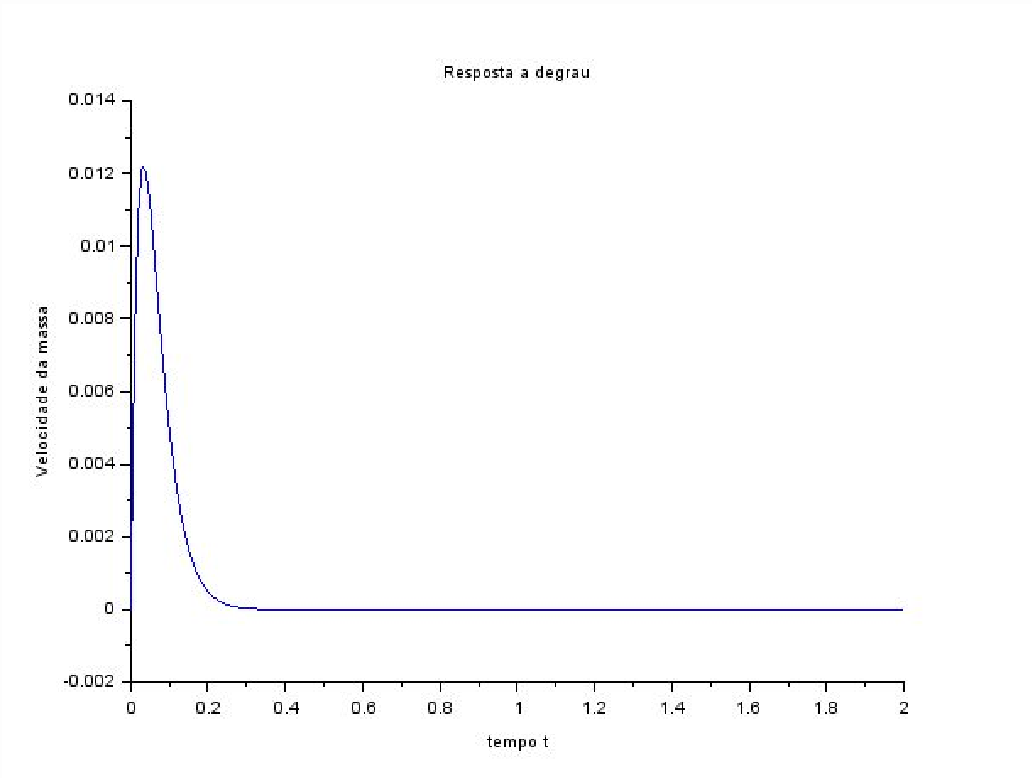




Para $\zeta = 1$:

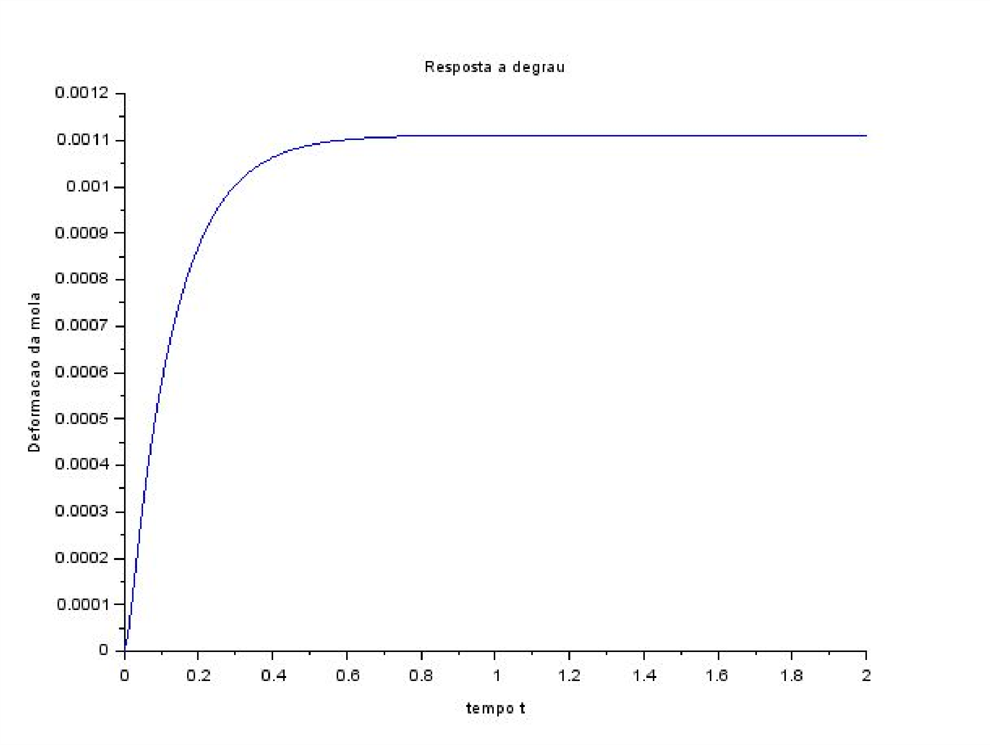


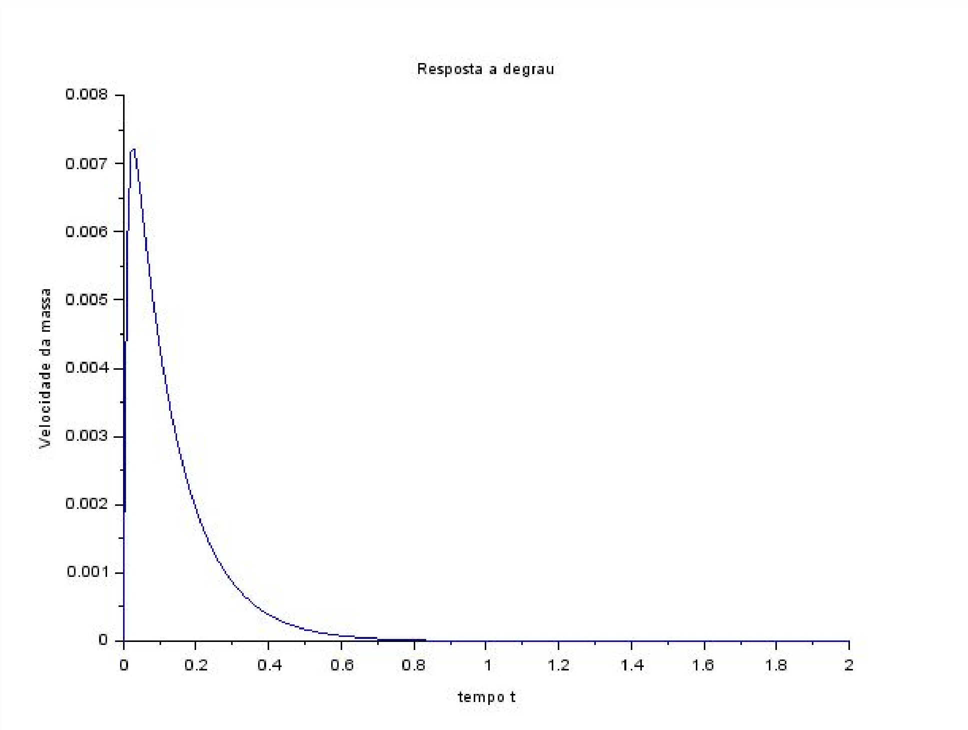
Resposta a degrau
0.0012
0.0011
0.001
0.0009
0.0008
0.0007
0.0006
0.0005
0.0004
0.0003
0.0002
0.0001
0
Deformacao da mola
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
tempo t



Resposta a degrau
0.014
0.012
0.01
0.008
0.006
0.004
0.002
0
-0.002
Velocidade da massa
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2
tempo t

Para $\zeta = 2$:

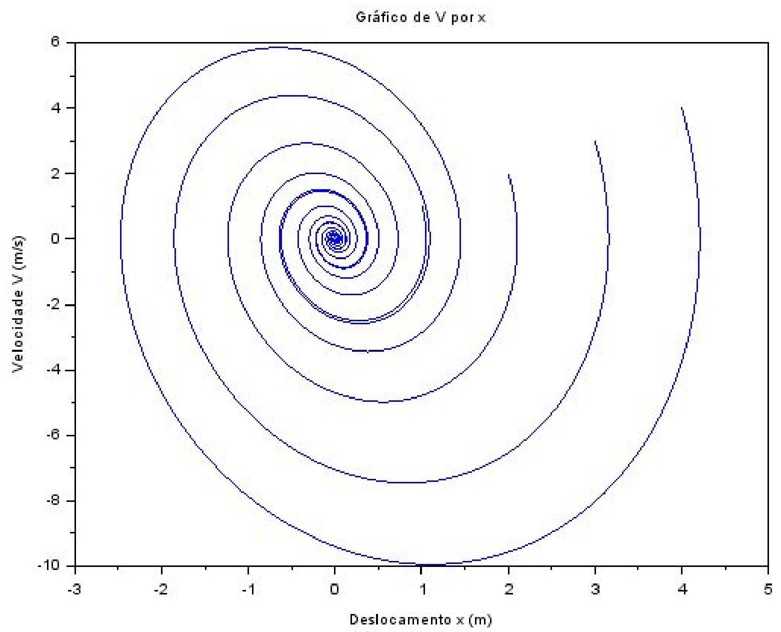

Resposta a degrau
Deformacao da mola
tempo t
0.0012
0.0011
0.001
0.0009
0.0008
0.0007
0.0006
0.0005
0.0004
0.0003
0.0002
0.0001
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2


Resposta a degrau
Velocidade da massa
tempo t
0.008
0.007
0.006
0.005
0.004
0.003
0.002
0.001
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2

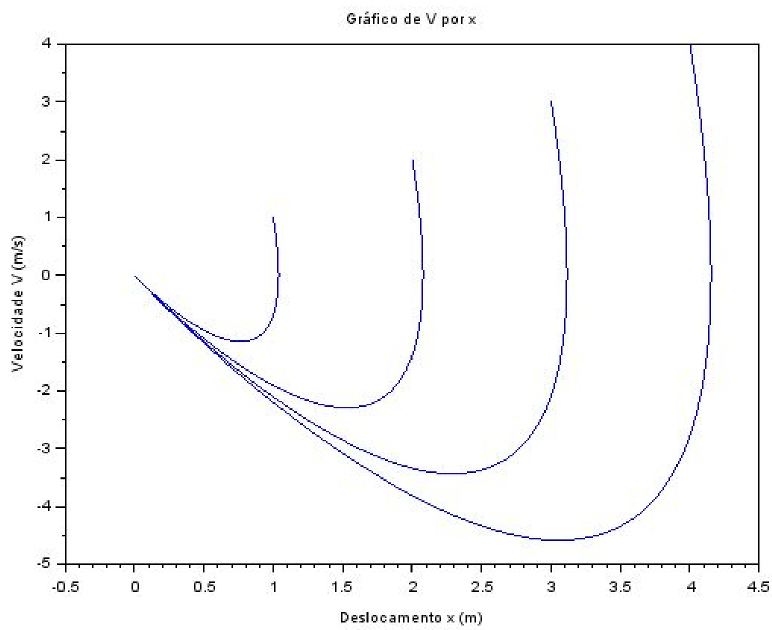
Exercício 2:

Simulando o sistema para diferentes condições iniciais nas situações de pólos complexos, pólos reais e iguais, e pólos reais e distintos, para obter os gráficos de v por x seguintes:

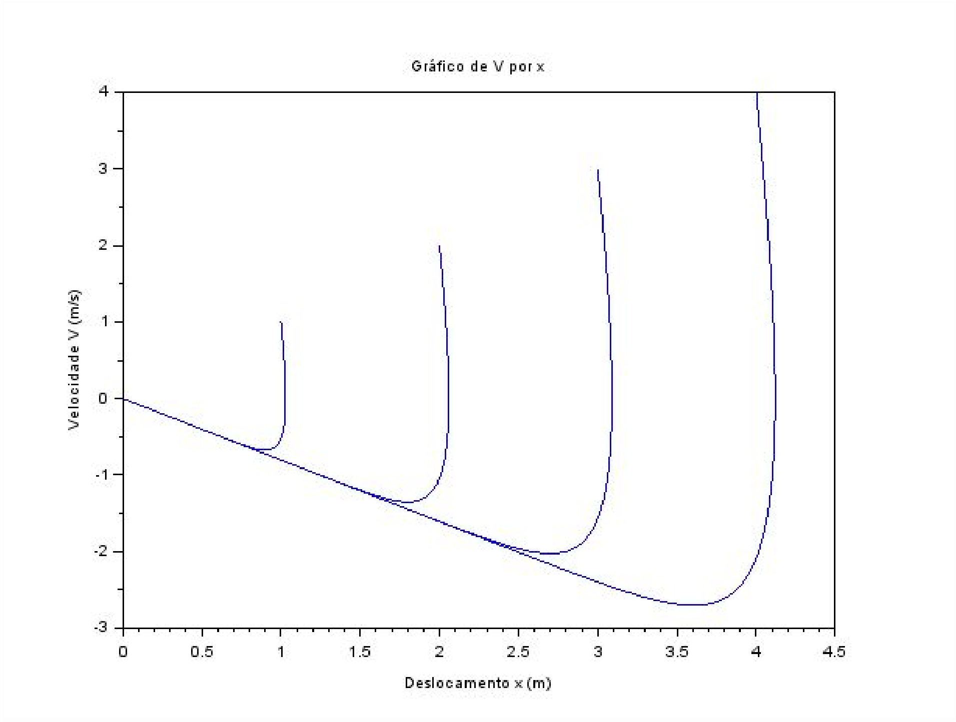
Pólos complexos (ζ =0.1):



Pólos reais e iguais (ζ =1):



Pólos reais e distintos (ζ =2):



Códigos

Exercício 1

```
clear all
// Definindo os parametros do sistema:
m=1;Cc=0.1;k=900;
c=2*sqrt(k*m)*Cc;
// Matrizes do sistema:
A=[0 1; -k/m -c/m];
B=[0;1/m];
C=[0 0];
D=[0];
// Montando o sistema:
suspensao=syslin('c',A,B,C,D);
// Definindo o vetor tempo:
t=0:0.01:2;
// Definindo a entrada:
u=ones(t);
// No espaco de estados temos 2 variaveis de estado:
x0e=[0;0]; // neste caso, x1(0)=0 e x2(0)=0
// Alem de calcular a saida y, a função csim também permite obter o estado x:
[y,x]=csim(u,t,suspensao,x0e);
// Abrindo uma nova janela de graficos:
xset('window',1)
// Mostrando o resultado da simulacao:
plot2d(t,x(1,:),2)
xtitle('Resposta a degrau','tempo t','Deformacao da mola')
// Podemos plotar o grafico do estado x2, por exemplo:
// Abrindo uma nova janela de graficos:
xset('window',2)
// Mostrando o resultado da simulacao:
plot2d(t,x(2,:),2)
xtitle('Resposta a degrau','tempo t','Velocidade da massa')
```

Exercício 2

```
clear all;
winsid(xdel());
// Definindo os parametros do sistema:
m=1;k=9;
//Escolhendo do caso:
caso=3;
//Raízes complexas:
if caso==1 then
 c=1;
//Raízes reais e iguais:
elseif caso==2 then
 c=6;
//Raízes reais e distintas:
else
 c=12;
end
// Definindo o vetor tempo:
t=0:0.01:2;
//Condições iniciais:
x0=1:1:4;
xp0=1:1:4;
//Definindo o vetor de estados:
funcprot(0)
 function dy=edo(t, y)
 dy(1)=y(2);
 dy(2)=-y(1)*k/m-y(2)*c/m;
 endfunction
//Solução da ODE:
for i=1:length(x0)
 solve=ode([x0(i);xp0(i)],0,t,edo);
 for j=1:length(t)
 x(i,j)=solve(1,j);
 xp(i,j)=solve(2,j);
 end
end
//Plotando os resultados:
scf(1);
for i=1:length(x0)
 plot(x(i,:),xp(i,:));
end
xtitle("Gráfico de V por x");
xlabel("Deslocamento x (m)");
ylabel("Velocidade V (m/s)");
```