Escola Politécnica da Universidade de São Paulo

Lista B de Modelagem de Sistemas Dinâmicos

Prof. Dr. Agenor de Toledo Fleury

Prof. Dr. Decio Crisol Donha

Gabriela Gomes Valejo Sanches NUSP: 10772592

Sumário

Exemplo 1

Executou-se o código apresentado no exemplo 1:

```
clear
function [ydot]=funcao(y)
ydot=(1-y)/2;
endfunction

//Conjunto de comandos para solucao numerica de equacao diferencial dada pela funcao.sci
//Apagando dados anteriores:

//Carregando a equacao diferencial:
//Carregue a funcao usando o comando Load do Scilab
//Instante inicial:
t(1)=0;
//Instante final:
tf=10;
//Condicao inicial:
y(1)=0;
//Valor inicial da solucao exata:
ye(1)=0;
//Passo de integracao
h=0.5;
//Calculo de numero de passos:
n=round(tf/h);
//Integracao numerica usando o metodo de Euler:
  //Comando for:
  for i=1:n
    //Vetor de tempo
  t(i+1)=t(i)+h;
    //Solucao numerica
  y(i+1)=y(i)+h*funcao(y(i));
    //Solucao exata:
  ye(i+1)=1-%e^(-t(i+1)/2);
    //Termino do comando for
  end
//Plotando a solucao numerica y versus vetor de tempo t e solucao exata ye versus vetor de tempo t:
plot2d([t,t],[y,ye],[-1 -2]);
//Colocando uma legenda na parte inferiordireito da figura (parametro 4):
legends(["Solucao numerica","Solucao exata"],[-1,-2],4)
//Colocando um titulo na figura e nomeando os eixos:
xtitle("Comparacao entre solucao numerica e solucao exata","Tempo t","Solucao")
//Abrindo uma nova janela de graficos:
set("current_figure",1);
//Desenhando outro grafico com linhas diferentes:
plot2d([t,t],[y,ye],[1 2]);
//Usando a variavel do tipo 'lista':
T=list("Comparacao entre solucao numerica e solucao exata","Tempo t","SOlucao","Solucao numerica","Solucao
exata");
//Colocando uma legenda na parte superior esquerda da figura (parametro 2):
legends([T(4),T(5)],[1,2],2);
//Colocando um titulo na figura e nomeando os eixos:
xtitle(T(1),T(2),T(3));
```

E obteve-se os seguintes resultados

:

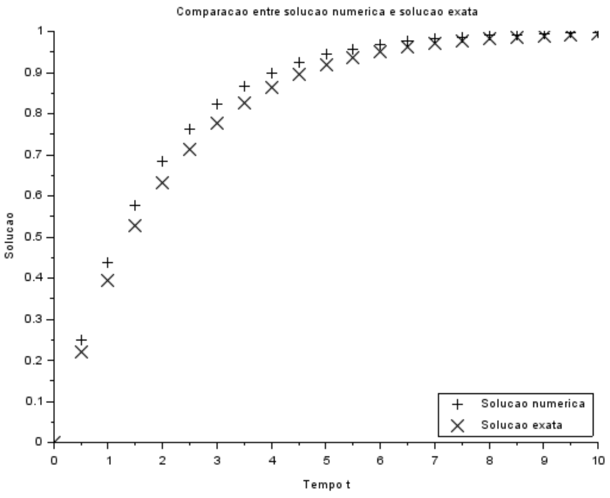

Figura 1: Gráfico com asteriscos do exemplo 1

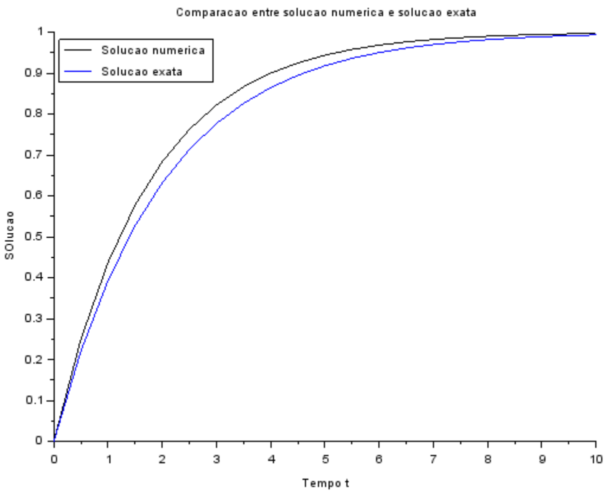

Figura 2: Gráfico de linha do exemplo 1

Exemplo 2

O código executado neste exemplo foi:

```
// Conjunto de comandos para solucao numerica de equacao diferencial [1-y(i)]/2
// Apagando dados anteriores:
clear
// Instante inicial:
t(1)=0;
// Instante final:
tf=10;
// Condicao inicial:
y(1)=0;
// Valor inicial da solucao exata:
```

```
ye(1)=0;
// Passo de integracao (experimente alterar o passo):
h=0.5;
// Calculo de numero de passos):
n=round((tf-t(1))/h);
// Integracao numerica usando o metodo de Runge Kutta:
  // Comando for:
  for i=1:n
// Vetor de tempo:
  t(i+1)=t(i)+h;
// Solucao numerica:
  k1=h*(1-(y(i)))/2;
  k2=h*(1-(y(i)+k1/2))/2;
  k3=h*(1-(y(i)+k2/2))/2;
  k4=h*(1-(y(i)+k3))/2;
  y(i+1)=y(i)+((k1+2*k2+2*k3+k4)/6);
  // Solucao exata:
  ye(i+1)=1-%e^(-t(i+1)/2);
  // Termino do comando for:
  end
// Plotando solucao numerica y versus vetor de tempo t e solucao exata ye versus vetor de tempo t:
plot2d([t,t],[y,ye],[-1 -2]);
// Colocando uma legenda na parte inferior direito da figura (parametro 4):
legends(["Solucao numerica","Solucao exata"],[-1,-2],4)
// Colocando um titulo na figura e nomeando os eixos:
xtitle("Comparacao entre solucao numerica e solucao exata","Tempo t","Solucao")
// Abrindo uma nova janela de graficos:
set("current_figure", 1);
// Aumentando a espessura das linhas:
  xset("thickness",2)
// Aumentando o tamanho da fonte:
  xset("fontsize",4)
// Desenhando outro grafico com linhas diferentes:
plot2d([t,t],[y,ye],[1 2]);
// Usando a variavel do tipo 'lista':
T=list("Comparacao entre solucao numerica e solucao exata","Tempo t","Solucao","Solucao numerica","Solucao
exata");
// Diminuindo a espessura das linhas:
xset("thickness",1)
// Colocando uma legenda na parte superior esquerda da figura (parametro 2):
legends([T(4),T(5)],[1,2],2);
// Colocando um titulo na figura e nomeando os eixos:
xtitle(T(1),T(2),T(3));
// Colocando uma grade no grafico:
xgrid(1)
```

Ao executá-lo, obteve-se o seguinte resultado:

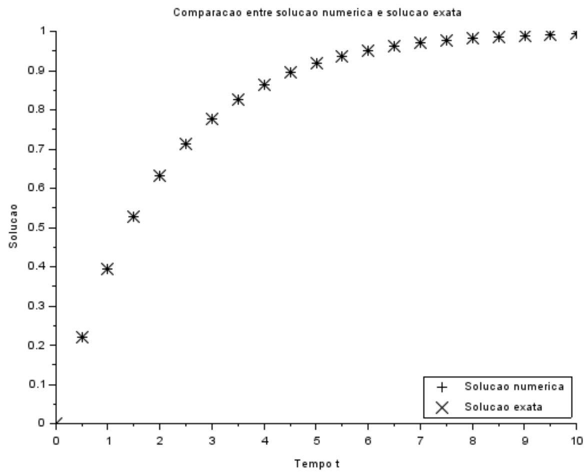

Figura 3: Gráfico do Exemplo2

Percebe-se uma precisão maior do método numérico, tendo em vista que está mais próximo da solução exata.

Exercício 1

Neste exercício o objetivo era modelar a equação diferencial que regia o sistema apresentado de um reservatório de água.

$$\dot{h} = \left(-\sqrt{\frac{\rho g h}{R}} + Q_e \right) \frac{1}{S}$$

Método de Euler

Primeiro o sistema foi resolvido através do Método de Euler, chegando ao seguinte resultado:

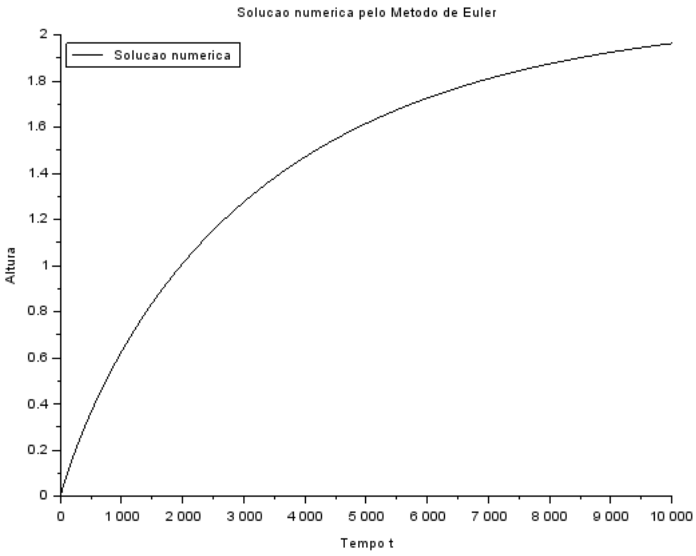

Figura 4: Resultado pelo Método de Euler do sistema com um reservatório.

O código utilizado é apresentado a seguir:

```
clear
function [hdot]=funcao(h)
rho=1000;
g=10;
R=2*(10^8);
Qe=0.010247;
S=10

hdot=((-sqrt(rho*g*h/R)+Qe)/S);
endfunction

//Instante inicial:
t(1)=0;
//Instante final:
tf=10000;
//Condicao inicial:
h(1)=0;
//Valor inicial da solucao exata:
he(1)=0;
//Passo de integracao
p=0.5;
//Calculo de numero de passos:
n=round(tf/p);
//Integracao numerica usando o metodo de Euler:
  //Comando for:
  for i=1:n
    //Vetor de tempo
  t(i+1)=t(i)+p;
    //Solucao numerica
  h(i+1)=h(i)+p*funcao(h(i));
    //Termino do comando for
  end
//Plotando a solucao numerica y versus vetor de tempo t e solucao exata ye versus vetor de tempo t:
plot2d([t],[h]);
//Colocando uma legenda na parte inferiordireito da figura (parametro 4):
legends(["Solucao numerica"],[-1,-2],4)
//Colocando um titulo na figura e nomeando os eixos:
xtitle("Solucao numerica pelo Metodo de Euler","Tempo t","Solucao")
//Abrindo uma nova janela de graficos:
```

```
set("current_figure",1);
//Desenhando outro grafico com linhas diferentes:
plot2d([t],[h]);
//Usando a variavel do tipo 'lista':
T=list("Solucao numerica pelo Metodo de Euler","Tempo t","Solucao","Solucao numerica");
//Colocando uma legenda na parte superior esquerda da figura (parametro 2):
legends([T(4)],[1,2],2);
//Colocando um titulo na figura e nomeando os eixos:
xtitle(T(1),T(2),T(3));
```

Método de Runge-Kutta

Em seguida o mesmo sistema foi resolvido pelo Método de Runge-Kutta, apresentando o seguinte resultado:

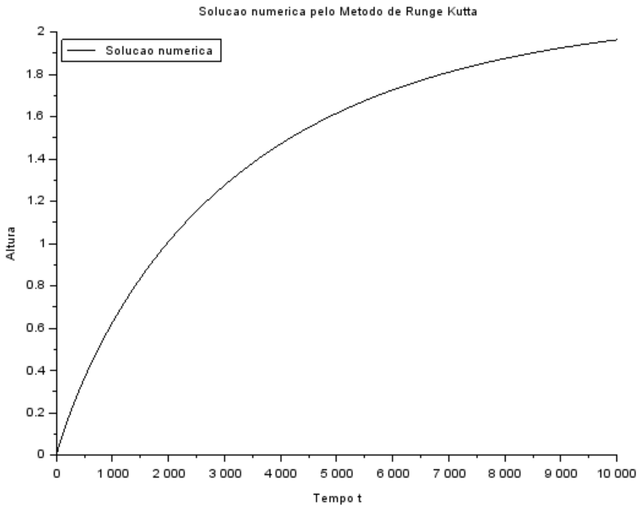

Figura 5 Resultado pelo Método de Runge-Kutta do sistema com um reservatório.

O código utilizado é apresentado a seguir:

```
// Conjunto de comandos para solucao numerica de equacao diferencial [1-y(i)]/2
// Apagando dados anteriores:
clear
// Instante inicial:
t(1)=0;
// Instante final:
tf=10000;
// Condicao inicial:
y(1)=0;
// Valor inicial da solucao exata:
ye(1)=0;
// Passo de integracao (experimente alterar o passo):
h=0.5;
// Calculo de numero de passos):
n=round((tf-t(1))/h);
//Determinando as variaveis do problema
rho=1000;
g=10;
R=2*(10^8);
Qe=0.010247;
S=10;
// Integracao numerica usando o metodo de Runge Kutta:
```

```
   // Comando for:
   for i=1:n
// Vetor de tempo:
   t(i+1)=t(i)+h;
// Solucao numerica:
   k1=h*((-sqrt(rho*g*y(i)/R)+Qe)/S);
   k2=h*((-sqrt(rho*g*(y(i)+k1/2)/R)+Qe)/S);
   k3=h*((-sqrt(rho*g*(y(i)+k2/2)/R)+Qe)/S);
   k4=h*((-sqrt(rho*g*(y(i)+k3)/R)+Qe)/S);
   y(i+1)=y(i)+((k1+2*k2+2*k3+k4)/6);
   // Termino do comando for:
   end
// Plotando solucao numerica y versus vetor de tempo t e solucao exata ye versus vetor de tempo t:
plot2d([t],[y],[-1 -2]);
// Colocando uma legenda na parte inferior direito da figura (parametro 4):
legends(["Solucao numerica"],[-1,-2],4)
// Colocando um titulo na figura e nomeando os eixos:
xtitle("Solucao numerica pelo Metodo de Runge Kutta","Tempo t","Solucao")
// Abrindo uma nova janela de graficos:
set("current_figure", 1);
// Aumentando a espessura das linhas:
   //xset("thickness",2)
// Aumentando o tamanho da fonte:
   //xset("fontsize",4)
// Desenhando outro grafico com linhas diferentes:
plot2d([t],[y],[1 2]);
// Usando a variavel do tipo 'lista':
T=list("Solucao numerica pelo Metodo de Runge Kutta","Tempo t","Solucao","Solucao numerica");
// Diminuindo a espessura das linhas:
   //xset("thickness",1)
// Colocando uma legenda na parte superior esquerda da figura (parametro 2):
legends([T(4)],[1,2],2);
// Colocando um titulo na figura e nomeando os eixos:
xtitle(T(1),T(2),T(3));
// Colocando uma grade no grafico:
//xgrid(1)
```

Exercício 2

Neste exercício o objetivo era modelar a equação diferencial que regia o sistema apresentado de dois reservatórios de água acoplados.

$$\begin{cases} \dot{h_1} = \left(-\sqrt{\dfrac{\rho g(h_1 - h_2)}{R_a}} + Q_e\right)\dfrac{1}{S_1} \\ \dot{h_2} = \left(\sqrt{\dfrac{\rho g(h_1 - h_2)}{R_a}} - \sqrt{\dfrac{\rho g h_2}{R_s}}\right)\dfrac{1}{S_2} \end{cases}$$

Considerou-se o Segundo reservatório menor do que o primeiro.

Método de Euler

Primeiro o sistema foi resolvido através do Método de Euler, chegando ao seguinte resultado:

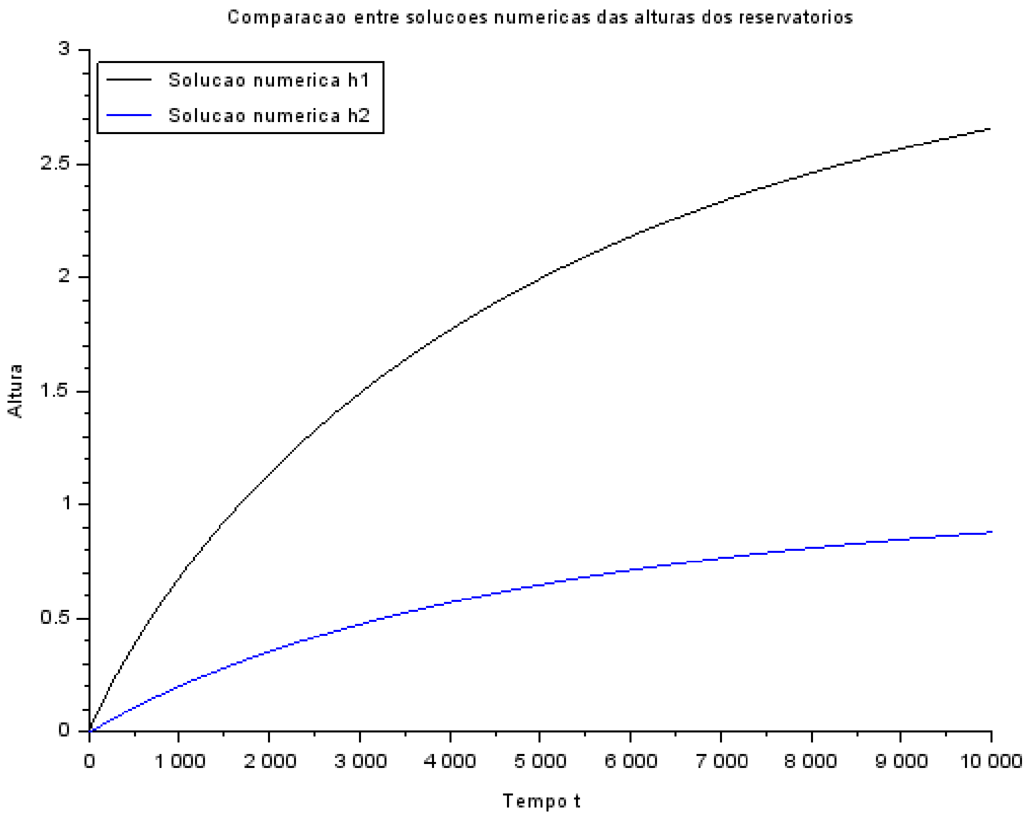

Figura 6: Resultado pelo Método de Euler do sistema com dois reservatórios acoplados.

O código utilizado é apresentado a seguir:

```
clear

//Considerando o primeiro reservatorio como o mesmo do exercicio 1
rho=1000
g=10
Ra=2*(10^8)
Qe=0.010247
S1=10
Rs=10^8
//Considerando o raio do segundo reservatorio como metade do primeiro
S2=2.5

function [y1dot]=funcao(y1, y2)
y1dot=((-sqrt(rho*g*(y1-y2)/Ra)+Qe)/S1);
endfunction

function [y2dot]=func(y1, y2)
y2dot=(sqrt(rho*g*(y1-y2)/Ra)-sqrt(rho*g*y2/Rs))/S2;
endfunction

//Instante inicial:
t(1)=0;
//Instante final:
tf=10000;
//Condicao inicial equacao 1:
y1(1)=0;
//COndicao inicial equacao 2:
y2(1)=0;
```

```
//Passo de integracao
p=0.5;
//Calculo de numero de passos:
n=round(tf/p);
//Integracao numerica da equacao 1 usando o metodo de Euler:
  //Comando for:
  for i=1:n
    //Vetor de tempo
  t(i+1)=t(i)+p;
    //Solucao numerica
  y1(i+1)=y1(i)+p*funcao(y1(i),y2(i));
  y2(i+1)=y2(i)+p*func(y1(i),y2(i));
    //Termino do comando for
  end


//Plotando a solucao numerica y versus vetor de tempo t e solucao exata ye versus vetor de tempo t:
plot2d([t,t],[y1,y2],[-1 -2]);
//Colocando uma legenda na parte inferiordireito da figura (parametro 4):
legends(["Solucao numerica de h1","Solucao numerica de h2"],[-1,-2],4)
//Colocando um titulo na figura e nomeando os eixos:
xtitle("Comparacao entre solucoes numericas das alturas dos reservatorios","Tempo t","Solucao")
//Abrindo uma nova janela de graficos:
set("current_figure",1);
//Desenhando outro grafico com linhas diferentes:
plot2d([t,t],[y1,y2],[1 2]);
//Usando a variavel do tipo 'lista':
T=list("Comparacao entre solucoes numericas das alturas dos reservatorios","Tempo t","Solucao","Solucao numerica h1","Solucao numerica h2");
//Colocando uma legenda na parte superior esquerda da figura (parametro 2):
legends([T(4),T(5)],[1,2],2);
//Colocando um titulo na figura e nomeando os eixos:
xtitle(T(1),T(2),T(3));
```

Método de Runge-Kutta

Então, o mesmo sistema foi resolvido pelo método de Runge-Kutta, chegando ao seguinte resultado:

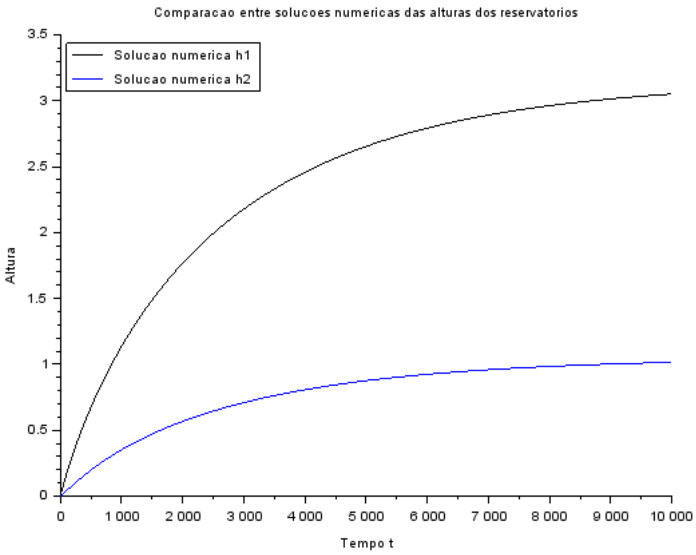

Figura 7: Resultado pelo Método de Runge-Kutta do sistema com dois reservatórios acoplados.

O código utilizado é apresentado a seguir:

```
clear

//Considerando o primeiro reservatorio como o mesmo do exercicio 1
rho=1000
g=10
Ra=2*(10^8)
Qe=0.010247
S1=10
Rs=10^8
//Considerando o raio do segundo reservatorio como metade do primeiro
S2=2.5

function [y1dot]=funcao(y1, y2)
y1dot=((-sqrt(rho*g*(y1-y2)/Ra)+Qe)/S1);
endfunction

function [y2dot]=func(y1, y2)
y2dot=(sqrt(rho*g*(y1-y2)/Ra)-sqrt(rho*g*y2/Rs))/S2;
endfunction

//Instante inicial:
t(1)=0;
//Instante final:
tf=10000;
//Condicao inicial equacao 1:
y1(1)=0;
//COndicao inicial equacao 2:
y2(1)=0;
//Passo de integracao
p=0.5;
//Calculo de numero de passos:
n=round(tf/p);
//Integracao numerica da equacao 1 usando o metodo de Euler:
  //Comando for:
//Comando for:
  for i=1:n
    //Vetor de tempo
  t(i+1)=t(i)+p;
    //Solucao numerica
```

```
   k11=funcao(y1(i),y2(i));
   k12=func(y1(i),y2(i));

   k21=funcao(y1(i)+k11/2,y2(i)+k12/2);
   k22=func(y1(i)+k11/2,y2(i)+k12/2);

   k31=funcao(y1(i)+k21/2,y2(i)+k22/2);
   k32=func(y1(i)+k21/2,y2(i)+k22/2);

   k41=funcao(y1(i)+k31,y2(i)+k32);
   k42=func(y1(i)+k31,y2(i)+k32);

   y1(i+1)=y1(i)+((k11+2*k21+2*k31+k41)/6);
   y2(i+1)=y2(i)+((k12+2*k22+2*k32+k42)/6);
     //Termino do comando for
   end

//Plotando a solucao numerica y versus vetor de tempo t e solucao exata ye versus vetor de tempo t:
plot2d([t,t],[y1,y2],[-1 -2]);
//Colocando uma legenda na parte inferiordireito da figura (parametro 4):
legends(["Solucao numerica de h1","Solucao numerica de h2"],[-1,-2],4)
//Colocando um titulo na figura e nomeando os eixos:
xtitle("Comparacao entre solucoes numericas das alturas dos reservatorios","Tempo t","Solucao")
//Abrindo uma nova janela de graficos:
set("current_figure",1);
//Desenhando outro grafico com linhas diferentes:
plot2d([t,t],[y1,y2],[1 2]);
//Usando a variavel do tipo 'lista':
T=list("Comparacao entre solucoes numericas das alturas dos reservatorios","Tempo t","Solucao","Solucao numerica h1","Solucao numerica h2");
//Colocando uma legenda na parte superior esquerda da figura (parametro 2):
legends([T(4),T(5)],[1,2],2);
//Colocando um titulo na figura e nomeando os eixos:
xtitle(T(1),T(2),T(3));
```