

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

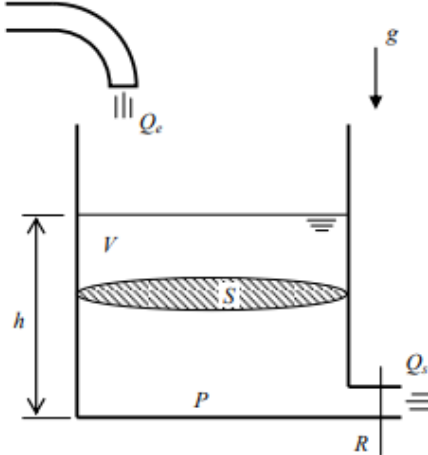
Pedro Pires Sulzer

10705940

27 de agosto de 2020

Exercício:

Implemente um programa no Scilab que resolva numericamente a equação diferencial que modela o sistema abaixo, tanto pelo método de Euler como Runge Kutta.



Reservatório com água

Parâmetros:

$S = 10 \text{ m}^2$ - área da seção transversal (constante)

$R = 2 \times 10^8 \text{ Pa}/(\text{m}^3/\text{s})^2$ - parâmetro que relaciona vazão com queda de pressão (perda de carga)

$\rho = 1000 \text{ kg}/\text{m}^3$ - massa específica da água

$G = 10 \text{ m}/\text{s}^2$ - aceleração da gravidade na superfície da terra

Variáveis:

$Q_e = 0,010247 \text{ m}^3/\text{s}$ - vazão de entrada

h : nível do reservatório [m]

V : volume de água no reservatório [m^3]

P : pressão relativa à atmosférica, no fundo do reservatório [Pa]

Q_s : vazão de saída [m^3/s]

Admite-se que a água seja incompressível.

Pela equação da continuidade:

$$\frac{dV}{dt} = Q_e - Q_s$$

Vamos admitir que a perda de carga na saída é modelada pela expressão:

$$P = RQ_s^2 \Rightarrow Q_s = \sqrt{\frac{P}{R}}$$

Por outro lado, a pressão no fundo do reservatório é:

$$P = \rho gh$$

Considere uma entrada Q_e constante.

Volume de água no reservatório:

$$V = Sh \Rightarrow \dot{V} = S\dot{h}$$

Substituindo:

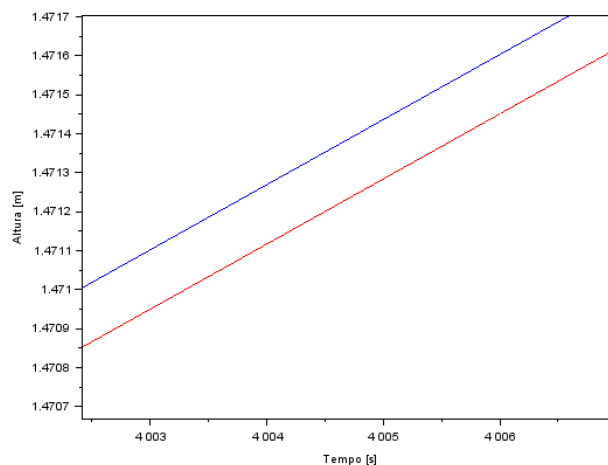
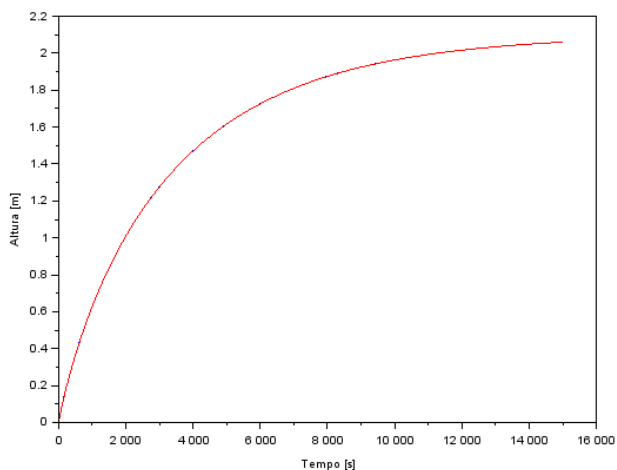
$$S\dot{h} = Q_e - \sqrt{\frac{\rho gh}{R}}$$

Resultando na seguinte equação diferencial ordinária não linear (modelo de 1 reservatório):

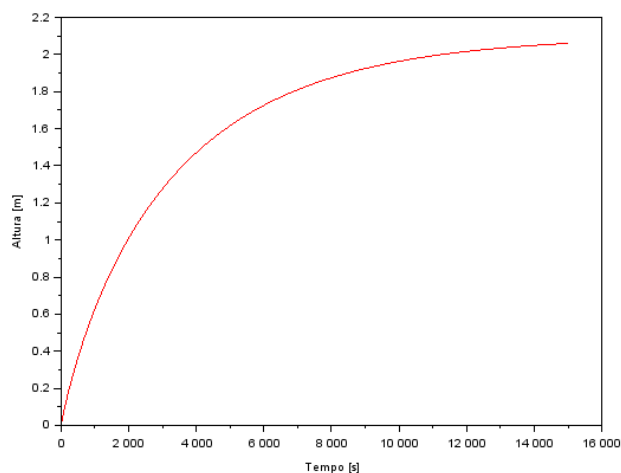
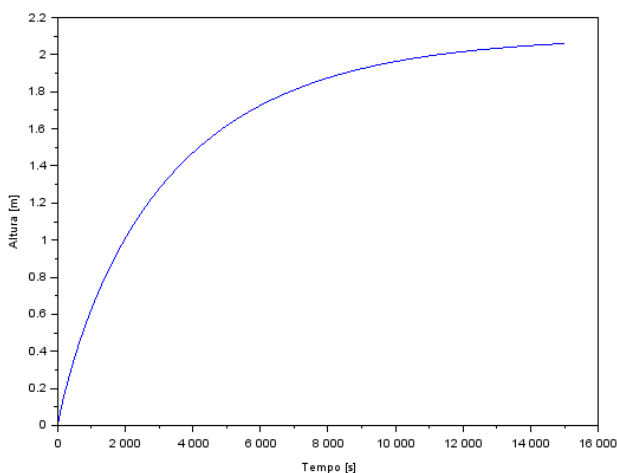
$$\dot{h} = \left(-\sqrt{\frac{\rho gh}{R}} + Q_e \right) \frac{1}{S}$$

Resultados obtidos: Exercício com 1 reservatório

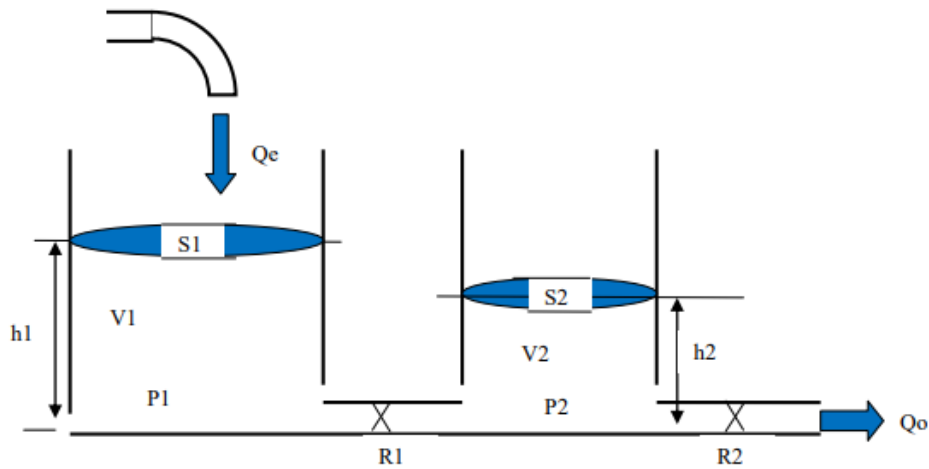
Os dois primeiros graficos representam as soluções pelo metodo de Euler, em azul, e Runge-Kutta, em vermelho, onde fica evidente que sim existe uma diferença entre os dois metodos, mas tal diferença se encontra em um nivel de precisão que é com certeza muito menor do que o necessário para a solução desse tipo de problema, tanto que para o primeiro grafico, em um tempo total de 250 minutos, é imperceptível qualquer diferença. Só é possível visualizar uma diferença quando se aproxima o grafico para uma passagem de tempo segundo a segundo e tal difereça não chega a 2 milímetros.



Os dois graficos a seguir foram colocados apenas para mostrar as soluções plotadas separadamente:



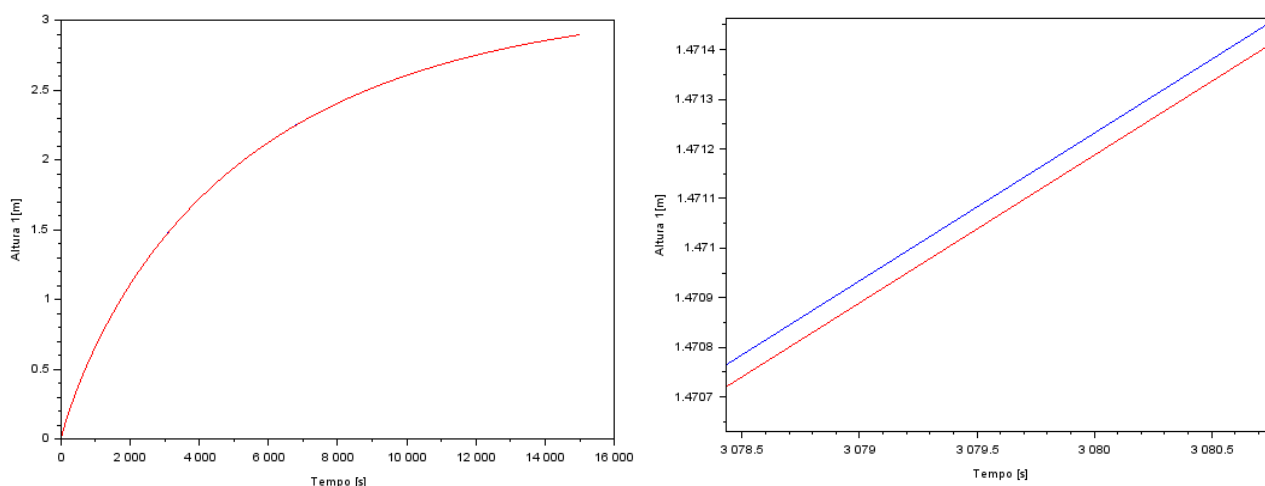
Desenvolva um programa em Scilab que resolva numericamente o sistema de equações diferenciais que modela o sistema com dois reservatórios, usando tanto Euler como Runge Kutta. Dica: raciocine com vetores.



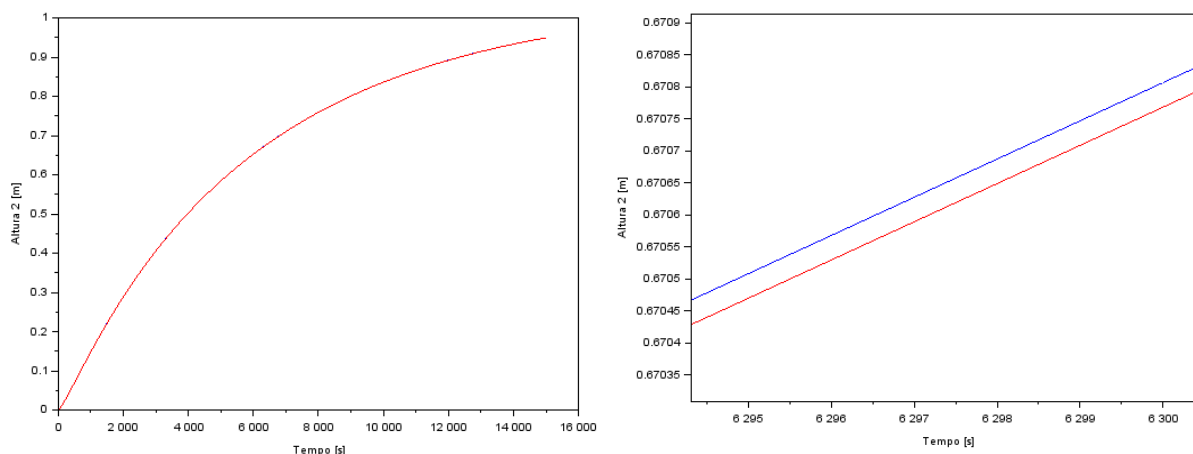
Modelo do sistema de 2 reservatórios (considere a entrada constante e perdas de carga não lineares como no caso do ex. de 1 tanque).

$$\begin{cases} \dot{h}_1 = \left[Q_e - \sqrt{\frac{\rho g}{R_a}} (h_1 - h_2) \right] \frac{1}{S_1} \\ \dot{h}_2 = \left[\sqrt{\frac{\rho g}{R_a}} (h_1 - h_2) - \sqrt{\frac{\rho g}{R_s}} h_2 \right] \frac{1}{S_2} \end{cases}$$

Resultados obtidos: Exercício com 2 reservatórios

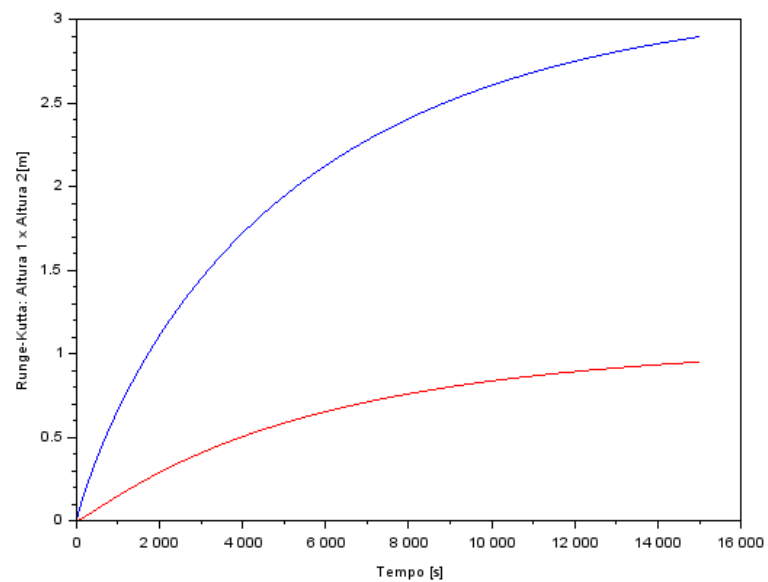
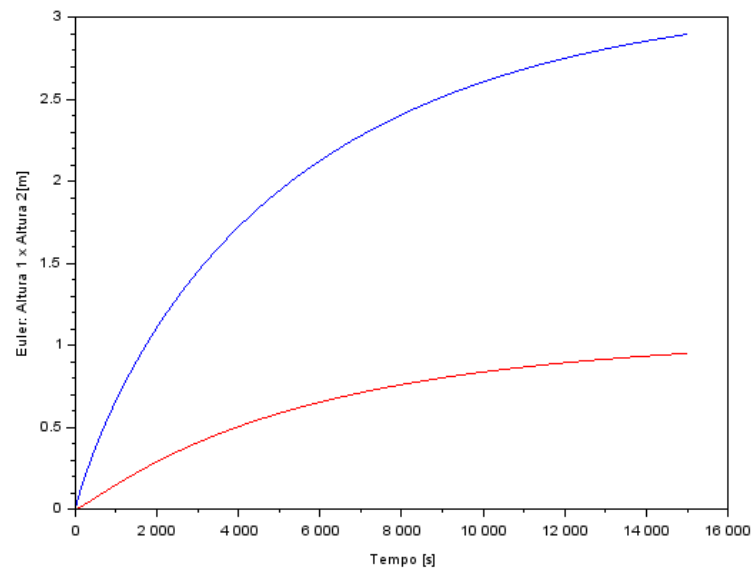


Os dois graficos acima representam as soluções pelo metodo de Euler, em azul, e Runge-Kutta, em vermelho, para a altura do primeiro reservatorio, onde fica evidente que sim existe uma diferença entre os dois metodos, mas tal diferença se encontra em um nivel de precisão que é com certeza muito menor do que o necessário para a solução desse tipo de problema, tanto que para o primeiro grafico, em um tempo total de 250 minutos, é imperceptível qualquer diferença. Só é possível visualizar uma diferença quando se aproxima o grafico para uma passagem de tempo segundo a segundo e tal difereça não chega a 1 milimetro.



Os dois graficos acima representam as soluções pelo metodo de Euler, em azul, e Runge-Kutta, em vermelho, para a altura do segundo reservatorio, onde fica evidente que sim existe uma diferença entre os dois metodos, mas tal diferença se encontra em um nivel de precisão que é com certeza muito menor do que o necessário para a solução desse tipo de problema, tanto que para o primeiro grafico, em um tempo total de 250 minutos, é imperceptível qualquer diferença. Só é possível visualizar uma diferença quando se aproxima o grafico para uma passagem de tempo segundo a segundo e tal difereça não chega a 1 milimetro.

Os graficos a seguir foram colocados apenas para mostrar as soluções plotadas juntas para o metodo de Euler e Runge-Kutta, respectivamente:



Código utilizado:

Exercício 1:

```
1 //definição das constantes e das variáveis dadas no enunciado
2 //área da seção transversal
3 S = .10;
4 //perda de carga
5 R = .200000000;
6 //massa específica da água
7 p = .1000;
8 //gravidade
9 g = .10;
10 //vazão de entrada
11 Qe = .0.010247;
12 //tempo inicial
13 ti = 0;
14 //tempo final
15 tf = .15000;
16 //intervalo
17 n = .15000;
18 //passo
19 h = tf/n;
20
21 //definição da passagem do tempo para ambas as soluções
22 ta = zeros(n+1,1)
23 for i = 2:n+1
24     ta(i,1) = ta(i-1,1) + h
25 end
26
27 //definição da função para integração pelo método de Euler
28 function [Euler]=funcao1(h, p, g, R, Qe, S)
29     Euler = (Qe - sqrt(p*g*h/R))/S;
30 endfunction
31
32 //aplicação da integração pelo método de Euler
33 hEuler = zeros(n+1,1);
34 for i = 2:n+1
35     hEuler(i,1) = hEuler(i-1,1) + h*funcao1(hEuler(i-1,1), p, g, R, Qe, S)
36 end
37
38 //definição da função para integração pelo método de Runge-Kutta
39 function [RungeKutta]=funcao2(h, p, g, R, Qe, S)
40     RungeKutta = (Qe - sqrt(p*g*h/R))/S;
41 endfunction
42
43 //aplicação da integração pelo método de Runge-Kutta
44 hRK = zeros(n+1,1);
45 for i = 2:n+1
46     k1 = funcao2(hRK(i-1,1), p, g, R, Qe, S);
47     k2 = funcao2(hRK(i-1,1) + h*0.5*k1, p, g, R, Qe, S);
48     k3 = funcao2(hRK(i-1,1) + h*0.5*k2, p, g, R, Qe, S);
49     k4 = funcao2(hRK(i-1,1) + h*k3, p, g, R, Qe, S);
50     hRK(i,1) = hRK(i-1,1) + (h/6)*(k1+2*k2+2*k3+k4);
51 end
52
53 //plot das duas funções juntas e separadas para análise da sobreposição e
54 //evidenciação das diferenças entre os métodos
55 scf(1)
56 plot(ta,hEuler);
57 xlabel("Tempo. [s]")
58 ylabel("Altura. [m]")
59 plot(ta,hRK,'r');
60 xlabel("Tempo. [s]")
61 ylabel("Altura. [m]")
62
63 scf(2)
64 plot(ta,hEuler);
65 xlabel("Tempo. [s]")
66 ylabel("Altura. [m]")
67
68 scf(3)
69 plot(ta,hRK,'r');
70 xlabel("Tempo. [s]")
71 ylabel("Altura. [m]")
```

Exercicio 2:

```

1 //definição das constantes e das variáveis dadas no enunciado
2 //área da seção transversal
3 S1 = 10;
4 S2 = 8;
5 //perda de carga
6 R1 = 200000000;
7 R2 = 100000000;
8 //massa específica da água
9 p = 1000;
10 //gravidade
11 g = 10;
12 //vazão de entrada
13 Qe = 0.010247;
14 //tempo inicial
15 ti = 0;
16 //tempo final
17 tf = 6000;
18 //intervalo
19 n = 30000;
20 //passo
21 h = tf/n;
22
23 //definição da passagem do tempo para ambas as soluções
24 ta = zeros(n+1,1)
25 for i = 2:n+1
26     ta(i,1) = ta(i-1,1) + h
27 end
28
29 //definição da função para integração pelo método de Euler
30 function [Euler1]=funcao1(h1, h2, Qe, rho, g, Ra, S1)
31     Euler1 = (Qe-sqrt(rho*g*(h1-h2)/Ra))/S1;
32 endfunction
33
34 function [Euler2]=funcao2(h1, h2, rho, g, Ra, Rs, S2)
35     Euler2 = (sqrt(rho*g*(h1-h2)/Ra)-sqrt(rho*g*h2/Rs))/S2;
36 endfunction
37
38 //aplicação da integração pelo método de Euler
39 hEuler = zeros(n+1,1);
40 for i = 2:n+1
41     hEuler(i,1) = hEuler(i-1,1) + h*funcao1(hEuler(i-1,1), p, g, R, Qe, S)
42 end
43
44 hEuler1 = zeros(n+1,1);
45 hEuler2 = zeros(n+1,1);
46 for i = 2:n+1
47     hEuler1(i,1) = hEuler1(i-1,1) + h*funcao1(hEuler1(i-1,1), hEuler2(i-1,1), Qe, p, g, R1, S1);
48     hEuler2(i,1) = hEuler2(i-1,1) + h*funcao2(hEuler1(i-1,1), hEuler2(i-1,1), p, g, R1, R2, S2)
49 end
50
51 //definição da função para integração pelo método de Runge-Kuta
52 function [RungeKutta]=funcao3(h, p, g, R, Qe, S)
53     RungeKutta = (Qe - sqrt(p*g*h/R))/S;
54 endfunction
55
56 function [RungeKutta1]=funcao4(h1, h2, Qe, p, g, Ra, S1)
57     RungeKutta1 = (Qe-sqrt(p*g*(h1-h2)/Ra))/S1;
58 endfunction
59
60 function [RungeKutta2]=funcao5(h1, h2, p, g, Ra, Rs, S2)
61     RungeKutta2 = (sqrt(p*g*(h1-h2)/Ra)-sqrt(p*g*h2/Rs))/S2;
62 endfunction

```



```

62 //aplicação da integração pelo método de Runge-Kutta
63 hRK1 = zeros(n+1,1);
64 hRK2 = zeros(n+1,1);
65 for i = 2:n+1
66     k1 = funcao3(hRK1(i-1,1), hRK2(i-1,1), Qe, p, g, R1, S1);
67     k2 = funcao3(hRK1(i-1,1) + h*0.5*k1, hRK2(i-1,1) + h*0.5*k1, Qe, p, g, R1, S1);
68     k3 = funcao3(hRK1(i-1,1) + h*0.5*k2, hRK2(i-1,1) + h*0.5*k2, Qe, p, g, R1, S1);
69     k4 = funcao3(hRK1(i-1,1) + h*k3, hRK2(i-1,1) + h*k3, Qe, p, g, R1, S1);
70     hRK1(i,1) = hRK1(i-1,1) + (h/6)*(k1+2*k2+2*k3+k4);
71     k1 = funcao4(hRK1(i-1,1), hRK2(i-1,1), p, g, R1, R2, S2);
72     k2 = funcao4(hRK1(i-1,1) + h*0.5*k1, hRK2(i-1,1) + h*0.5*k1, p, g, R1, R2, S2);
73     k3 = funcao4(hRK1(i-1,1) + h*0.5*k2, hRK2(i-1,1) + h*0.5*k2, p, g, R1, R2, S2);
74     k4 = funcao4(hRK1(i-1,1) + h*k3, hRK2(i-1,1) + h*k3, p, g, R1, R2, S2);
75     hRK2(i,1) = hRK2(i-1,1) + (h/6)*(k1+2*k2+2*k3+k4);
76 end
77
78 //plot das duas funções juntas para análise da sobreposição e
79 //evidenciação das diferenças entre os métodos
80 scf(1)
81 plot(ta,hEuler1);
82 plot(ta,hRK1,'r');
83 xlabel("Tempo [s]")
84 ylabel("Altura-1[m]")
85
86 scf(2)
87 plot(ta,hEuler2);
88 plot(ta,hRK2,'r');
89 xlabel("Tempo [s]")
90 ylabel("Altura-2 [m]")
91
92 scf(3)
93 plot(ta,hEuler1);
94 plot(ta,hEuler2,'r');
95 xlabel("Tempo [s]")
96 ylabel("Euler: -Altura-1 x -Altura-2 [m]")
97
98 scf(4)
99 plot(ta,hRK1);
100 plot(ta,hRK2,'r');
101 xlabel("Tempo [s]")
102 ylabel("Runge-Kutta: -Altura-1 x -Altura-2 [m]")

```