Escola Politécnica da
Universidade de São Paulo

PME3380

Lista B

Professor: Décio Crisol e Agenor Fleury

Aluno: Maurício Chung Leiman - 10772571

São Paulo

2020

SUMÁRIO

1 PRIMEIRO EXERCÍCIO

Inicialmente modelamos o sistema composto apenas por 1 tanque, com determinados valores numéricos e ilustrado na figura abaixo:

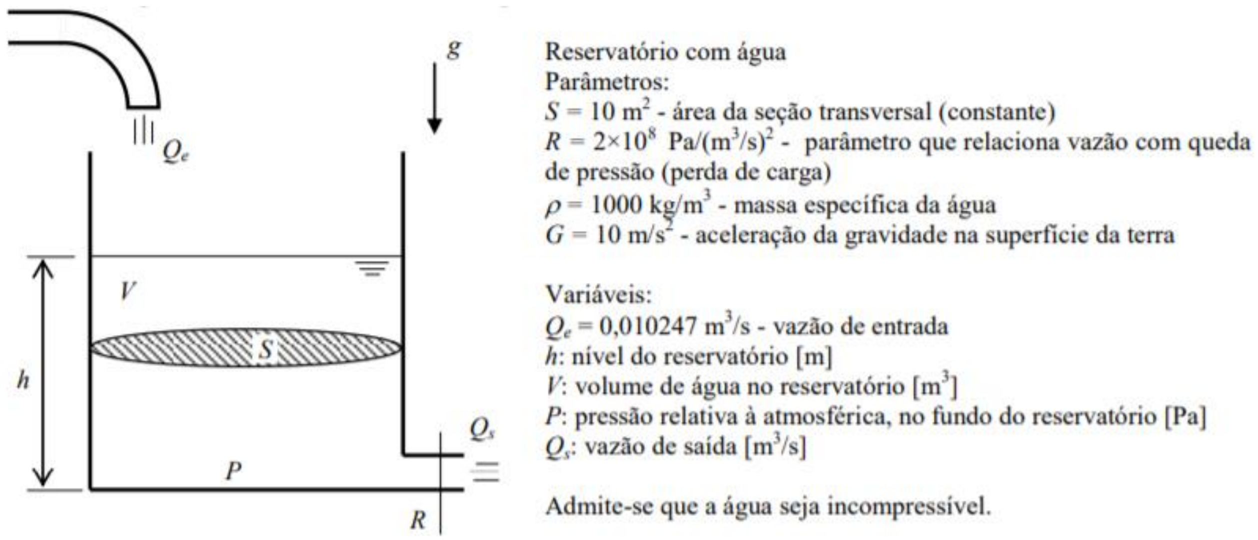
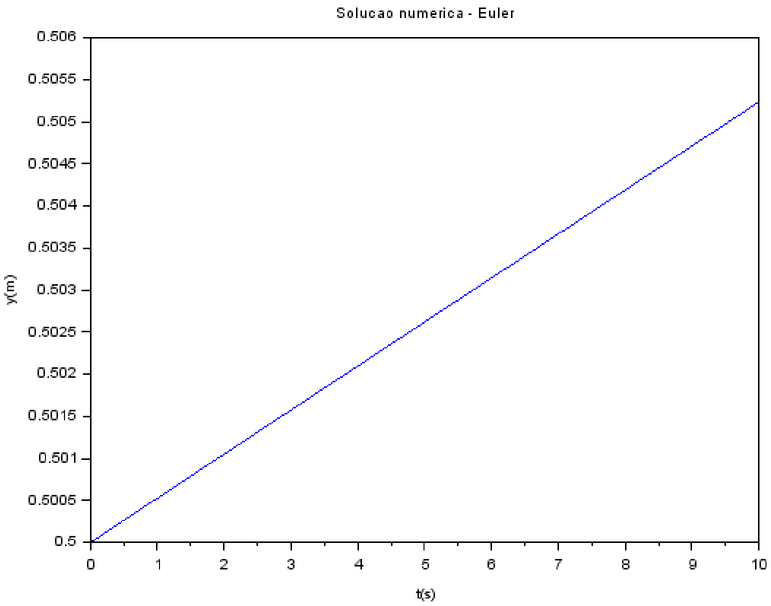


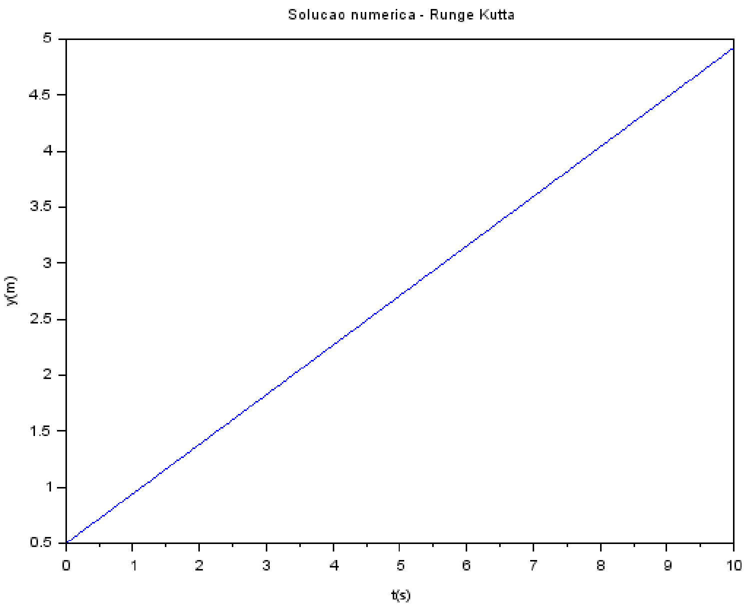
Figura 1: Sistema e valores a serem utilizados.

Elaboramos dois códigos no ambiente computacional *Scilab* para modelar este sistema, um fazendo uso do método de Euler e o outro do método de Runge-Kutta para a integração numérica. A equação diferencial utilizada é a seguinte:

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

A seguir estão os resultados obtidos para a altura do nível do reservatório utilizando cada um dos métodos e considerando altura inicial de 0.5m:



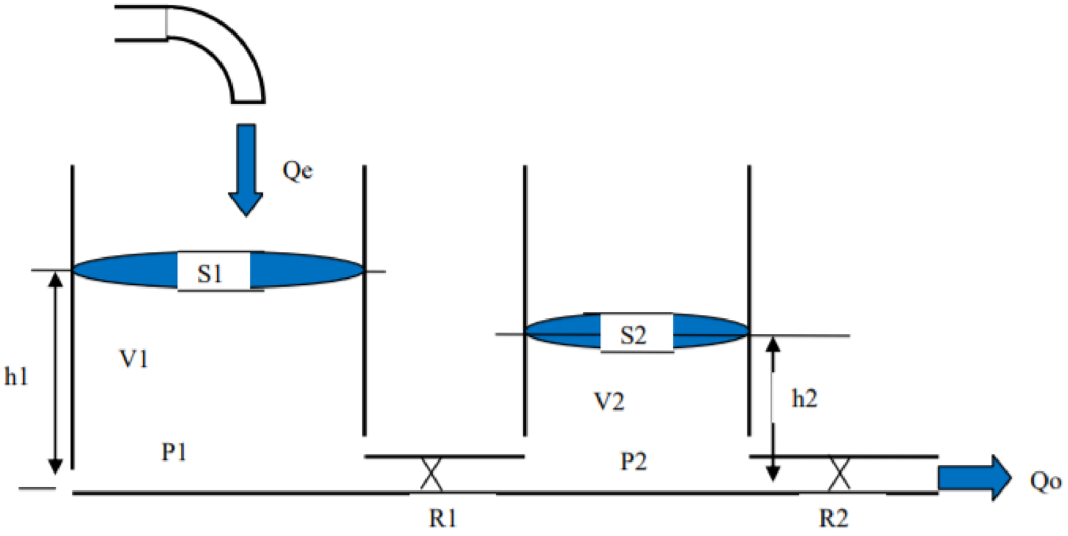


Figuras 2 e 3: Resultados obtidos.

Nota-se que em ambos os casos temos um comportamento semelhante. Porém, a taxa de variação obtida ao longo do tempo é significativamente maior no resultado obtido por Runge-Kutta.

2 SEGUNDO EXERCÍCIO

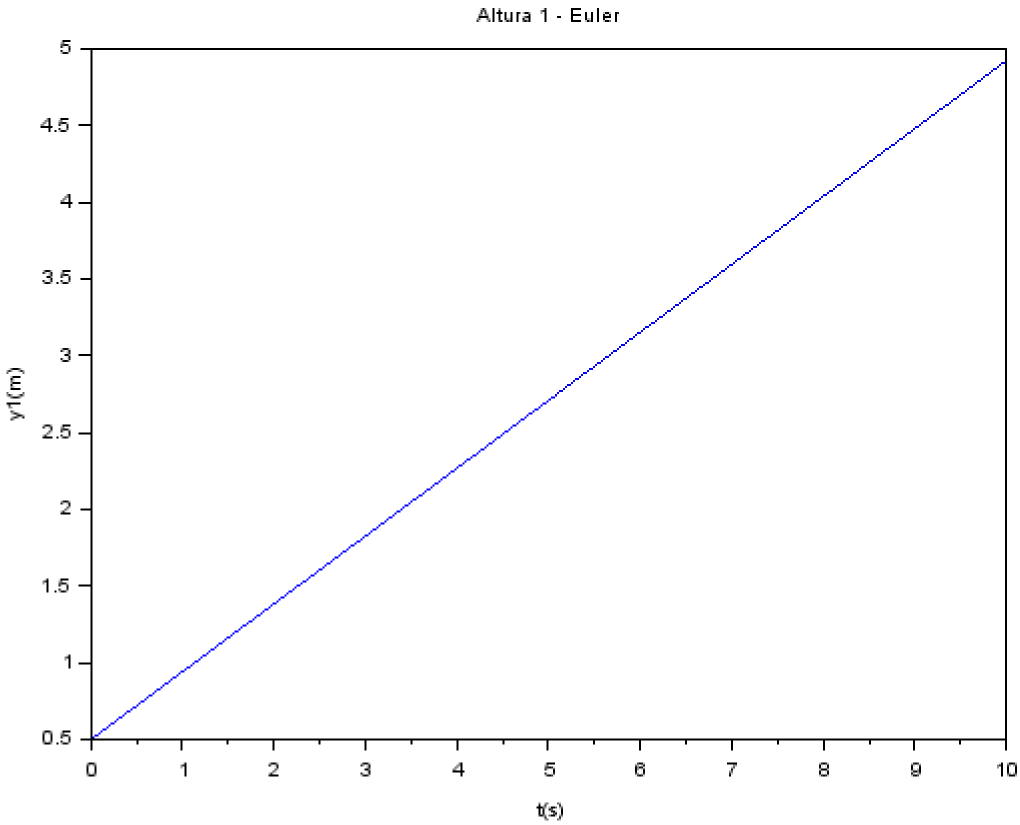
No segundo exercício proposto devemos modelar um sistema com 2 tanques, conforme ilustrado abaixo:

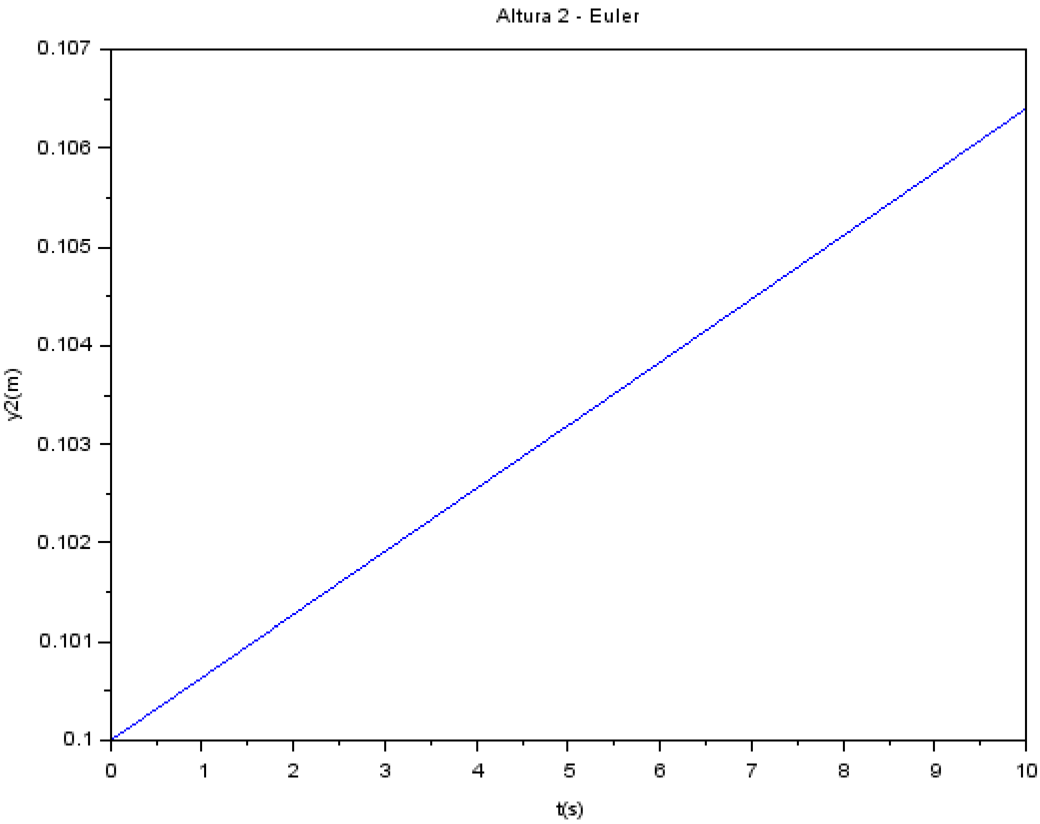


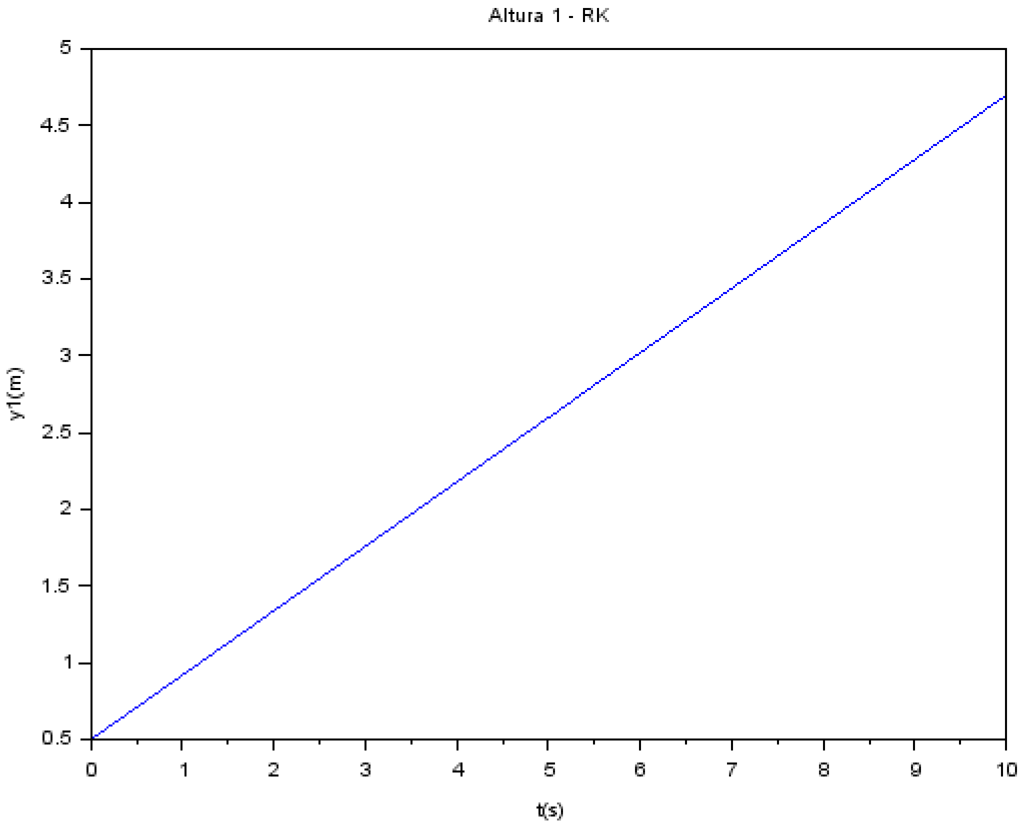
Fazemos uso das seguintes equações diferenciais e dos valores iniciais de 0.5m para o tanque 1 e 0.1m para o tanque 2:

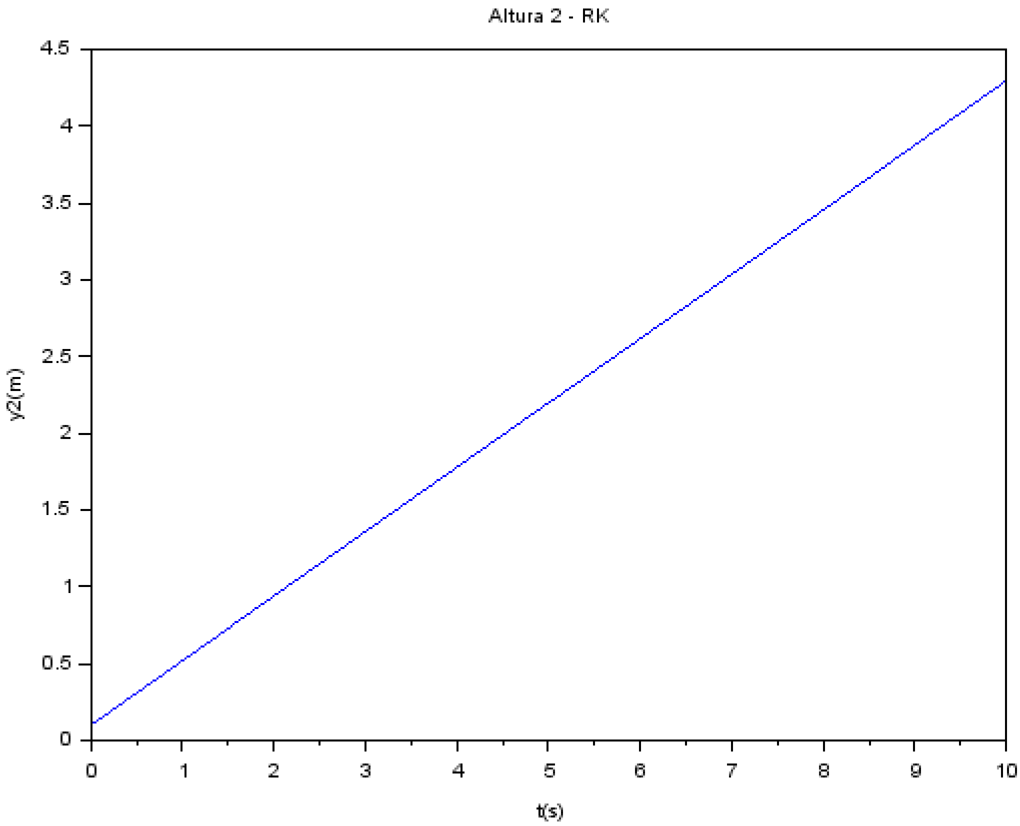
$$\begin{cases} \dot{h}_1 = \left[Q_e - \sqrt{\dfrac{\rho g}{R_a}(h_1 - h_2)} \right] \dfrac{1}{S_1} \\ \dot{h}_2 = \left[\sqrt{\dfrac{\rho g}{R_a}(h_1 - h_2)} - \sqrt{\dfrac{\rho g}{R_s} h_2} \right] \dfrac{1}{S_2} \end{cases}$$

Por fim, a seguir estão os resultados obtidos:


Altura 1 - Euler
y1(m)
t(s)
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
1
2
3
4
5
6
7
8
9
10


Altura 2 - Euler
y2(m)
t(s)
0.107
0.106
0.105
0.104
0.103
0.102
0.101
0.1
0
1
2
3
4
5
6
7
8
9
10


Altura 1 - RK
y1(m)
t(s)
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
1
2
3
4
5
6
7
8
9
10


Altura 2 - RK
y2(m)
t(s)
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
0
1
2
3
4
5
6
7
8
9
10

Figuras 4, 5, 6 e 7: Resultados obtidos.

Nota-se que os resultados referentes ao tanque 1 são semelhantes para os dois métodos. Porém, há discrepâncias significativas entre os resultados referentes ao tanque 2.

3 APÊNDICE 1

```
// Instante inicial:
t(1)=0;
// Instante final:
tf=10;
// Condicao inicial:
y(1)=0.5;
// Passo de integracao (experimente alterar o passo):
h=0.5;
// Calculo de numero de passos):
n=round(tf/h);

//Definindo valores a serem utilizados
rho = 1000;
s = 10;
g = 10;
R = 2*(10^8);
Qe = 0.010247;

//Estabelecendo a equação diferencial
function [ydot]=funcao(y)
ydot=(-sqrt((rho*g*y)/R) + Qe)/s;
endfunction

// 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));
// Termino do comando for:
end

//plotando o gráfico da solução numérica
f1 = scf(1)
plot(t,y);
xtitle("Solucao numerica - Euler", "t(s)","y(m)");
```

4 APÊNDICE 2

```
// Instante inicial:
t(1)=0;
// Instante final:
tf=10;
// Condicao inicial:
y(1)=0.5;
// Passo de integracao (experimente alterar o passo):
h=0.5;
// Calculo de numero de passos):
n=round(tf/h);

//Definindo valores a serem utilizados
rho = 1000;
s = 10;
g = 10;
R = 2*(10^8);
Qe = 0.010247;

//Estabelecendo a equação diferencial
function [ydot]=funcao(y)
ydot=(-sqrt((rho*g*y)/R) + Qe)/s;
endfunction

// 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-(funcao(y(i))))/2;
   k2=h*(1-(funcao(y(i))+k1/2))/2;
   k3=h*(1-(funcao(y(i))+k2/2))/2;
   k4=h*(1-(funcao(y(i))+k3))/2;
   y(i+1)=y(i)+((k1+2*k2+2*k3+k4)/6);
// Termino do comando for:
end

//plotando o gráfico da solução numérica
f1 = scf(1)
plot(t,y);
xtitle("Solucao numerica - Runge Kutta", "t(s)","y(m)");
```

5 APÊNDICE 3

```
// Instante inicial:
t(1)=0;
// Instante final:
tf=10;
// Passo de integracao:
h=0.5;
// Calculo de numero de passos:
n=round(tf/h);
//Definindo valores a serem utilizados
rho = 1000;
s = 10;
g = 10;
Ra = 2*(10^8);
Rb = 2*(10^8);
Qe = 0.010247;

//vetor de tempo
t = linspace(0, tf, n);

y1 = zeros(1,n);
y2 = zeros(1,n);
y1(1)=0.5;
y2(1)=0.1;

//definindo as equações diferenciais
function [ydot]=funcao1(i)
ydot=((Qe-sqrt((rho*g*(y1(i)-y2(i))/Ra)))/s);
endfunction

function [ydot]=funcao2(i)
ydot=((sqrt((rho*g*(y1(i)-y2(i))/Ra)) + (sqrt((rho*g*y2(i))/Rb)))/s);
endfunction

for i=1:(n-1)
  k11=h*(1-(funcao1(i)))/2;
  k21=h*(1-(funcao1(i)+k11/2))/2;
  k31=h*(1-(funcao1(i)+k21/2))/2;
  k41=h*(1-(funcao1(i)+k31))/2;
  y1(i+1)=y1(i)+((k11+2*k21+2*k31+k41)/6);

  k12=h*(1-(funcao2(i)))/2;
  k22=h*(1-(funcao2(i)+k12/2))/2;
  k32=h*(1-(funcao2(i)+k22/2))/2;
  k42=h*(1-(funcao2(i)+k32))/2;
  y2(i+1)=y2(i)+((k12+2*k22+2*k32+k42)/6);
end

f1 = scf(1)
plot(t,y1);
xtitle("Altura 1 - RK", "t(s)","y1(m)");

f2 = scf(2)
plot(t,y2);
xtitle("Altura 2 - RK", "t(s)","y2(m)");
```

6 APÊNDICE 4

```
// Instante inicial:
t(1)=0;
// Instante final:
tf=10;
// Passo de integracao:
h=0.5;
// Calculo de numero de passos:
n=round(tf/h);
//Definindo valores a serem utilizados
rho = 1000;
s = 10;
g = 10;
Ra = 2*(10^8);
Rb = 2*(10^8);
Qe = 0.010247;

//vetor de tempo
t = linspace(0, tf, n);

//vetor de valores de y e valores iniciais
y1 = zeros(1,n);
y2 = zeros(1,n);
y1(1)=0.5;
y2(1)=0.1;

//definindo as equações diferenciais
function [ydot]=funcao1(i)
ydot=((Qe-sqrt((rho*g*(y1(i)-y2(i))/Ra)))/s);
endfunction

function [ydot]=funcao2(i)
ydot=((sqrt((rho*g*(y1(i)-y2(i))/Ra)) + (sqrt((rho*g*y2(i))/Rb)))/s);
endfunction

//método de euler
for i=1:(n-1)
  y1(i+1)=y1(i)+h*funcao1(i);
  y2(i+1)=y2(i)+h*funcao2(i);
end

//plot de graficos
f1 = scf(1)
plot(t,y1);
xtitle("Altura 1 - Euler", "t(s)","y1(m)");

f2 = scf(2)
plot(t,y2);
xtitle("Altura 2 - Euler", "t(s)","y2(m)");
```