Henrique Kuhlmann 10772672

Lista B – PME3380

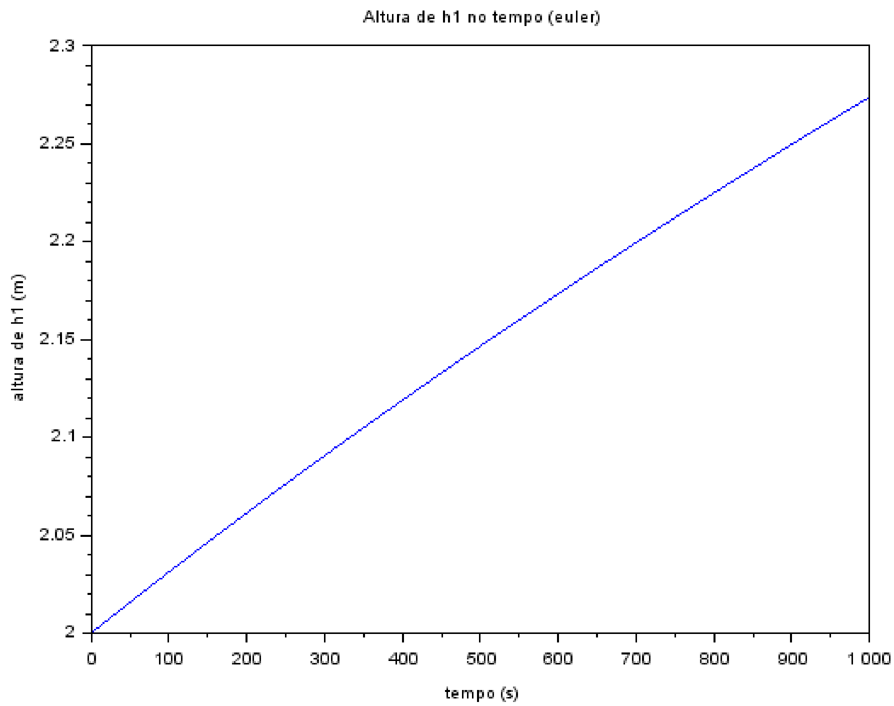
Dado o problema fornecido na Lista B, pretende-se resolver o seguinte problema de valor inicial em torno das variáveis $h_1(t)$ e $h_2(t)$:

$$\begin{cases} \dot{h_1}(t) = \dfrac{Q_e - \sqrt{\dfrac{\rho g}{R_1}(h_1 - h_2)}}{S_1} \\ \dot{h_2}(t) = \dfrac{\sqrt{\dfrac{\rho g}{R_1}(h_1 - h_2)} - \sqrt{\dfrac{\rho g}{R_2} h_2}}{S_2} \\ h_1(0) = 2 \\ h_2(0) = 1 \end{cases}$$

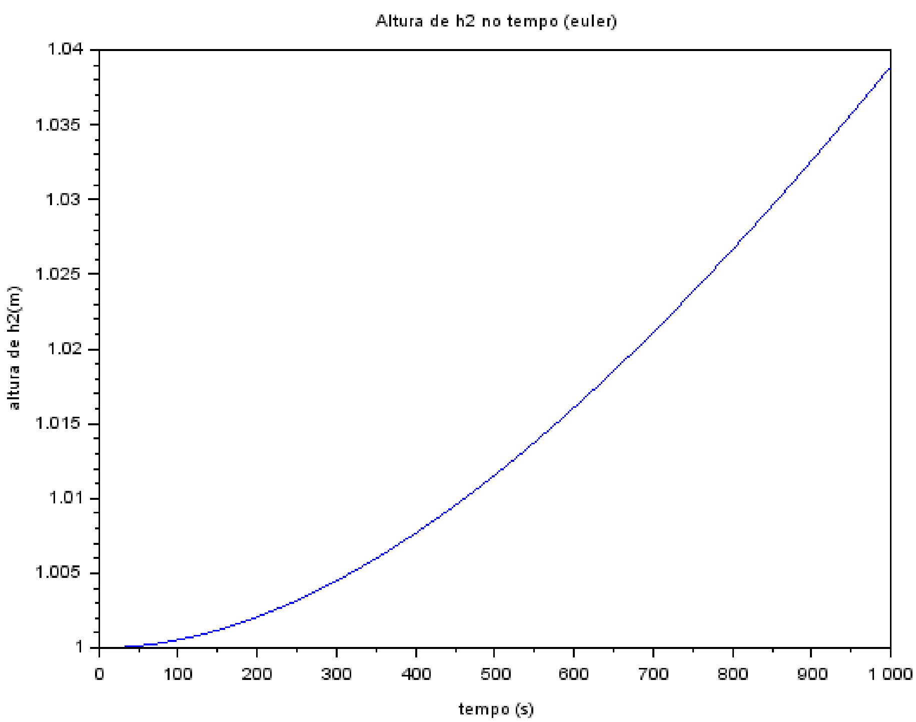
Note que as condições iniciais foram arbitradas, uma vez que não foram fornecidas no problema.

A solução do problema exposto acima é feita por através de métodos numéricos, por meio do método de Euler e de Runge Kutta de quarta ordem. Ambos os métodos foram realizados com um passo de 0,1s e tempo de simulação total de 1000s. Os códigos utilizados podem ser verificados no final do relatório.

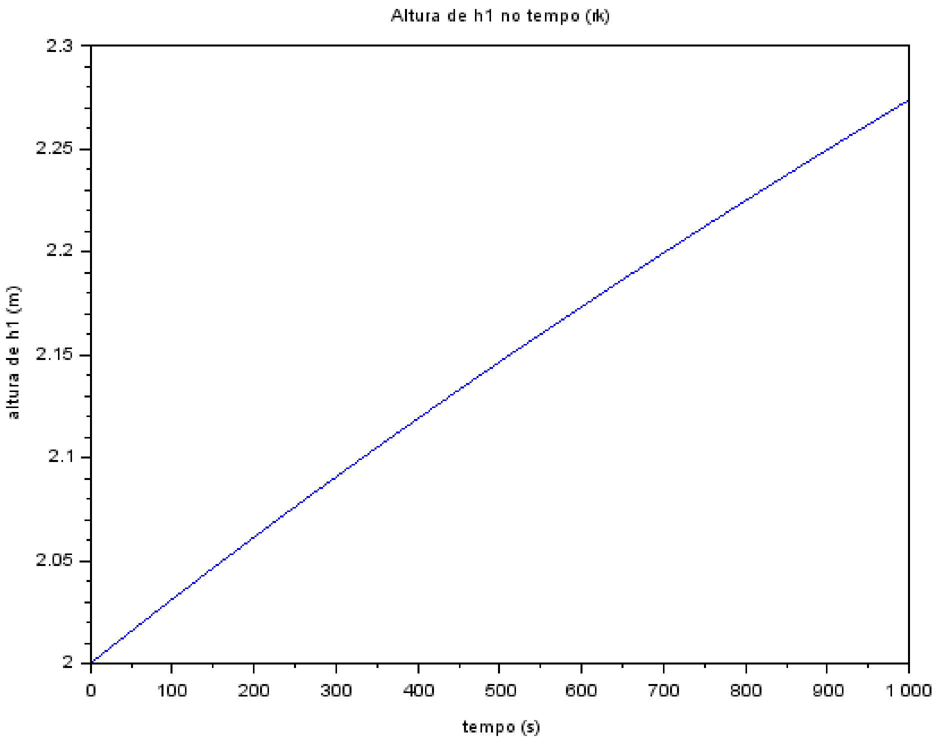
Os gráficos das soluções obtidas para cada um dos métodos são expostos a seguir:

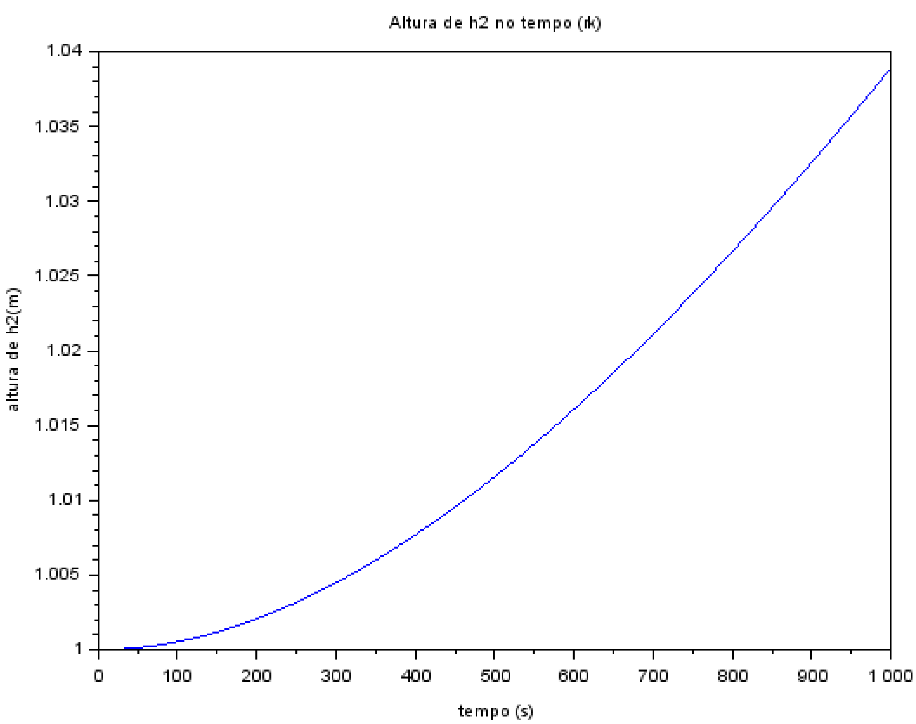


Altura de h1 no tempo (euler)
altura de h1 (m)
tempo (s)
2
2.05
2.1
2.15
2.2
2.25
2.3
0
100
200
300
400
500
600
700
800
900
1 000



Altura de h2 no tempo (euler)
altura de h2(m)
tempo (s)
1
1.005
1.01
1.015
1.02
1.025
1.03
1.035
1.04
0
100
200
300
400
500
600
700
800
900
1 000


Altura de h1 no tempo (rk)
altura de h1 (m)
2.3
2.25
2.2
2.15
2.1
2.05
2
0
100
200
300
400
500
600
700
800
900
1 000
tempo (s)


Altura de h2 no tempo (rk)
altura de h2(m)
1.04
1.035
1.03
1.025
1.02
1.015
1.01
1.005
1
0
100
200
300
400
500
600
700
800
900
1 000
tempo (s)

O desvio entre os resultados obtidos em cada método não é perceptível, o que indica que o passo de integração utilizado é adequado.

Códigos utilizados:

Euler:

```
clear;
'todos as unidades no SI'
S=10;
R=2*10^8;
rho=1000;
g=10;
R1=R;
R2=R;
S1=S;
S2=S;
Qe=0.010247

'valores iniciais'
h10=2;
h20=1;

'tempo de simulacao'
tn=10000;

'vetor tempo'
n=1000;
T=linspace(0,tn,n);
dt=T(2)-T(1);

'vetores de altura'
H1=zeros(1,n);
H2=zeros(1,n)
H1(1)=h10;
H2(1)=h20;

'formulas fornecidas'
function y=h1l(i)
    y = (Qe-sqrt(rho*g*(H1(i)-H2(i))/R1))/S1;
endfunction

function y=h2l(i)
    y=(sqrt(rho*g*(H1(i)-H2(i))/R1)-sqrt(rho*g*H2(i)/R2))/S2;
endfunction

'integracao pelo metodo de Euler'
for i=2:n
    H1(i)=H1(i-1)+dt*h1l(i-1)
    H2(i)=H2(i-1)+dt*h2l(i-1)
end

'resultados'
scf(1);
xtitle("Altura de h1 no tempo (euler)","tempo (s)","altura de h1 (m)")
plot(T,H1);
```

```
k2=scf(2);
plot(T,H2);
xtitle("Altura de h2 no tempo (euler)","tempo (s)","altura de h2(m)")
```

Runge Kutta:

```
clear;
'todos as unidades no SI'
S=10;
R=2*10^8;
rho=1000;
g=10;
R1=R;
R2=R;
S1=S;
S2=S;
Qe=0.010247

'valores iniciais'
h10=1;
h20=0.5;

'tempo de simulacao'
tn=10000;

'vetor tempo'
n=101;
T=linspace(0,tn,n);
dt=T(2)-T(1);

'vetores de altura'
H1=zeros(1,n);
H2=zeros(1,n)
H1(1)=h10;
H2(1)=h20;

'formulas fornecidas'
function y=h1l(i)
    y = (Qe-sqrt(rho*g*(H1(i)-H2(i))/R1))/S1;
endfunction

function y=h2l(i)
    y=(sqrt(rho*g*(H1(i)-H2(i))/R1)-sqrt(rho*g*H2(i)/R2))/S2;
endfunction

'integracao RK4'
for i=2:n;
    k1=dt*h1l(i-1);
    k2=dt*h1l(i-1+k1/2);
    k3=dt*h1l(i-1+k2/2);
    k4=dt*h1l(i-1+k3);
    H1(i)=H1(i-1)+(k1+2*k2+2*k3+k4)/6

    k1=dt*h2l(i-1);
```

```
    k2=dt*h2l(i-1+k1/2);
    k3=dt*h2l(i-1+k2/2);
    k4=dt*h2l(i-1+k3);
    H2(i)=H2(i-1)+(k1+2*k2+2*k3+k4)/6
end

'resultados'
scf(1);
xtitle("Altura de h1 no tempo","tempo (s)","altura de h1 (m)")
plot(T,H1);
k2=scf(2);
plot(T,H2);
xtitle("Altura de h2 no tempo","tempo (s)","altura de h2(m)")
```