UNIVERSIDADE DE SÃO PAULO

ESCOLA POLITÉCNICA

VICTOR MANOEL FERREIRA ROSA DA COSTA

10772713

Modelagem de Sistemas Dinâmicos- Lista B

São Paulo

2020

Nestes exercícios pretende-se simular as soluções de equações diferenciais utilizando os métodos numéricos de Euler e Runge-Kutta de 4ª ordem.

1- Sistema com um reservatório

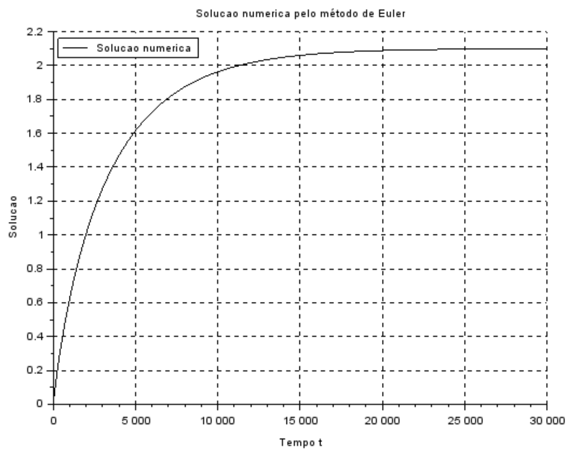
Ao estudar o sistema chega-se a seguinte equação diferencial:

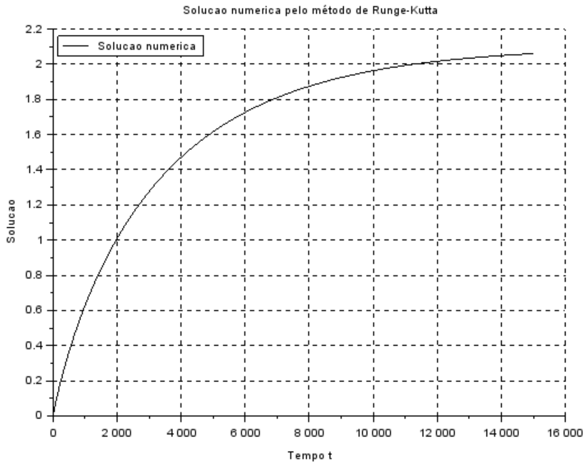
$$\dot{h} = \frac{1}{S}\left(Q_e - \sqrt{\frac{\rho g h}{R}}\right)$$

Onde:

- h é o nível do reservatório [m]
- S é a área da seção transversal do reservatório [10 m^2]
- Q_e é a vazão de entrada [m^3/s]
- ρ é a massa específica da água [1000 kg/m^3]
- g é a aceleração da gravidade [10 m/s^2]
- R é a perda de carga [$2\cdot10^8$ $Pa/(m^3/s)^2$]

Usando-se os algoritmos dos métodos numéricos, têm-se os seguintes resultados:





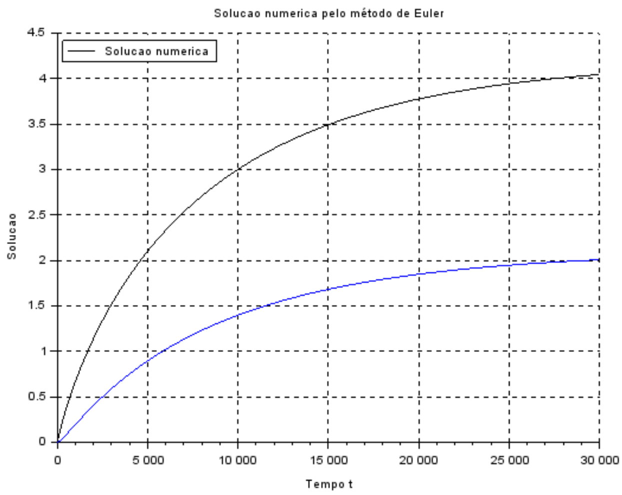
2- Sistema com dois reservatórios

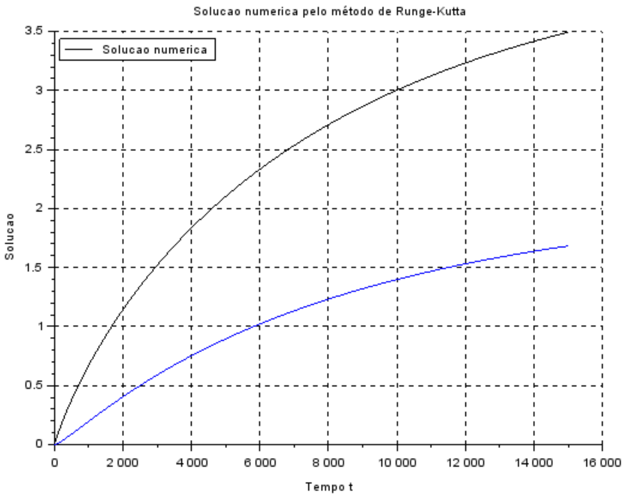
Adotando-se o mesmo procedimento do caso anterior, encontram-se duas equações diferenciais:

$$\dot{h_1} = \frac{1}{S_1}\left(Q_e - \sqrt{\frac{\rho g(h_1 - h_2)}{R_a}}\right)$$

$$\dot{h_2} = \frac{1}{S_2}\left(\sqrt{\frac{\rho g(h_1 - h_2)}{R_a}} - \sqrt{\frac{\rho g h_2}{R_s}}\right)$$

Os resultados foram:





3- Códigos:

Exercício 1 pelo método de Euler:

```
function [ydot]=ex(y)
        ydot=(1/S)*(Qe-sqrt(C*y(i)));
endfunction
save('ex.sci','ex');
```

```
clear;

load('ex.sci','ex');

S=10;
R=2*(10^8);
g=10;
p=1000;
Qe=0.010247;
C=p*g/R;

t(1)=0;
tf=30000;
y(1)=0;
h=0.5;
n=round(tf/h);

for i=1:n
    t(i+1)=t(i)+h;
    y(i+1)=y(i)+h*ex(y);
end

plot2d([t],[y],[1 2]);
T=list("Solucao numerica","Tempo t","Solucao","Solucao numerica");
xset("thickness",1);
legends(T(4),[1,2],2);
xtitle(T(1),T(2),T(3));
xgrid(1);
```

Exercício 1 pelo método de Runge-Kutta:

```
clear;

S=10;
R=2*(10^8);
g=10;
p=1000;
Qe=0.010247;
C=p*g/R;

t(1)=0;
tf=30000;
y(1)=0;
h=0.5;
n=round((tf-t(1)/h));

for i=1:n
    t(i+1)=t(i)+h;
    k1=h*((1/S)*(Qe-sqrt(C*y(i))));
    k2=h*((1/S)*(Qe-sqrt(C*((y(i)+k1/2)))));
    k3=h*((1/S)*(Qe-sqrt(C*((y(i)+k2/2)))));
    k4=h*((1/S)*(Qe-sqrt(C*((y(i)+k3)))));
    y(i+1)=y(i)+((k1+2*k2+2*k3+k4)/6);
end

plot2d(t,y);
T=list("Solucao numerica pelo método de Runge-Kutta","Tempo t","Solucao","Solucao numerica");
xset("thickness",1);
legends(T(4),[1,2],2);
xtitle(T(1),T(2),T(3));
xgrid(1);
```

Exercício 2 pelo método de Euler:

```
clear;

load('ex2A.sci','ex2A');
load('ex2B.sci','ex2B');

S1=10;
S2=8;
R=2*(10^8);
g=10;
p=1000;
Qe=0.010247;
C=p*g/R;

t(1)=0;
tf=30000;
y1(1)=0;
y2(1)=0;
h=0.5;
n=round(tf/h);

for i=1:n
    t(i+1)=t(i)+h;
    y1(i+1)=y1(i)+h*ex2A(y1,y2);
    y2(i+1)=y2(i)+h*ex2B(y1,y2);
end

plot2d([t],[y1,y2],[1 2]);
T=list("Solucao numerica pelo método de Euler","Tempo t","Solucao","Solucao numerica");
xset("thickness",1);
legends(T(4),[1,2],2);
xtitle(T(1),T(2),T(3));
xgrid(1);
```

Exercício 2 pelo método de Runge-Kutta:

```
clear;

S1=10;
S2=8;
R=2*(10^8);
g=10;
p=1000;
Qe=0.010247;
C=p*g/R;

t(1)=0;
tf=30000;
y1(1)=0;
y2(1)=0;
h=0.5;
n=round((tf-t(1)/h));

for i=1:n
    t(i+1)=t(i)+h;
    k1=h*((1/S1)*(Qe-sqrt(C*(y1(i)-y2(i)))));
    k2=h*((1/S1)*(Qe-sqrt(C*(((y1(i)-y2(i))+k1/2)))));
    k3=h*((1/S1)*(Qe-sqrt(C*(((y1(i)-y2(i))+k2/2)))));
    k4=h*((1/S1)*(Qe-sqrt(C*(((y1(i)-y2(i))+k3)))));
    y1(i+1)=y1(i)+((k1+2*k2+2*k3+k4)/6);
    
    q1=h*((1/S2)*(sqrt(C*(y1(i)-y2(i)))-sqrt(C*y2(i))));
    q2=h*((1/S2)*(sqrt(C*(((y1(i)-y2(i))+q1/2)))-sqrt(C*(y2(i)+q1/2))));
    q3=h*((1/S2)*(sqrt(C*(((y1(i)-y2(i))+q2/2)))-sqrt(C*(y2(i)+q2/2))));
    q4=h*((1/S2)*(sqrt(C*(((y1(i)-y2(i))+q3)))-sqrt(C*(y2(i)+q3))));
    y2(i+1)=y2(i)+((q1+2*q2+2*q3+q4)/6);
end
```