ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO
DEPARTAMENTO DE ENGENHARIA MECÂNICA

PME 3380 - MODELAGEM DE SISTEMAS DINÂMICOS

LISTA B

Aluno:
Caio Shohei Uemura Fujinaka 8040879

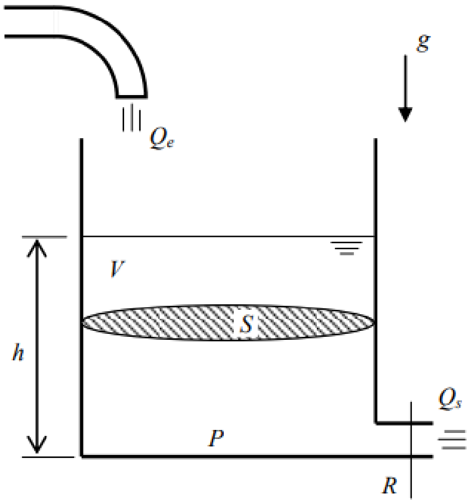
Docentes:
Prof. Dr. Agenor de Toledo Fleury
Prof. Dr. Décio Crisol Donha

São Paulo, SP
2020

A Lista B consiste em 2 exercícios: (i) Implementar um programa que resolva numericamente a equação diferencial que modela o sistema de reservatório com água pelos métodos de Euler e de Runge-Kutta e (ii) Implementar um programa que modele o sistema com 2 reservatórios com água, novamente pelos métodos de Euler e de Runge-Kutta.

I. Sistema de 1 Reservatório com Água

a. Apresentação do Sistema



Reservatório com água
Parâmetros:
S = 10 m^2 - área da seção transversal (constante)
R = 2×10^8 Pa/(m^3/s)^2 - parâmetro que relaciona vazão com queda de pressão (perda de carga)
ρ = 1000 kg/m^3 - massa específica da água
G = 10 m/s^2 - aceleração da gravidade na superfície da terra

Variáveis:
Q_e = 0,010247 m^3/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.

b. Método de Euler

i. Criação da Função B1 – EDO modelo de 1 reservatório

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

```
function dy = f(y)
S = 10;        %Área da seção transversal(m²)
R = 2E8;       %Perda de carga (Pa/(m³/s²))
ro = 1000;     %Massa específica da água (kg/m³)
G = 10;        %Aceleração da gravidade (m/s²)
Qe = 0.010247; %Vazão de entrada (m³/s)

dy = ( -( ro*G*y / R )^(1/2) + Qe) / S;
end
```

ii. Código Básico

```
t(1)  = 0;                    % T inicial
tf = 10;                      % T final
y(1)  = 0;                    % Condição inicial
h = 0.5;                      % Passo
n = round( (tf - t(1)) / h );  % Cálculo # passos

%Método de Euler
for i = 1:n
    t(i+1) = t(i) + h;                   %Vetor de tempo
    y(i+1) = y(i) + h*function_B1(y(i)); %Solução numérica;
end

% Plotando a solução numérica y vs tempo
plot(t,y);
% Legenda inferior direita da figura:
```

```
legend('Solução Numérica')
% Título e eixos:
title('Solução numérica usando método de Euler')
xlabel('Tempo t')
ylabel('Solução numérica')

grid on;    %Colocando grade no gráfico
```

iii. Gráfico Gerado

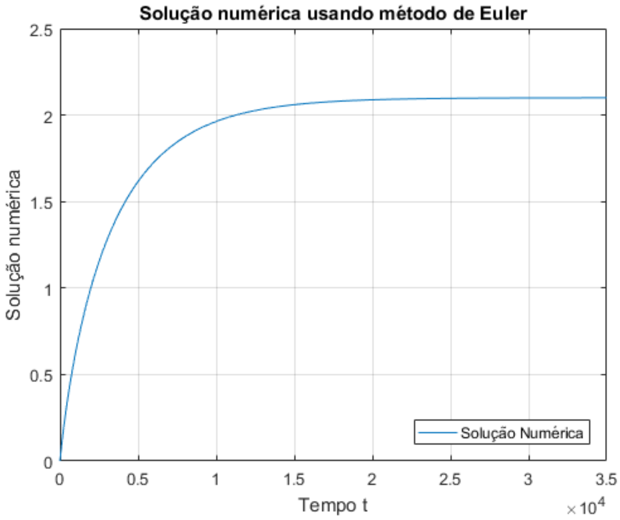


Figura 1-Solução para 1 Reservatório com Água através do Método de Euler

c. Método de Runge Kutta

i. Código Básico

```
(1) = 0;                          % T inicial
tf = 10000;                            % T final
y(1) = 0;                          % Condição inicial
h = 0.5;                           % Passo
n = round( (tf - t(1)) / h );     % Cálculo # passos

% Método Runge-Kutta
  for i = 1:n
      t(i+1) = t(i) + h;   %Vetor de tempo

      %Solução numérica
      k1 = function_B1( y(i)          );
      k2 = function_B1( y(i) + h*k1/2 );
      k3 = function_B1( y(i) + h*k2/2 );
      k4 = function_B1( y(i) + k3*h   );
      y(i+1) = y(i) + ( h*( k1 + 2*k2 + 2*k3 + k4 )/ 6 );
  end

% Plotando a solução numérica y vs tempo
plot(t,y);
% Legenda inferior direita da figura:
legend('Solução Numérica')
% Título e eixos:
title('Solução numérica usando Runge-Kutta')
xlabel('Tempo t')
ylabel('Solução numérica')

grid on;    %Colocando grade no gráfico
```

ii. Gráfico Gerado

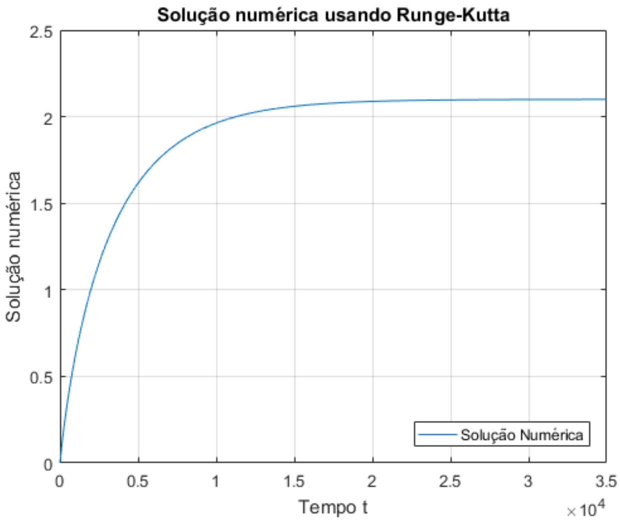
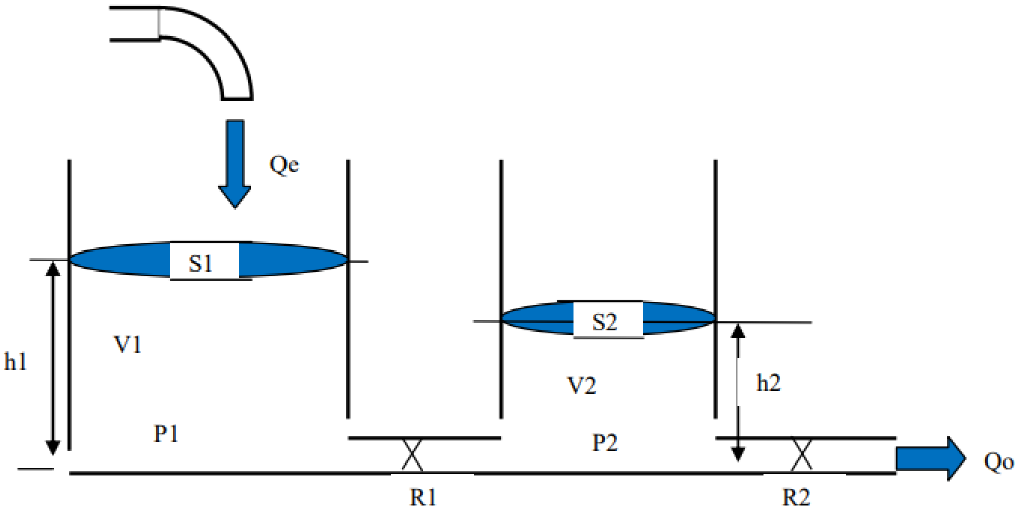


Figura 2-Solução para 1 Reservatório com Água através do Método de Runge-Kutta

II. Sistema de 2 Reservatórios com Água

a. Apresentação do Sistema



b. Método de Euler

i. Criação da Função B2 – EDO modelo de 2 reservatório

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

```
function dy = f(y)
S1 = 10;      %Área da seção transversal do primeiro reservatório(m²)
S2 = 5;       %Área da seção transversal do segundo reservatório (m²)
R = 2E8;      %Perda de carga (Pa/(m³/s²))
ro = 1000;    %Massa específica da água (kg/m³)
```

```
G = 10;        %Aceleração da gravidade na superfície da terra (m/s²)
Qe = 0.010247; %Vazão de entrada (m³/s)

dy(1) = ( Qe -( ro*G*(y(1) - y(2)) / R )^(1/2) ) / S1;
dy(2) = ( ( ro*G*(y(1) - y(2)) / R )^(1/2) - ( ro*G*y(2) / R )^(1/2) ) / S2;
end
```

ii. Código Básico

```
t(1) = 0;                    % T inicial
tf   = 10;                   % T final
y1(1) = 0;                   % Condição inicial reservatório 1
y2(1) = 0;                   % Condição inicial reservatório 2
h = 0.5;                     % Passo
n = round( (tf - t(1)) / h );  % Cálculo # passos

% Método de Euler
for i = 1:n
    t(i+1) = t(i) + h;

    % Solução numérica
    x = function_B2( [ y1(i), y2(i) ] );
    y1(i+1) = y1(i) + h * x(1);
    y2(i+1) = y2(i) + h * x(2);
end

% Plotando a solução numérica y vs tempo
plot( t, y1, t, y2 );
% Legenda na parte inferior direita da figura:
legend('Reservatório 1', 'Reservatório 2')
% Título na figura e eixos:
title('Solução numérica usando método de Euler')
xlabel('Tempo t')
ylabel('Solução numérica')

grid on;    %Colocando grade no gráfico
```

iii. Gráfico Gerado

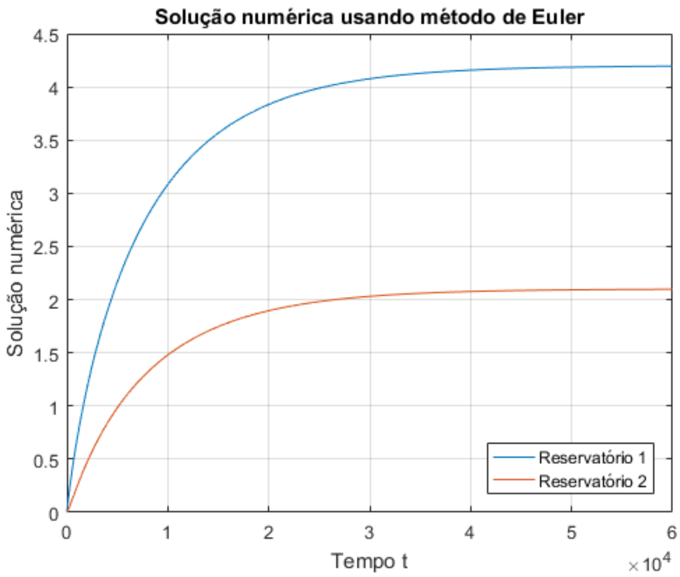


Figura 3-Solução para 2 Reservatórios com Água através do Método de Euler

c. Método de Runge Kutta

i. Código Básico

```
t(1) = 0;                    % T inicial
tf = 10000;                   % T final
y1(1) = 0;                   % Condição inicial reservatório 1
y2(1) = 0;                   % Condição inicial reservatório 2
h = 0.5;                     % Passo
n = round( (tf - t(1)) / h );    % Cálculo # passos

%Método de Runge-Kutta
  for i = 1:n
     t(i+1) = t(i) + h;   %Vetor de tempo

     k1 = function_B2( [y1(i)          , y2(i)          ] );
     k2 = function_B2( [y1(i) + h*k1(1)/2 , y2(i) + h*k1(2)/2 ] );
     k3 = function_B2( [y1(i) + h*k2(1)/2 , y2(i) + h*k2(2)/2 ] );
     k4 = function_B2( [y1(i) + k3(1)*h   , y2(i) + k3(2)*h   ] );

     y1(i+1) = y1(i) + ( h*( k1(1) + 2*k2(1) + 2*k3(1) + k4(1) )/ 6 );
     y2(i+1) = y2(i) + ( h*( k1(2) + 2*k2(2) + 2*k3(2) + k4(2) )/ 6 );
  end

% Plotando a solução numérica y vs tempo
plot(t, y1, t, y2);
% Legenda parte inferior direita da figura:
legend('Reservatório 1', 'Reservatório 2')
% Título e eixos:
title('Solução numérica usando Runge-Kutta')
xlabel('Tempo t')
ylabel('Solução numérica')

grid on;    %Colocando grade no gráfico
```

ii. Gráfico Gerado

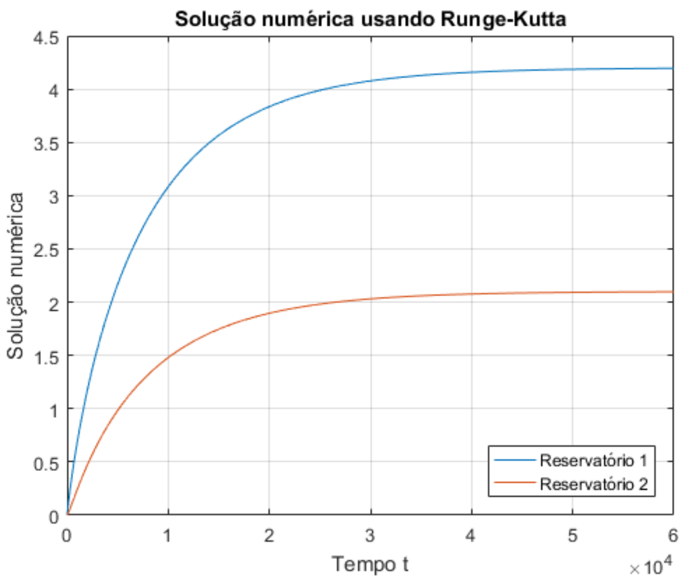


Figura 4-Solução para 2 Reservatórios com Água através do Método de Runge-Kutta