# LISTA B- PME3380


Nome: Gabriel Rodrigues Camargo

NUSP: 10772460


## EXERCÍCIO 1

Nesse exercício foi implementado métodos numéricos de Runge Kutta e Euler para a solução do problema de enchimento de um reservatório a partir da seguinte equação:

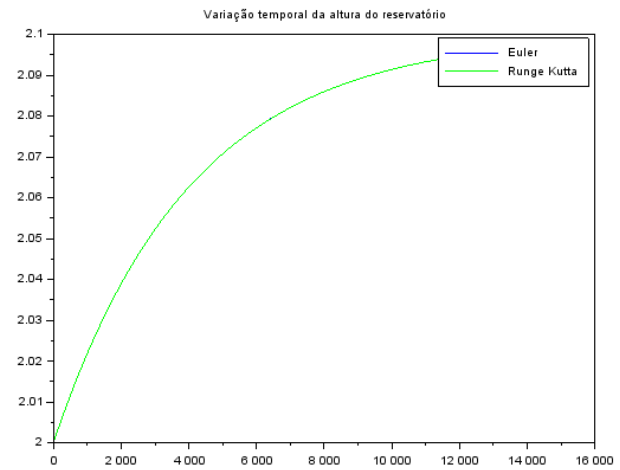
$$\dot{h} = \left( -\sqrt{\frac{\rho g h}{R}} + Q_e \right) \frac{1}{S}$$

Regida pelos seguintes parâmetros e variáveis:

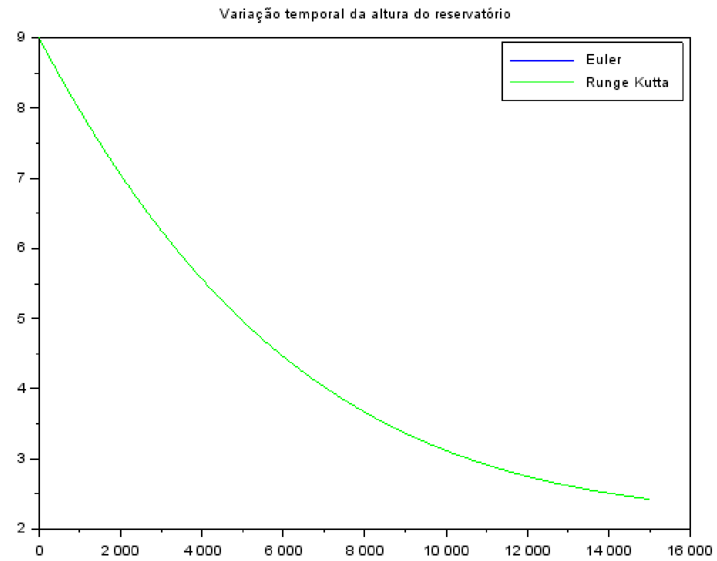
Parâmetros:
$S = 10\ \mathrm{m}^2$ - área da seção transversal (constante)
$R = 2 \times 10^8\ \ \mathrm{Pa/(m^3/s)^2}$ - parâmetro que relaciona vazão com queda de pressão (perda de carga)
$\rho = 1000\ \mathrm{kg/m^3}$ - massa específica da água
$G = 10\ \mathrm{m/s^2}$ - aceleração da gravidade na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

Variáveis:
$Q_e = 0{,}010247\ \mathrm{m^3/s}$ - vazão de entrada
$h$: nível do reservatório [m]
$V$: volume de água no reservatório [$\mathrm{m^3}$]
$P$: pressão relativa à atmosférica, no fundo do reservatório [Pa]
$Q_s$: vazão de saída [$\mathrm{m^3/s}$]

A aplicação dos métodos de Euler e Runge Kutta sob condição inicial $h_0 = 2m$, sob um intervalo de tempo de 15000 segundos com um passo de integração igual a 0,1s nos fornece os seguintes resultados:




Os resultados praticamente são idênticos, mostrando a eficiência do passo escolhido. Além disso no início há um aumento da altura do reservatório devido à vazão superior ser maior em relação a vazão inferior, resultado da uma baixa carga relacionada a altura. Quando aumenta-se o valor de $h_0$ para 9 metros, pode-se perceber comportamento inverso:



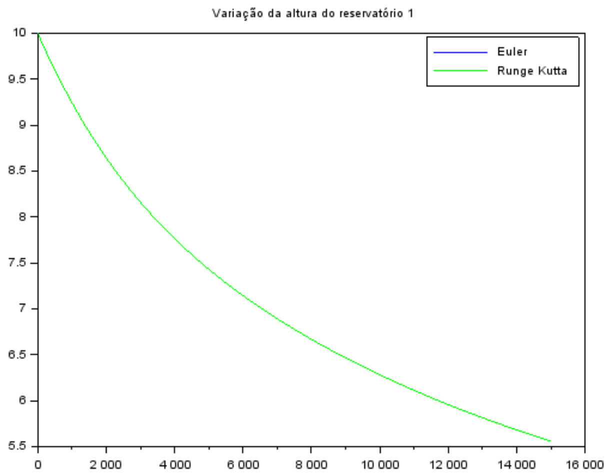

Esse resultado decorre da altura que por ser muito maior favorece a sua carga em relação à vazão superior. Além disso, assim como o outro caso, esse resultado mostra a precisão de ambos os métodos.

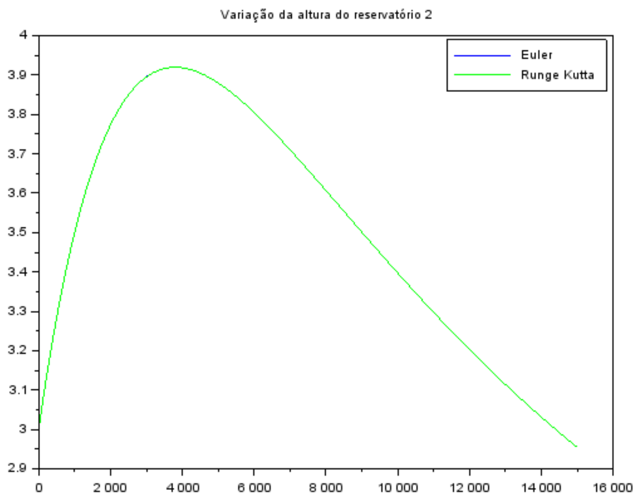
# EXERCÍCIO 2

Nesse problema usa-se os mesmos parâmetros do problema anterior mas com 2 reservatórios que seguem o seguinte sistema de equações:

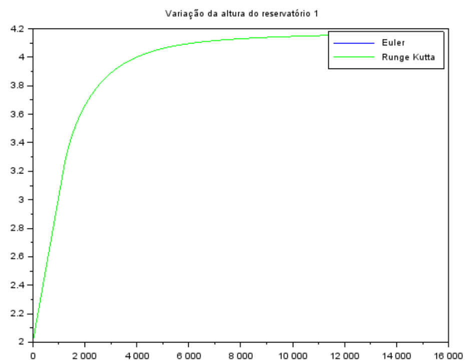
$$\begin{cases} \dot{h}_1 = \left[ Q_e - \sqrt{\dfrac{\rho g}{R_a}(h_1 - h_2)} \right] \dfrac{1}{S_1} \\ \dot{h}_2 = \left[ \sqrt{\dfrac{\rho g}{R_a}(h_1 - h_2)} - \sqrt{\dfrac{\rho g}{R_s} h_2} \right] \dfrac{1}{S_2} \end{cases}$$

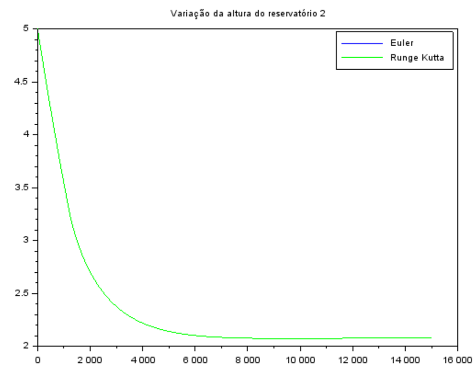
Foi adotado como condição inicial $h_1(0) = 10m$ e $h_2(0) = 3m$, um intervalo de tempo igual a 15000s com um passo de integração igual a 0,1 segundos que fornece o seguinte resultado para os métodos de Runge Kutta e Euler:




Os valores novamente ficam praticamente iguais, alterando $h_1(0) = 2m$ e $h_2(0) = 5m$ para nova tentativa obtém-se os seguintes resultados:



## CÓDIGO UTILIZADO:

```
clear
//Constantes Utilizadas
S = 10;
R = 2*10^8;
rho = 1000;
g = 10;
Qe = 0.010247;
"--------EXERCICIO 1-------------"

function hponto=funcao(h)
 hponto= (-sqrt(rho*g*h/R)+Qe)/S
endfunction
// Instante inicial:
t(1)=0;
// Instante final:
tf=15000;
// Condicao inicial para Euler e Runge Kutta:
h_eu(1)=9;
h_rk(1)=9;
// Passo de integracao:
passo=0.1;
// Calculo de numero de passos:
n=round(tf/passo);
// Integracao numerica usando o metodo de Euler:
for i=1:n
// Vetor de tempo:
   t(i+1)=t(i)+passo;
// Solucao numerica:
   h_eu(i+1)=h_eu(i)+passo* funcao(h_eu(i));
end

// Integracao numerica usando o metodo de Runge Kutta:
for i=1:n
// Vetor de tempo:
   t(i+1)=t(i)+ passo;
// Solucao numerica:
   k1=funcao(h_rk(i));
   k2=funcao(h_rk(i) + (passo/2)*k1);
   k3=funcao(h_rk(i) + (passo/2)*k2);
   k4=funcao(h_rk(i) + passo*k3);
   h_rk(i+1)=h_rk(i) + passo*((k1+2*k2+2*k3+k4)/6);
end

scf(1)
xtitle("Variação temporal da altura do reservatório")
plot(t,h_eu);
plot(t,h_rk,'g');
legend(['Euler';'Runge Kutta']);

"--------EXERCICIO 2-------------"
function hponto=funcao1(h1, h2)
 hponto= (Qe - sqrt(rho*g*(h1-h2)/R))*(1/S)
endfunction
function hponto=funcao2(h1, h2)
 hponto= (sqrt(rho*g*(h1-h2)/R) - sqrt(rho*g*h2/R))*(1/S)
endfunction
// Instante inicial:
t(1)=0;
// Instante final:
tf=15000;
// Condicao inicial da altura do reservatório:
h1_0 = 2;
h2_0 = 5;
h1_e(1) = h1_0;
```

```
h2_e(1) = h2_0;
h1_rk(1) = h1_0;
h2_rk(1) = h2_0;
// Passo de integracao:
passo=0.1;
// Calculo de numero de passos:
n = round((tf-t(1))/passo);
//Método de Euler:
for i = 1:n
   t(i+1) = t(i) + passo
   h1_e(i+1) = h1_e(i) + passo*funcao1(h1_e(i),h2_e(i));
   h2_e(i+1) = h2_e(i) + passo*funcao2(h1_e(i),h2_e(i));
end
//Método de Runge Kutta:
for i = 1:n
 k11 = funcao1(h1_rk(i),h2_rk(i));
 k21 = funcao2(h1_rk(i),h2_rk(i));
 k12 = funcao1(h1_rk(i) + (passo/2)*k11, h2_rk(i) + (passo/2)*k21)
 k22 = funcao2(h1_rk(i) + (passo/2)*k11, h2_rk(i) + (passo/2)*k21)
 k13 = funcao1(h1_rk(i) + (passo/2)*k12, h2_rk(i) + (passo/2)*k22)
 k23 = funcao2(h1_rk(i) + (passo/2)*k12, h2_rk(i) + (passo/2)*k22)
 k14 = funcao1(h1_rk(i) + passo*k13, h2_rk(i) + passo*k23)
 k24 = funcao2(h1_rk(i) + passo*k13, h2_rk(i) + passo*k23)
 h1_rk(i+1) = h1_rk(i) + (passo/6)*(k11 + 2*k12 + 2*k13 + k14)
 h2_rk(i+1) = h2_rk(i) + (passo/6)*(k21 + 2*k22 + 2*k23 + k24)
end

scf(2)
xtitle("Variação da altura do reservatório 1")
plot(t,h1_e)
plot(t,h1_rk,'g')
legend(['Euler';'Runge Kutta']);
scf(3)
xtitle("Variação da altura do reservatório 2")
plot(t,h2_e)
plot(t,h2_rk,'g')
legend(['Euler';'Runge Kutta']);
```