

1. Exemplo 1

No exemplo 1, inicia-se com a criação da função que representa a equação diferencial. Em seguida, são utilizados os comandos para a resolver através do Método de Euler. Também são plotados os seguintes gráficos, que comparam os resultados da solução exata aos do Método de Euler.

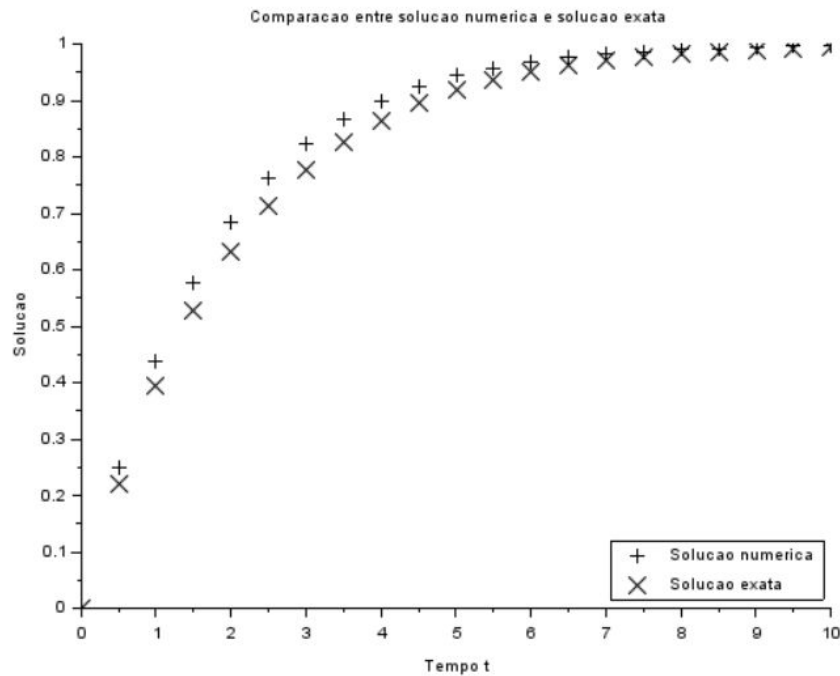


Imagem 1 - Janela gráfica 0 do exemplo 1

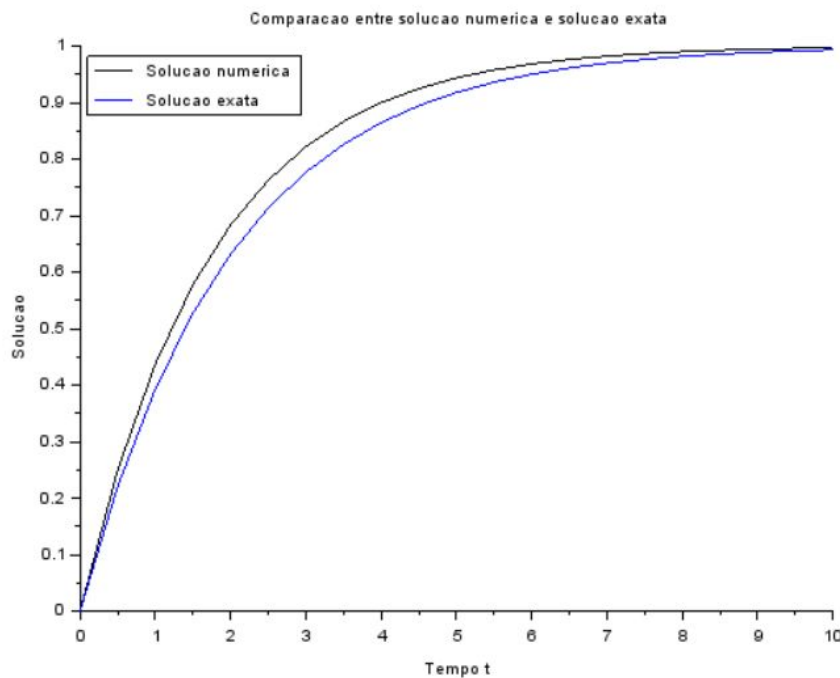


Imagem 2 - Janela gráfica 1 do exemplo 1

2. Exemplo 2

Já no exemplo 2, utiliza-se novamente a função que representa a equação diferencial. Entretanto, em seguida, são utilizados os comandos para a resolver através do Método de Runge Kutta. Também são plotados os seguintes gráficos, que comparam os resultados da solução exata aos do método pedido.

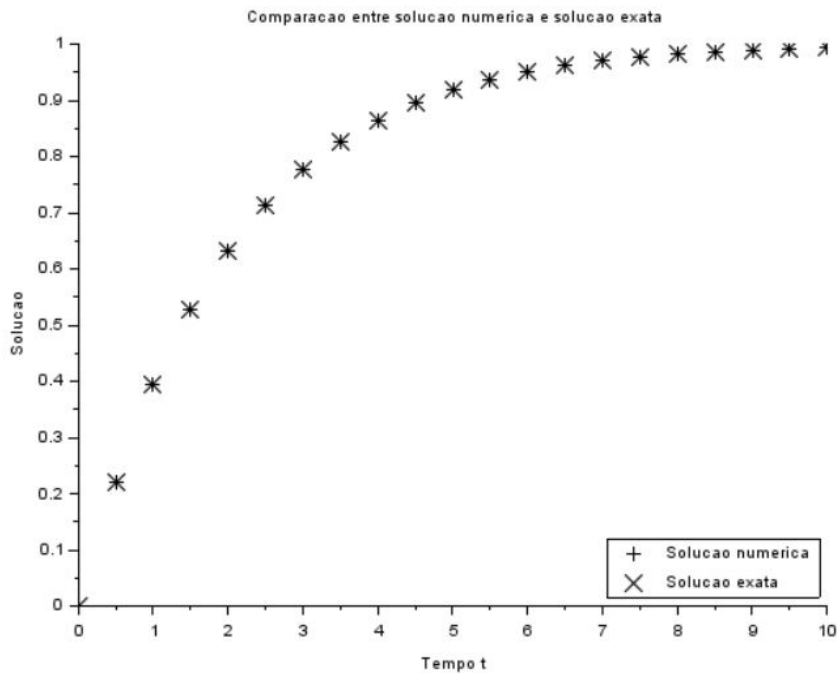


Imagem 3 - Janela gráfica 0 do exemplo 2

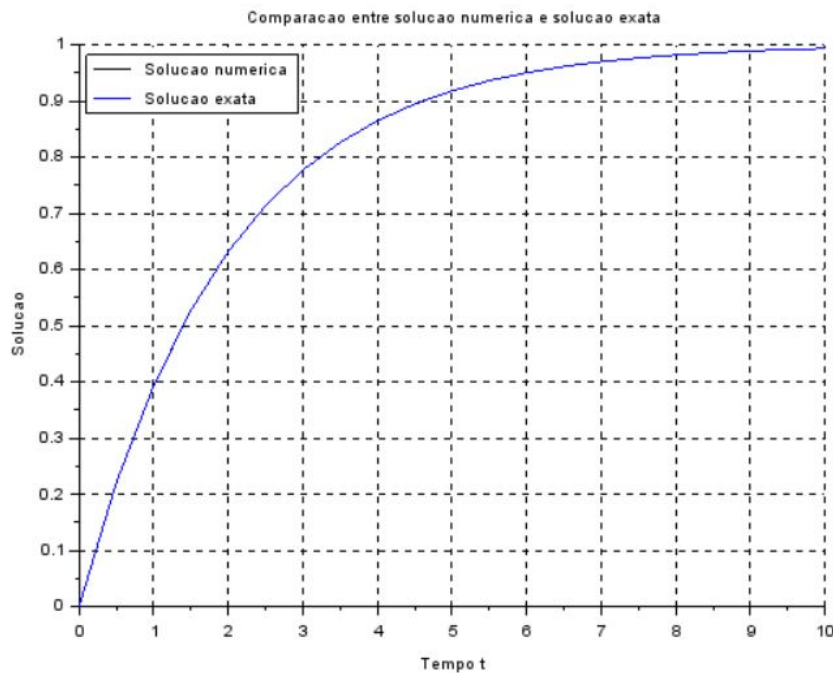


Imagem 4 - Janela gráfica 1 do exemplo 2

3. Exercício 1

No exercício 1, pede-se a altura da coluna d'água no reservatório ao longo do tempo. Primeiramente, usa-se o Método de Euler e, em seguida, o Método de Runge Kutta de quarta ordem. O resultado obtido está apresentado no gráfico abaixo :

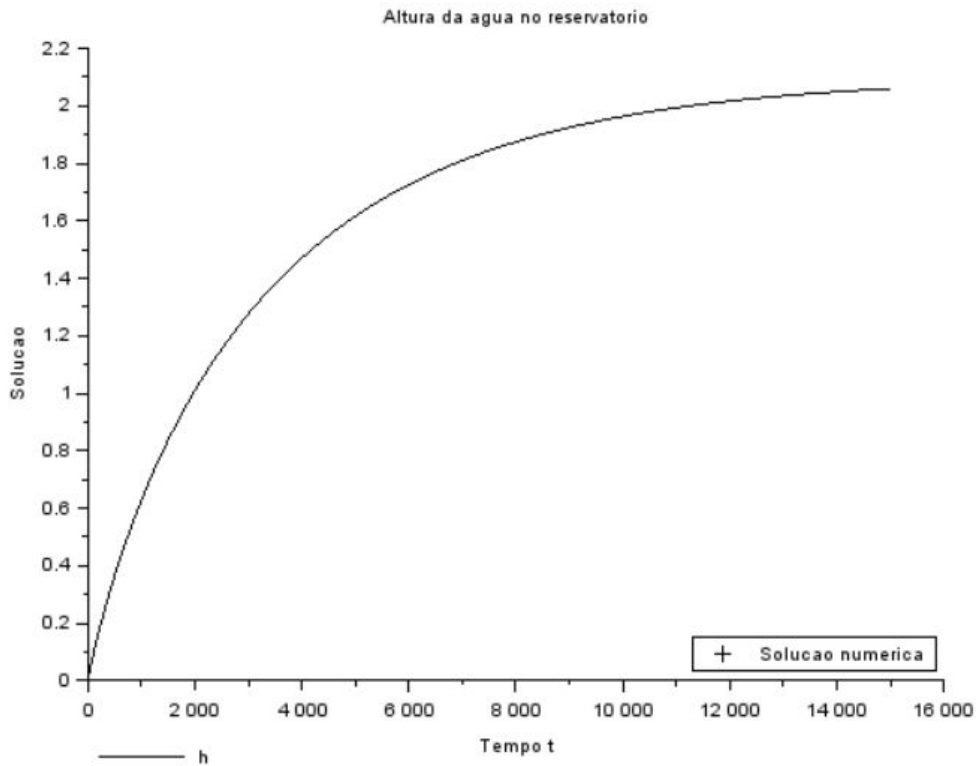


Imagem 5 - Janela gráfica 0 do exercício 1 pelo Método de Euler

4. Códigos

a. Exercício 1

i. Euler

//Parâmetros:

S = 10 // área da seção transversal (constante) em m²

*R = 2*10⁸ // relaciona vazão com queda de pressão (perda de carga) em Pa/(m³/s)²*

Rho = 1000 // massa específica da água em kg/m³

g = 10 // aceleração da gravidade na superfície da terra m/s²

Qe = 0.010247 // vazão de entrada (constante) em m³/s

function [h_dot]=funcao(h)

*h_dot = (-sqrt(Rho*g*h/R)+Qe)/S; // Qs = sqrt(Rho*G*h/R)*

endfunction

```

t(1)=0; // Instante inicial
t_f=15000; // Instante final
delta_t=0.1; // Passo de integracao
n=round((t_f-t(1))/delta_t); // Calculo do numero de passos

h(1)=0; // Condicao inicial

// Integracao numerica pelo metodo de Euler:

for i=1:n
    t(i+1)=t(i)+delta_t; // Vetor de tempo
    h(i+1)=h(i)+delta_t*funcao(h(i)); // Solucao numerica
end

plot2d(t,h',leg="h");
legends(["Solucao numerica"],-1,4)
xlabel("Altura da agua no reservatorio","Tempo t","Solucao")

```

ii. Runge Kutta

```

clear;
//Parâmetros:
S = 10 // área da seção transversal (constante) em m^2
R = 2*10^8 // relaciona vazão com queda de pressão (perda de carga) em Pa/(m^3/s)^2
Rho = 1000 // massa específica da água em kg/m^3
g = 10 // aceleração da gravidade na superfície da terra m/s^2

//Variáveis:
Qe = 0.010247 // vazão de entrada (constante) em m^3/s
//h: nível do reservatório [m]
//V: volume de água no reservatório [m^3]
//P: pressão relativa à atmosférica, no fundo do reservatório [Pa]
//Qs: vazão de saída [m^3/s]

function [h_dot]=funcao(h)
h_dot = (-sqrt(Rho*g*h/R)+Qe)/S; // Qs = sqrt(Rho*G*h/R)
endfunction

t(1)=0; // Instante inicial
t_f=15000; // Instante final
delta_t=0.1; // Passo de integracao
n=round((t_f-t(1))/delta_t); // Calculo do numero de passos

h(1)=0; // Condicao inicial

```

// Integracao numerica pelo metodo de Euler:

```
for i=1:n
    t(i+1)=t(i)+delta_t; // Vetor de tempo
    k1=funcao((h(i)));
84 k2=funcao((h(i)+delta_t*k1/2));
85 k3=funcao((h(i)+delta_t*k2/2));
86 k4=funcao((h(i)+delta_t*k3));
87 h(i+1)=h(i)+((k1+2*k2+2*k3+k4)/6);
end
```

Códigos

Exemplo 1

```
clear
```

```
function [ydot]=funcao(y)  
ydot=(1-y)/2;  
endfunction
```

```
t(1)=0; // Instante inicial  
tf=10; // Instante final  
h=0.5; // Passo de integracao  
n=round(tf/h); // Calculo do numero de passos
```

```
y(1)=0; // Condicao inicial  
ye(1)=0; // Valor inicial da solucao exata
```

```
// Integracao numerica pelo metodo de Euler:
```

```
for i=1:n  
    t(i+1)=t(i)+h; // Vetor de tempo  
    y(i+1)=y(i)+h* funcao(y(i)); // Solucao numerica  
    ye(i+1)=1-%e-(t(i+1)/2); // Solucao exata  
end
```

```
plot2d([t,t],[y,ye],[-1 -2]);  
legends(["Solucao numerica","Solucao exata"],[-1,-2],4)  
xtitle("Comparacao entre solucao numerica e solucao exata","Tempo  
t","Solucao")
```

```
set("current_figure",1);
```

```
plot2d([t,t],[y,ye],[1 2]);
```

```
T=list("Comparacao entre solucao numerica e solucao exata","Tempo  
t","Solucao","Solucao numerica","Solucao exata");
```

```
legends([T(4),T(5)],[1,2],2);
```

```
xtitle(T(1),T(2),T(3));
```