



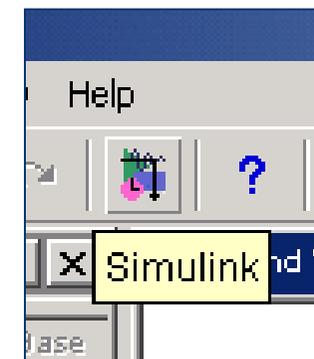
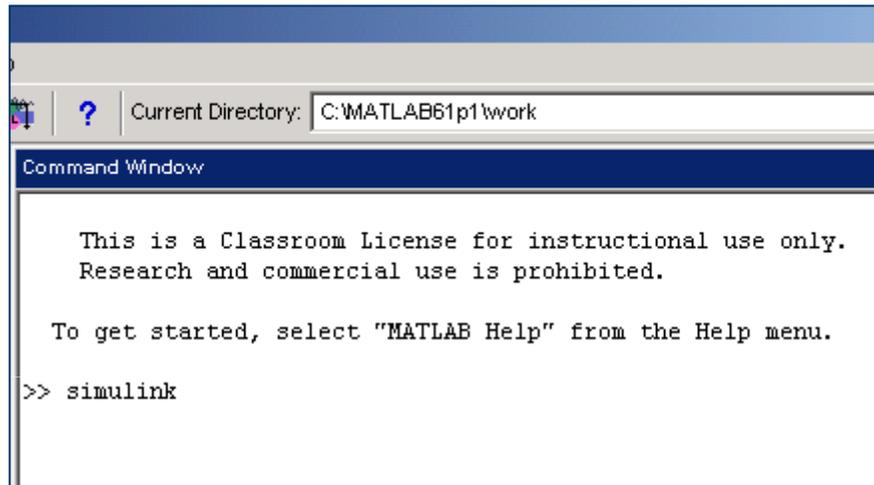
Simulink

Carlos André Vaz Junior

cavazjunior@gmail.com

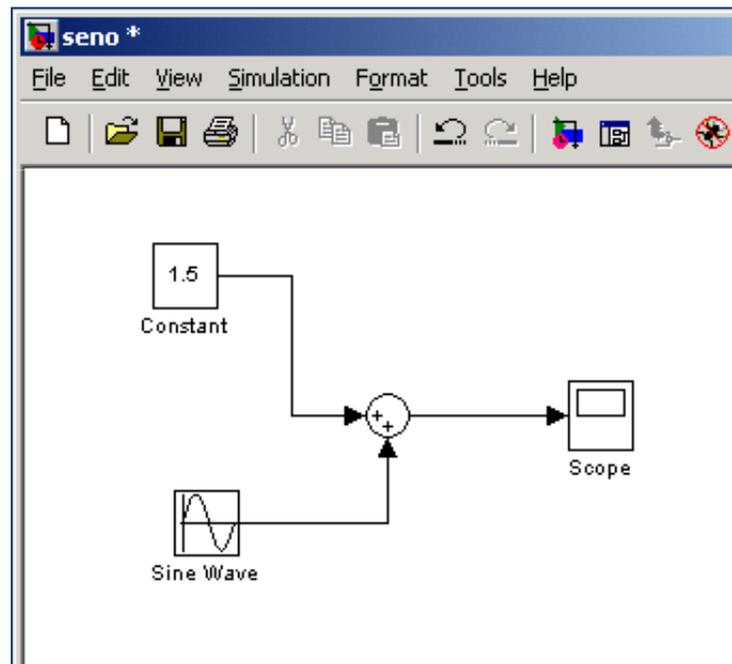
<http://www.eq.ufrj.br/links/h2cin/carlosandre>

Acessando o Simulink



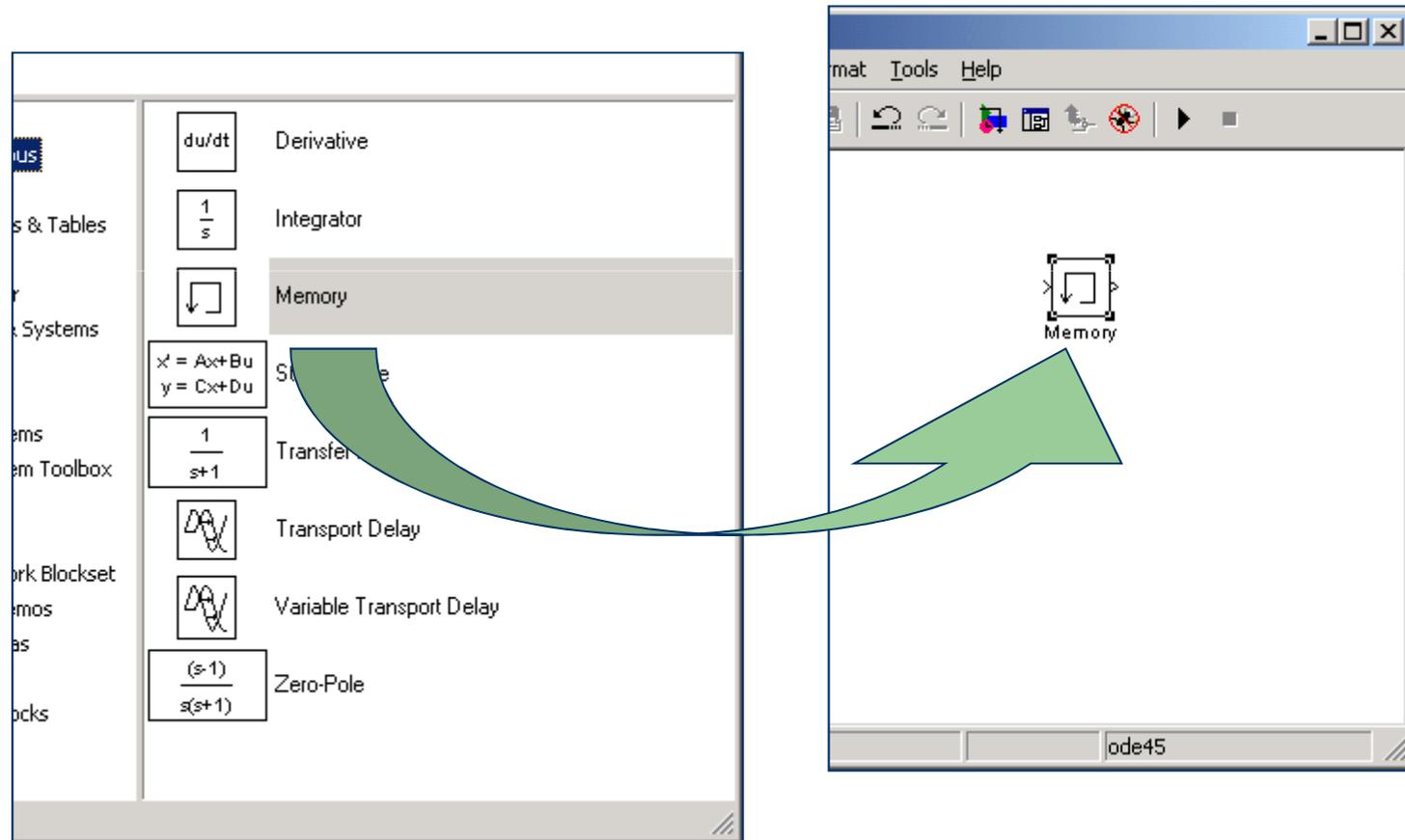


A programação no Simulink segue uma interface gráfica muito mais intuitiva e fácil de usar:





Ambiente de Trabalho Simulink





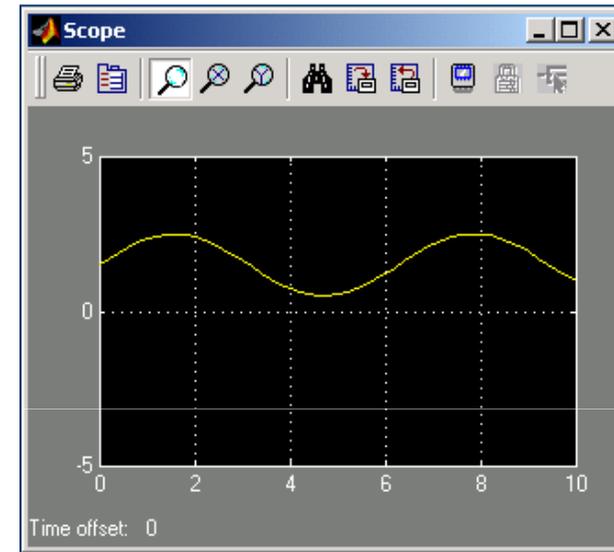
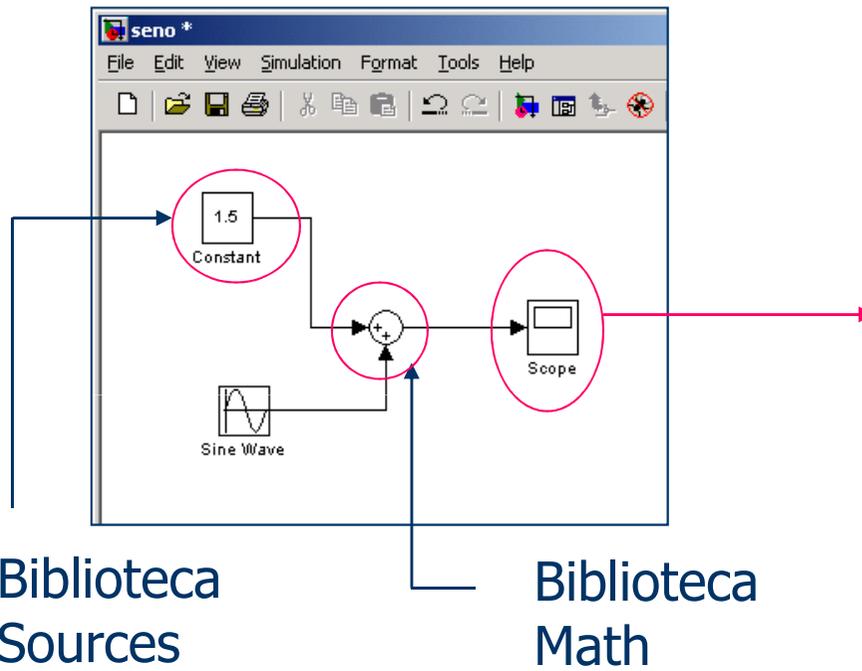
Exemplos



Exemplo

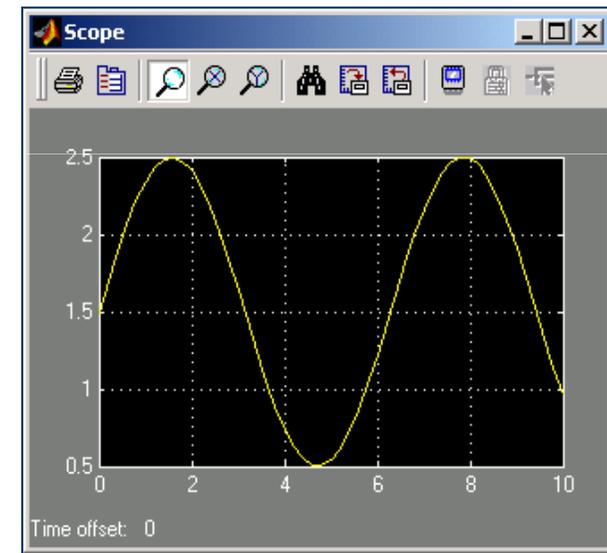
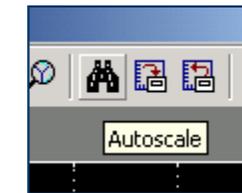
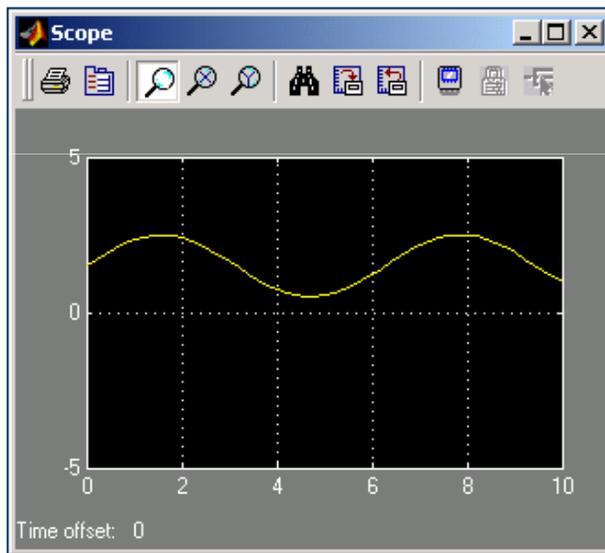
1

Exemplo 1 – Comportamento Senoidal



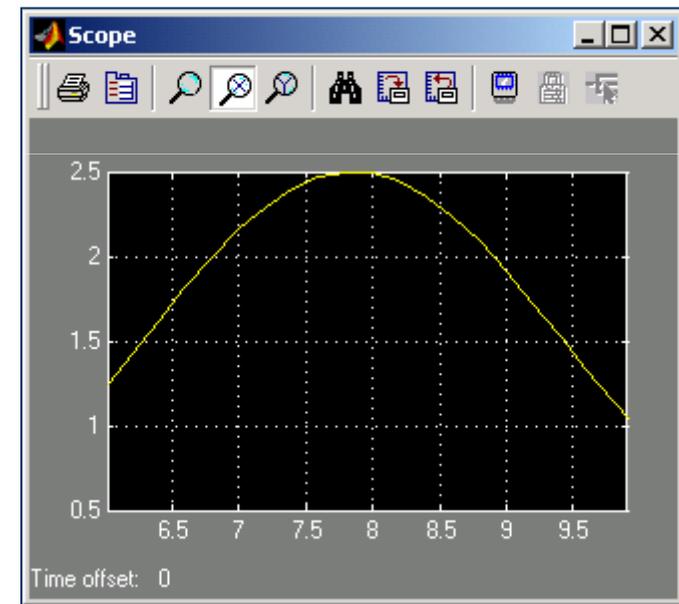
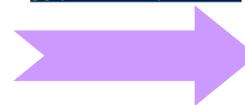
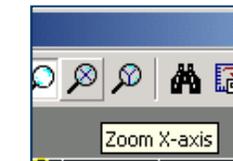
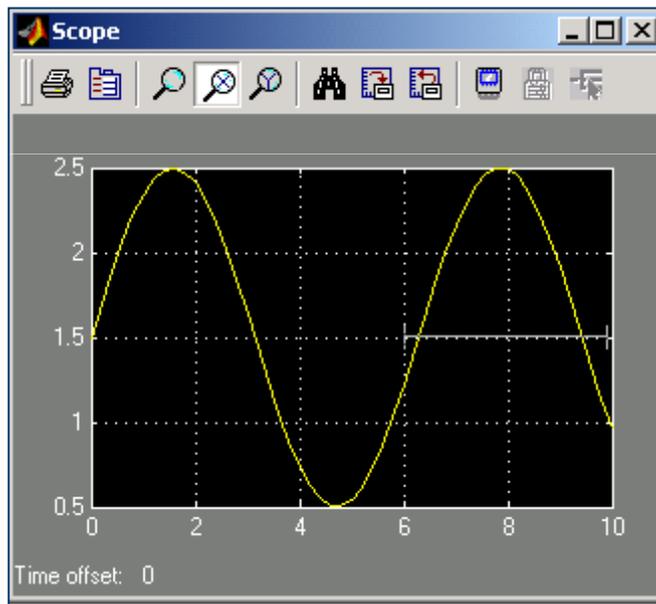


Ajuste automático da escala do gráfico:





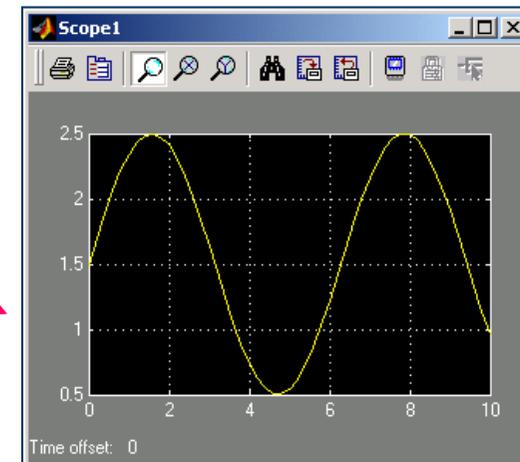
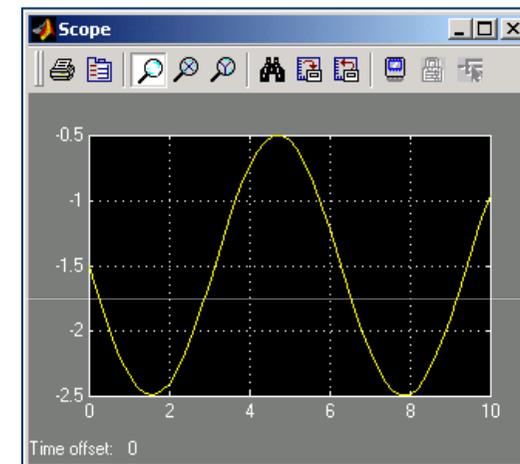
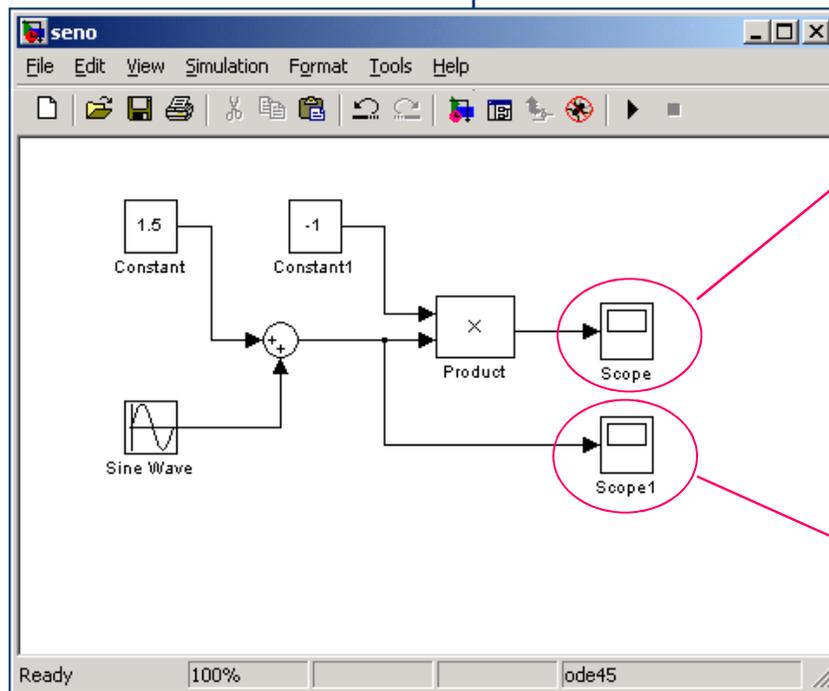
Ajuste manual da escala do gráfico:





Agora quero multiplicar o resultado por -1:

Biblioteca
Math

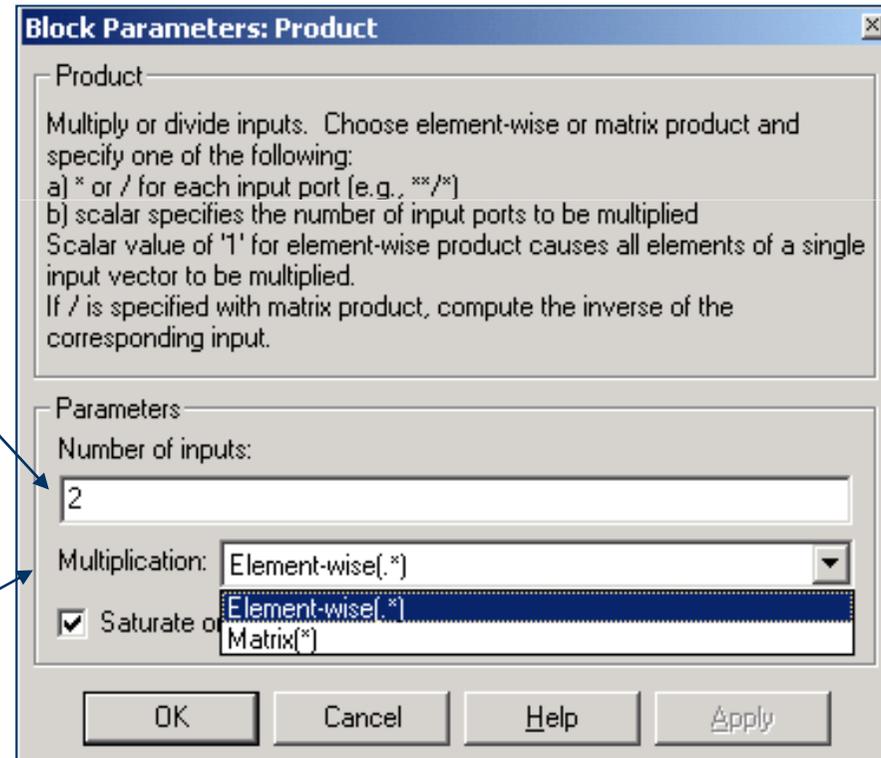




Configuração do bloco Product:

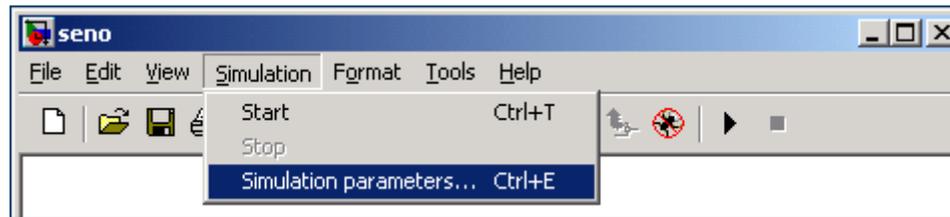
Número de termos da multiplicação.

Multiplicação de matrizes ou termo a termo.



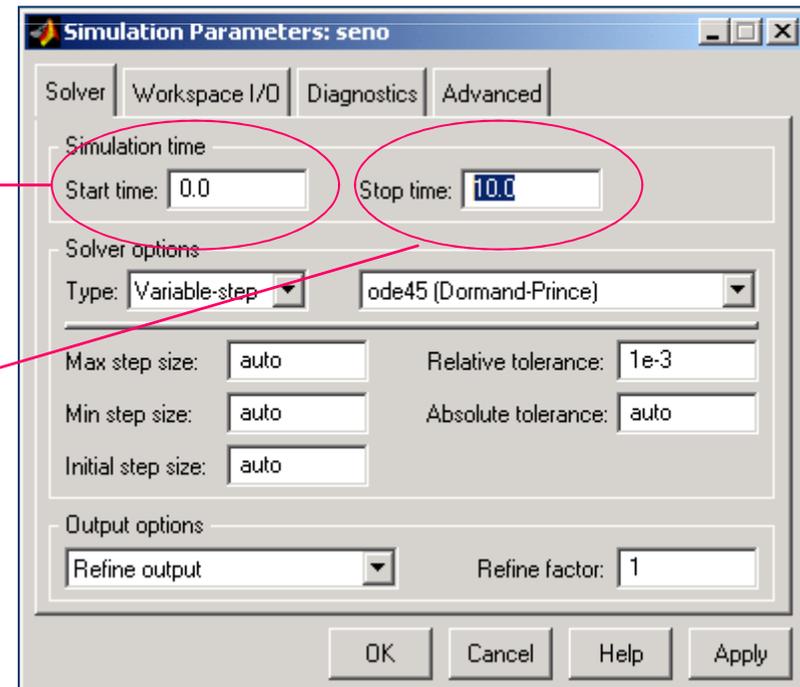


Alterando os parâmetros de simulação:



Tempo inicial

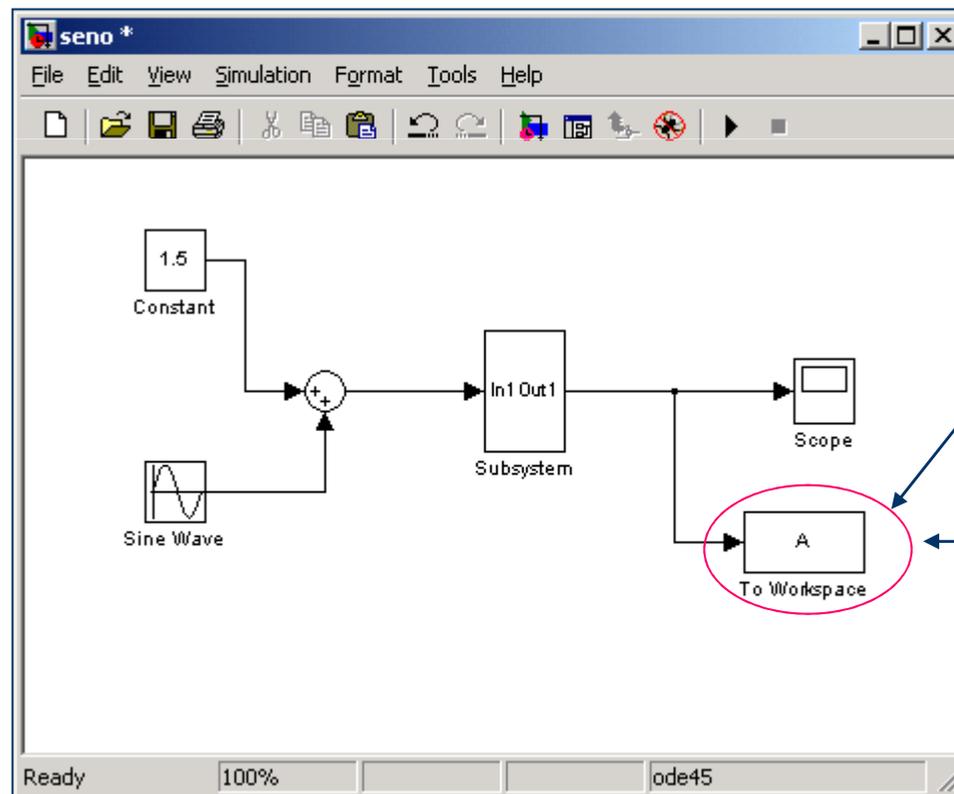
Tempo final





Algumas vezes é mais fácil tratar os dados gerados no ambiente Matlab.

Usamos o bloco “to workspace”:

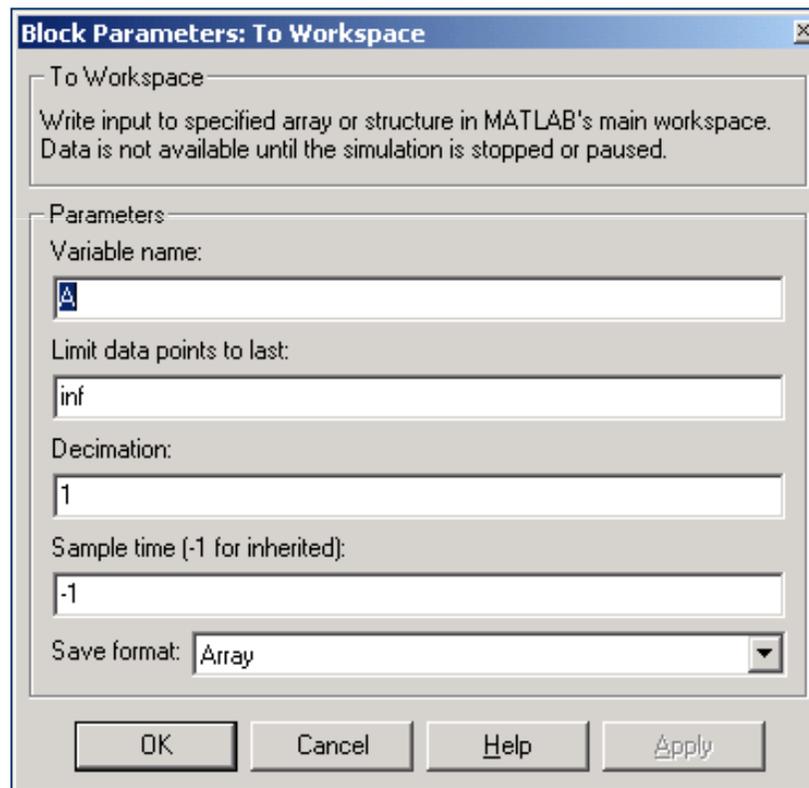


Biblioteca Sinks

Cria a variável A no workspace



Configuração do bloco "To Workspace":

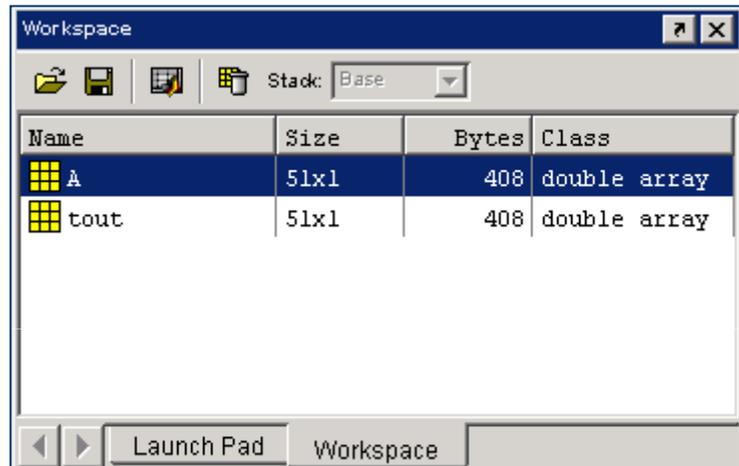


Cria a variável
A no workspace

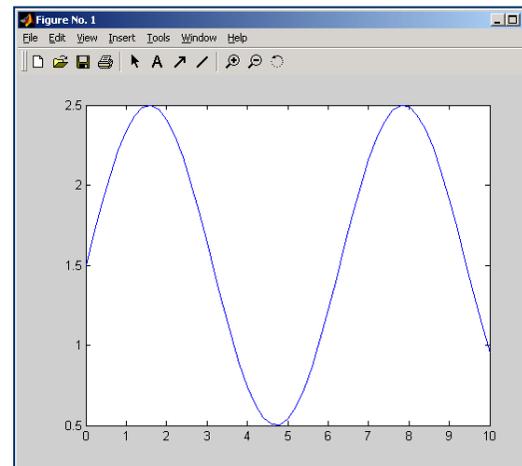
Formato da variável



No Workspace...

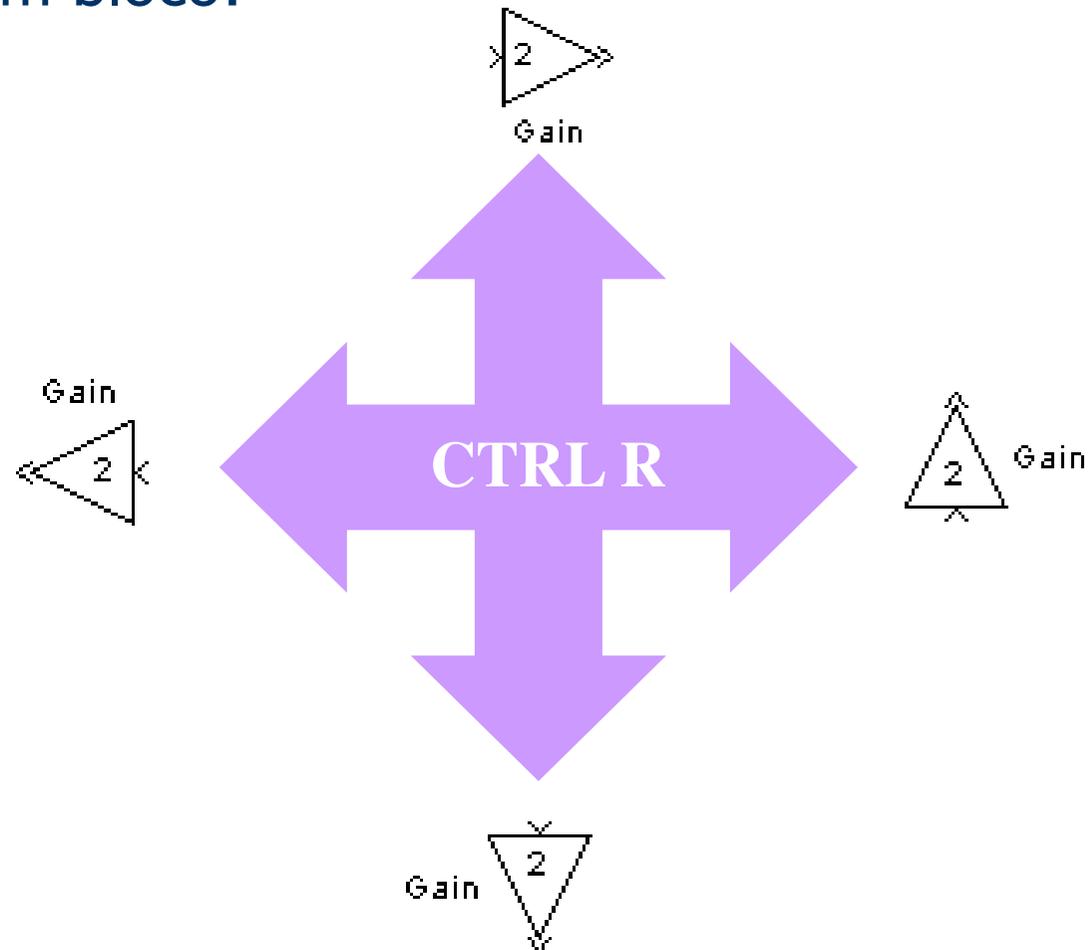


```
>> plot(tout,A)
```





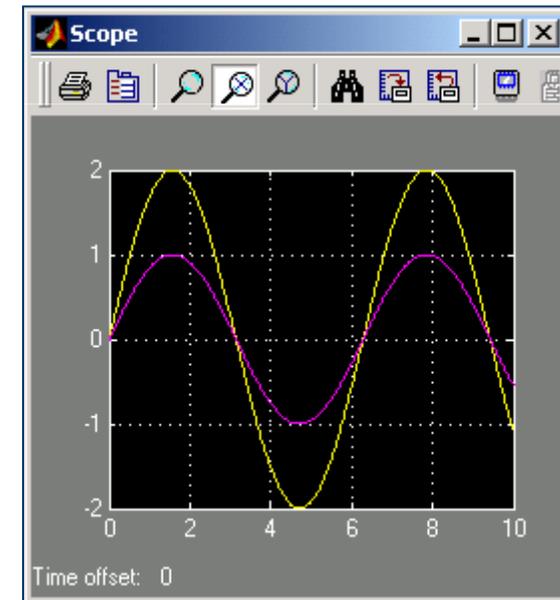
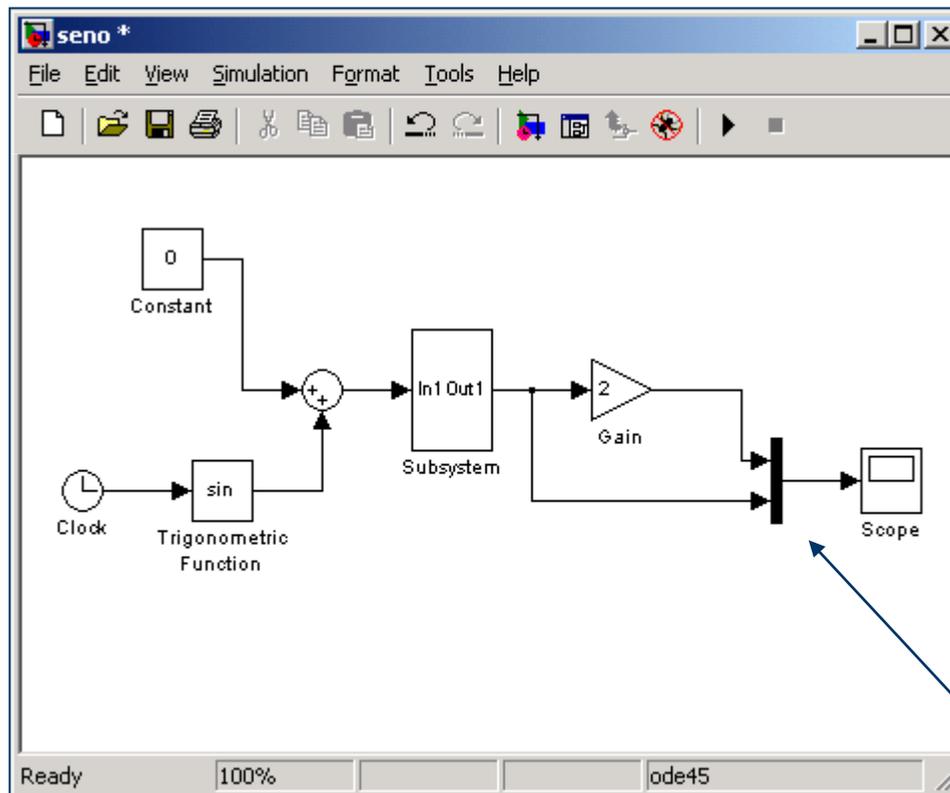
Rodando um bloco:





Combinando dois sinais:

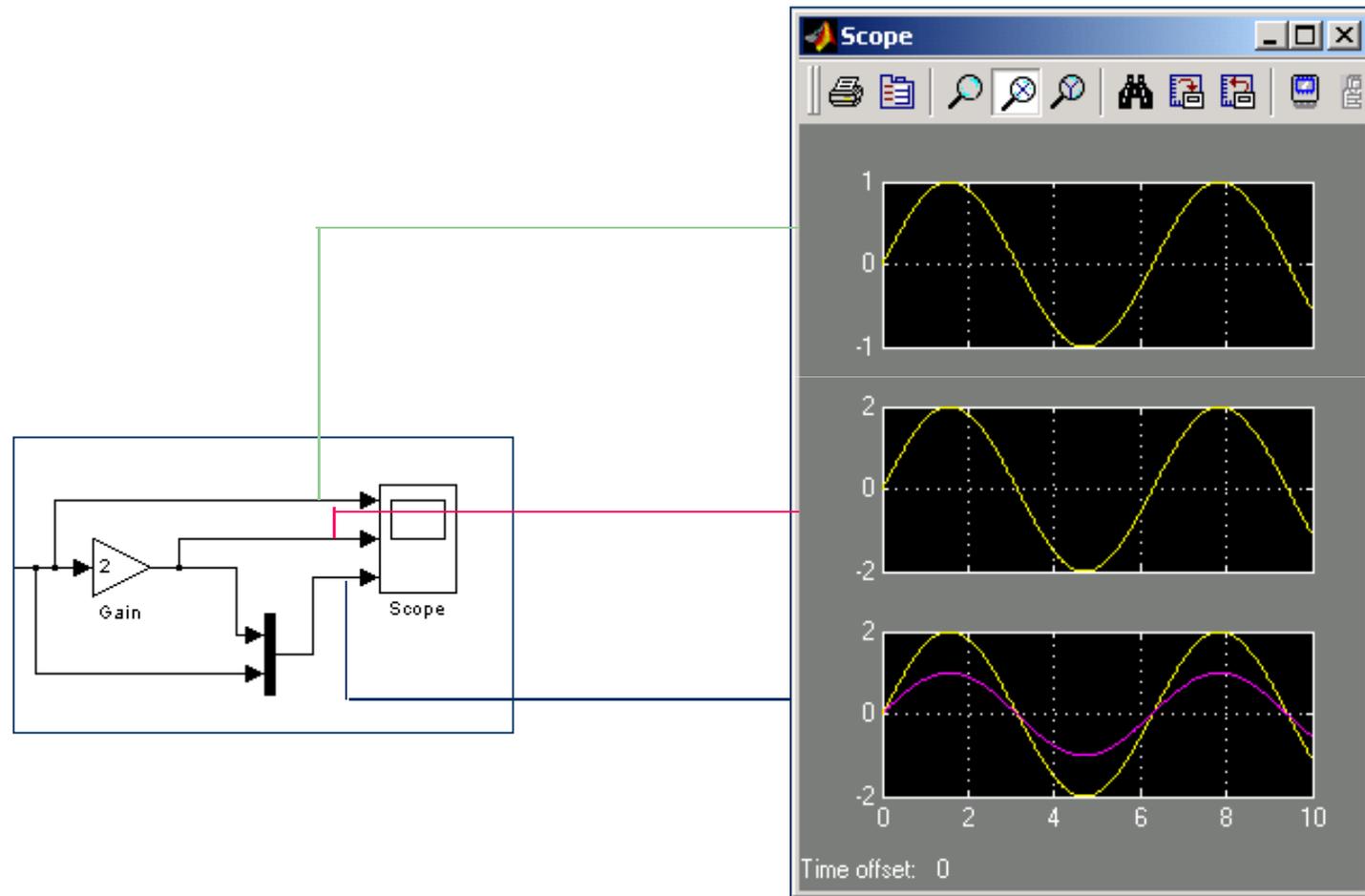
Entre outras aplicações, permite exibir duas ou mais curvas no mesmo gráfico.



Bloco MUX
Biblioteca Signals & Sys.



Dois ou mais gráficos:





Dois ou mais gráficos:
Configurando...

The diagram illustrates the configuration of a Scope component. On the left, a Scope component is circled in red, with three arrows pointing to it. A 'Scope parameters' dialog box is open, showing settings for 3 axes, auto time range, and decimation of 1. A large blue arrow points from the dialog to a screenshot of the Scope window displaying three stacked sine wave plots.

The 'Scope parameters' dialog box shows the following settings:

- General: Tip: try right clicking on axes
- Axes: Number of axes: 3, floating scope:
- Time range: auto
- Tick labels: bottom axis only
- Sampling: Decimation: 1

The Scope window displays three stacked plots of sine waves. The top plot shows a single yellow sine wave. The middle plot shows a single yellow sine wave. The bottom plot shows two sine waves, one yellow and one purple. The x-axis is labeled from 0 to 10, and the y-axis is labeled from -2 to 2. The time offset is 0.

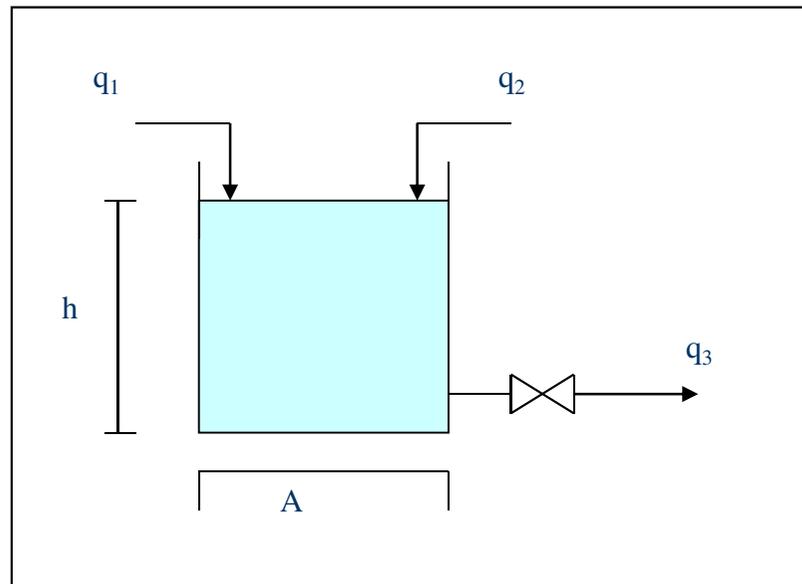


Exemplo

2



Temos a simulação de um tanque de nível sob a influência de uma perturbação degrau na vazão da alimentação. A figura descreve o sistema físico que será simulado.





Deduzindo o modelo matemático que descreve o tanque:

Assumindo que:

- a densidade do líquido e a área da seção transversal do tanque A são constantes.
- a relação entre a vazão e a carga é linear:

$$q_3 = h / R$$



O modelo é descrito por uma equação de balanço transiente de massa no tanque:

$$\rho A \frac{dh}{dt} = \rho q_1 + \rho q_2 - \rho q_3$$

Substituindo a hipótese ii na equação anterior ficamos com:

$$\rho A \frac{dh}{dt} = \rho q_1 + \rho q_2 - \rho \frac{h}{R}$$



Introduzindo as variáveis-desvio e aplicando a Transformada de Laplace, chegamos as funções de transferência:

$$\frac{h'(s)}{q_1'(s)} = G_1(s) = \frac{K_p}{\tau s + 1} \quad \leftarrow$$

$$\frac{h'(s)}{q_2'(s)} = G_2(s) = \frac{K_p}{\tau s + 1} \quad \leftarrow$$

onde:

$$K_p = R$$

$$\tau = AR$$



Para o exemplo em questão considere um tanque de 0.5 m de diâmetro e uma válvula na saída na linha atuando sob uma resistência linear (R) de 6.37 min/m².

Serão simulados um degrau de 1 ft³ na vazão q₁ a partir do tempo igual a 0 min (step) e um degrau de 1 ft³ na vazão q₂ a partir do tempo igual a 10 min(step1).

$$A = 3.1415 * (0.5/2)^2$$

$$A = 0.196$$

$$R = 6.37$$

$$K_p = R = 6.37$$

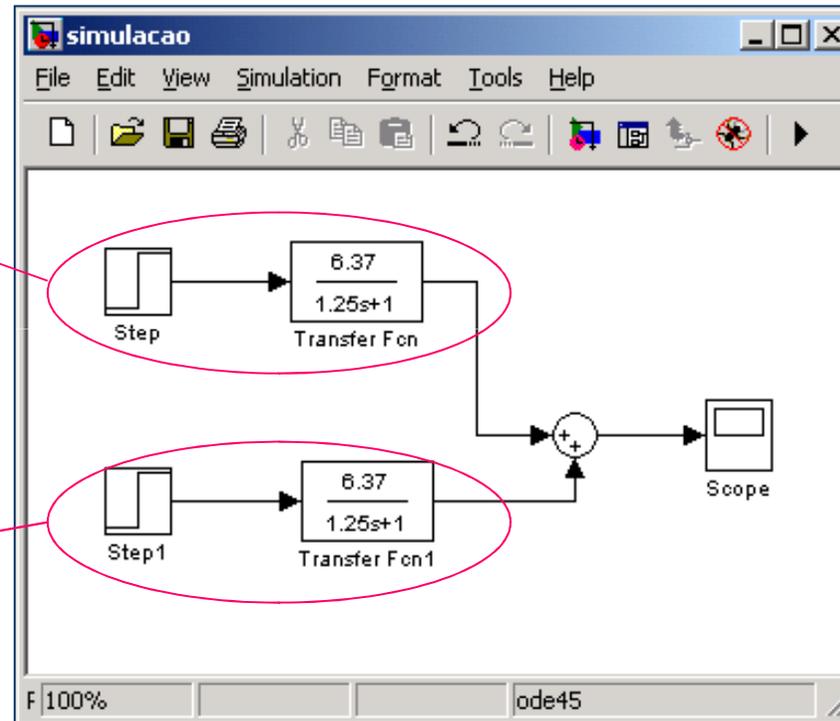
$$\tau = AR = 1.25$$

Exemplo 2 – Simulação de um modelo dinâmico



Corrente q1

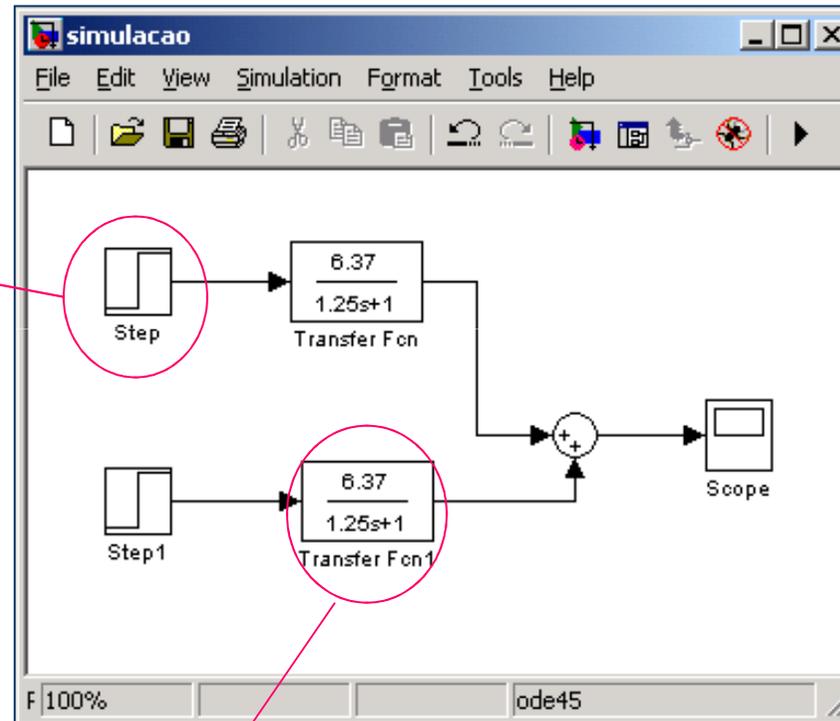
Corrente q2



Exemplo 2 – Simulação de um modelo dinâmico



Biblioteca
Source



Biblioteca
Continuous

Exemplo 2 – Simulação de um modelo dinâmico



Block Parameters: Step

Step
Output a step.

Parameters
Step time:
0

Initial value:
0

Final value:
1

Sample time:
0

Interpret vector parameters as 1-D

OK Cancel Help Apply

Degrau começa
no tempo zero



Degrau começa
No tempo dez



Block Parameters: Step1

Step
Output a step.

Parameters
Step time:
10

Initial value:
0

Final value:
1

Sample time:
0

Interpret vector parameters as 1-D

OK Cancel Help Apply

Bloco Função
de Transferência



Block Parameters: Transfer Fcn

Transfer Fcn
Matrix expression for numerator, vector expression for denominator.
Output width equals the number of rows in the numerator. Coefficients are
for descending powers of s.

Parameters
Numerator:
[6.37]

Denominator:
[1.25 1]

Absolute tolerance:
auto

OK Cancel Help Apply

Exemplo 2 – Simulação de um modelo dinâmico



Block Parameters: Step

Step
Output a step.

Parameters

Step time:
0

Initial value:
0

Final value:
1

Sample time:
0

Interpret vector parameters as 1-D

OK Cancel Help Apply

A amplitude do degrau é 1

Block Parameters: Step1

Step
Output a step.

Parameters

Step time:
0

Initial value:
0

Final value:
1

Sample time:
0

Interpret vector parameters as 1-D

OK Cancel Help Apply

Block Parameters: Transfer Fcn

Transfer Fcn
Matrix expression for numerator, vector expression for denominator.
Output width equals the number of rows in the numerator. Coefficients are for descending powers of s.

Parameters

Numerator:
[6.37]

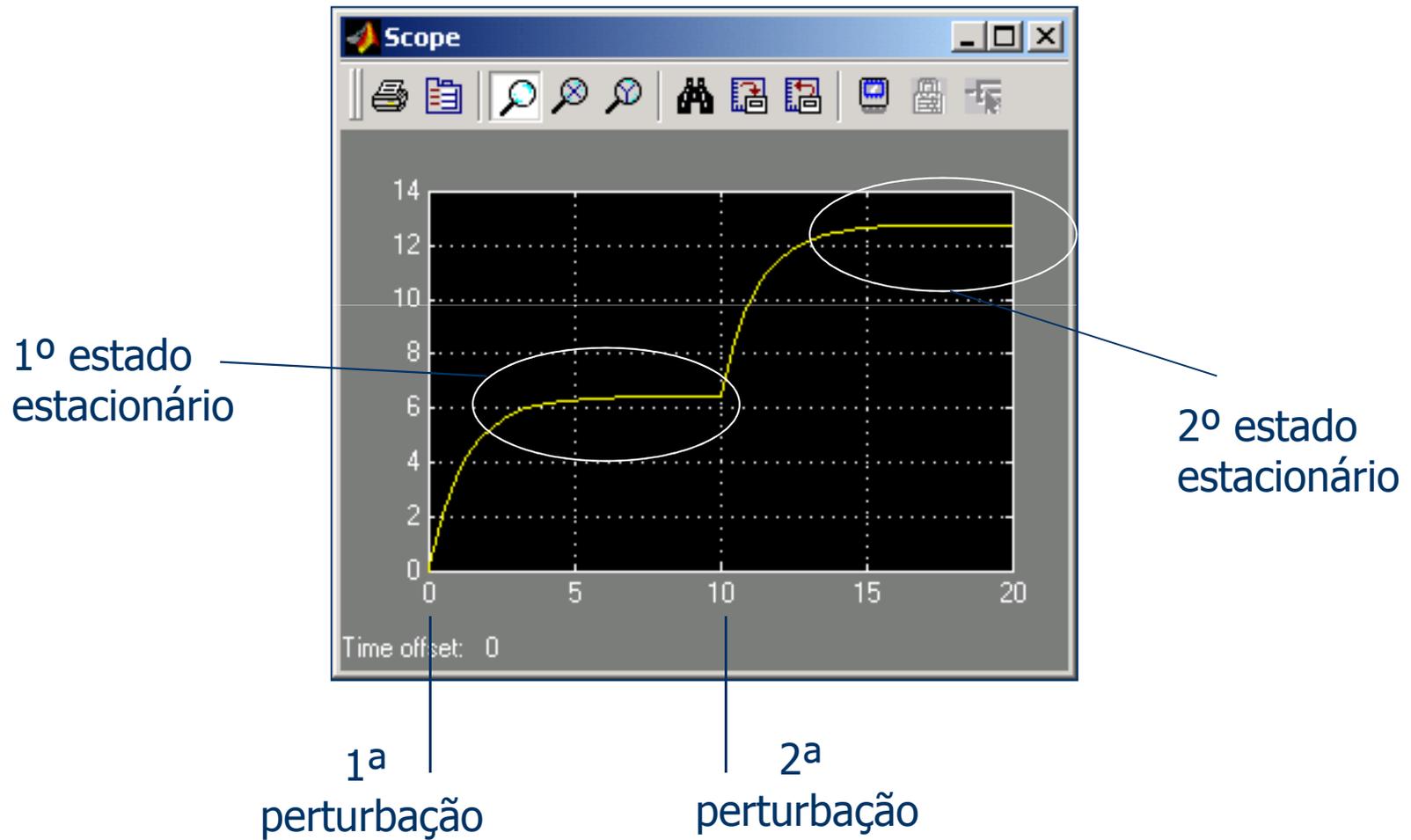
Denominator:
[1.25 1]

Absolute tolerance:
auto

OK Cancel Help Apply



Resultado obtido:





Exemplo

3



Equações para modelar um CSTR:

$$\frac{dV}{dt} = F^A - F$$

$$\frac{dC_A}{dt} = \frac{F^A}{V} (C_A^A - C_A) - k_0 e^{-\frac{E}{RT}} C_A$$

$$\frac{dT}{dt} = \frac{\rho F^A C_P (T^A - T) + \Delta H V k_0 e^{-\frac{E}{RT}} C_A - UA(T - T_c)}{\rho V C_P}$$



Passando as equações para o formato Matlab:

$$dCa = (Fi*(cai-Ca)/V) - k*Ca;$$

$$dV = Fi-F;$$

$$dT = (Fi*Cp*ro*(Ti-T) + DeltaH*k*Ca*V - U*A*(T-Tc)) / (V*ro*Cp);$$



onde:

Fi: vazão de alimentação do reator (ft³/h)

Cai: concentração da alimentação do reator (lbm/ft³)

Ca: concentração no reator (variável)

k: é dado pela equação $k = k_0 \cdot \exp(-E/(R \cdot T))$

V: volume do reator

F: vazão de saída (ft³/h)

Cp: calor específico = 0.75 btu/lbm.R

ro: densidade = 50 lb/ft³

Ti: temperatura de alimentação (R)

T: temperatura do reator

DeltaH: calor de reação = -30000 BTU/ lbm

U: coeficiente de troca térmica = 150 BTU/(h.ft².R)



onde:

A: área de troca térmica = 250 ft²

T_c: temperatura do fluido de alimentação (R)

E: energia de ativação = 30000 BTU/lbm

R: constante dos gases = 1.99 BTU/lbm.R

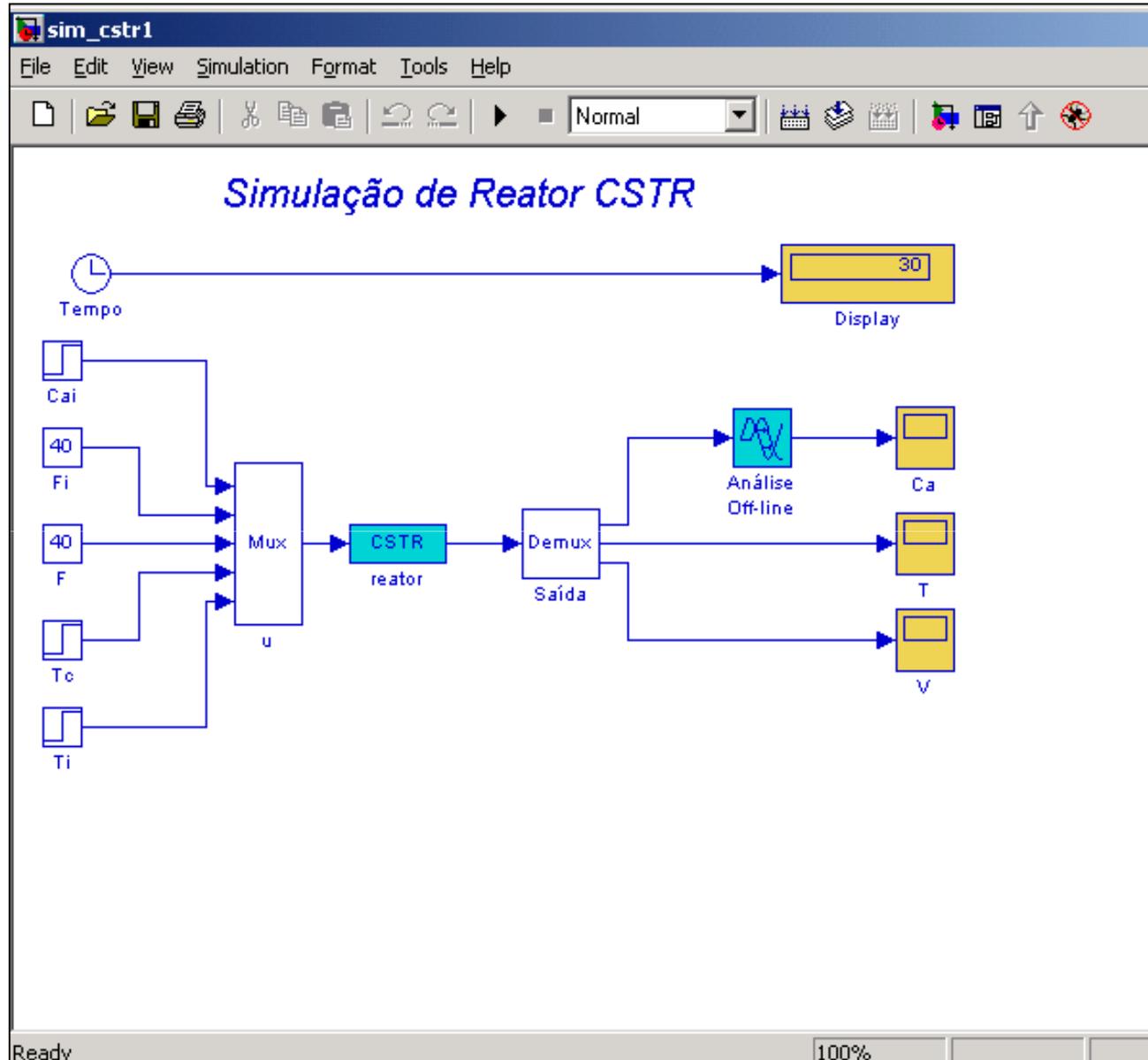
Legenda:

parâmetros freqüentemente alterados

parâmetros raramente alterados

parâmetros calculados

Exemplo 3 – Bloco S-function

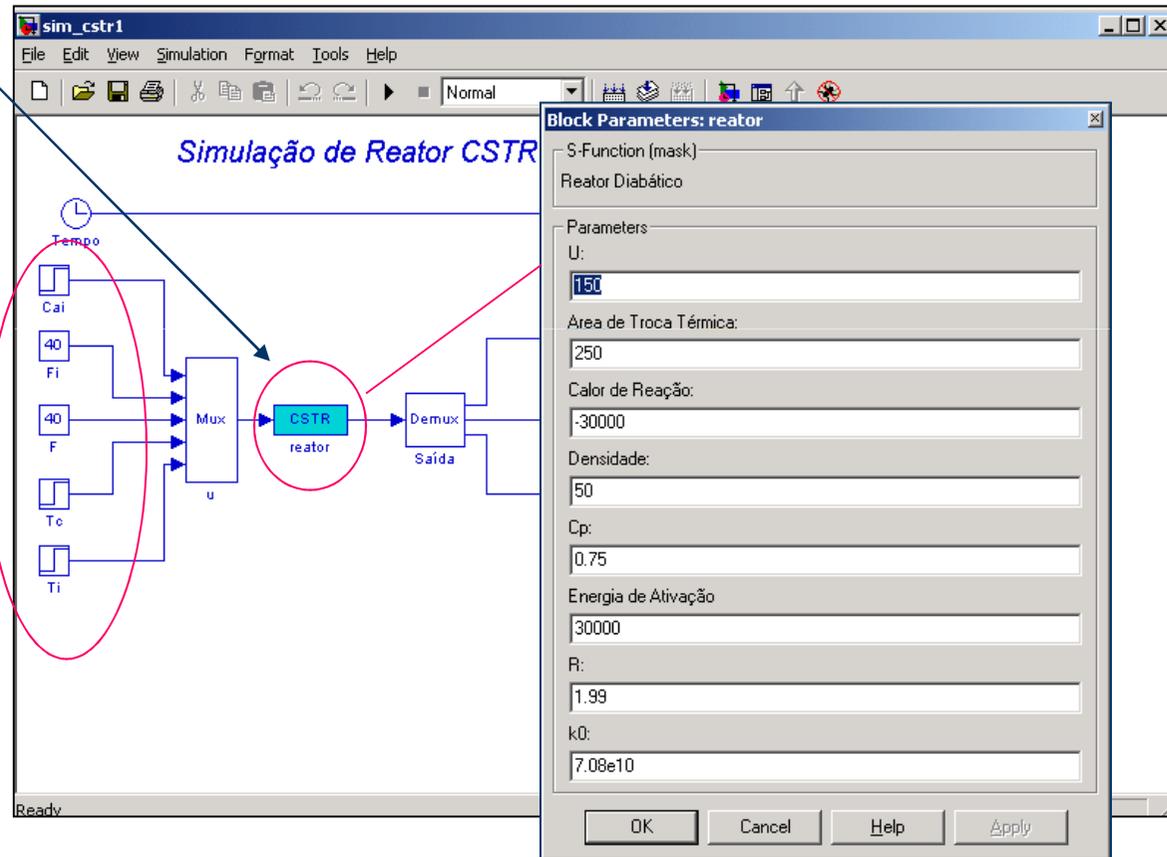


Exemplo 3 – Bloco S-function



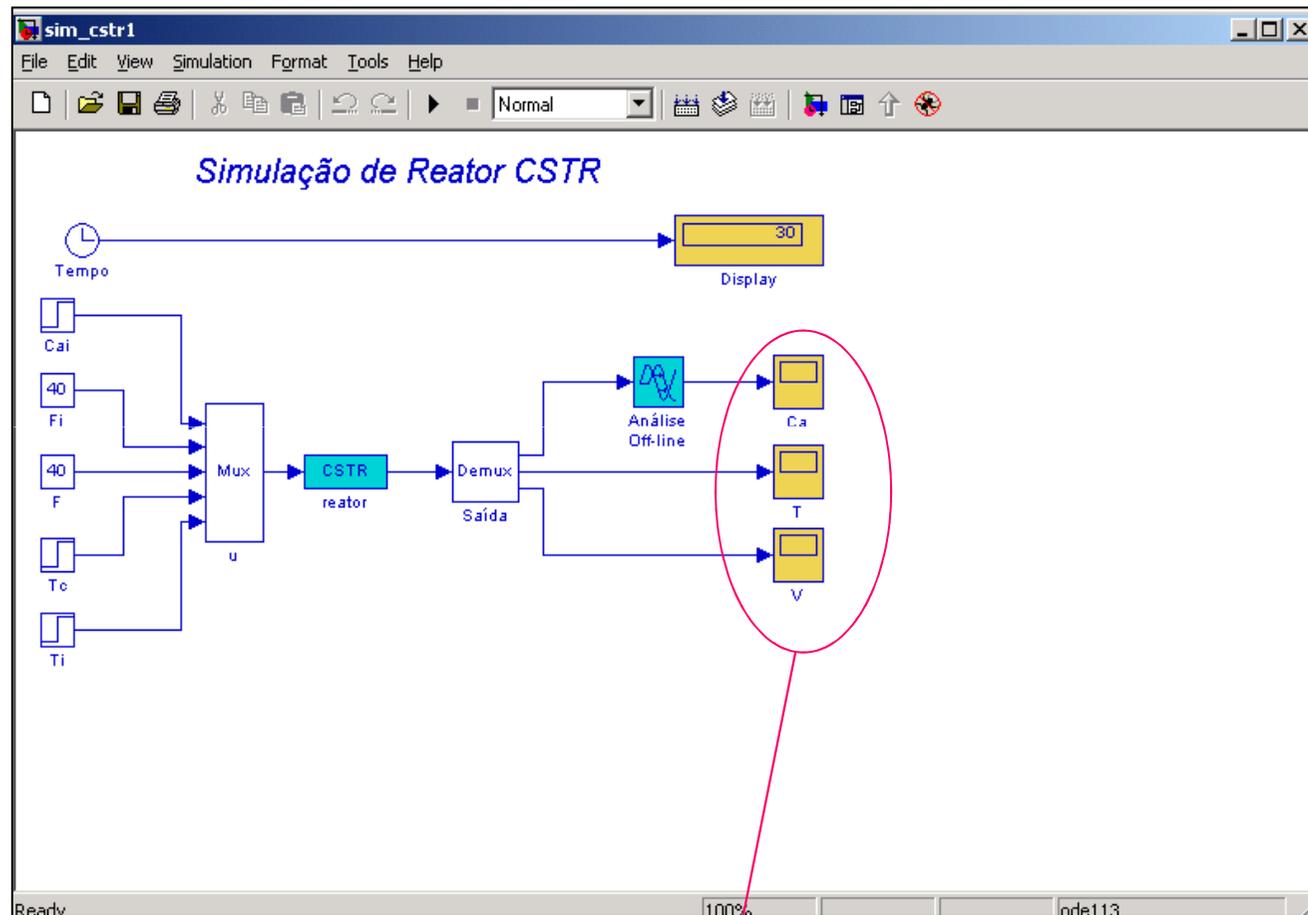
Biblioteca
Functions & Tables

Parâmetros
freqüentemente
alterados



Parâmetros raramente alterados
(máscara)

Exemplo 3 – Bloco S-function



Parâmetros calculados



Em resumo:

| | | | |
|-----|---|--------|---|
| Fi |  | T |  |
| Cai |  | DeltaH |  |
| Ca |  | U |  |
| Ko |  | A |  |
| V |  | Tc |  |
| F |  | E |  |
| Cp |  | R |  |
| ro |  | | |
| Ti |  | | |



Configurando o bloco S-function:

Block Parameters: S-Function

S-Function
User-definable block. Blocks may be written in M, C, Fortran or Ada and must conform to S-function standards. t,x,u and flag are automatically passed to the S-function by Simulink. "Extra" parameters may be specified in the 'S-function parameters' field.

Parameters
S-function name:
reator

S-function parameters:
U A DeltaH ro Cp E R k0

OK Cancel Help Apply

Nome do arquivo com as equações

Parâmetros alterados pela máscara



Criando uma máscara:

The image illustrates the process of creating a mask for an S-function block. It consists of several key components:

- Context Menu:** A menu with options: Cut, Copy, Clear, Block parameters..., Block properties..., **Mask s-function** (highlighted), Look under mask, Link options, Format, Foreground color, Background color, and Help.
- Mask Editor (Left):** A window titled "Mask Editor: sim1/S-Function" with tabs for Icon, Initialization, and Documentation. The "Mask type" field is empty. The "Drawing commands" area contains the text `disp('CSTR')`. Below this are settings for "Icon frame" (Visible), "Icon transparency" (Opaque), "Icon rotation" (Fixed), and "Drawing coordinates" (Autoscale). Buttons for OK, Cancel, Unmask, and Help are at the bottom.
- Mask Editor (Right):** The same window with the "Initialization" tab selected. It shows a table of prompts and variables:

| Prompt | Type | Variable |
|-----------------------------|-------------|----------|
| k0: | edit | k0 |
| R: | edit | R |
| Energia de Ativação: | edit | E |
| Cp: | edit | Cp |
| Densidade: | edit | ro |
| Calor de Reação: | edit | Del |

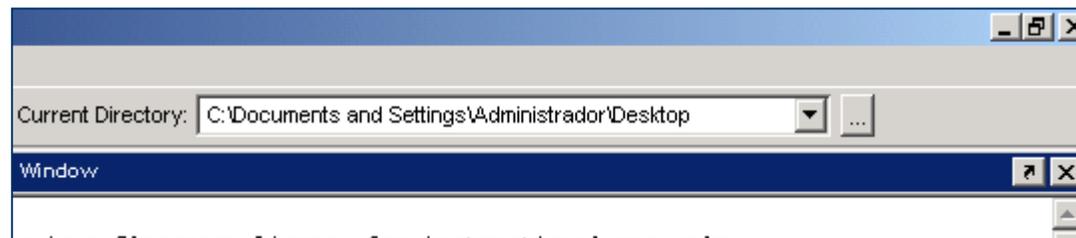
Below the table, "Prompt" is set to "Energia de Ativação:", "Control type" is "Edit", "Variable" is "E", and "Assignment" is "Evaluate". There are also fields for "Popup strings" and "Initialization commands".
- Block Parameters (Bottom Right):** A dialog titled "Block Parameters: S-Function" for the "Reator CSTR". It lists parameters with values: k0 (7.08e1), R (1.99), Energia de Ativação (30000), Cp (0.75), Densidade (50), Calor de Reação (-30000), Área de Troca Térmica (250), U (150). Buttons for OK, Cancel, Help, and Apply are at the bottom.
- Block Diagram:** A small diagram showing a block labeled "CSTR" with "S-Function" written below it, with an arrow pointing to the right.



Localização do arquivo com as equações:

O arquivo com as equações deve estar localizado no mesmo local dos arquivos Simulink!

O Current Directory do Matlab deve apontar para esse local!





Criando o arquivo com as equações:

```
function [sys,x0] = reator(t,x,u,flag,U,A,DeltaH,ro,Cp,E,R,k0)
%
% Simula um reator CSTR (mistura perfeita) no qual se conduz uma
% reação exotérmica (A->B), resfriado por serpentina

%
switch flag
    case 0 % Dimensiona o sistema e inicializa os estados
        % sys=[estados,0,saídas,entradas,0,0]
        sys = [3,0,3,5,0,0];
        % Condições iniciais
        ca = 0.1315;           %lbm/ft3, concentração inicial no reator
        T = 584.4115;         %R, temperatura do reator
        V = 200;              %ft3, volume do reator
        x0 = [ca T V]';
```



Criando o arquivo com as equações:

```
function [sys,x0] = reator(t,x,u,flag,U,A,DeltaH,ro,Cp,E,R,k0)
%
% Simula um reator CSTR (mistura perfeita) no qual se conduz uma
% reação exotérmica (A->B), resfriado por serpentina
```

- **sys** é a saída do modelo, cujo significado depende de flag
- **x0** é o vetor de condições iniciais (funciona apenas quando flag = 0)
- **t** é o tempo de simulação
- **x** é o vetor de estados do modelo
- **u** é o vetor de entradas do modelo (recebido do bloco Mux)
- **flag** é um parâmetro que informa o tipo de informação que o integrador espera receber a cada chamado
- **U,...,k0** são os parâmetros adicionais que podem ser passados à função através de uma máscara (devem estar declarados na configuração do bloco S-function).



Criando o arquivo com as equações:

```
function [sys,x0] = reator(t,x,u,flag,U,A,DeltaH,ro,Cp,E,R,k0)
%
% Simula um reator CSTR (mistura perfeita) no qual se conduz uma
% reação exotérmica (A->B), resfriado por serpentina

%
switch flag
case 0 % Dimensiona o sistema e inicializa os estados

    % sys=[estados,0,saídas,entradas,0,0]
    sys = [3,0,3,5,0,0];
    % Condições iniciais
    ca = 0.1315;           %lbm/ft3, concentração inicial no reator
    T = 584.4115;         %R, temperatura do reator
    V = 200;              %ft3, volume do reator
    x0 = [ca T V]';
```



```
sys = [ número de estados contínuos  
        número de estados discretos  
        número de saídas  
        número de entradas  
        marcador de alimentação direta  
        tempo de amostragem ]
```

```
case 0 % Dimensiona o sistema e inicializa os estados
```

```
% sys=[estados,0,saídas,entradas,0,0]
```

```
sys = [3,0,3,5,0,0];
```

```
% Condições iniciais
```

```
ca = 0.1315;           %lbm/ft3, concentração inicial no reator
```

```
T = 584.4115;         %R, temperatura do reator
```

```
V = 200;              %ft3, volume do reator
```

```
x0 = [ca T V]';
```

continua...



Estimativas iniciais para o cálculo do sistema de equações diferenciais
(cálculo numérico)

```
case 0 % Dimensiona o sistema e inicializa os estados
```

```
% sys=[estados,0,saídas,entradas,0,0]
```

```
sys = [3,0,3,5,0,0];
```

```
% Condições iniciais
```

```
ca = 0.1315; %lbm/ft3, concentração inicial no reator
```

```
T = 584.4115; %R, temperatura do reator
```

```
V = 200; %ft3, volume do reator
```

```
x0 = [ca T V]';
```

continua...



case 1 % Calcula as derivadas

```
% Atualiza entradas
cai = u(1);          %lbm/ft3, concentração da alimentação=0.5;
Fi = u(2);          %ft3/hr, vazão de alimentação=40
F = u(3);           %vazão de retirada=40
Tc = u(4);          %R, temperatura do fluido de refrigeração=594.6
Ti = u(5);          %R, temperatura da alimentação=530
% Cálculo das derivadas

Ca = x(1);
T = x(2);
V = x(3);

k = k0*exp(-E/(R*T));

dCa = (Fi*(cai-Ca)/V) - k*Ca;
dV = Fi-F;
dT = (Fi*Cp*ro*(Ti-T) + DeltaH*k*Ca*V - U*A*(T-Tc)) / (V*ro*Cp);

sys = [dCa; dT; dV];
```

Exemplo 3 – Bloco S-function



case 1 % Calcula as derivadas

% Atualiza entradas

cai = u(1);

Fi = u(2);

F = u(3);

Tc = u(4);

Ti = u(5);

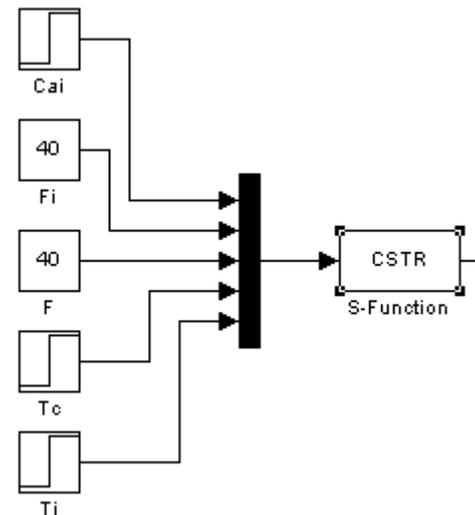
%lbm/ft³, concentração da alimentação=0.5;

%ft³/hr, vazão de alimentação=40

%vazão de retirada=40

%R, temperatura do fluido de refrigeração=594.6

%R, temperatura da alimentação=530





case 1 % Calcula as derivadas

```
% Atualiza entradas
cai = u(1);          %lbm/ft3, concentração da alimentação=0.5;
Fi = u(2);          %ft3/hr, vazão de alimentação=40
F = u(3);           %vazão de retirada=40
Tc = u(4);          %R, temperatura do fluido de refrigeração=594.6
Ti = u(5);          %R, temperatura da alimentação=530
% Cálculo das derivadas

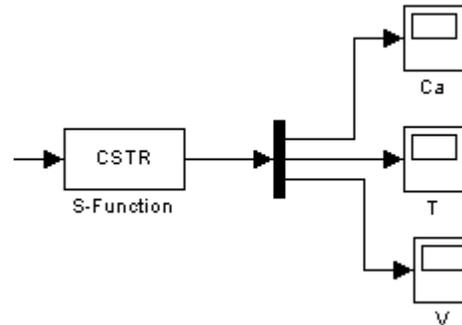
Ca = x(1);
T = x(2);
V = x(3);

k = k0*exp(-E/(R*T));

dCa = (Fi*(cai-Ca)/V) - k*Ca;
dV = Fi-F;
dT = (Fi*Cp*ro*(Ti-T) + DeltaH*k*Ca*V - U*A*(T-Tc)) / (V*ro*Cp);

sys = [dCa; dT; dV];
```

Exemplo 3 – Bloco S-function



% Cálculo das derivadas

$$Ca = x(1);$$

$$T = x(2);$$

$$V = x(3);$$

$$k = k_0 \cdot \exp(-E/(R \cdot T));$$

$$dCa = (F_i \cdot (c_{ai} - Ca) / V) - k \cdot Ca;$$

$$dV = F_i - F;$$

$$dT = (F_i \cdot C_p \cdot r_o \cdot (T_i - T) + \Delta H \cdot k \cdot Ca \cdot V - U \cdot A \cdot (T - T_c)) / (V \cdot r_o \cdot C_p);$$

$$\text{sys} = [dCa; dT; dV];$$



```
case 3 % Calcula as saídas
    sys = [x(1) x(2) x(3)];
otherwise
    sys = [];
end
```



Simulink

Carlos André Vaz Junior

cavazjunior@gmail.com

<http://www.eq.ufrj.br/links/h2cin/carlosandre>