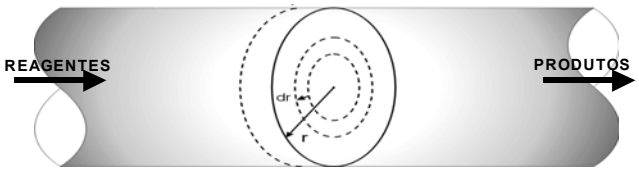
AINST

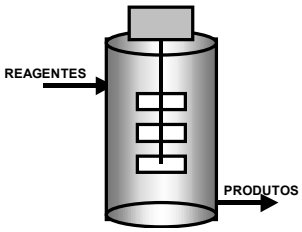
Laboratório de Controle e Otimização Industrial
LACOI
TREINAMENTO E CONSULTORIA
Atendimento: [71] 9984-3316
email: lacoi@ufba.br

Curso de Extensão:

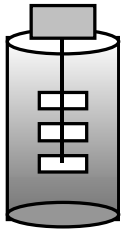
PROGRAMAÇÃO EM MATLAB COM APLICAÇÃO EM REATORES QUÍMICOS



REATOR TUBULAR PFR



CSTR



BATELADA

Prof. Dr. Ricardo de Araújo Kalid

Departamento de Engenharia Química da UFBA

Escola Politécnica da UFBA

UFBA

Laboratório de Controle e Otimização Industrial
LACOI

PROGRAMAÇÃO EM MATLAB COM APLICAÇÃO EM REATORES QUÍMICOS

ÍNDICE

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Laboratório de Controle e Otimização Industrial
LACOI

CURRÍCULO DO INSTRUTOR

Ricardo de Araújo Kalid, D. Sc. 04/09/64
kalid@ufba.br
(0xx71) 247.5123 / 9984.3316
Prof. Deptº Engenharia Química da UFBA
Graduação em Engenharia Química – UFBA (88)
Mestrado em Engenharia Química - UFBA (91)
Doutorado em Engenharia Química – USP (99)

ÁREAS DE ATUAÇÃO E LINHAS DE PESQUISA

Simulação em Regime Estacionário e Transiente de Processos
Identificação de Processos
Controle de Processos
Otimização de Processos
Simulação, Controle e Otimização de Reatores e Colunas de Destilação

OUTROS

Professor do Mestrado em Engenharia Química da UFBA
Professor (anos 92 e 93) do Curso de Especialização em Instrumentação e Controle (CEINST) promovido pelo Departamento de Engenharia Mecânica da UFBA
Professor de Cursos de Educação Continuada (Controle Avançado, Controle Preditivo Multivariável, Identificação de Processos, Otimização de Processos Químicos, Controle de Colunas de Destilação) para DOW, PETROBRAS, GRIFFIN, EDN, CIQUINE, OXITENO, COPENE.
Professor (98) do Curso de Especialização em Automação de Sistemas Industriais (CEASI) promovido pelo Deptº de Engenharia Elétrica da UFBA
Professor e Coordenador (99) do Curso de Especialização em Controle e Automação de Processos Industriais (CECAPI) promovido pelos Deptº de Engenharia Química e Elétrica da UFBA
Professor e Coordenador (2000) do Curso de Especialização em Instrumentação, Automação, Controle e Otimização de Processos Contínuos (CICOP) promovido pelo Deptº de Engenharia Química e UFBA e AINST.

PROJETOS COOPERATIVOS E/OU CONSULTORIAS PARA INDÚSTRIAS

DETEN: simulação do reator radial para desidrogenação de parafinas
EDN: participou da equipe de desenvolvimento do plano diretor de automação
COPENE: identificação de processos, sintonia de controladores industriais, simulação, controle e otimização do conversor de acetileno da ETENO II (em andamento)

PDAI-BA - Programa de Desenvolvimento da Automação Industrial, participantes: UFBA, UNIFACS, CEFET-BA, CETIND-SENAI, FIEB, SEPLANTEC, PETROBRAS, NITROCARBONO, DETEN, OXITENO, OPP, POLIBRASIL, POLITENO, COPENE

PROCESSAMENTO DE POLÍMEROS, participantes: UFBA, CETIND, BAPLASTIL
GRIFFIN: Sistema de controle de pH. Modelagem e Otimização do Reator de DCA

INDICADORES DE PRODUÇÃO CIENTÍFICA

Trabalhos Apresentados em Congressos:	8
Trabalhos Publicados em Periódicos:	2
Tese de Mestrado Defendida e Aprovada:	1
Tese de Doutorado Defendida e Aprovada:	1
Orientação de Iniciação Científica:	15 (concluídas) e 3 em andamento
Orientação de Dissertações de Mestrado:	5 (em andamento)

Trabalhos em parceria com indústrias:

N	Tema	Participantes	Instituição
01	Modelagem, simulação do reator de desidrogenação de n-parafinas da DETEN	Gian Carlo Gangemi	DETEN
		Ricardo Kalid	UFBA
02	Modelagem por redes neurais híbridas e otimização de reatores de CPD	Tatiana Freitas	UFBA
		Luiz Alberto Falcon	COPENE
		Ricardo Kalid	UFBA
03	Estimativa do tempo de campanha de fornos de pirólise da COPENE	Murilo F. de Amorim	COPENE
		Eliane Santanta	UFBA
		Ricardo Kalid	UFBA
04	Estimativa do tempo de campanha de fornos de pirólise da TRIKEM	José Milton	TRIKEM
		Milton Thadeu	TRIKEM
		Ricardo Kalid	UFBA
05	Módulo multimídia para treinamento em controle de processos contínuos	Gustavo Gadelha	SENAI-CETIND
		Jadson Aragão	SENAI-CETIND
		Admilson Casé	SENAI-CETIND
		Ricardo Kalid	UFBA
06	Modelagem, simulação, controle e otimização de conversores de acetileno da COPENE	Fabrício Brito	UFBA
		Tatiana Marucci	UFBA
		Mauricio Moreno	COPENE
		Paulo Freitas	COPENE
		Ricardo Kalid	UFBA
07	Simulação e controle de colunas de destilação de BTX da COPENE	Lueci V. do Vale	UFBA
		Mark Langerhost	COPENE
		Ricardo Kalid	UFBA
08	"Plantwide control" de um trem de separação de xilenos (3 colunas de destilação em série/paralelo) da COPENE	Fábio Carrilho	COPENE
		Mauricio Moreno	COPENE
		Ricardo Kalid	UFBA
09	Simulação e controle de colunas de destilação de sulfolane da COPENE	Cathia R. Apenburg	COPENE
		Williane Carneiro	COPENE
		Ricardo Kalid	UFBA
10	Sistema de controle de pH dos efluentes da GRIFFIN	Nelson Siem Velarde	GRIFFIN
		Ricardo Kalid	UFBA
11	Sintonia do controlador de topo da coluna de destilação de 3,4 DCA da GRIFFIN	Klauss Villalva Serra	GRIFFIN
		Almir Viana Cotias Filho	GRIFFIN
		Ricardo Kalid	UFBA
12	Modelagem, simulação e otimização do reator de 3,4 DCA da GRIFFIN	Klauss Villalva Serra	GRIFFIN
		Ricardo Kalid	UFBA

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

CURSOS E APOSTILAS DO INSTRUTOR SOBRE MODELAGEM DE PROCESSOS

1. ***Operações Unitárias em Regime Transiente – Balanços de Massa, Energia e Momentum Aplicados a Processo Químicos***.
   Ricardo de Araújo Kalid. DEQ-UFBA.
2. ***Identificação de Processos Químicos.***
   Ricardo de Araújo Kalid. DEQ-UFBA.

CURSOS E APOSTILAS DO INSTRUTOR SOBRE SIMULAÇÃO DE PROCESSOS

3. ***Programação em MATLAB com Aplicação em Reatores Químicos.*** Ricardo de Araújo Kalid. DEQ-UFBA.

CURSOS E APOSTILAS DO INSTRUTOR SOBRE CONTROLE DE PROCESSOS

4. ***Sistemas de Controle dos Principais Equipamentos da Indústria de Processos Químicos e Petroquímicos.***
   Ricardo de Araújo Kalid. DEQ-UFBA.
5. ***Controle de Processos Químicos***. Ricardo de Araújo Kalid. DEQ-UFBA.
6. ***Definição da Estrutura do Sistema de Controle Multimalha de Processos Multivariáveis***. Ricardo de Araújo Kalid. DEQ-UFBA.
7. ***Controle Avançado de Processos***. Ricardo de Araújo Kalid. DEQ-UFBA.
8. ***Controle de Coluna de Destilação***. Ricardo de Araújo Kalid. DEQ-UFBA.
9. ***Controle Preditivo Multivariável: DMC - Controle por Matriz Dinâmica***.
   Ricardo de Araújo Kalid. DEQ/UFBA

CURSOS E APOSTILAS DO INSTRUTOR SOBRE OTIMIZAÇÃO DE PROCESSOS

10. ***Otimização de Processos Químicos.***
    Ricardo de Araújo Kalid. DEQ-UFBA.

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

• INTRODUÇÃO

O MATLAB (MATrix LABoratory) é um ambiente de programação adequado para resolver problemas que envolvam operações matemáticas com matrizes.

O MATLAB é composto por duas ferramentas básicas e várias auxiliares. As básicas, o próprio MATLAB e o SIMULINK constituem o núcleo mínimo que deve estar disponível e possibilitam executar as ferramentas auxiliares, denominadas *toolbox*.
Os *toolbox* disponíveis na versão 5.3[&] são os seguintes:

```
daq\daq                 -  Data Acquisition Toolbox.
toolbox\dials           -  Dials & Gauges Blockset
toolbox\rptgenext       -  Simulink Report Generator
toolbox\rptgen          -  MATLAB Report Generator
database\database       -  Database Toolbox.
database\vqb            -  Visual Query Builder functions.
powersys\powersys       -  Power System Blockset
toolbox\compiler        -  MATLAB Compiler (and Compiler 1.2.1)
comm\comm               -  Communications Toolbox.
toolbox\symbolic        -  Symbolic Math Toolbox.
nag\nag                 -  NAG - Numerical & Statistical Library
map\map                 -  Mapping Toolbox
wavelet\wavelet         -  Wavelet Toolbox.
toolbox\pde             -  Partial Differential Equation Toolbox.
finance\finance         -  Financial Toolbox.
lmi\lmictrl             -  LMI Control Toolbox: Control Applications
lmi\lmilab              -  LMI Control Toolbox
qft\qft                 -  QFT Control Design Toolbox.
toolbox\fixpoint        -  Fixed-Point Blockset
dspblks\dspblks         -  DSP Blockset.
fuzzy\fuzzy             -  Fuzzy Logic Toolbox.
mpc\mpccmds             -  Model Predictive Control Toolbox.
fdident\fdident         -  Frequency Domain Identification Toolbox.
hosa\hosa               -  Higher-Order Spectral Analysis Toolbox.
toolbox\stats           -  Statistics Toolbox.
toolbox\ncd             -  Nonlinear Control Design Blockset
images\images           -  Image Processing Toolbox.
nnet\nnet               -  Neural Network Toolbox.
mutools\subs            -  Mu-Analysis and Synthesis Toolbox.
signal\signal           -  Signal Processing Toolbox.
toolbox\splines         -  Spline Toolbox.
toolbox\optim           -  Optimization Toolbox.
toolbox\robust          -  Robust Control Toolbox.
toolbox\ident           -  System Identification Toolbox.
toolbox\control         -  Control System Toolbox.
toolbox\rtw             -  Real-Time Workshop
rtw\rtwdemos            -  (No table of contents file)
stateflow\stateflow     -  Stateflow
```

[&] A versão mais recente é a 6.0

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Podemos observar que a área de cobertura dos *toolbox* é muito vasta cobrindo, por exemplo, os temas relacionados com

- finanças
- tratamento de imagem
- tratamento de sinais
- estatística
- redes neurais
- controle
- otimização

Mas além desses *toolbox*, que são adquiridos na MATHWORKS (empresa fornecedora do MATLAB/SIMULINK), vários outros podem ser encontrados gratuitamente na internet.

Devido o MATLAB cobrir várias áreas do conhecimento da engenharia e pela forma mais fácil de programar, quando comparado com outras linguagens de alto nível, como por exemplo FORTRAN ou C, esta plataforma tem sido largamente utilizado no meio acadêmico e industrial.

A facilidade da programação do MATLAB vem principalmente pela maneira intuitiva como as operações com vetores e matrizes são tratadas, dispensando quase♣ que completamente o uso de indexadores, *loops* tipo *for* ou *do* na soma, multiplicação, inversão ou qualquer outra operação com matrizes e/ou vetores, por exemplo:

EX 1: Seja os vetores v1 = [1 ; 2 ; 3 ; 4] e v2 = [2 ; 3 ; 4 ; 5].

A soma v3 = v1 + v2 no MATLAB é realizada exatamente da maneira como escrevemos na linguagem matemática usual.

EX 2: Seja a matriz $m1 = \begin{bmatrix} 2 & 3 \\ 5 & 1 \end{bmatrix}$

O quadrado dessa matriz é escrito por m1^2 ou m1*m1.

O dobro dessa matriz é dado por 2*m1

evitando, dessa forma, a chamada de subrotinas para realizar cálculos corriqueiros nos problemas de engenharia.

Outro ponto que facilita a programação é que o MATLAB é um ambiente de programação que realiza instrução por instrução, o que ajuda muito na etapa de desenvolvimento dos programas. Mas que também pode gerar um código compilado, ou seja gerar um executável, que é executado independente do MATLAB.

♣ A única situação na qual somos obrigados a utilizar os indexadores para realizar operação com matrizes é quando um elemento de uma matriz ou vetor depende de resultado de um elemento anterior dessa mesma matriz ou vetor.

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

- **Vetorização**

O MATLAB realiza os cálculos utilizando o recurso da vetorização das matrizes e vetores automaticamente e de forma transparente para o usuário, desta forma o tempo computacional é minimizado sem necessitar de esforços adicionais do programador. Por exemplo, definindo o vetor v1

```
v1 = ones(10000,1);
```

e o vetor v2

```
v2= 2*ones(size(v1)) ;
```

a soma utilizando a vetorização é dada por

```
tic
v3 = v1 + v2;
toc

elapsed_time =
     0
```

enquanto que utilizando a estrutura por indexadores, temos:

```
tic
for i = 1:length(v2)
     v4(i) = v1(i) + v2(i);
end
toc

elapsed_time =
    0.4300 ]
```

Obs: os tempos acima são diferentes a depender da carga que a CPU está submetida no instante do processamento.
Observe que os textos em verde são exatamente os comandos que devem ser digitados no *prompt* do MATLAB. Enquanto que os textos e azul é o resultado que aparecerá na tela.
Ainda nos comando acima verificamos a sintaxe do comando ***ones(nl,nc)*** , no qual definimos o número de linhas ***nl*** e colunas ***nc*** que o vetor de uns v1 terá.
Verificamos como gerar um novo vetor ***v2*** com as mesmas dimensões do vetor ***v1***.
Ainda aprendemos a utilizar os comandos ***tic*** e ***toc***. Aliás sempre que tiver dúvidas sobre como usar um comando, o MATLAB tem um help on-line, para tanto basta digitar ***help***[δ] seguido do comando que uma ajuda na tela aparecerá. Por exemplo, se quiser saber como os comandos supracitados trabalham, basta digitar ***help tic***

[δ] Um **help** mais completo é acionado através da barra principal do MATLAB, nesse **help** você pode fazer uma pesquisa por nome, ou navegar por todo a documentação do MATLAB.

```
help tic
 TIC Start a stopwatch timer.
    The sequence of commands
        TIC, operation, TOC
    prints the number of seconds required for the operation.

    See also TOC, CLOCK, ETIME, CPUTIME.
```

Na definição de uma matriz, a mudança de coluna é caracterizada pela inserção de uma vírgula e a de linha pelo ponto e vírgula. Por exemplo a matriz identidade 2 por 2 é criada pelo seguinte comando

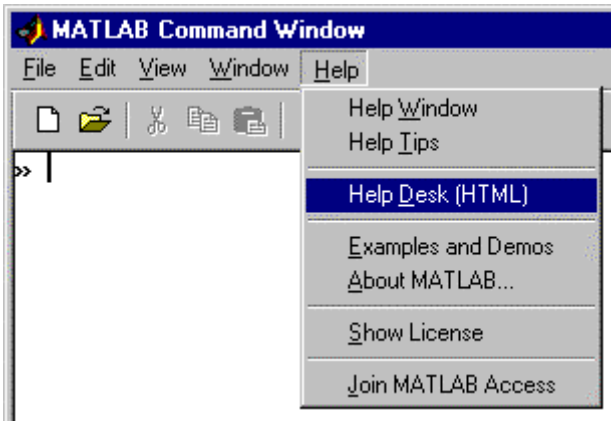
```
matriz_identidade = [ 1 , 0 ; 0 1 ]
matriz_identidade =
     1     0
     0     1
```

ou então usando o comando ***eye***:

```
mat_ident = eye(2)
mat_ident =
     1     0
     0     1
```

A inversa de uma matriz é calculada utilizando o comando ***inv*** :

```
A = [ 1 2 ; 4  10 ]

A =
     1     2
     4    10

iA = inv(A)
```



Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

```
iA =
    5.0000   -1.0000
   -2.0000    0.5000
```

verificando

```
iA*A

ans =
     1     0
     0     1
```

- **Operações elemento a elemento**

Podemos operar a multiplicação de duas matrizes elemento a elemento:

```
iA.*A

ans =
     5    -2
    -8     5
```

Observe que antes do sinal de multiplicação colocamos um ponto. É dessa forma que é indicado para o MATLAB se a operação a ser realizada, que neste caso foi a multiplicação, é elemento a elemento ou segue as regras de multiplicação de duas matrizes. Essa mesma regra do ponto serve para potência(.^) e divisão(./).

Temos que ter cuidado com a multiplicação de matrizes e vetores, pois a operação matricial ou elemento a elemento darão resultados completamente diferentes.

Por exemplo, a operação A*A gera a matriz

```
A = [ 1 2 ; 4  10 ];
A*A
ans =
     9    22
    44   108
```

enquanto que a operação **A.*A** gera a matriz

```
A.*A
ans =
     1     4
    16   100
```

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

• EQUAÇÕES ALGÉBRICAS

Muitos problemas são representados por sistemas de equações algébricas lineares ou não lineares. Como por exemplo

- cálculo da capacidade calorífica de substâncias e misturas
- obtenção dos coeficientes estequiométricos de um sistema reacional
- modelagem de um reator CSTR em estado estacionário

As equações algébricas podem ser classificadas como lineares ou não-lineares. Neste último caso, podemos resolver o sistema através da inversão de matrizes sem recorrer a procedimentos iterativos, porém no caso não-linear às vezes temos que utilizar de algoritmos numéricos mais sofisticados, como método de Newton, secante, etc.

• Sistemas de equações algébricas não-lineares que não necessitam de procedimentos iterativos

Por exemplo, no cálculo da capacidade calorífica de substâncias e misturas não há necessidade de utilizar métodos numéricos. Tomemos a título de ilustração uma mistura formada por C_2H_2, C_2H_4, C_2H_6, CO, H_2 e CH_4 a uma certa temperatura (T = 30 ºC) e a baixas pressões, a capacidade calorífica da mistura é dada por:

$$Cp_{mistura} = \sum_{i=1}^{nc} C_i . Cp_i(T) \qquad (02.01)$$

onde

C_i - concentração da espécie i

Cp_i - capacidade calorífica molar da espécie i

T - temperatura

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

```
% Definição dos índices das substâncias
C2H2 = 1 ;
C2H4 = 2 ;
C2H6 = 3 ;
CO   = 4 ;
H2   = 5 ;
CH4  = 6 ;

% Condições da mistura
T = 30 + 273.15 ; % [=] K
ConcMol = [ 0.01  0.8  0.18 0.001 0.008 0.001 ] ;

% Coeficientes das capacidades caloríficas
Cp(C2H2,1) =  2.682e+1 ; % [=] J/(mol.K)
Cp(C2H2,2) =  7.578e-2 ; % [=] J/(mol.K)
Cp(C2H2,3) = -5.007e-5 ; % [=] J/(mol.K)
Cp(C2H2,4) =  1.412e-8 ; % [=] J/(mol.K)
Cp(C2H4,1) =  3.806e+0 ; % [=] J/(mol.K)
Cp(C2H4,2) =  1.566e-1 ; % [=] J/(mol.K)
Cp(C2H4,3) = -8.348e-5 ; % [=] J/(mol.K)
Cp(C2H4,4) =  1.755e-8 ; % [=] J/(mol.K)
Cp(C2H6,1) =  5.409e+0 ; % [=] J/(mol.K)
Cp(C2H6,2) =  1.781e-1 ; % [=] J/(mol.K)
Cp(C2H6,3) = -6.938e-5 ; % [=] J/(mol.K)
Cp(C2H6,4) =  8.713e-9 ; % [=] J/(mol.K)
Cp(CO,1)   =  3.087e+1 ; % [=] J/(mol.K)
Cp(CO,2)   = -1.285e-2 ; % [=] J/(mol.K)
Cp(CO,3)   =  2.789e-5 ; % [=] J/(mol.K)
Cp(CO,4)   = -1.272e-8 ; % [=] J/(mol.K)
Cp(H2,1)   =  2.714e+1 ; % [=] J/(mol.K)
Cp(H2,2)   =  9.274e-3 ; % [=] J/(mol.K)
Cp(H2,3)   = -1.381e-5 ; % [=] J/(mol.K)
Cp(H2,4)   =  7.645e-9 ; % [=] J/(mol.K)
Cp(CH4,1)  =  1.925e+1 ; % [=] J/(mol.K)
Cp(CH4,2)  =  5.213e-2 ; % [=] J/(mol.K)
Cp(CH4,3)  =  1.197e-5 ; % [=] J/(mol.K)
Cp(CH4,4)  = -1.132e-8 ; % [=] J/(mol.K)

% Capacidade calorífica média
T2 = T  * T ;
T3 = T2 * T ;
T4 = T3 * T ;

Cpaux = Cp(:,1) + Cp(:,2)*T + Cp(:,3)*T2 + Cp(:,4)*T3 ;

CCpi  = ConcMol * Cpaux

CCpi =
   45.6172
```

Esses comandos podem todos estar contidos num arquivo texto, que deve ter a extensão **.m** . Dessa forma podemos calcular a propriedade à temperatura e/ou concentrações diferentes, sem ter a necessidade de digitar cada linha novamente.

- **Sistema de equações algébricas lineares**

Por exemplo queremos resolver o seguinte sistema de equações:

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

$$\begin{cases} 2x_1 + 3x_2 = 9 \\ 2{,}5x_1 + 1{,}5x_2 = 7 \end{cases} \qquad (02.02)$$

Solução: $A.x = b \Rightarrow x = A^{-1}.B$

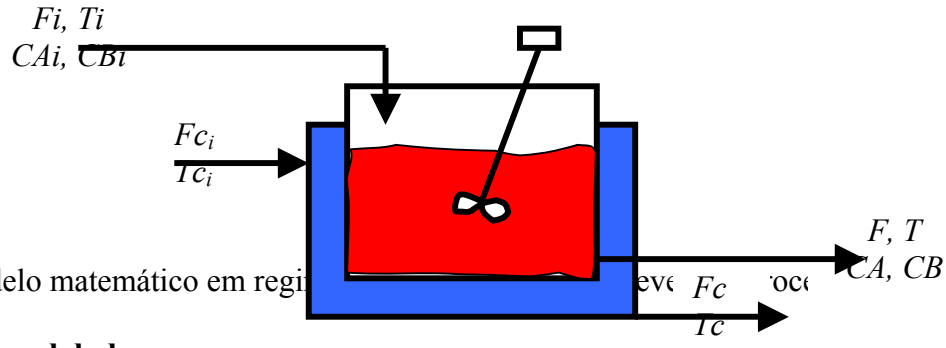
No MATLAB

```
A = [ 2 3 ; 2.5 1.5]
b = [ 9 ; 7]
x = inv(A) * b
```

```
A =
    2.0000    3.0000
    2.5000    1.5000
b =
     9
     7
x =
    1.6667
    1.8889
```

- **Sistema de equações algébricas não-lineares (SEANL)**

Outra classe de problemas bastante frequente que um engenheiro químico enfrenta é a solução de problemas formado por equações algébricas não lineares (SEANL). Por exemplo, ao modelar um reator CSTR não isotérmico em estado estacionário: Considere um reator CSTR, veja Figura 2.01, no qual ocorre uma reação química irreversível exotérmica, $A \rightarrow B$. Este reator tem uma camisa de resfriamento e encontra-se em estado estacionário.



O modelo matemático em regi… …eve… …oce…

Balanço global:

$$0 = F_i(t) - F(t) \qquad (02.03)$$

Balanço de massa para o reagente A:

$$0 = \left(F_i(t).C_{Ai}(t) - F(t).C_A(t)\right) + \Gamma_A(t).V(t) \qquad (02.04)$$

Balanço de massa para o reagente B:

$$0 = \left(F_i(t).C_{Bi}(t) - F(t).C_B(t)\right) + \Gamma_B(t).V(t) \qquad (02.05)$$

Balanço de energia no reator:

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

$$0 = (F_i(t).H_i(t) - F(t).H(t)) + V(t).\Gamma_A(t).\Delta H_R \qquad (02.06)$$

Balanço de energia na camisa do reator:

$$0 = (Fc_i(t).Hc_i(t) - Fc(t).Hc(t)) + U.A.(T(t) - Tc(t)) \qquad (02.07)$$

As equações (02.03) a (02.07) constituem o modelo fenomenológico não-linear do CSTR. Considerando algumas hipóteses adicionais:

Hipóteses:

H1. Seção transversal do tanque é constante e de área A_t. Então $V(t) = A_t.L(t)$

H2. As propriedades dos componentes e da mistura não variam com a temperatura.

H3. Energia = Entalpia

H4. Os efeitos do calor de mistura são desprezíveis

H5. Reação do tipo: A $\rightarrow$ B de 2ª ordem, irreversível. Portanto podemos escrever:

$$\Gamma_B = -\Gamma_A = k(T).C_A^2 = k_o.e^{\left(-\frac{E}{R.(T+273.15)}\right)}.C_A^2$$

H6. A camisa está completamente preenchida pelo fluido de refrigeração, ou seja

$$F_c = F_{ci}$$

Então, omitindo a explicitação da dependência com o tempo (t) e considerando as hipóteses acima, podemos escrever:

$$0 = \frac{F_i}{A_t} - \frac{F}{A_t} \qquad (02.08)$$

$$0 = \frac{F_i}{A_t}\cdot\frac{1}{L}\cdot(C_{Ai} - C_A) - k_o.e^{\left(-\frac{E}{R.(T+273.15)}\right)}.C_A^2 \qquad (02.09)$$

$$0 = \frac{F_i}{A_t}\cdot\frac{1}{L}\cdot(C_{Bi} - C_B) + k_o.e^{\left(-\frac{E}{R.(T+273.15)}\right)}.C_A^2 \qquad (02.10)$$

$$0 = \frac{F_i}{A_t}\cdot\frac{1}{L}\cdot(T_i - T) - \frac{\Delta H_R.k_o}{\rho.Cp}\cdot e^{\left(-\frac{E}{R.(T+273.15)}\right)}.C_A^2 - \frac{U.A}{\rho.Cp.A_t}\cdot\frac{1}{L}\cdot(T - T_c) \qquad (02.11)$$

$$0 = \frac{F_c}{V_c}\cdot(T_{ci} - T_c) + \frac{U.A}{\rho_c.Cp_c.V_c}\cdot(T - T_c) \qquad (02.12)$$

Portanto o modelo é formado por um sistema de 5 equações algébricas não lineares.

Para resolver esse sistema de equações vamos utilizar o comando ***fsolve*** .

A sintaxe do comando ***fsolve*** é a seguinte:

solucao = fsolve('seanl',estimativa_inicial)

onde ***seanl*** :

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

- é uma ***M-function*** que contém o Sistema de Equações Algébricas Não Lineares a ser resolvido;
- uma ***m-function*** é um arquivo com extensão ***.m*** apresenta na primeira linha a palavra reservada ***function*** , neste caso o nome do arquivo é ***seanl.m***

Para o CSTR teremos, por exemplo

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

nome do arquivo: CSTR_estado_estacionario.m

```
function [f] = CSTR_estado_estacionario(x)

V     =    7.08    ; % volume do reator [=] m3
ro    =   19.20    ; % [=] kmol/m3
Cp    =    1.815e5 ; % [=] J/(kmol.oC)
A     =    5.40    ; % área de troca térmica [=] m2
roc   = 1000.00    ; % [=] kg/m3
ko    =    0.0744  ; % [=] m3/(s.kmol)
dHR   =   -9.86e7  ; % [=] J/kmol
U     = 3550.00    ; % [=] J/(s.m2.oC)
Vc    =    1.82    ; % [=] m3
Cpc   = 4184.00    ; % [=] K/(kg.oC)
E     =    1.182e7 ; % [=] J/kmol
Fcmax =    0.02    ; % [=] m3/s
Rg    = 8314.39    ; % [=]
g     =    9.81    ; % [=] m/s2

Rt    = ( 2 * V / ( 2 * pi ) ) ^ (1/3) ; % Lmax = 2R , e L = 50% Lmax
At    = pi * Rt ^ 2                    ; % área transversal do tanque (L=2R) [=] m2
Lt    = V / At                         ; % nível de projeto [=] m

Ft    = 7.5e-3                   ; % vazão de descarga de projeto [=] m3/s
alfaL = Ft / sqrt( ro * g * Lt ) ; % constante de proporcionalidade para F

L   =   V / At  ; % [=] m
CAi =   2.88    ; % [=] kmol/m3
CBi =   0       ; % [=] kmol/m3
Ti  =  66       ; % [=] oC
Tci =  27       ; % [=] oC
Fi  =  7.5e-3   ; % [=] m3/h
Fci =  7.392e-3 ; % [=] m3/h
Fc  =  Fci      ;

F  = x(1) ;
CA = x(2) ;
CB = x(3) ;
T  = x(4) ;
Tc = x(5) ;

V        = At * L                          ;
ex       = exp( - E / ( Rg * ( T + 273.15 ) ) ) ;
k        = ko * ex                         ;
kCA      = k * CA                          ;
G        = kCA * CA                        ;
UA       = U * A                           ;
roCp     = ro * Cp                         ;
VroCp    = V * roCp                        ;
FiV      = Fi / V                          ;
FiVL     = FiV * L                         ;
FcVc     = Fc / Vc                         ;
VcrocCpc = Vc * roc * Cpc                  ;

f(1) =        ( Fi  - F  ) / At                                    ;
f(2) = FiV  * ( CAi - CA ) -       G                               ;
f(3) = FiV  * ( CBi - CB ) +       G                               ;
f(4) = FiV  * ( Ti  - T  ) - dHR * G / roCp - UA * ( T - Tc ) / VroCp    ;
f(5) = FcVc * ( Tci - Tc )                  + UA * ( T - Tc ) / VcrocCpc ;
```

Então, após criar o arquivo CSTR_estado_estacionario.m e mudar para sua pasta de trabalho, para tanto use o comando ***cd*** ou use o botão  , execute as seguintes instruções:

```
estimativa_inicial = [ 0.0075  1.1303  1.7497  87.9218  50.3117];
x=fsolve('CSTR_estado_estacionario',estimativa_inicial)
```

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

• SISTEMA DE EQUAÇÕES DIFERENCIAIS ORDINÁRIAS - SEDO

Muito frequente também são os sistemas de equações diferenciais ordinárias.
Por exemplo no caso do CSTR em regime transiente, o modelo é dado por um SEDO:

$$f_1 = \frac{dL}{dt} = \frac{F_i}{A_t} - \frac{F}{A_t} \tag{03.01}$$

$$f_2 = \frac{dC_A}{dt} = \frac{F_i}{A_t} \cdot \frac{1}{L} \cdot \left(C_{Ai} - C_A\right) - k_o . e^{\left(-\frac{E}{R.(T+273.15)}\right)} . C_A^2 \tag{03.02}$$

$$f_3 = \frac{dC_B}{dt} = \frac{F_i}{A_t} \cdot \frac{1}{L} \cdot \left(C_{Bi} - C_B\right) + k_o . e^{\left(-\frac{E}{R.(T+273.15)}\right)} . C_A^2 \tag{03.03}$$

$$f_4 = \frac{dT}{dt} = \frac{F_i}{A_t} \cdot \frac{1}{L} \cdot \left(T_i - T\right) - \frac{\Delta H_R . k_o}{\rho . Cp} \cdot e^{\left(-\frac{E}{R.(T+273.15)}\right)} . C_A^2 - \frac{U . A}{\rho . Cp . A_t} \cdot \frac{1}{L} \cdot \left(T - T_c\right) \tag{03.04}$$

$$f_5 = \frac{dT_c}{dt} = \frac{F_c}{V_c} \cdot \left(T_{ci} - T_c\right) + \frac{U . A}{\rho_c . Cp_c . V_c} \cdot \left(T - T_c\right) \tag{03.05}$$

Existem 3 formas diferentes de resolver esse problema utilizando o MATLAB/SIMULINK:

1. Através do uso do ***Differential Equation Editor***;
2. Através do uso do **SIMULINK** e **S-FUNCTIONS;**
3. Através do uso dos comandos ***ode23***, ***ode45***, ***ode113***, ***ode15s***, ***ode23s***, ***ode23t***, ***ode23tb***, ***rk45***, ***rk23*** e **M-FUNCTIONS**;

Na Tabela 03.01 observamos em que situações cada alternativa de solução de SEDOs é a mais recomendada.

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Tabela 03.01: Alternativas para resolução de SEDOs e recomendações

Situação	*Differential Equation Editor*	***odeXX* ou *rkXX* e M-FUNCTIONS**	**SIMULINK e S-FUNCTIONS**
Número de equações diferenciais	poucas (até 2)	poucas ou muitas	poucas ou muitas
Uso em problemas de simulação	sim	sim	sim
Uso em problemas de controle	não	não	sim
Uso em problemas de otimização	não	sim	não
Exemplo de utilização	digite ***dee*** na linha de comando	comando ***help odeXX*** ou ***rkXX***	pasta **CSTR_transiente** ou pasta **pH** ou pasta **BATCH**

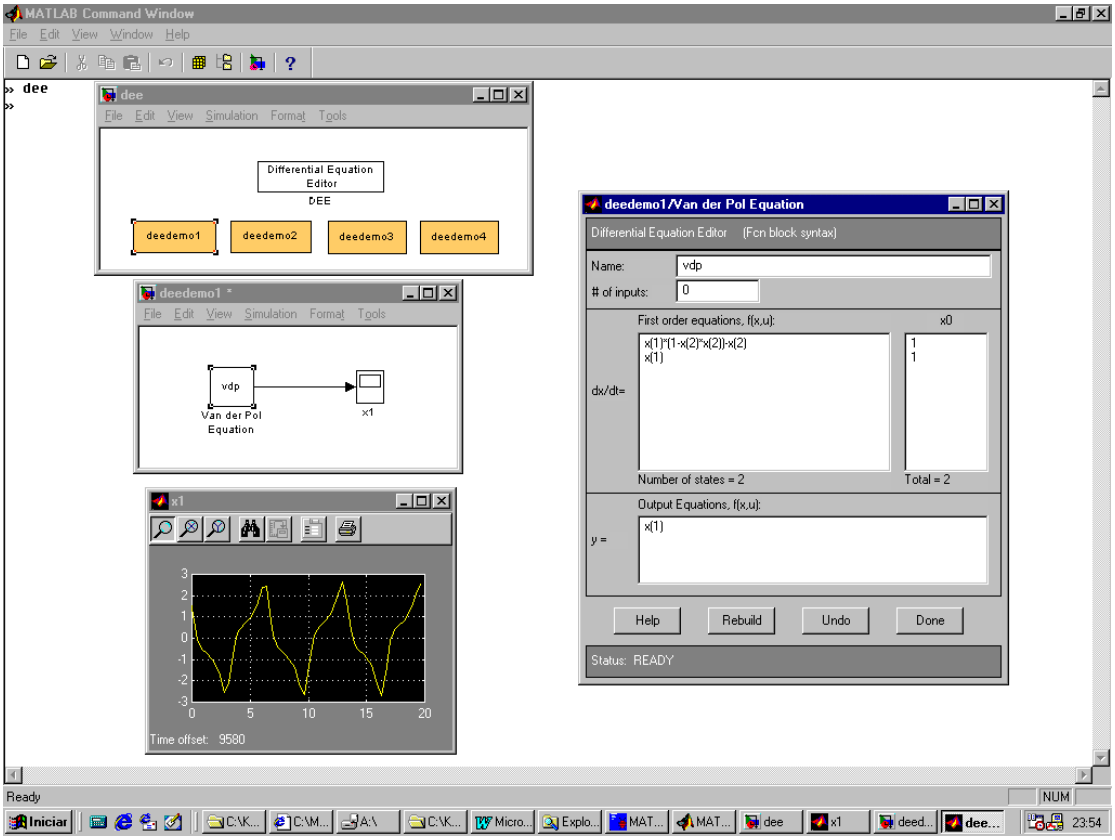


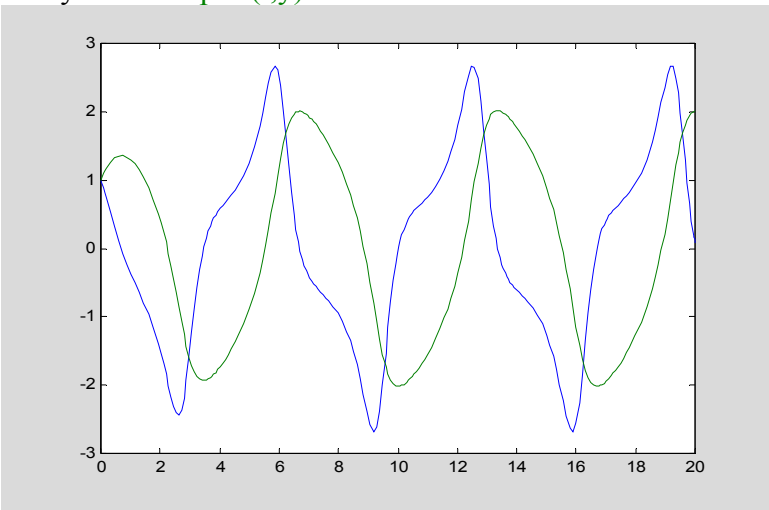
Figura 3.01: Problema VDP (Van Der Pol) utilizando o ***dee***

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Resolvendo o VDP por ode45:
a) Crie uma M-FUNCTION denominada de **vdp.m**:

```
function derivada = vdp(t,x,u)
derivada(1) = x(1) .* (1 - x(2).^2) - x(2)    ;
derivada(2) = x(1)                             ;
derivada = derivada'                           ; % criando vetor coluna
```

b) Digite o seguinte comando [t,y] = ode45('vdp',[0;20],[1;1]) ;
c) Plote o gráfico y versus t: plot(t,y)



Para observar um exemplo do uso do SIMULINK para resolver equações diferenciais mude para a pasta **CSTR_transiente** e digite ***cstr*** na linha de comando. Neste caso o modelo de um reator CSTR formado por 5 equações diferenciais ordinárias é resolvido.

Outro exemplo é dado na pasta **pH** na qual o sistema de neutralização em malha fechada da GRIFFIN é simulado. Neste caso digite **pH** na linha de comando do MATLAB para executar a simulação.

Utilizando o MATLAB podemos, também, simular processos a batelada, como podemos observar no exemplo da pasta BATCH. Neste caso digite **batch** na linha de comando do MATLAB para executar a simulação.

- ## Uso de M-FUNCTION – Exemplo: Reator a Batelada

Para ilustrar um exemplo de uso da instrução **ode45** considere um processo a batelada. Utilizaremos o reator descrito por Luyben[β].

[β] Luyben, W. L. *Process Modeling, Simulation and Control for Chemical Engineers*. McGraw-Hill, páginas 57-61, 150-157. 1989 .

Considere um reator químico a batelada no qual a temperatura da parede do reator pode ser controlada com perfeição e no qual ocorre a seguinte reação em fase líquida:

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$

Equação 1

Sistemas reacionais como o da equação 1 são relativamente comuns em processos químicos, por exemplo, oxidação de hidrocarbonetos ou cloração de aromáticos. E nesses casos o produto intermediário é sempre o mais desejado.

O reator pode ser visto na Figura 5.01. Existe uma camisa de refrigeração ou de aquecimento que garante o calor necessário para esfriar ou aquecer a parede até a temperatura T_M desejada.

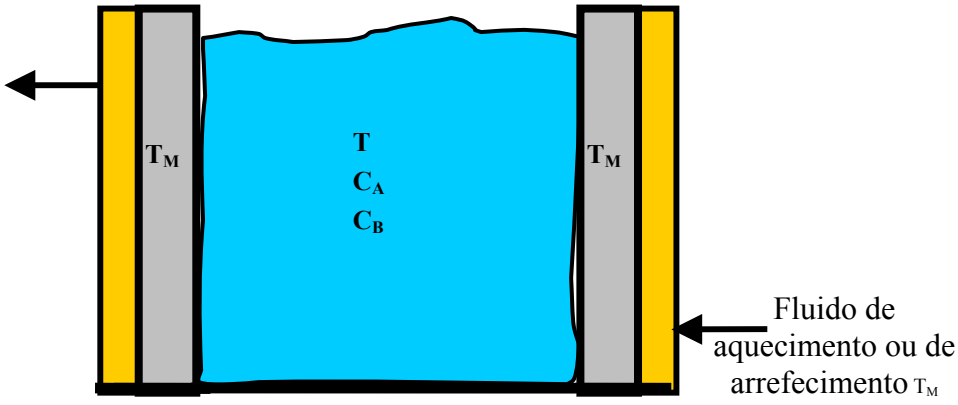


Figura 5.01: Reator a batelada com camisa de aquecimento/arrefecimento

A cinética e sua dependência com a temperatura é conhecida. O objetivo deste processo é produzir a substância B. O tempo de batelada é fixo e conhecido (200 minutos).

As taxas de consumo das substâncias (cinética das reações) podem ser assim descritas:

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

$$\Gamma_A = -k_1(T)C_A$$

Equação 2

$$\Gamma_B = k_1(T)C_A - k_2(T)C_B$$

Equação 3

onde $$k_i(T) = k_i^o \exp\left(-\frac{E_i}{RgT}\right) \quad , \quad i = 1,2$$

Os dados e valores dos parâmetros deste modelo é dado abaixo:

C_A^o = 0.80 lb-mol/ft^3 C_B^o = 0 lb-mol/ft^3

k_1^o = 729.55 min^{-1} k_2^o = 6567.6 min^{-1}

E_1 = 15.000 Btu/lb-mol E_2 = 20.000 Btu/lb-mol

Modelo matemático

A) Balanço de massa:

$$\frac{dVC_A}{dt} = V\Gamma_A = -Vk_1(T)C_A$$

Equação 4

$$\frac{dVC_B}{dt} = V\Gamma_B = V\left[k_1(T)C_A - k_2(T)C_B\right]$$

Equação 5

como o volume do meio reacional é constante (reação em fase líquida), obtemos:

$$\frac{dC_A}{dt} = \Gamma_A = -k_1(T)C_A \quad , \quad C_A(0) = C_A^o$$

Equação 6

$$\frac{dC_B}{dt} = \Gamma_B = k_1(T)C_A - k_2(T)C_B \quad , \quad C_B(0) = C_B^o$$

Equação 7

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

B) Balanço de energia:

$$\frac{dT}{dt} = -\frac{\Delta H_1}{\rho Cp} k_1(T) C_A + \frac{\Delta H_2}{\rho Cp} k_2(T) C_B - \frac{Q_M}{V \rho Cp}$$

Equação 8

onde

$$Q_M = h_i A_i \left(T - T_M\right)$$

Equação 9

h_i = 160 Btu/(h.^{o}F.ft^2)

A_i = 56.5 ft^2

ρ = 50 lb$_m$/ft^3

ΔH_1 = -40.000 Btu/lb-mol

ΔH_2 = -50.000 Btu/lb-mol

Utilizando o MATLAB podemos, também, simular processos a batelada, como podemos observar no exemplo da pasta BATCH. Neste caso digite **batch** na linha de comando do MATLAB para executar a simulação.

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

O código do programa principal (**batch.m**)é o seguinte:

```
clear all
% Universidade Federal da Bahia
% Escola Politécnica
% Departamento de Engenharia Química
% Professor:  Ricardo de Araujo Kalid (e-mail: kalid@ufba.br)
% Data:       31 de janeiro de 2001
% Reator a Batelada
% 3 Equações diferenciais
% Exercício Luyben pg151
global A1 A2 En1 En2 dHr1 dHr2 ro Cp V hi Area Rg TM
% Temperatura do fluido refrigerante
  TM = 50 ;
% Tempo de simulação
  tempo = 40 ;
% Condições iniciais
  CAo =  0.8 ;
  CBo =  0.0 ;
  To  = 80.0 ;
% Constantes cinéticas e outras constantes
  A1   =    729.55  ;
  A2   =   6567.60  ;
  En1  =  15000     ;
  En2  =  20000     ;
  dHr1 = -40000     ;
  dHr2 = -50000     ;
  ro   =     50     ;
  Cp   =      1     ;
  V    =     42.5   ;
  hi   =    160.0   ;
  Area =     56.5   ;
  Rg   =      1.987 ;
% Simulação
  x0 = [ CAo CBo To ] ;
  [time,x] = ode45('balancos',[0,tempo],x0) ; % Integração balanços de massa e energia
  CA = x(:,1) ;
  CB = x(:,2) ;
  T  = x(:,3) ;
% Gráficos
  figure(1)
  set(1,'Name','Perfil Temp. no Tempo')
  plot(time,T,'r',time,TM,'b')
  title('Perfil da Temperatura ao Longo do Tempo')
  ylabel('Temperat. (oF)')
  xlabel('T (verm)  TM (azul)         tempo (min)')
  figure(2)
  set(2,'Name','Perfil da Concetração de CA')
  plot(time,CA,'b')
  title('Perfil de CA ao Longo do Tempo')
  ylabel('CA (lb-mol/ft3)')
  xlabel('CA (azul)    -     tempo (min)')
  figure(3)
  set(2,'Name','Perfil da Concetração de CB')
  plot(time,CB,'r')
  title('Perfil de CB ao Longo do Tempo')
  ylabel('CB (lb-mol/ft3)')
  xlabel('CB (verm)    -     tempo (min)')
% Salvamento
  save simulacao
% Fim deste arquivo
```

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

O programa principal chama a rotina de integração (**ode45**), que tem como um de seus argumentos a **function** que tem o modelo do processo. Ou seja, é necessário escrever os balanços de massa e energia do sistema na **function** denominada **balancos.m** .

```
function [ dxdt ] = balancos( time , x )

% Universidade Federal da Bahia
% Escola Politécnica
% Departamento de Engenharia Química
% Professor:  Ricardo de Araujo Kalid (e-mail: kalid@ufba.br)
% Data:       31 de janeiro de 2001

% Reator a Batelada
% 3 Equações diferenciais
% Exercício Luyben pg151

% Function que define os balanços de massa

global A1 A2 En1 En2 dHr1 dHr2 ro Cp V hi Area Rg TM

CA = x(1) ;
CB = x(2) ;
T  = x(3) ;

k1 = A1 * exp( - En1/(Rg*(T+460)) ) ;
k2 = A2 * exp( - En2/(Rg*(T+460)) ) ;
QM = hi * Area * ( T - TM )         ;

dC1dt = - k1 * CA ;
dC2dt = + k1 * CA - k2 * CB  ;
dTdt  = - dHr1*k1*CA/(ro*Cp) + dHr2*k2*CB/(ro*Cp) - QM/(V*ro*Cp) ;

dxdt = [ dC1dt
         dC2dt
         dTdt  ] ;

% Fim deste arquivo
```

Essa **function** que é chamada por uma rotina que resolve um SEDO obrigatoriamente deve ter:

a) dois argumentos de entrada, o tempo e a variável de estado, no mínimo;
b) um argumento de saída, as equações diferenciais;
c) para passar parâmetros para a **function** foi utilizado a instrução **global**
d) para evitar confusão denomine o nome da **function** o mesmo nome do arquivo que a contém, pois a relação do MATLAB é dada pelo nome do arquivo e não pelo nome da **function** .

- **S-function**

Os exemplos utilizando o SIMULINK fazem uso de uma estrutura de programação denominada **S-function.**
Embora seja possível utilizar o SIMULINK com **M-function** a forma mais eficiente do ponto de vista computacional é empregando a **S-function**, por isso nessa seção nos concentraremos no estudo dessa estrutura de programação do MATLAB/SIMULINK.

Uma **S-function** começa com a instrução function e é formada por seções:

- Seção 0 (zero) para **flag = 0**
  - inicialização do sistema formado por SEDO (Sistema de Equações Diferenciais Ordinárias)
  - definição do tamanho do SEDO a ser resolvido
  - definição da condição inicial do SEDO

- Seção 1 (um) para **flag = 1**
  - definição das equações do modelo SEDO
  - utilizada quando temos modelos contínuos
  - retorna as derivadas das variáveis de estado $\frac{dx(t)}{dt}$

- Seção 2 (dois) para **flag = 2**
  - definição das equações de diferenças
  - utilizada quando temos modelos discretos
  - retorna os estados discretos $x(n+1)$

- Seção 3 (três) para **flag = 3**
  - definição das equações de saída do sistema
  - retorna um vetor com as variáveis de saída do sistema $y(t)$

- Seção 4 (quatro) para **flag = 4**
  - retorna o próximo intervalo de tempo para atualização do modelo discreto

Nos exemplos que estudaremos utilizaremos modelo contínuos.

- **S-function: Exemplo 1:**
  SEDOL - Espaço de estados lineares

Seja o seguinte modelo em espaço de estados lineares e invariantes no tempo:

$$\begin{cases} \dot{\underline{x}} = \underline{\underline{A}}.\underline{x} + \underline{\underline{B}}.\underline{u} \\ \underline{y} = \underline{\underline{C}}.\underline{x} + \underline{\underline{D}}.\underline{u} \end{cases}$$

onde $\underline{\underline{A}}$, $\underline{\underline{B}}$, $\underline{\underline{C}}$, e $\underline{\underline{D}}$ são matrizes. Por exemplo:

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

```
A  =  [  -0.3   0   0 ;  2.9   -0.62 -2.3 ;   0   2.3   0 ]
B  =  [   1    ;   0    ;  0     ]
C  =  [   1    1   0    ;  1   -3     1    ]
D  =  [   1    ;   0       ]
A =
   -0.3000         0         0
    2.9000   -0.6200   -2.3000
         0    2.3000         0
B =
     1
     0
     0
C =
     1     1     0
     1    -3     1
D =
     1
     0
```

Para resolver esse SEDO-L também precisamos de uma condição inicial:

```
xInicial   =   [   1   ;   1   ;   1   ]
xInicial =
     1
     1
     1
```

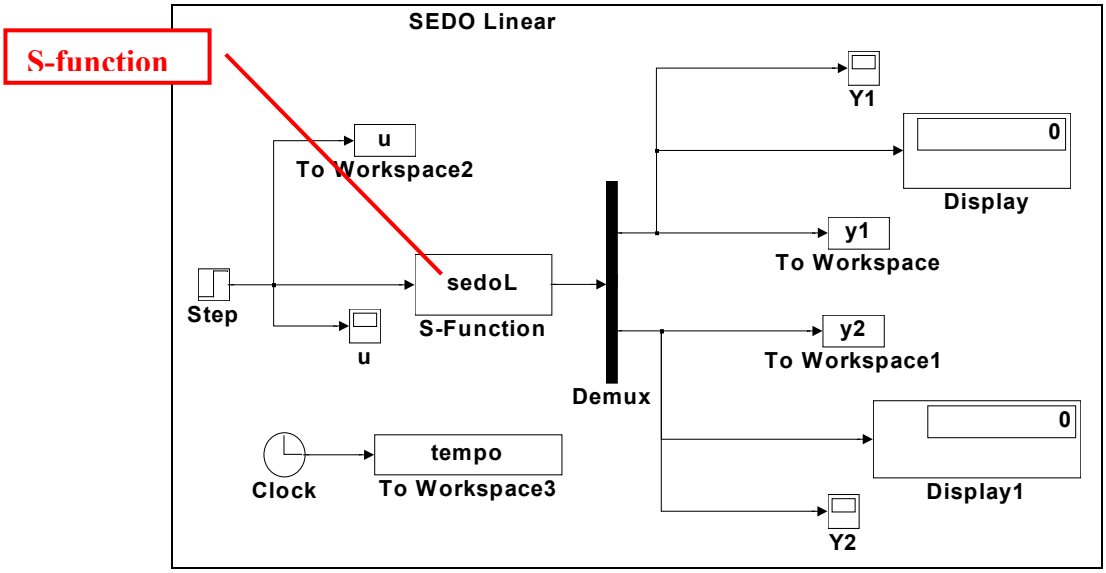
Este problema tem 3 estados , 1 entrada e 2 variáveis de saída

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Escrevendo a **S-function : sedoL.m** :

```
function [sys,x0] = sedoL(t,x,u,flag)

% Definição das constantes
A = [ -0.3 0  0 ; 2.9  -0.62 -2.3 ; 0  2.3  0 ] ;
B = [  1  ;  0  ; 0  ]                          ;
C = [  1  1  0  ; 1  -3   1  ]                  ;
D = [  1  ;  0   ]                                ;

if abs(flag) == 1
  % Cálculo das derivadas dos estados
  sys(1,1) = A(1,1)*x(1) + A(1,2)*x(2) + A(1,3)*x(3) + B(1)*u(1) ;  % Equação  dx1/dt
  sys(2,1) = A(2,1)*x(1) + A(2,2)*x(2) + A(2,3)*x(3) + B(2)*u(1) ;  % Equação  dx2/dt
  sys(3,1) = A(3,1)*x(1) + A(3,2)*x(2) + A(3,3)*x(3) + B(3)*u(1) ;  % Equação  dx3/dt
  % Ou poderia ser:  sys = A*x + B*u ;

elseif abs(flag) == 3
  % Transformando as variáveis de estado em variáveis de saída
  sys(1,1) = C(1,1)*x(1) + C(1,2)*x(2) + C(1,3)*x(3) + D(1)*u(1) ;  % Equação  y1(t)
  sys(2,1) = C(2,1)*x(1) + C(2,2)*x(2) + C(2,3)*x(3) + D(2)*u(1) ;  % Equação  y1(t)
  % Ou poderia ser:  sys = C*x + D*u ;

elseif flag == 0
  % Inicialização do sistema
  x0       = [  1  ;  1  ;  1   ]  ; % Condições inciais
  sys(1,1) = 3                     ; % Número de estados contínuos
  sys(2,1) = 0                     ; % Número de estados discretos
  sys(3,1) = 2                     ; % Número de saídas
  sys(4,1) = 1                     ; % Número de entradas
  sys(5,1) = 0                     ; % Número de raizes discretas
  sys(6,1) = 0                     ; % Usado em sistemas de sistemas
else
  sys = [];
end
```

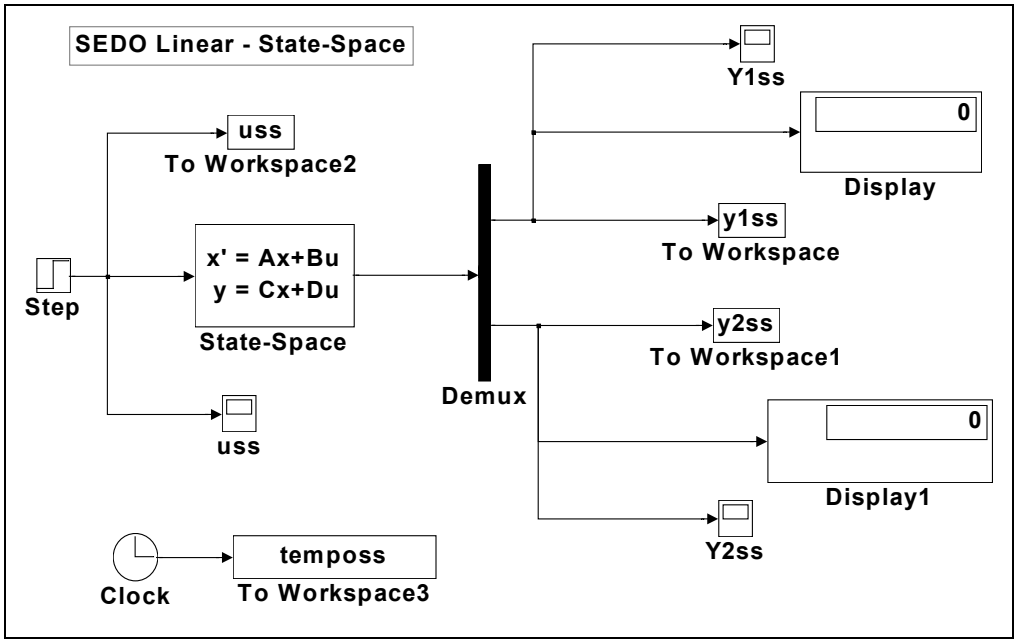
Para editar o arquivo **sedoL.m** utilize o editor de programas que vem associado ao MATLAB.
Para utilizar a **S-function** execute o modelo **sedo** digitando esse nome na linha de comando do MATLAB. Então aparecerá o seguinte sinótico:



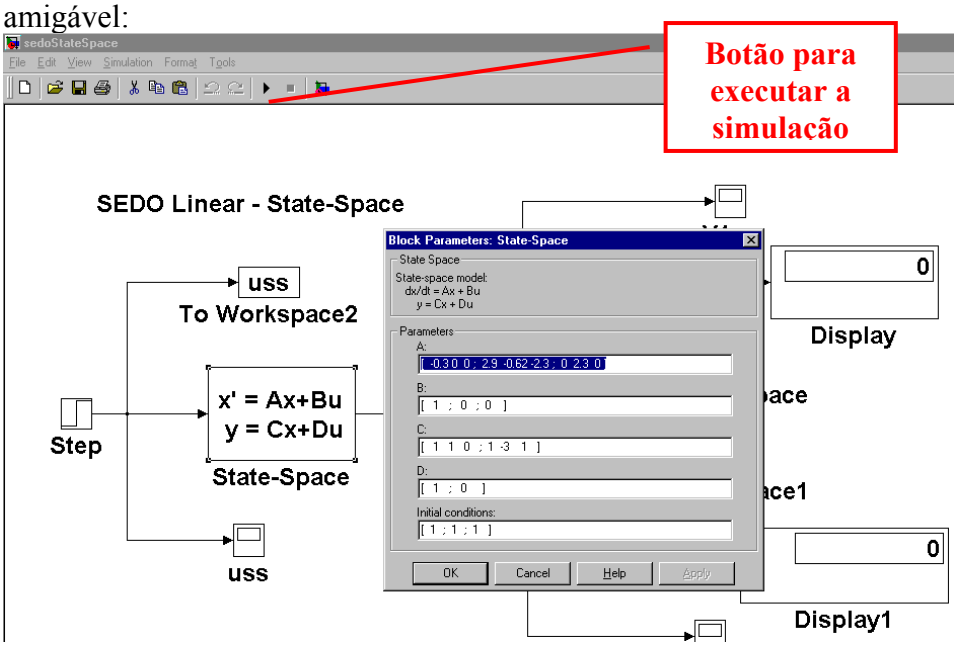
Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: lacoi@ufba.br

Para criar um sinótico utilize o comando **SIMULINK** na linha de comando do MATLAB, ou click no botão

Outra forma de resolver sistemas de equações diferenciais lineares pode ser observada digitando **sedoStateSpace** na linha de comando do MATLAB, aparecerá então o seguinte sinótico



Se você executar esse sinótico ou o anterior, o resultado será o mesmo, pois ambos fazem a mesmíssima coisa. Apenas a máscara do sinótico **sedoStateSpace** é mais amigável:



Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: lacoi@ufba.br

Laboratório de Controle e Otimização Industrial
LACOI

- **S-function: Exemplo 2: Espaço de estados não-lineares SEDONL : CSTR em regime transiente**

Após mudar para pasta **CSTR_transiente** digite **CSTR** na linha de comando do MATLAB. A **S-function** dessa simulação **S_sedo.m** é a seguinte:

```
function [sys,x0] = S_sedo(t,x,u,flag)
% Planta: CSTR
% Autor:  Ricardo Kalid
%         kalid@ufba.br
% Interface no Simulink para  S_sedo.m
% Inputs:
% t : tempo em segundos
% x : estados do sistema
% u : perturbações externas ou variáveis manipuladas
% Outputs:   sys  e  x0  como descritos no manual do SIMULINK
%            quando  flag é 0 , sys contem as dimensões dos vetores e
%                    x0  contem as condições iniciais,
%            quando  flag  é  1 , sys contem as derivadas,
%            quando  flag  é  3 , sys contem as saídas:
global VetorAdim
global CondicoesIniciais
global NumEstados NumVarManip NumVarDistMed
global ro Cp A roc ko dHR U Vc Cpc E Rg
global At g  alfaL VetorAdim   TipoVazaoDescarga
if abs(flag) == 1
  % Cálculo das derivadas dos estados
  x   = x .* [ x > 0 ] ;
  sys = M_cstr(t,x,u)   ;
elseif abs(flag) == 3
  % Transformando as variáveis de estado em variáveis de saída
  sys(1,1) = x(1) * VetorAdim(1) ; % L  = nível              do reator     [=] m
  sys(2,1) = x(2) * VetorAdim(2) ; % CA = concentração       do reagente A [=] kmol/m3
  sys(3,1) = x(3) * VetorAdim(3) ; % CB = concentração       do produto  B [=] kmol/m3
  sys(4,1) = x(4) * VetorAdim(4) ; % T  = temperatura        do reator     [=] oC
  sys(5,1) = x(5) * VetorAdim(5) ; % Tc = temperatura        da camisa     [=] oC
  % Vazão de descarga do reator
  % Se  F = F(L)                                          ==>  F = alfaL * P^0.5
  % F  independe de L e não existe controle de nível      ==>  F = ve(12)
  % Controle de nível é perfeito                         ==>  F = Fi
  L  = sys(1,1) ; % nível                [=] m
  Fi = u(1)     ; % vazão da alimentação [=] m3/s
  if TipoVazaoDescarga == 1
     P = ro * g * L         ; % pressão na tubulação de descarga
     F = alfaL * sqrt ( P ) ; % vazão proporcional ao nível (L)
  elseif TipoVazaoDescarga == 2
     F   = u(7)             ; % vazão da descarga fornecida,
                              % fixada pelo usuário ou pelo sistema de controle [=] m3/s
  elseif TipoVazaoDescarga == 3
     F = Fi                 ; % controle perfeito  do nível (L)
  else
     disp( ' Problemas em S_SEDO.M' )
     disp( ' Escolha TipoVazaoDescarga = 1 , 2 ou 3' )
     break
  end
  sys(6,1) = F                  ; % F  = vazão de descarga do reator     [=] m3/s
elseif flag == 0
  % Inicialização do sistema
  x0       = CondicoesIniciais              ;
  sys(1,1) = NumEstados                     ; % Número de estados contínuos
  sys(2,1) = 0                              ; % Número de estados discretos
  sys(3,1) = NumEstados + 1                 ; % Número de saídas
  sys(4,1) = NumVarManip + NumVarDistMed - 1 ; % Número de entradas
  sys(5,1) = 0                              ; % Número de raizes discretas
  sys(6,1) = 0                              ; % Usado em sistemas de sistemas
else
  sys = [];
end
% Fim deste arquivo
```

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

As equações diferenciais do modelo matemático do CSTR estão na **M-function** M_cstr.m listada abaixo.

```
function Derivadas = M_cstr(t,x,u)

% Função para cálculo do Sistema de Equações Diferenciais Ordinárias
% que representa a PLANTA do CSTR

% Dados passados
global ro Cp A roc ko dHR U Vc Cpc E Rg
global At g  alfaL VetorAdim   TipoVazaoDescarga

% Estados do sistema
L   = x(1) * VetorAdim(1) ; % nível        do reator     [=] m
CA  = x(2) * VetorAdim(2) ; % concentração do reagente A [=] kmol/m3
CB  = x(3) * VetorAdim(3) ; % concentração do produto  B [=] kmol/m3
T   = x(4) * VetorAdim(4) ; % temperatura  do reator     [=] oC
Tc  = x(5) * VetorAdim(5) ; % temperatura  da camisa     [=] oC

% Entradas do sistema
Fi  = u(1) ; % vazão        da carga               [=] m3/s
CAi = u(2) ; % concentração do reagente A na carga [=] kmol/m3
CBi = u(3) ; % concentração do produto  B na carga [=] kmol/m3
Ti  = u(4) ; % temperatura  da carga               [=] oC
Tci = u(5) ; % temperatura  do fluido refrigerante [=] oC
Fc  = u(6) ; % vazão        do fluido refrigerante [=] m3/s

% Vazão de descarga do reator
% Se  F = F(L)                                     ==>  F = alfaL * P^0.5
% F  independe de L e não existe controle de nível ==>  F = ve(12)
% Controle de nível é perfeito                     ==>  F = Fi

if TipoVazaoDescarga == 1
   P = ro * g * L         ; % pressão na tubulação de descarga
   F = alfaL * sqrt ( P ) ; % vazão proporcional ao nível (L)
elseif TipoVazaoDescarga == 2
   F   = u(7)             ; % vazão da descarga fornecida,
                            % fixada pelo usuário ou pelo sistema de controle [=] m3/s
elseif TipoVazaoDescarga == 3
   F = Fi                 ; % controle perfeito  do nível (L)
else
   disp( ' Problemas em M_SEDO.M' )
   disp( ' Escolha TipoVazaoDescarga = 1 , 2 ou 3' )
   break
end

if ( L <= eps | CA <= eps | CB <= eps | T <= eps | Tc <= eps )
   disp( '      F        Fc       L        CA        CB        T        Tc ' )
   disp( [ F   Fc   L   CA   CB   T   Tc ] )
   disp( ' Problemas em SEDO.M' )
   break
end

% Sistema de Equações Diferenciais Ordinárias (SEDO)
[redacted]

% Adimensionalização, quando for o caso
Derivadas = Derivadas ./ VetorAdim ;

% Fim deste arquivo
```

M-file sedo.m :

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

```
% SEDO

V         = At * L                                  ;
ex        = exp( - E / ( Rg * ( T + 273.15 ) ) ) ;
k         = ko * ex                                 ;
kCA       = k * CA                                  ;
G         = kCA * CA                                ;
UA        = U * A                                   ;
roCp      = ro * Cp                                 ;
VroCp     = V * roCp                                ;
FiV       = Fi / V                                  ;
FiVL      = FiV * L                                 ;
FcVc      = Fc / Vc                                 ;
VcrocCpc = Vc * roc * Cpc                           ;

dLdt  =         ( Fi  - F  ) / At                                        ;
dCAdt = FiV  * ( CAi - CA ) -       G                                    ;
dCBdt = FiV  * ( CBi - CB ) +       G                                    ;
dTdt  = FiV  * ( Ti  - T  ) - dHR * G / roCp - UA * ( T - Tc ) / VroCp   ;
dTcdt = FcVc * ( Tci - Tc )                  + UA * ( T - Tc ) / VcrocCpc ;

Derivadas = [ dLdt
              dCAdt
              dCBdt
              dTdt
              dTcdt ] ;
```

Portanto, de maneira *claríssima* observamos que

sedo.m é um **M-file** que será incluído na **M-function** **M_cstr.m** que, por sua vez, é uma subrotina da **S-function** **S_sedo.m**, chamada pelo sinótico **cstr1sim.mdl** (ou **cstr2sim.mdl** ou **cstr3sim.mdl**)
ou melhor

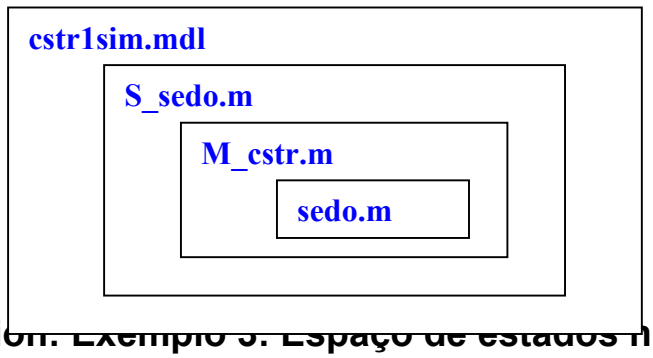
sedo.m é um **M-file** que será incluído na **M-function** **M_cstr.m**
M_cstr.m é uma **M-function,** subrotina da **S-function** **S_sedo.m**
S_sedo.m é uma **S-function**, executada pelo sinótico **cstr1sim.mdl**

ou ainda

cstr1sim.mdl chama **S_sedo.m** que chama **M_cstr.m** que inclui **sedo.m**

ou seja

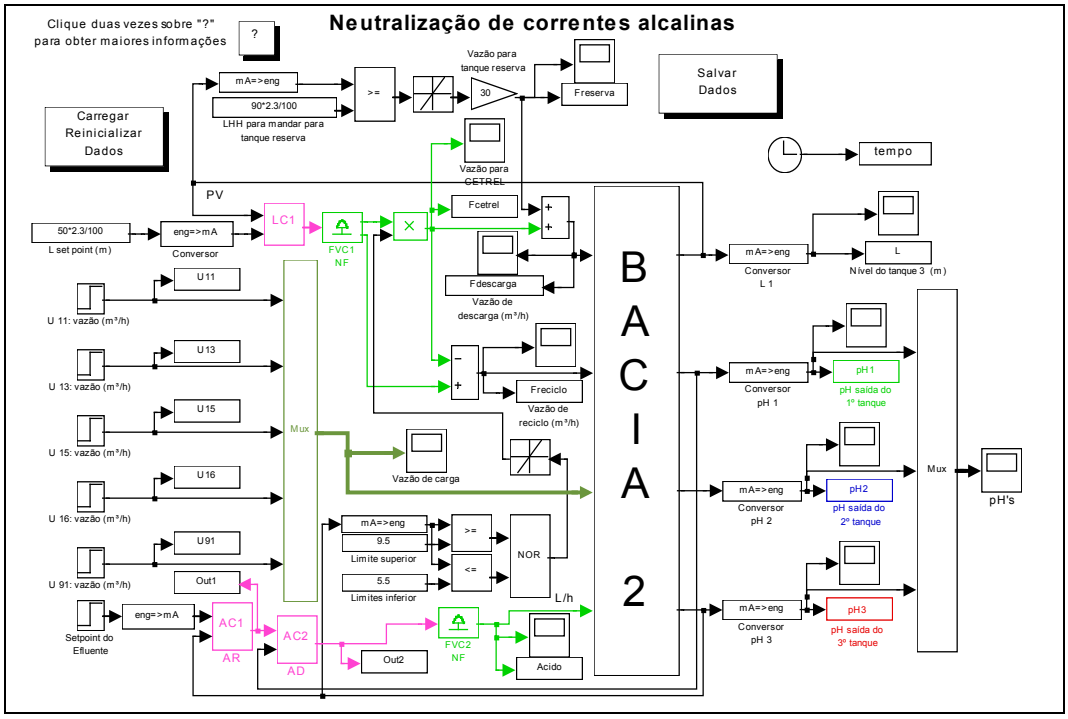
cstr1sim.mdl manda em **S_sedo.m**, que manda **M_cstr.m**, que manda em **sedo.m**, que calcula as derivadas (é o peão).



1.1.1 S-function. Exemplo 5. Espaço de estados não-lineares SEDONL : neutralização de pH em malha fechada

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Abaixo vemos o sinótico de um sistema de controle de neutralização de correntes alcalinas.



O sistema têm duas malhas de controle, uma de nível (LC1) e uma de pH (AC1 controlador mestre do AC2).

Os controladores são do tipo PID (Proporcional + Integral + Derivativo).

Além do sistema de controle também está simulado o sistema de intertravamento, que atua se o pH estiver fora da faixa e/ou se o nível do 3º compartimento estiver muito alto ou muito baixo.

Esta simulação demonstra a grande versatilidade do **MATLAB/SIMULINK** para representar processos. Além disso essa plataforma computacional é muito mais fácil de programar que as tradicionais linguagens de alto nível, tipo FORTRAN, C etc. Mesmo quando comparada com o Visual C++ ou Visual Basic, o MATLAB leva grande vantagens no desenvolvimento de simuladores para a processos contínuos e/ou discretos.
O coração do modelo desenvolvido para representar o pH da mistura a partir do pH das correntes individuais:

```
% pH de entrada para o 1° tanque de neuralização
% Média ponderada com as vazões das correntes de alimentação (alcalinas e ácido)
% Considero que a mistura é perfeita xxk

pH_mistura = ( vaz_U11 / (vaz_ent_base + eps) - Us_na_mist_01(1) ) * pH_U_11_12i + ...
             ( vaz_U13 / (vaz_ent_base + eps) - Us_na_mist_01(2) ) * pH_U_13_14i + ...
             ( vaz_U15 / (vaz_ent_base + eps) - Us_na_mist_01(3) ) * pH_U_15i    + ...
```

```
             ( vaz_U16 / (vaz_ent_base + eps) - Us_na_mist_01(4) ) * pH_U_16i    + ...
             ( vaz_U91 / (vaz_ent_base + eps) - Us_na_mist_01(5) ) * pH_U_91labi + ...
            pH_U_mist_01i                                                         ;
vol_mistura = (vaz_U11/(vaz_ent_base+eps) - Us_na_mist_01(1)) * vol_PSA_U_11_12i + ...
              (vaz_U13/(vaz_ent_base+eps) - Us_na_mist_01(2)) * vol_PSA_U_13_14i + ...
              (vaz_U15/(vaz_ent_base+eps) - Us_na_mist_01(3)) * vol_PSA_U_15i    + ...
              (vaz_U16/(vaz_ent_base+eps) - Us_na_mist_01(4)) * vol_PSA_U_16i    + ...
              (vaz_U91/(vaz_ent_base+eps) - Us_na_mist_01(5)) * vol_PSA_U_91labi + ...
              vol_PSA_mist_01i                                                    ;

% interpolação linear para considerar o reciclo xxk
pH_mistura  = ( pH_mistura * vaz_ent_base  + pH_reciclo * vaz_rec )  / ...
              vaz_ent_base_reciclo                                    ;
vol_mistura = ( vol_mistura * vaz_ent_base + vol_reciclo * vaz_rec ) / ...
              vaz_ent_base_reciclo                                    ;

% o pH do reciclo acho mais correto considerar como sendo o pH da saída do 3° tanque
% na iteração anterior xxk
pH_reciclo  = pH_mistura  ;
vol_reciclo = vol_mistura ;
pH_mistura_matriz  = [ pH_mistura_matriz ; pH_mistura' ]   ;
vol_mistura_matriz = [ vol_mistura_matriz ; vol_mistura' ] ;

% Cálculo da cuuva de neutralização (ganho do processo) do 1° tanque
% falta considerar o efeito da variação da vazão e qualidade das alimentações e do
% ácido no pH da mistura
x            = vol_mistura                ;
y            = pH_mistura                 ;
metodo       = 'linear'                   ;
vaz_aux      = vaz_A / fator_de_escala    ; % correção da diferença de conc. do ácido

pH_tq_ent(1) = interp1(x,y,vaz_aux,metodo) ;

% pH de entrada para os demais tanques de neuralização

pH_tq_ent(2) = pH_tq(1) ;
pH_tq_ent(3) = pH_tq(2) ;

% Sistema de Equações Diferenciais Ordinárias (SEDO)

tal(1) = vol_util(1) / vaz_ent ;
tal(2) = vol_util(2) / vaz_ent ;

% existe um certo erro,
% mas se o controle de nível for perfeito então o tal(3) está correto
tal(3) = vol_util(3) / vaz_sai ;

dnivel_tq3dt = ( vaz_ent - vaz_sai )       / area_total(3) ;
dpH_tqdt(1)  = ( pH_tq_ent(1) - pH_tq(1) ) / tal(1)        ;
dpH_tqdt(2)  = ( pH_tq_ent(2) - pH_tq(2) ) / tal(2)        ;
dpH_tqdt(3)  = ( pH_tq_ent(3) - pH_tq(3) ) / tal(3)        ; % xxk

Derivadas = [ dnivel_tq3dt ; dpH_tqdt(1) ; dpH_tqdt(2) ; dpH_tqdt(3) ] ;

% Fim deste arquivo
```

1.1.2 S-function: Exemplo 4: Espaço de estados não-lineares SEDONL : reator PFR em estado estacionário

Considere um reator tubular em estado estacionário, no qual ocorrem as seguintes reações químicas:

(R1) Hidrogenação do acetileno a etileno: $C_2H_2 + H_2 \rightarrow C_2H_4$

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

$$\Gamma_1 = \frac{k_1 K_{H_2} P_{H_2} K_{C_2H_2} P_{C_2H_2}}{\left(1 + K_{C_2H_2} P_{C_2H_2} + K_{CO} P_{CO}\right)\left(1 + K_{H_2} P_{H_2}\right)}$$

(R2) Hidrogenação do etileno a etano: $C_2H_4 + H_2 \rightarrow C_2H_6$

$$\Gamma_2 = \frac{k_2 K_{H_2} P_{H_2} K_{C_2H_4} P_{C_2H_4}}{\left(1 + K_{C_2H_4} P_{C_2H_4} + K_{CO} P_{CO}\right)\left(1 + K_{H_2} P_{H_2}\right)}$$

O reator catalítico, adiabático cuja carga é composta por C_2H_2, C_2H_4, C_2H_6, CO, H_2 e CH_4. O C_2H_2 é a substância que deve ser convertida a C_2H_4, contudo parte desta subtância é convertida em C_2H_6 (reação indesejada). O monóxido de carbono é um agente que aumenta a seletividade do catalisador. O metano é inerte.
O modelo em regime estacionário para esse processo é dado por

$$\frac{1}{\pi R^2}\frac{dF_{C_2H_2}}{dz} = -\Gamma_{G_{C_2H_2}}$$

$$F_{C_2H_6} = F^o_{C_2H_6} + F^o_{H_2} - F_{H_2} - \left(F^o_{C_2H_2} - F_{C_2H_2}\right)$$

$$F_{C_2H_4} = F^o_{C_2H_4} + \left(F^o_{C_2H_2} - F_{C_2H_2}\right) - \left(F_{C_2H_6} - F^o_{C_2H_6}\right)$$

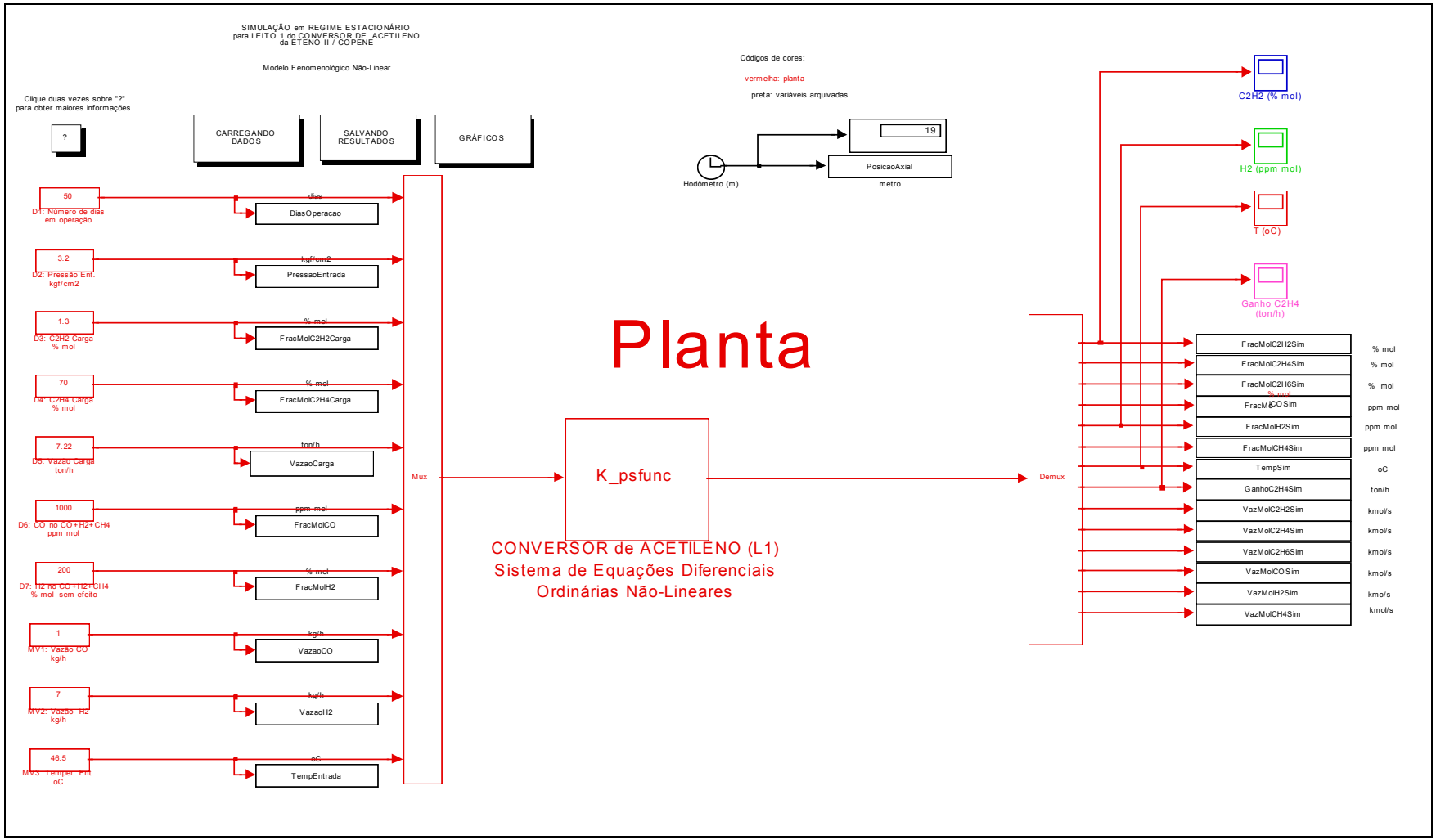
$$\frac{1}{\pi R^2}\frac{dF_{H_2}}{dz} = -\Gamma_{G_{H_2}}$$

$$v\left(\sum_{i=1}^{nc} C_i \overline{Cp_i}\right)\frac{dT}{dz} = -\left(\Delta H_1 \sigma_1 \Gamma_1 + \Delta H_2 \sigma_2 \Gamma_2\right)$$

Na próxima página observamos o sinótico co reator tubular. Para executar o simulador, mude para a pasta **pfr_estacionário** e digite **pfr** na linha de comando do MATLAB.

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Laboratório de Controle e Otimização Industrial
LACOI



Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (0 71) 9984.3316
e-mail: lacoi@ufba.br

Laboratório de Controle e Otimização Industrial
LACOI

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Laboratório de Controle e Otimização Industrial
LACOI

- **SISTEMA DE EQUAÇÕES DIFERENCIAIS PARCIAIS**
  S-function: espaço de estados não-lineares
  SEDPNL: reator PFR em regime transiente

Considere um reator tubular em regime dinâmico, no qual ocorrem as seguintes reações químicas:

(R1) Hidrogenação do acetileno a etileno: $C_2H_2 + H_2 \rightarrow C_2H_4$

(R2) Hidrogenação do etileno a etano: $C_2H_4 + H_2 \rightarrow C_2H_6$

O reator catalítico, adiabático cuja carga é composta por C_2H_2, C_2H_4, C_2H_6, CO, H_2 e CH_4. O C_2H_2 é a substância que deve ser convertida a C_2H_4, contudo parte desta subtância é convertida em C_2H_6 (reação indesejada). O monóxido de carbono é um agente que aumenta a seletividade do catalisador. O metano é inerte.
O modelo em regime transitório para esse processo é dado por 7 equações diferenciais parciais não-lineares:

$$\varepsilon_t \frac{\partial C_{C_2H_2}}{\partial t} = -\frac{1}{\pi R^2}\frac{\partial F_{C_2H_2}}{\partial z} - \Gamma_{G_{C_2H_2}}$$

$$\varepsilon_t \frac{\partial C_{C_2H_4}}{\partial t} = -\frac{1}{\pi R^2}\frac{\partial F_{C_2H_4}}{\partial z} - \Gamma_{G_{C_2H_4}}$$

$$\varepsilon_t \frac{\partial C_{C_2H_6}}{\partial t} = -\frac{1}{\pi R^2}\frac{\partial F_{C_2H_6}}{\partial z} - \Gamma_{G_{C_2H_6}}$$

$$\varepsilon_t \frac{\partial C_{CO}}{\partial t} = -\frac{1}{\pi R^2}\frac{\partial F_{CO}}{\partial z}$$

$$\varepsilon_t \frac{\partial C_{H_2}}{\partial t} = -\frac{1}{\pi R^2}\frac{\partial F_{H_2}}{\partial z} - \Gamma_{G_{H_2}}$$

$$\varepsilon_t \frac{\partial C_{CH_4}}{\partial t} = -\frac{1}{\pi R^2}\frac{\partial F_{CH_4}}{\partial z}$$

$$\left[\varepsilon_t \sum_{i=1}^{nc} C_i \overline{Cp}_i + (1-\varepsilon_L)\rho_p Cp_p\right]\frac{\partial T}{\partial t} = -v\left(\sum_{i=1}^{nc} C_i \overline{Cp}_i\right)\frac{\partial T}{\partial z} - \sum_{j=1}^{nr} \Delta H_j \theta_j \eta_j \Gamma_j$$

onde

Γ_{G_i}	-	taxa de consumo efetivo do componente *i*
Γ_j	-	taxa intrínseca da reação *j*
θ_j	-	função atividade catalítica
η_j	-	fator de efetividade

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

$\varepsilon_t \sum_{i=1}^{nc} \left(C_i \overline{Cp}_i \right)$ - termo referente ao acúmulo de energia no fluido

$(1-\varepsilon_L)\rho_p Cp_p$ - termo referente ao acúmulo de energia no catalisador

$$k_1 = k_1^o \exp\left(-\frac{E_1}{T}\right)$$

$$k_2 = k_2^o \exp\left(-\frac{E_2}{T}\right)$$

$$K_{C_2H_2} = k_{C_2H_2}^o \exp\left(\frac{Q_{C_2H_2}}{T}\right)$$

$$K_{C_2H_4} = k_{C_2H_4}^o \exp\left(\frac{Q_{C_2H_4}}{T}\right)$$

$$K_{H_2} = k_{H_2}^o \exp\left(\frac{Q_{H_2}}{T}\right)$$

$$K_{CO} = k_{CO}^o \exp\left(\frac{Q_{CO}}{T}\right)$$

Os valores dos parâmetros apresentados por Bos (1993) são os seguintes:

$\kappa_1^o = 2{,}74_x 10^6 \; kmol/kg_s s$

$E_1 = 7517 \; K$

$\kappa_2^o = 3{,}16_x 10^8 \; kmol/kg_s s$

$E_2 = 8479 \; K$

$K_{C_2H_2}^o = 1{,}97_x 10^{-1} \; kPa^{-1}$

$Q_{C_2H_2} = 1508 \; K$

$K_{C_2H_4}^o = 5{,}33_x 10^{-6} \; kPa^{-1}$

$Q_{C_2H_4} = 1810 \; K$

$K_{H_2} = 1{,}28_x 10^{-6} \; kPa^{-1}$

$Q_{H_2} = 1861 \; K$

$K_{CO} = 4{,}0_x 10^{-9} \; kPa^{-1}$

$Q_{CO} = 8963 \; K$

As condições inicial e de contorno para as equações (2.II.01-07) são as seguintes:

condição inicial: para $t = 0 \rightarrow$ $C_i(0,z) = C_i^o(z)$

$T(0,z) = T(z)$

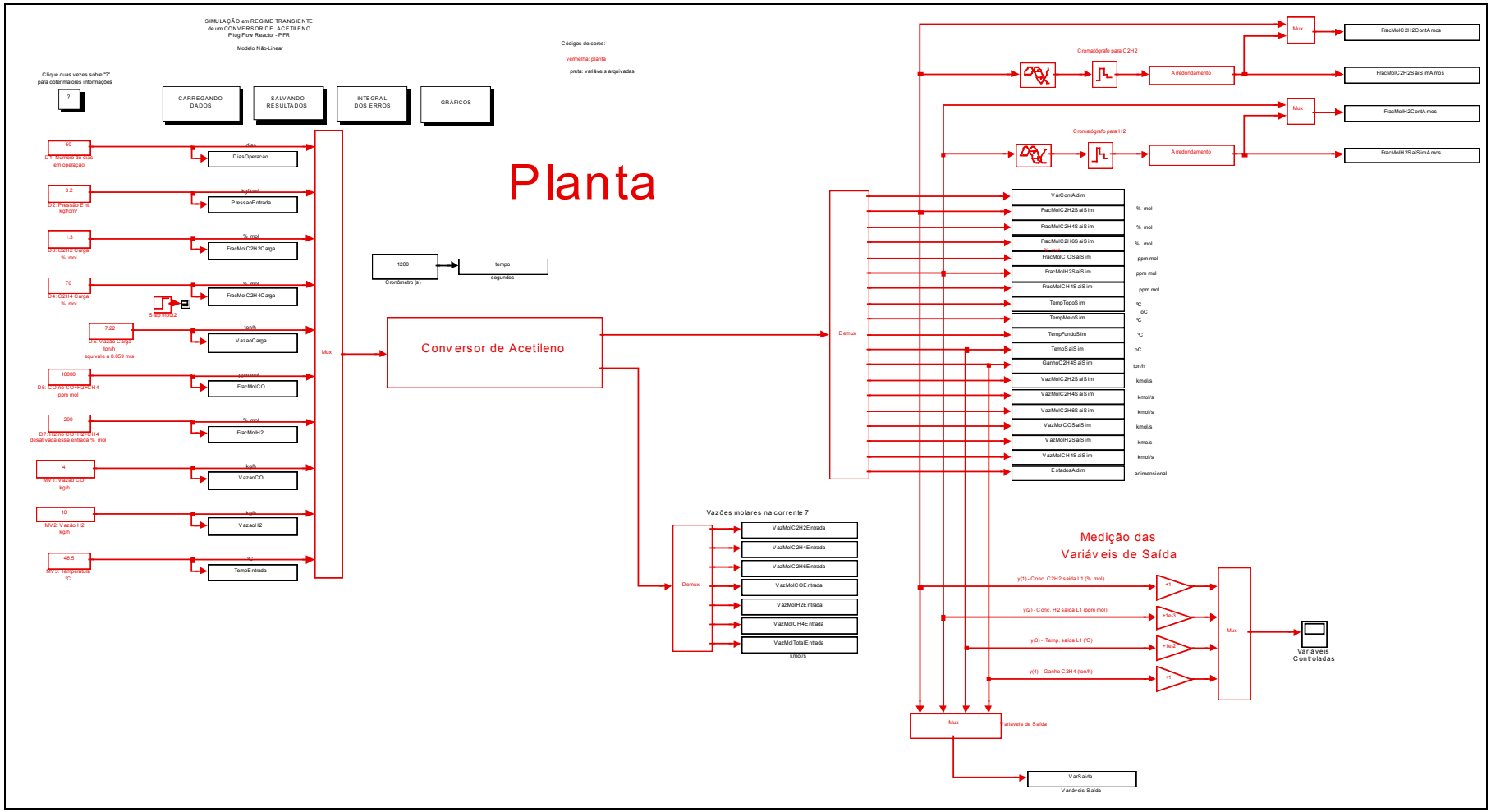
condição de contorno: para $z = 0 \rightarrow$ $F_i(t,0) = F_i(t)$

$T(t,0) = T(t)$

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

Laboratório de Controle e Otimização Industrial
LACOI

Nesta página observamos o sinótico do reator tubular. Para executar o simulador, mude para a pasta **pfr_transiente** e digite **pfr** na linha de comando do MATLAB.



Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

• DEPURADOR DE PROGRAMAS

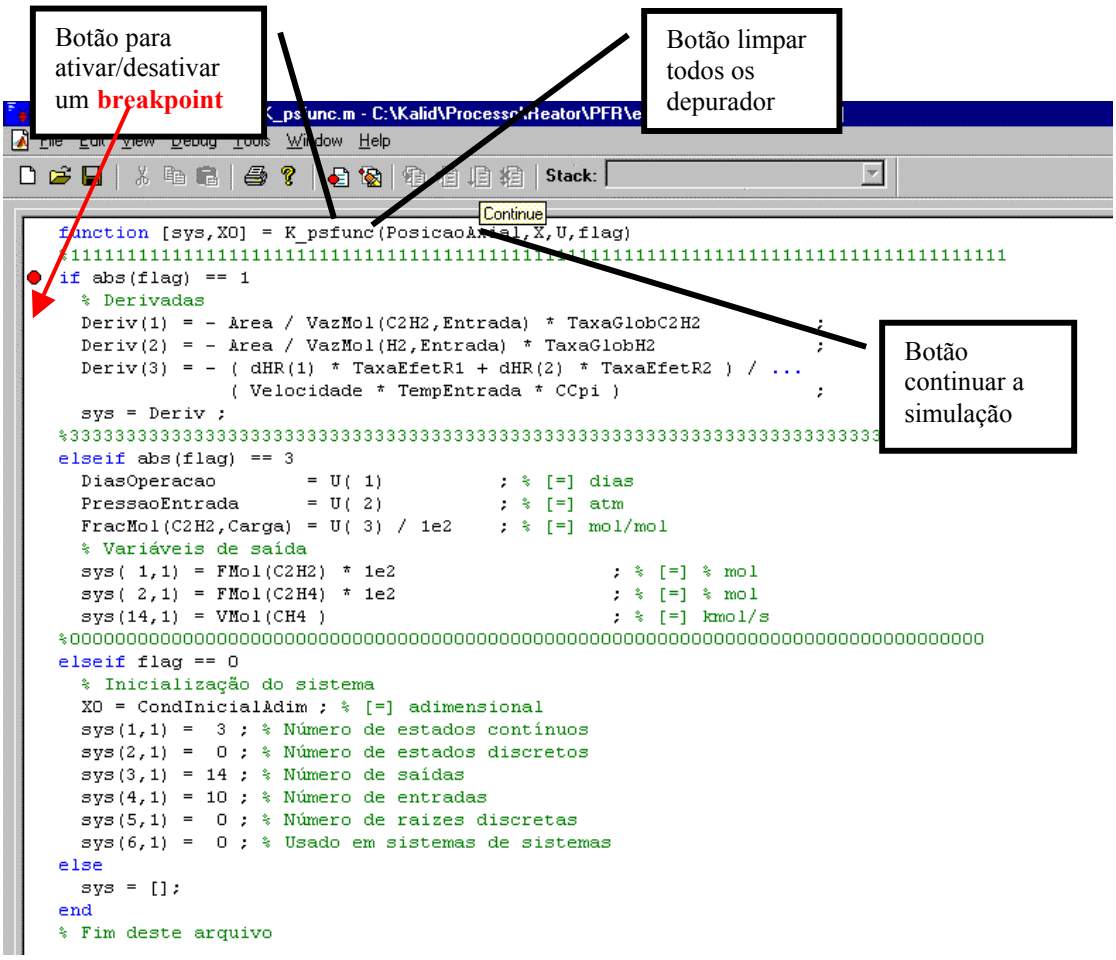
O MATLAB possui um depurador de programas que auxilia na detecção de erros de programação, sintaxe, divisão por zero, et.

Para saber suas opções digite ***help debug*** na linha de comando do MATLAB. O depurador também pode ser ativado a partir do editor de programas do MATLAB.

```
help debug
 Debugging commands.
    dbstop      - Set breakpoint.
    dbclear     - Remove breakpoint.
    dbcont      - Resume execution.
    dbdown      - Change local workspace context.
    dbmex       - Enable MEX-file debugging.
    dbstack     - List who called whom.
    dbstatus    - List all breakpoints.
    dbstep      - Execute one or more lines.
    dbtype      - List M-file with line numbers.
    dbup        - Change local workspace context.
    dbquit      - Quit debug mode.

    When a breakpoint is hit, MATLAB goes into debug mode, the
debugger window becomes active, and the prompt changes to a K>.  Any
MATLAB command is allowed at the prompt.

    To resume M-file function execution, use DBCONT or DBSTEP.

    To exit from the debugger use DBQUIT.
```

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r

O depurador de programas do MATLAB também pode ser utilizado através do editor/depurador associado ao mesmo.



É importante destacar que numa **function**, seja ela **M-function** ou **S-function** as variáveis são todas definidas apenas localmente. Assim se uma equação utiliza um parâmetro, o mesmo deve ser passado como argumento da **function**, utilizando a instrução **global** ou então o parâmetro deve ser definido dentro da mesma.

Para visualizar o valor de uma variável ou parâmetro que pertence a uma **function** a variável deve ser impressa ou utiliza-se o depurador. Neste último caso, um "breakpoint" deve ser acionado na linha imediatamente posterior ao aparecimento da variável, pois quando o MATLAB executar essa linha o processamento será suspenso, possibilitando a visualização do valor do parâmetro, através da colocação da seta do mouse sobre o parâmetro em questão.

Rua Aristides Novis, nº 2, 2º andar, 40.210-630, Federação, Salvador, Bahia
Programa de Pós-Graduação em Engenharia Química
Escola Politécnica - Universidade Federal da Bahia
Fax: (0xx71) 247.5123 - Tel: (071) 9984.3316
e-mail: l a c o i @ u f b a . b r