107484 – Controle de Processos

Aula: Simulação de sistemas dinâmicos no *Matlab & Simulink*

Prof. Eduardo Stockler Tognetti

Departamento de Engenharia Elétrica
Universidade de Brasília – UnB

1^o Semestre 2015

Introdução

Simulação de sistemas dinâmicos

1. Sistema não-linear

$$\begin{cases} \frac{d}{dt}x(t) &= f(x(t),u(t)) \\ y(t) &= g(x(t),u(t)) \end{cases} \quad \text{ou} \quad \begin{cases} \dot{x} &= f(x,u) \\ y &= g(x,u) \end{cases} \tag{1}$$

2. Sistema linear

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{2}$$

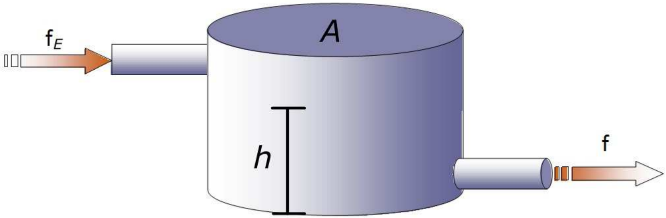
3. Função de transferência

$$Y(s) = H(s)U(s) \tag{3}$$

Ferramentas computacionais

1. Simulink
2. Matlab & Simulink
3. Matlab

Tanque de Nível



- Modelo matemático

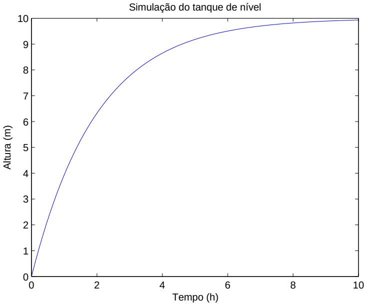
$$\frac{dh(t)}{dt} = \frac{1}{A}(f_E(t) - f(t)), \qquad h(t=0) = h_0 \tag{4}$$

- Considerando $f(t)$ proporcional à altura da coluna de líquido e inversamente proporcional a uma resistência ao escoamento (R), $f(t) = \dfrac{h(t)}{R}$,

$$\frac{dh(t)}{dt} = \frac{1}{A}(f_E(t) - \frac{h(t)}{R}), \qquad h(t=0) = h_0 \tag{5}$$

Tanque de Nível - Simulação da resposta analítica

```
% Definição das constantes do modelo
R = 1;   % h/m2
A = 2; % m2
Fe = 10; % m3/h

% Tempo de simulação
t = 0.0 : 0.01 : 10.0;   % h

% Simulação da altura de líquido
h = R*Fe*(1 - exp(-t/(R*A)));   % m

%  Visualização da simulação
plot(t,h);
title('Simulação do tanque de nível');
xlabel('Tempo (h)');
ylabel('Altura (m)');
```



Tanque de Nível - Simulação da EDO (Matlab)

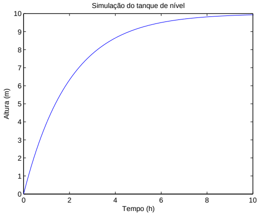
- Solvers para edo's (ode45, ode23, etc):

```
[T,Y] = solver(odefun,tspan,y0,options)
```

```
function output = tnivel3(R,A,Fe)

if nargin == 0
% Def. das ctes do modelo
R = 1;  % h/m2
A = 2; % m2
Fe = 10; % m3/h
end

% Simulação da altura de líquido
h0 = 0; t = 0.0:0.01:10.0;
[t,h] = ode45('dhdt',t, h0,[],[R A Fe]);

% Visualização da simulação
plot(t,h);
```

Arquivo `dhdt.m` :

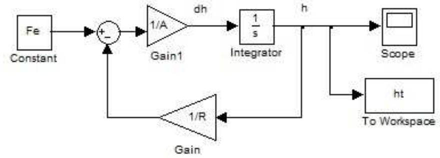
```
function dh = dhdt(t,h,flag,par)
R = par(1);
A = par(2);
Fe = par(3);
dh = (Fe-(h/R))/A;
```

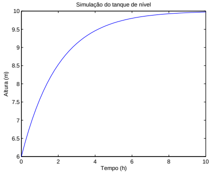


Tanque de Nível - Simulação (Simulink)

```
% Def. das ctes do modelo
R = 1;   % h/m2
A = 2; % m2
Fe = 10; % m3/h

% Tempo de simulação
tf = 10.0;   % h

% Condição inicial
h0 = 6;
```

```
% Simulação da altura de líquido
[T,H] = sim('tnivel1_sim',tf);

% Saida
t = ht.time;
h = ht.signals.values;

%  Visualização da simulação
plot(t,h);
```





- Observação:

Para executar um arquivo *Simulink* via comando `sim(.)` de dentro de uma função, é necessário carregar os parâmetros da simulação usando os comandos

```
options = simset('SrcWorkspace','current'); % ou evalin('base','h0=10');
[T,H] = sim('tnivel_sim2',tf,options);
```

Entrada variante no tempo

Simulação com entrada variante

- Como considerar uma entrada que varia no tempo ou dependente de outras variáveis (estado, parâmetros) ?

Ferramentas computacionais

1. Matlab
2. Simulink

Tanque de Nível - Simulação com Distúrbio de Entrada (Matlab)

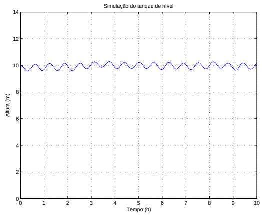
```
function output = tnivel5

% Simulação da altura de líquido
h0 = 10; t = 0.0:0.01:10.0;
[t,h] = ode45(@dhdt2,t,h0,[]);

% Visualização da simulação
plot(t,h);
```

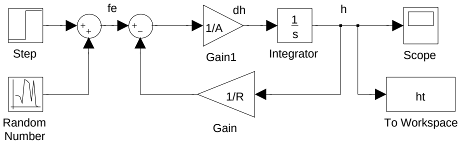
Mesmo arquivo (tnivel5.m):

```
function dh = dhdt2(t,h,flag,par)
R = 1;  % h/m2
A = 2; % m2
dh = (Fe(t)-(h/R))/A;
```

Mesmo arquivo (tnivel5.m):

```
function fe = Fe(t)
fe1 = 10*(1-0.5*sin(10*t));
fe =  awgn(fe1,10,0);
```





Representação na forma de espaço de estados

- Observe que o sistema (5) pode ser escrito na forma

$$\begin{aligned}\dot{x} &= \mathscr{A}x + \mathscr{B}u \\ y &= \mathscr{C}x + \mathscr{D}u\end{aligned} \tag{6}$$

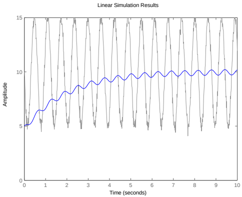
com

$$\mathscr{A} = -1/(RA); \qquad \mathscr{B} = 1/A; \qquad \mathscr{C} = 1; \qquad \mathscr{D} = 0. \tag{7}$$

Comandos matlab:

```
% Sistema
As = -1/(R*A); Bs = 1/A;
Cs = 1; Ds = 0;
sys = ss(As,Bs,Cs,Ds);

% Simulação à condição inicial
h0 = 10;
initial(sys,h0)

% Simulação ao degrau unitário (h0=0)
step(sys)
```

```
% Simulação ao distúrbio (h0=5)
t = 0.0:0.01:10.0;
fe1 = 10*(1-0.5*sin(10*t));
fe =  awgn(fe1,10,0);
lsim(sys,fe,t,h0)
```



Representação na forma de função de transferencia

- Representando o sistema (5) no domínio-s

$$H(s) = \frac{K}{\tau s + 1}, \qquad h(0) = 0 \tag{8}$$

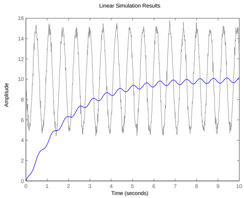
com

$$\tau = AR \quad \text{e} \quad K = R. \tag{9}$$

Comandos matlab:

```
% Sistema
[num,den] = ss2tf(As,Bs,Cs,Ds,1);
sys = tf(num,den);

% ou
sys=tf([R],[A*R 1])

% Simulação ao degrau unitário (h0=0)
step(sys)
```

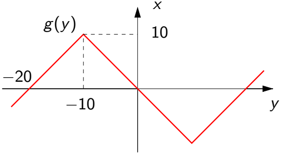
```
% Simulação ao distúrbio (h0=0)
t = 0.0:0.01:10.0;
fe1 = 10*(1-0.5*sin(10*t));
fe =  awgn(fe1,10,0);
lsim(sys,fe,t)
```



Simulando funções no Simulink

- Seja o sistema não linear

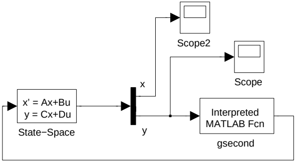
$$\begin{cases} \dot{x} = -\beta x - y \\ \dot{y} = x - g(y) \end{cases} \qquad (10)$$

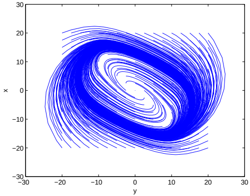


onde $g(y)$ é uma função linear por partes.

Simulação:

```
function y=gsecond(u)
% Implementa g(y)
if u < -10
    y=(u+20);
else if u > 10
        y=(u-20);
    else
        y=-u;
    end
end
```





Tanque de nível não-linear

- Considere a situação mais realística no qual o fluxo de saída é dado por

$$f(t) = C_v\sqrt{h(t)}, \qquad Cv \triangleq \frac{1}{R}. \tag{11}$$

Tem-se

$$\frac{dh(t)}{dt} = \frac{1}{A}(f_E(t) - \frac{\sqrt{h(t)}}{R}), \qquad\qquad h(t=0) = h_0. \tag{12}$$

- Em regime permanente,

$$\overline{h} = (R\overline{f}_E)^2. \tag{13}$$

- Linearizando em torno de $(\overline{f}_E, \overline{h})$, tem-se

$$\frac{d\tilde{h}(t)}{dt} = \frac{1}{A}\tilde{f}_E(t) - \frac{1}{2AR\sqrt{\overline{h}}}\tilde{h}(t) = \frac{1}{A}\tilde{f}_E(t) - \frac{1}{\tau}\tilde{h}(t), \tag{14}$$

onde

$$\tilde{h}(t) = h(t) - \overline{h}, \qquad \tilde{f}_E(t) = f_E(t) - \overline{f}_E \tag{15}$$

- No domínio-s,

$$\frac{\tilde{H}(s)}{\tilde{F}_E(s)} = \frac{K}{\tau s + 1}, \qquad K = \frac{\tau}{A}, \qquad \tau = 2AR\sqrt{\overline{h}}. \tag{16}$$

Simulação Matlab - Sistema Não-Linear

Comandos matlab:

```
[t,h] = ode23('dhdt_NL',ts,h0,
 opts,[R A fe0 fe1]);
function dh = dhdt_NL(t,h,flag,par)
% Sistema não-linear
R = par(1);
A = par(2);
fe0 = par(3:4); % degrau de entrada
dh = (f_Fe(t,fe0)-sqrt(h)/R)/A;
```

```
function fe = f_Fe(t,fe0)
%
disturbio de entrada (m3/h)
if t < 3 || t > 30
    fe = fe0(1);
else
    fe = fe0(2);
end
```

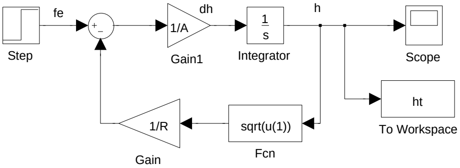


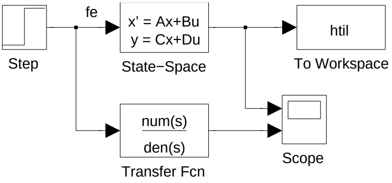
Figura: Diagrama Simulink do sistema não-linear.

Simulação Matlab - Sistema Linearizado

Comandos matlab:

```
[t,htil] = ode45('dhdt_NL_lin',
  ts, h0-hlin, opts, [R A
  fe0-felin fe1-felin hlin]);
function dhtil =
 dhdt_NL_lin(t,htil,flag,par)
R = par(1);
A = par(2);
fetil = par(3:4);
hlin = par(5);
dhtil = (f_Fe(t,fetil)
  -1/(2*R*sqrt(hlin))*htil)/A;
```

```
% Espaço de Estados:
tau = 2*A*R*sqrt(hlin);
Alin = -1/tau;
Blin = 1/A;
Clin = 1;
Dlin = 0;
% Função Transferência:
numlin = tau/A;
denlin = [tau 1];
% Simulação:
[T,H] =
 sim('tnivel_simNL_lin',tmax);
```



Simulação Matlab - Sistema Linearizado

Simulação de $h(t)$

- Simulação de $h(t)$ ao invés de $\tilde{h}(t)$
- Simulação da respota degrau
- Linearização em outros pontos de operação

Ferramentas computacionais

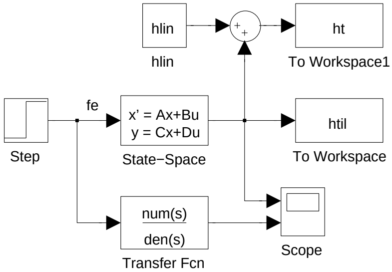
1. Matlab
2. Simulink

Simulação Matlab - Sistema Linearizado

Comandos matlab:

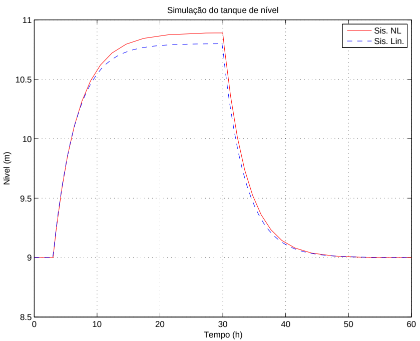
```
ht = htil +
 hlin*ones(length(htil),1);
plot(t,ht);

[t,ht2] = ode45('dhdt_NL_lin2',
 ts,h0, opts,[R A
 fe0 fe1 felin hlin]);
```

```
function dh =
 dhdt_NL_lin2(t,h,flag,par)
R = par(1);
A = par(2);
fe0 = par(3:4);
felin = par(5);
hlin = par(6);
fe = [fe0(1)-felin fe0(2)-felin];
dh = (f_Fe(t,fe)-
 1/(2*R*sqrt(hlin))*(h-hlin))/A;
```



Resposta ao degrau

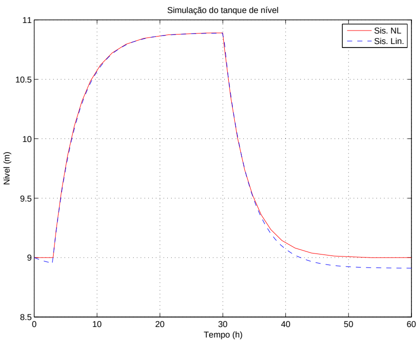
Sistema Não-linear *versus* Linearizado



- Regime Permanente $(f_e(0),\ h(0)) = (10.0,\ 9.0)$
- Linearizado em $(\overline{f}_e,\ \overline{h}) = (10.0,\ 9.0)$
- Simulação: Condição inicial $h(0) = 9.0$; Distúrbio $f_e(t) = 10.0 \implies 11.0$

Resposta ao degrau

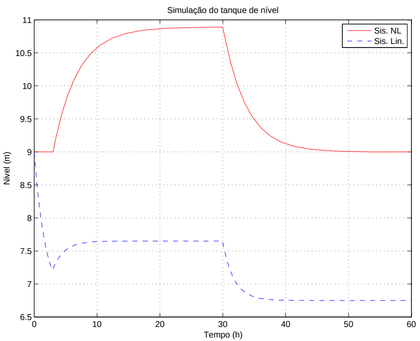
Sistema Não-linear *versus* Linearizado



- Regime Permanente $(f_e(0),\ h(0)) = (10.0,\ 9.0)$
- Linearizado em $(\overline{f}_e,\ \overline{h}) = (11.0,\ 10.9)$
- Simulação: Condição inicial $h(0) = 9.0$; Distúrbio $f_e(t) = 10.0 \Longrightarrow 11.0$

Resposta ao degrau

Sistema Não-linear *versus* Linearizado



- Regime Permanente $(f_e(0),\ h(0)) = (10.0,\ 9.0)$
- Linearizado em $(\overline{f}_e,\ \overline{h}) = (5.0,\ 2.3)$
- Simulação: Condição inicial $h(0) = 9.0$; Distúrbio $f_e(t) = 10.0 \Longrightarrow 11.0$

Sistema em malha fechada

Controle do nível pela válvula do fluxo de entrada

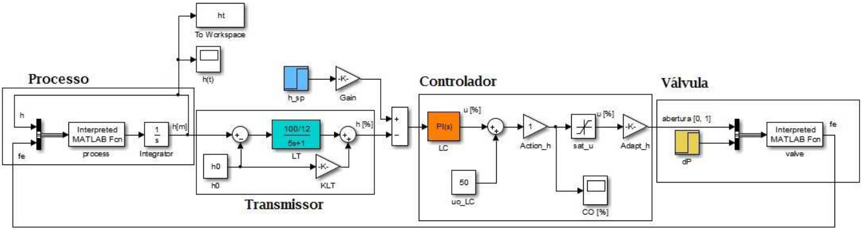
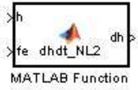


Figura: Sistema em malha fechada: sistema não-linear, transmissor de nível, controlador PI (ação direta) e válvula de controle na entrada do tanque.

ou




```
function dh = dhdt_NL2(h,fe)
% modelo não-linear do tq. de aquecimento
dh = (fe-sqrt(h)/R)/A;
```

Sistema em malha fechada

Controle do nível pela válvula do fluxo de saída do tanque

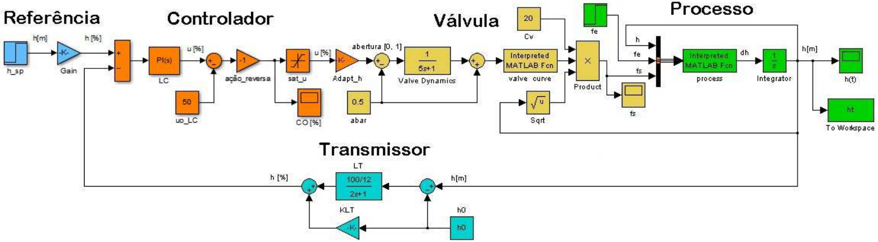


Figura: Sistema em malha fechada: sistema não-linear, transmissor de nível, controlador PI (ação reversa) e válvula de controle na saída do tanque.

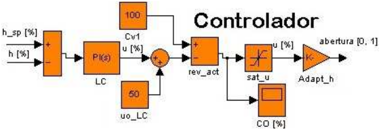


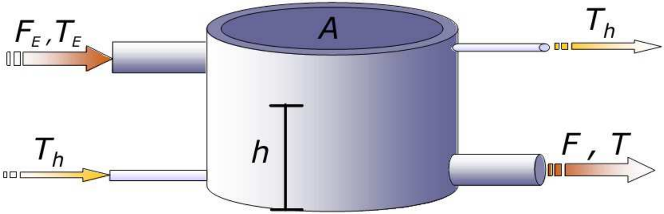
Figura: Implementação alternativa para ação reversa do controlador.

Representação de processos no *Simulink*

Observações

- O processo deve estar inicialmente em estado estacionário em um dado ponto de operação, ou seja, para entradas fixas a saída deve permanecer constante até que um degrau ou uma perturbação aconteça (geralmente as entradas e saídas são não nulas em estado estacionário);
- A referência e a saída do processo devem ser expressas em variáveis absolutas com unidades de engenharia definidas;
- O nível pode ser expresso em termos percentuais;
- Em cada parte do diagrama deve ficar claro a variável envolvida e sua correspondente unidade ou significado;
- Os ganhos também devem ter unidades de engenharia definida (ex.: ganho de um conversor I/P $[\mathrm{psi}/\mathrm{mA}]$);
- Recomenda-se que o ganho do controlador seja expresso de forma adimensional;
- A saída do transmissor pode ser expressa em mA, V ou %;
- É usual que o sinal de controle seja expresso na faixa de 0 a 100 % com saturação em 100 %.

Tanque de Aquecimento



- Modelo matemático ($\dot{Q}(t) = UA_q(T_h(t) - T(t))$, considerando-se $A_q = A$)

$$\begin{aligned}\frac{dh(t)}{dt} &= \frac{1}{A}\left(f_E(t) - \frac{\sqrt{h(t)}}{R}\right),\\ \frac{dT(t)}{dt} &= \frac{1}{h(t)}\left[\left(\frac{F_e(t)T_e(t)}{A} + \frac{UT_h(t)}{\rho c_p}\right) - T(t)\left(\frac{F_e(t)}{A} + \frac{U}{\rho c_p}\right)\right]\end{aligned} \qquad (17)$$

- Condições de regime permanente

$$\bar{h} = (R\bar{F}_e)^2, \qquad \bar{T} = \frac{\bar{F}_e\bar{T}_e\rho c_p + U\bar{T}_h A}{\bar{F}_e\rho c_p + UA} \qquad (18)$$

Tanque de Aquecimento - Simulação Matlab

```
function dx = dxdt(t,x,flag,par)
% modelo do tq. de aquecimento
U = par(1); A = par(2);
Ro = par(3); Cp  = par(4);
R = par(5);
Fe  = par(6:9);
Te = par(10:13);
Th  = par(14:17);
dx(1) = (f(t,Fe)-(sqrt(x(1))/R))/A;
dx(2) = (1/x(1))*((f(t,Fe)*f(t,Te)/A)+(U*f(t,Th)/(Ro*Cp)) -...
       ((f(t,Fe)/A)+(U/(Ro*Cp)))*x(2));
dx  = dx(:);

function out = f(t,x)
% disturbio de entrada: x = [x1 x2 t1 t2]
if t < x(3) || t > x(4)
    out = x(1);
else
    out = x(2);
end
```

Tanque de Aquecimento - Simulação Matlab

```
% Parâmetros
R = 0.3;  % h/m2
A = 2;  % m2
cp = 0.75;  % kJ/(kg.K)
rho = 1000; % kg/m3
U = 150;  % kJ/(m2.s.K)
param = [U A rho cp R];
```

```
% Entradas
t1 = 2; t2 = 30;
Fe = [10 11 t1 t2]; % m3/h
Te = [530 520 t1 t2]; % K
Th = [540 540 t1 t2];  % K
inputs = [Fe Te Th];
```

```
% Tempo de simulação
ts = 0.0:0.01:60.0; % h
% Regime permanente
h0 = (Fe(1)*R)^2;
T0 = (Fe(1)*Te(1)*rho*cp+U*Th(1)*A)/(Fe(1)*rho*cp+U*A);
% Simulação
[t,x] = ode45('dxdt',ts,[h0 T0],[],[param inputs]);
figure(1); plot(t,x(:,1)); ylabel ('Altura (m)');
figure(2); plot(t,x(:,2)); ylabel ('Temperatura (K)');
```

Tanque de Aquecimento - Simulação Matlab

Resposta á distúrbio em $F_e(t)$ e $T_e(t)$ em $t = 2$s e $t = 30$s