

# TCA: Controle de Processos

## Simulação de Sistemas Dinâmicos

Eduardo Stockler Tognetti

Departamento de Engenharia Elétrica  
Universidade de Brasília - UnB

8 de janeiro de 2013

# Introdução

## Simulação de sistemas dinâmicos

- 1 Sistema não-linear

$$\frac{d}{dt}x(t) = f(x(t)) \quad \text{ou} \quad \dot{x} = f(x, u) \quad (1)$$

- 2 Sistema linear

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

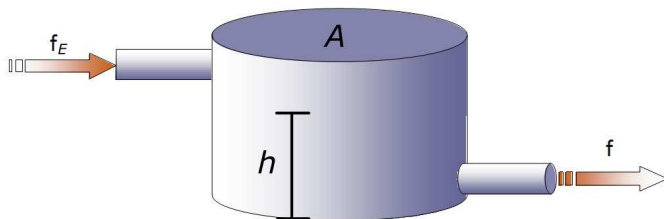
- 3 Função de transferência

$$Y(s) = H(s)U(s) \quad (3)$$

## Ferramentas computacionais

- 1 Simulink
- 2 Matlab & Simulink
- 3 Matlab

# Tanque de Nível



- Modelo matemático

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

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

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

# 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
```

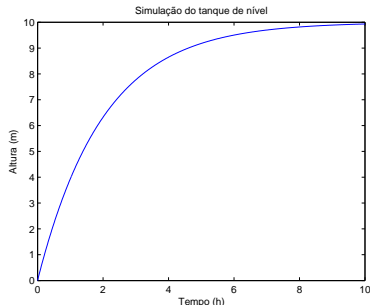
```
% Visualização da simulação
```

```
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):

$$[T,Y] = \text{solver}(\text{odefun}, \text{tspan}, \text{y0}, \text{options})$$

```
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 :

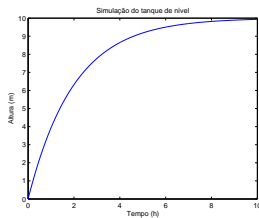
```
function dh = dhdt(t,h,flag,par)
```

```
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);
```

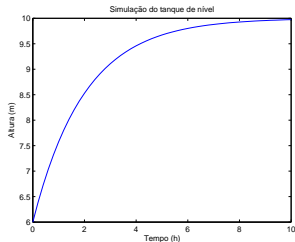
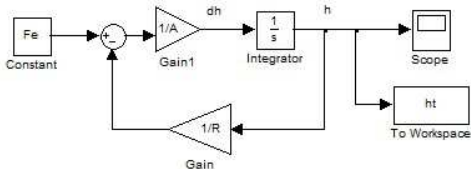
```
% Saida
```

```
t = ht.time;
```

```
h = ht.signals.values;
```

```
% Visualização da simulação
```

```
plot(t,h);
```



# 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
```

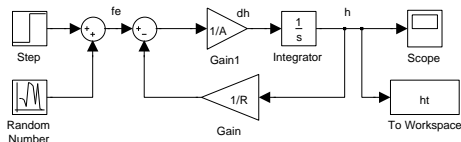
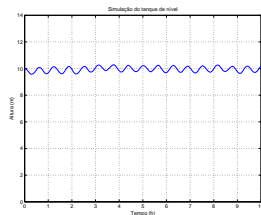
```
dh = (Fe(t)-(h/R))/A;
```

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

$$\begin{aligned}\dot{x} &= \mathcal{A}x + \mathcal{B}u \\ y &= \mathcal{C}x + \mathcal{D}u\end{aligned}\quad (6)$$

com

$$\mathcal{A} = -1/(RA); \quad \mathcal{B} = 1/A; \quad \mathcal{C} = 1; \quad \mathcal{D} = 0. \quad (7)$$

Comandos matlab:

```
% Sistema
```

```
As = -1/(R*A); Bs = 1/A;
```

```
Cs = 1; Ds = 0;
```

```
sys = ss(As,Bs,Cs,Ds);
```

```
% 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)
```

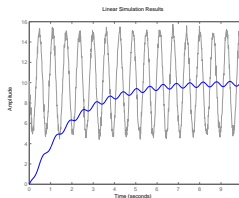
```
% Simulação à condição inicial
```

```
h0 = 10;
```

```
initial(sys,h0)
```

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

```
step(sys)
```



# Representação na forma de função de transferencia

- Representando o sistema (5) no domínio-s

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

com

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

Comandos matlab:

```

% 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)

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

```

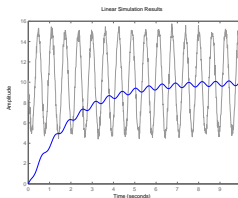
% ou

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

```

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

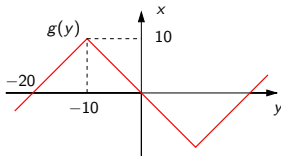
```



# Simulando funções no Simulink

- Seja o sistema não linear

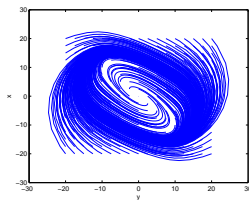
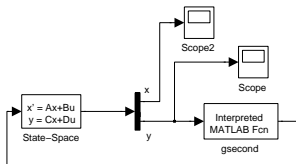
$$\begin{cases} \dot{x} = -\beta x - y \\ \dot{y} = x - g(y) \end{cases} \quad (10)$$



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

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

Tem-se

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

- Em regime permanente,

$$\bar{h} = (R\bar{f}_E)^2. \quad (13)$$

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

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

onde

$$\tilde{h}(t) = h(t) - \bar{h}, \quad \tilde{f}_E(t) = f_E(t) - \bar{f}_E \quad (15)$$

- No domínio-s,

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

# 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
```

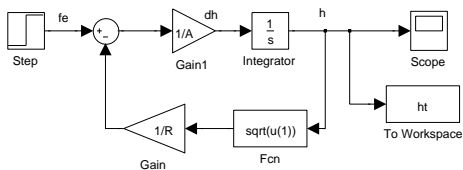


Figura : Diagrama Simulink do sistema não-linear.

# Simulação Matlab - Sistema Linearizado

Comandos matlab:

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

**% Espaço de Estados:**

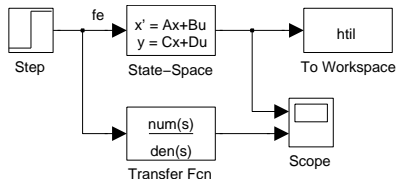
```
tau = 2*A*R*sqrt(hlin);
A_lin = -1/tau;
B_lin = 1/A;
C_lin = 1;
D_lin = 0;
```

**% 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 resposta degrau
- Linearização em outros pontos de operação

## Ferramentas computacionais

- 1 Matlab
- 2 Simulink

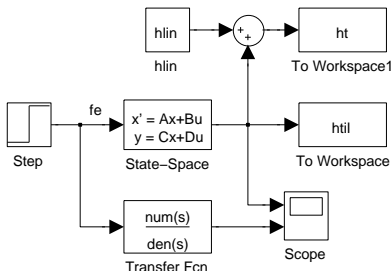
# Simulação Matlab - Sistema Linearizado

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]);
```

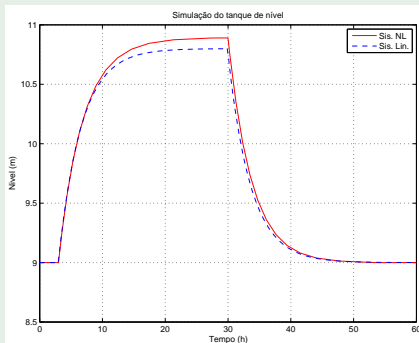
```
function dh =
    dhdt_NL_lin2(t,h,flag,par)
R = par(1);
A = par(2);
fe0 = par(3:4);
felin = par(5);
hlin = par(6);
fe = [fe0(1)-felin fe0(2)-felin];
dh = (f_Fe(t,fe)-
    1/(2*R*sqrt(hlin))*(h-hlin))/A;
```





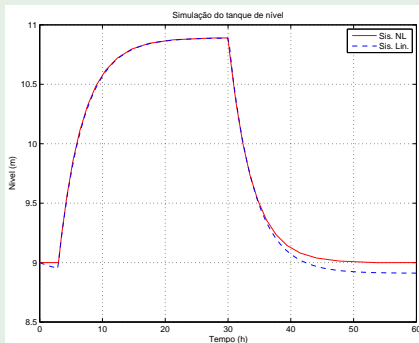
# Resposta ao degrau

## Sistema Não-linear versus Linearizado



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

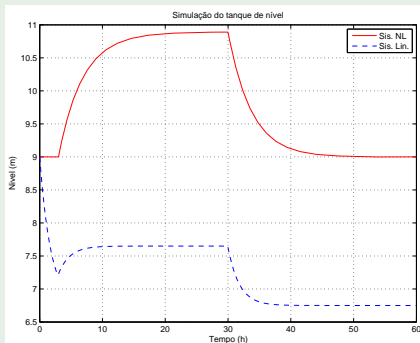
## Resposta ao degrau

Sistema Não-linear *versus* Linearizado

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

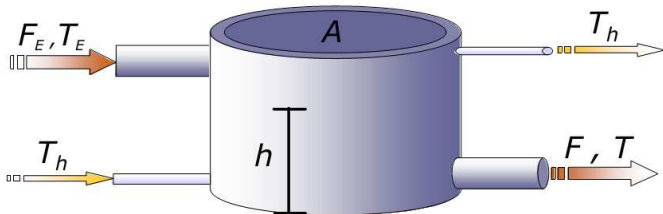
## Resposta ao degrau

## Sistema Não-linear versus Linearizado



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

# Tanque de Aquecimento



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

$$\frac{dh(t)}{dt} = \frac{1}{A} \left( f_E(t) - \frac{\sqrt{h(t)}}{R} \right), \quad (17)$$

$$\frac{dT(t)}{dt} = \frac{1}{h(t)} \left[ \left( \frac{F_e(t)T_e(t)}{A} + \frac{UT_h(t)}{\rho c_p} \right) - T(t) \left( \frac{F_e(t)}{A} + \frac{U}{\rho c_p} \right) \right]$$

- Condições de regime permanente

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

# 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)
% distúrbio 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];
```

```
% 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)');
```

```
% 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];
```

# Tanque de Aquecimento - Simulação Matlab

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

