

Identificação de Sistemas Dinâmicos – Caixa Cinza - idGrey

Consideração preliminar:

# Identificação das Matrizes de Estado

Adolfo Bauchspiess  
ENE/UnB

Representar as equações de Estado como

$$\mathbf{x}(k) = \text{Regressores} * \text{Parâmetros}$$

*Em analogia com:*  $y(k) = \psi(k-1)^T \theta + e(k)$

$$\mathbf{x}(k) = \Phi \mathbf{x}(k-1) + \Gamma \mathbf{u}(k-1) + \mathbf{e}(k)$$

$$\mathbf{x}(k) = [\Phi \quad \Gamma] \begin{bmatrix} \mathbf{x}(k-1) \\ \mathbf{u}(k-1) \end{bmatrix}$$

$$\mathbf{x}(k) = \Delta^T \mathbf{m}(k-1)^T$$

$$\mathbf{x}(k)^T = \mathbf{m}(k-1) \Delta$$

$$\mathbf{x}(k)^T = [\mathbf{x}(k-1)^T \quad \mathbf{u}(k-1)^T] \begin{bmatrix} \Phi^T \\ \Gamma^T \end{bmatrix}$$

$$(y(k) = \psi(k-1)^T \theta + e(k))$$

Para N instâncias desta equação:

$$\begin{bmatrix} \mathbf{x}(1)^T \\ \mathbf{x}(2)^T \\ \vdots \\ \mathbf{x}(N)^T \end{bmatrix} = \begin{bmatrix} \mathbf{m}(0) \\ \mathbf{m}(1) \\ \vdots \\ \mathbf{m}(N-1) \end{bmatrix} \Delta$$
$$X_N = M \Delta$$

$\Delta$  pode ser obtida resolvendo-se a pseudo-inversa:

$$\hat{\Delta}_{MQ} = [M^T M]^{-1} M^T X_N$$

Ou de forma recursiva:

$$\begin{cases} K_k = \frac{P_{k-1} \mathbf{m}(k-1)^T}{1 + \mathbf{m}(k-1)^T P_{k-1} \mathbf{m}(k-1)} \\ \hat{\Delta}_k = \hat{\Delta}_{k-1} + K_k [\mathbf{x}(k)^T - \mathbf{m}(k-1) \hat{\Delta}_{k-1}] \\ P_k = P_{k-1} - \frac{P_{k-1} \mathbf{m}(k-1)^T \mathbf{m}(k-1) P_{k-1}^T}{1 + \mathbf{m}(k-1)^T P_{k-1} \mathbf{m}(k-1)} \end{cases}$$

Identificação de Sistemas Dinâmicos

# Identificação Caixa Cinza

Adolfo Bauchspiess  
ENE/UnB

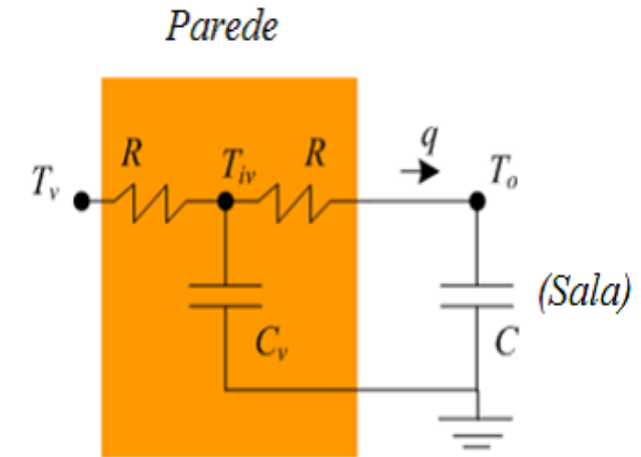
# Estimação Caixa Cinza

- Conhecimento à priori
  - estrutura,
  - leis físicas,
  - princípios fundamentais
  
- Informação auxiliar

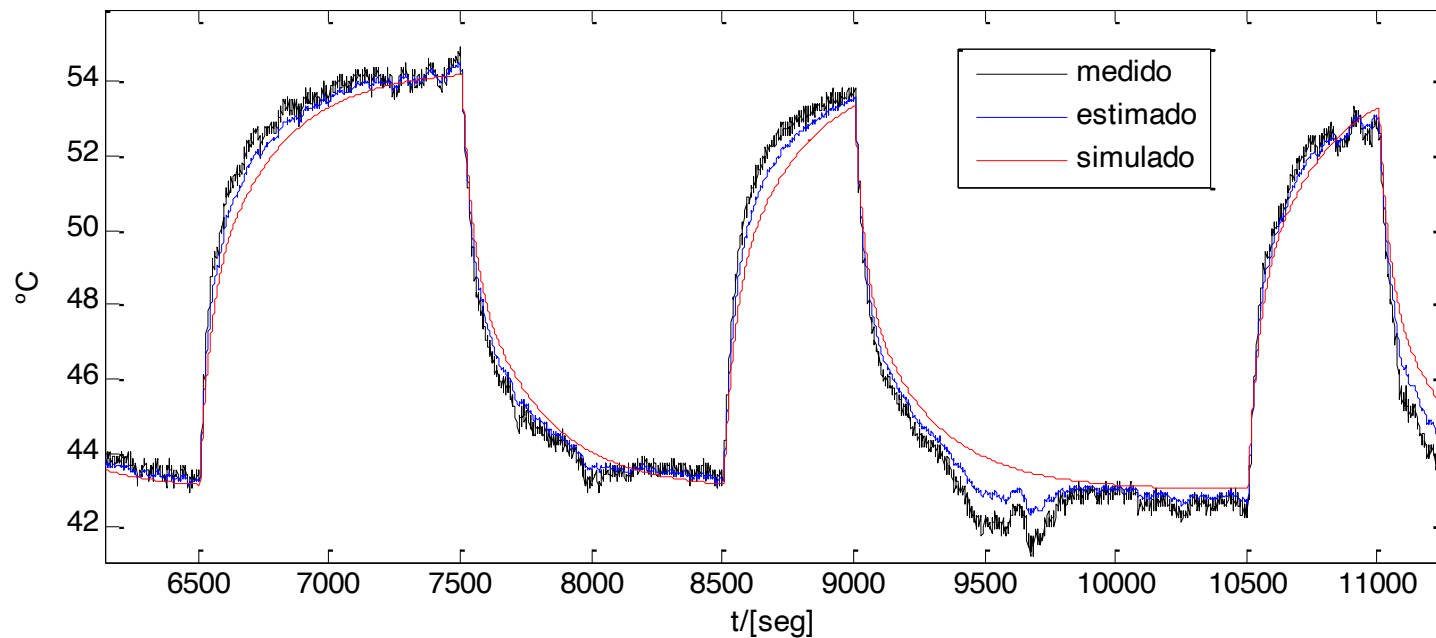
# Ex.1 – Processo Térmico

$$q = K(T_{iv} - T_o); q_v = K_v(T_v - T_{iv}) \text{ - "condução"}$$

- Princípio Fundamental: fluxo de calor é proporcional à diferença de temperaturas



Identificação Processo Térmico - Filtro de Kalman



Ex.: Modelo a Parâmetros Concentrados de climatização

## Analogia:

Capacitor elétrico - Capacitância Térmica

Resistor – transmitância térmica

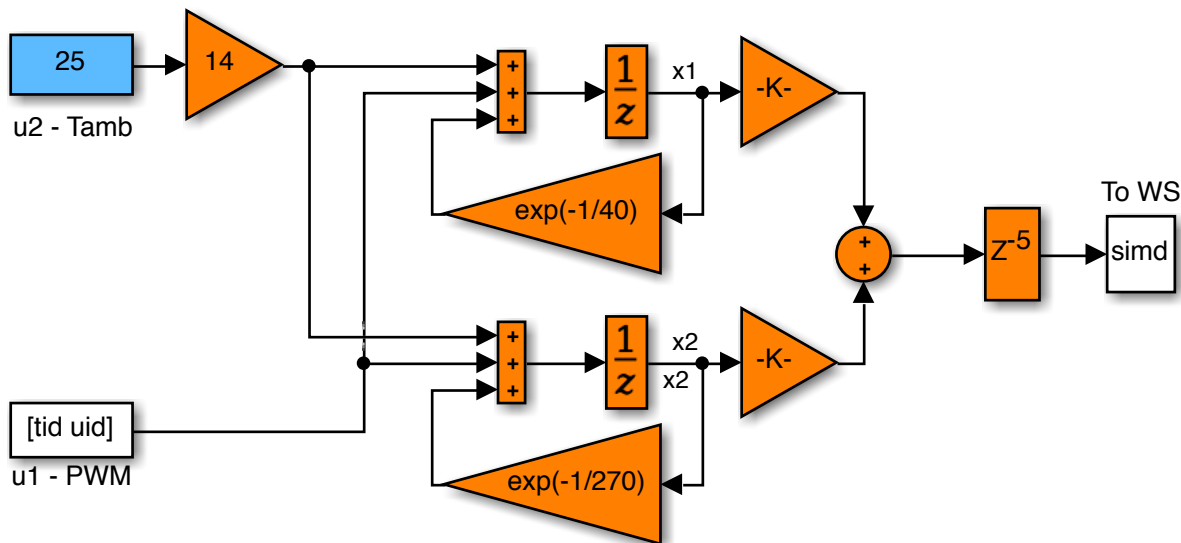
Corrente – fluxo de calor  $q$

# Forma Modal – desacoplamento entre modos

$$Y(z)*z^5 = G1(z)*U1(z) + G2(z)*U2(z)$$

- Expansão em frações
- ssest/n4sid

```
sys2=n4sid(ze,2,'Form','modal','InputDelay',5);
```

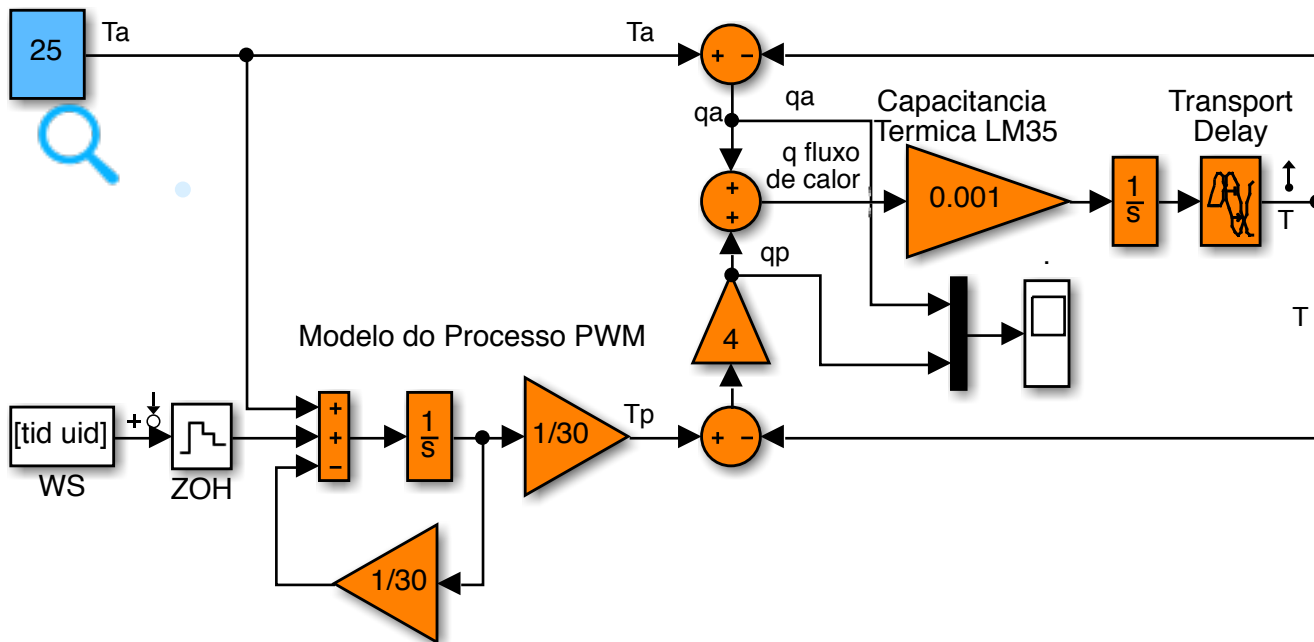


```
%% x(k+1)=phi*x(k)+gama*u(k)
% y(k) = H*x(k) + J*u(k)
phi=[exp(-1/40) 0 0 0 0 0
      0 exp(-1/270) 0 0 0 0
      (1-exp(-1/40))*0.0364 (1-exp(-1/270))*0.0521 0 0 0 0
      0 0 1 0 0 0
      0 0 0 1 0 0
      0 0 0 0 1 0];
gama=[1 14
      1 14
      0 0
      0 0
      0 0
      0 0
      0 0];
Hr=[0 0 0 0 0 0 1];
```

# Modelo Princípios Fundamentais

$$q_p = K (T_p - T_o)$$

$$q_a = K_a (T_a - T)$$



%Modelo discreto do processo térmico  
no espaço de estados

%  $x(k+1) = \text{phi} * x(k) + \text{gama} * u(k)$

%  $y(k) = C * x(k)$

phi = [exp(-1/30)

(1-exp(-1/30)) \* 4 \* 0.001

0

0

0

0

0

0

gama = [1 1

0 0.001

0 0

0 0

0 0

0 0

0 0

0 0]

H = [0 0 0 0 0 0 1];

0 0 0 0 0 0

1 0 0 0 0 -5\*0.001

1 0 0 0 0 0

0 1 0 0 0 0

0 0 1 0 0 0

0 0 0 1 0 0

0 0 0 0 1 0]

# MQR – MQ com Restrições

$$\mathbf{y} = \Psi \hat{\boldsymbol{\theta}} + \boldsymbol{\xi}$$

*minimizar a Função de Custo  $\boldsymbol{\xi}^T \boldsymbol{\xi}$*

*$n_r$  restrições sobre os parâmetros*

$$\mathbf{c} = S\boldsymbol{\theta}$$

$\mathbf{c} \in \mathbb{R}^{n_r \times 1}$  e  $S \in \mathbb{R}^{n_r \times n_\theta}$  são conhecidos.

$$\hat{\boldsymbol{\theta}}_{\text{MQR}} = \arg \min_{\boldsymbol{\theta} : \mathbf{c} = S\boldsymbol{\theta}} [\boldsymbol{\xi}^T \boldsymbol{\xi}]$$

$$\hat{\boldsymbol{\theta}}_{\text{MQR}} = (\Psi^T \Psi)^{-1} \Psi^T \mathbf{y} - (\Psi^T \Psi)^{-1} S^T [S (\Psi^T \Psi)^{-1} S^T]^{-1} (S \hat{\boldsymbol{\theta}}_{\text{MQ}} - \mathbf{c})$$



# Ex. MQR – Característica Estática

$$y(k) = 1,2796y(k-1) + 0,0178u(k-2)u(k-1) + 0,0408u(k-1)^2 - 0,3668y(k-2) - 0,2565u(k-2)y(k-1) + 0,2205u(k-2)y(k-2) + 0,0029u(k-2)^2,$$

$$\bar{y} = \frac{\Sigma_{u^2} \bar{u}^2}{1 - \Sigma_y - \Sigma_{yu} \bar{u}}$$

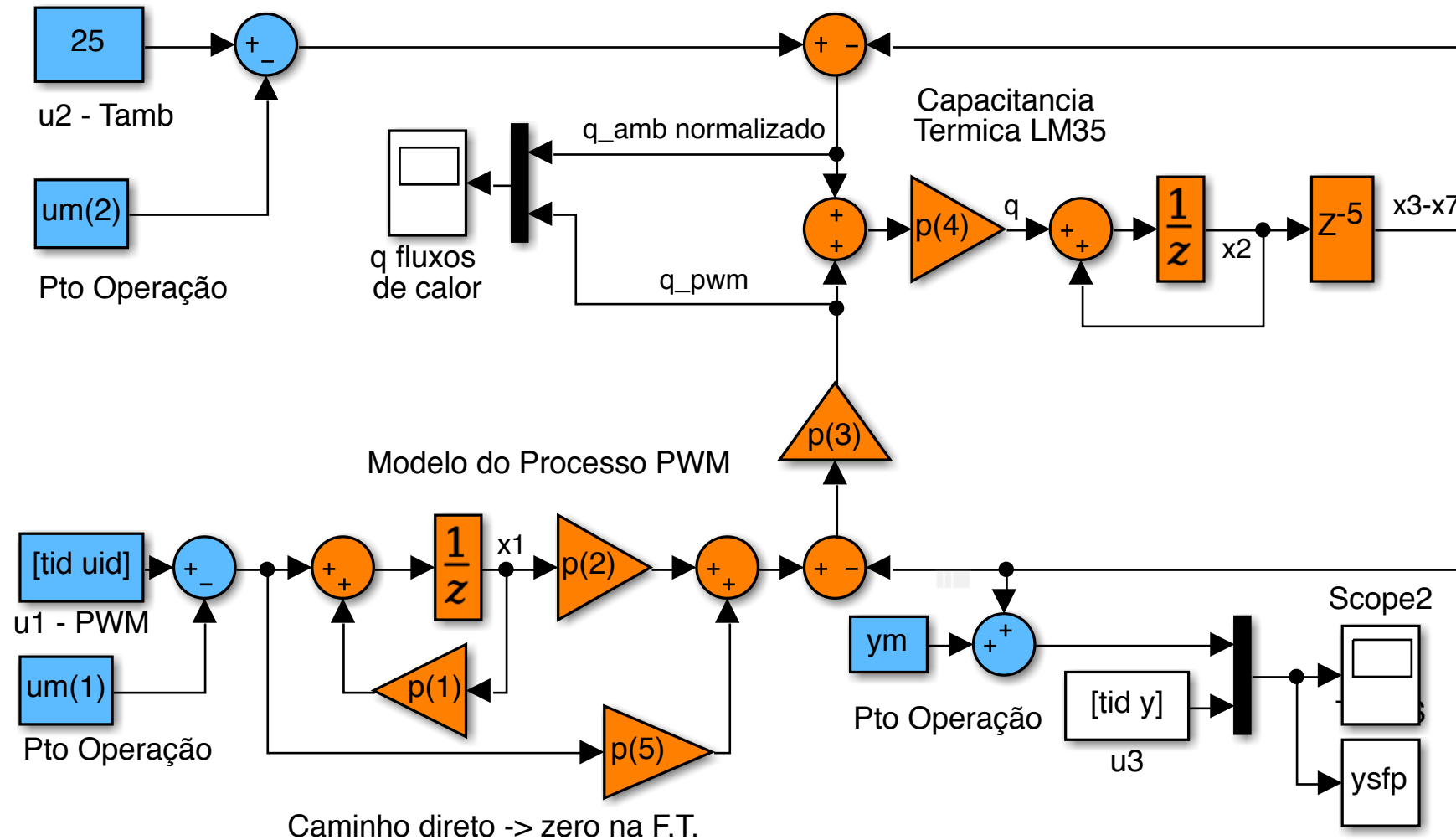
$$\mathbf{c} = [\Sigma_{u^2} \quad \Sigma_{yu} \quad \Sigma_y]^T = [0,0615 \quad -0,0360 \quad 0,9128]^T$$

$$\begin{bmatrix} 0,0615 \\ -0,0360 \\ 0,9128 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \end{bmatrix}$$

$$\bar{y} = \frac{0,0615 \bar{u}^2}{1 - 0,9128 + 0,0360 \bar{u}}, \quad \text{Característica Estática}$$

# Identificação EE com parametrização Estruturada



# Idss com parametrização Estruturada

```
gs.Structure.a.Free=...
```

```
[1 0 0 0 0 0 0  
1 0 0 0 0 0 1  
0 0 0 0 0 0 0  
0 0 0 0 0 0 0  
0 0 0 0 0 0 0  
0 0 0 0 0 0 0  
0 0 0 0 0 0 0  
0 0 0 0 0 0 0];
```

```
gs.Structure.b.Free=...
```

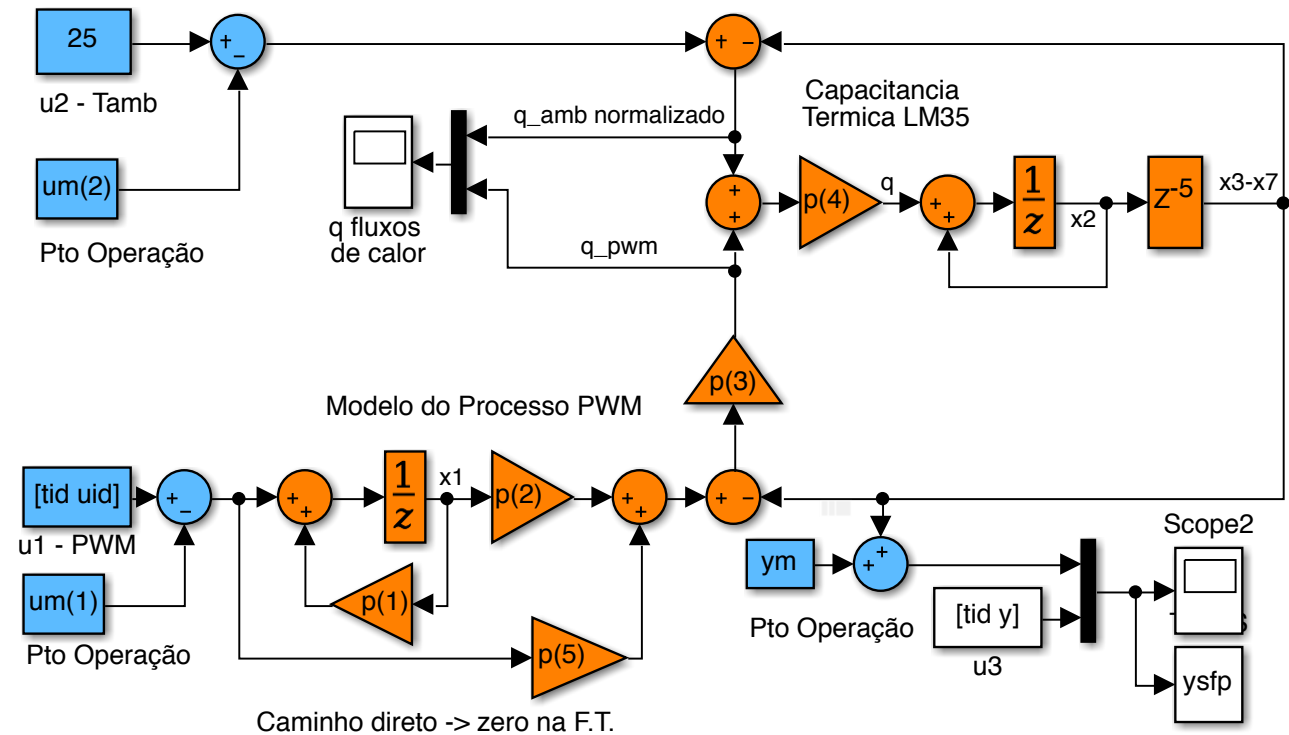
```
[0 0;1 1;0 0;0 0;0 0;0 0;0 0];
```

```
gs.Structure.c.Free=[0 0 0 0 0 0 0];
```

```
gs.Structure.d.Free=0;
```

```
set(gs, 'Ts', Ts); % modelo contínuo
```

```
gss=ssest(ze, gs);
```



# greyest com parametrização Estruturada

```
function [A,B,C,D,K,x0] = myFirst_id3 (par,Ts,aux)
```

```
A = [par(1) 0 0 0 0 0 0
      par(2)*par(3)*par(4) 1 0 0 0 0 -(par(3)+1)*par(4)
      0 1 0 0 0 0 0
      0 0 1 0 0 0 0
      0 0 0 1 0 0 0
      0 0 0 0 1 0 0
      0 0 0 0 0 1 0];
```

```
B = [1 0
      par(3)*par(4)*par(5) par(4)
```

```
% 0 par(4)
```

```
0 0
```

```
0 0
```

```
0 0
```

```
0 0
```

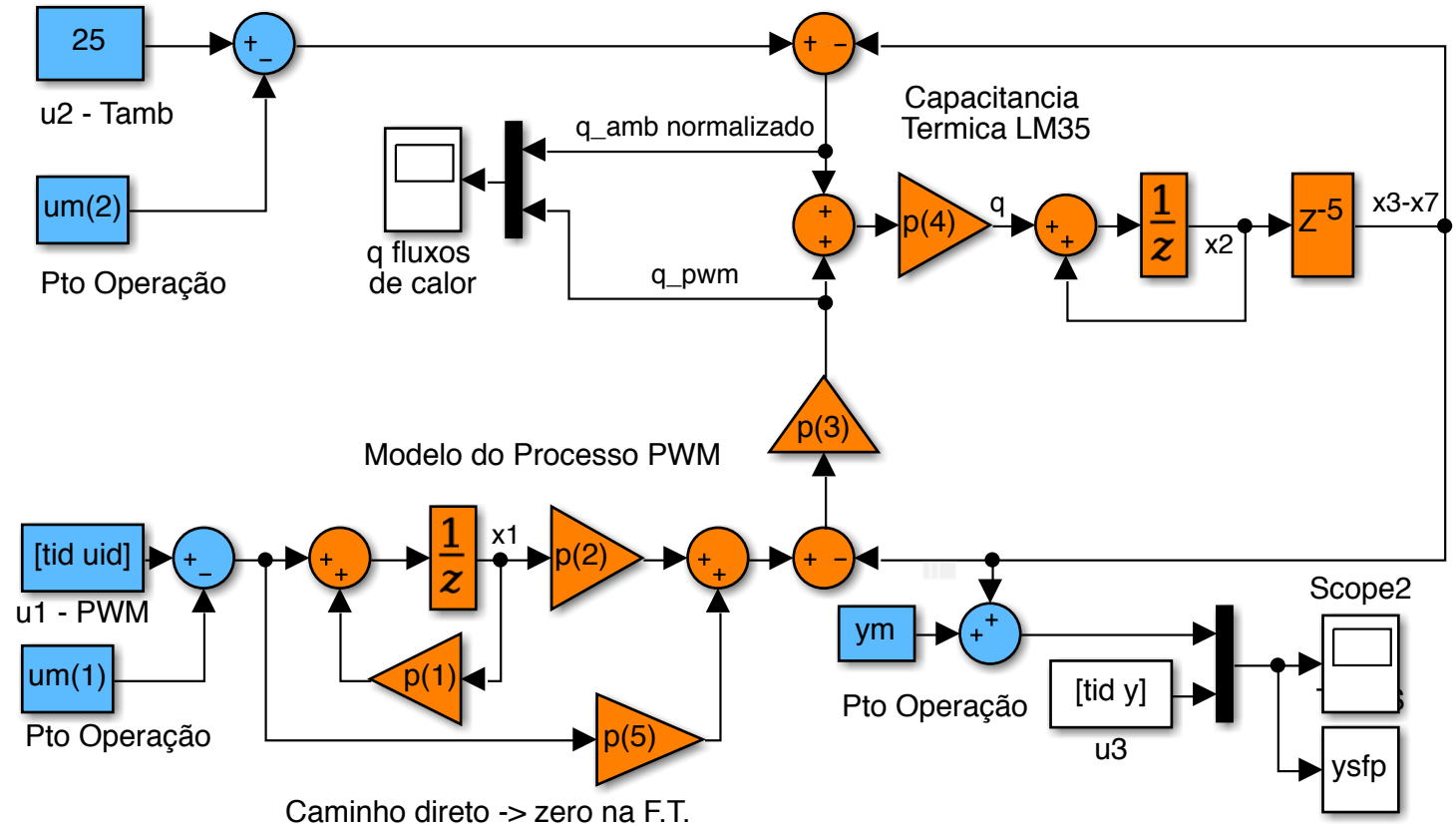
```
0 0];
```

```
C=[0 0 0 0 0 0 1];
```

```
D = zeros(1,2);
```

```
K = zeros(7,1);
```

```
x0 =[0; 0; 0; 0; 0; 0; 0];
```



# greyest com parametrização Estruturada

```
par=[exp(-1/40) 0.1 2.5 .001 1]; %I.C,
```

```
gr=idgrey('myFirst_id3',par,'cd',aux,Ts);
```

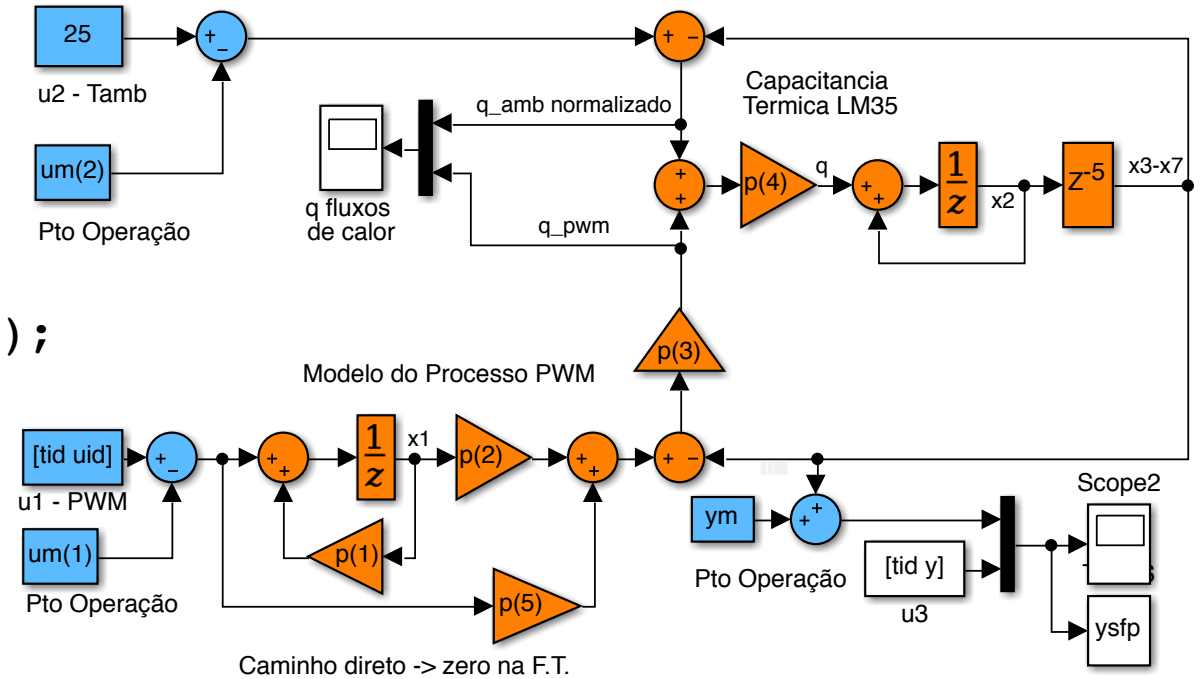
```
ggr=greyest(ze,gr);
```

```
% Simula sistema identificado
```

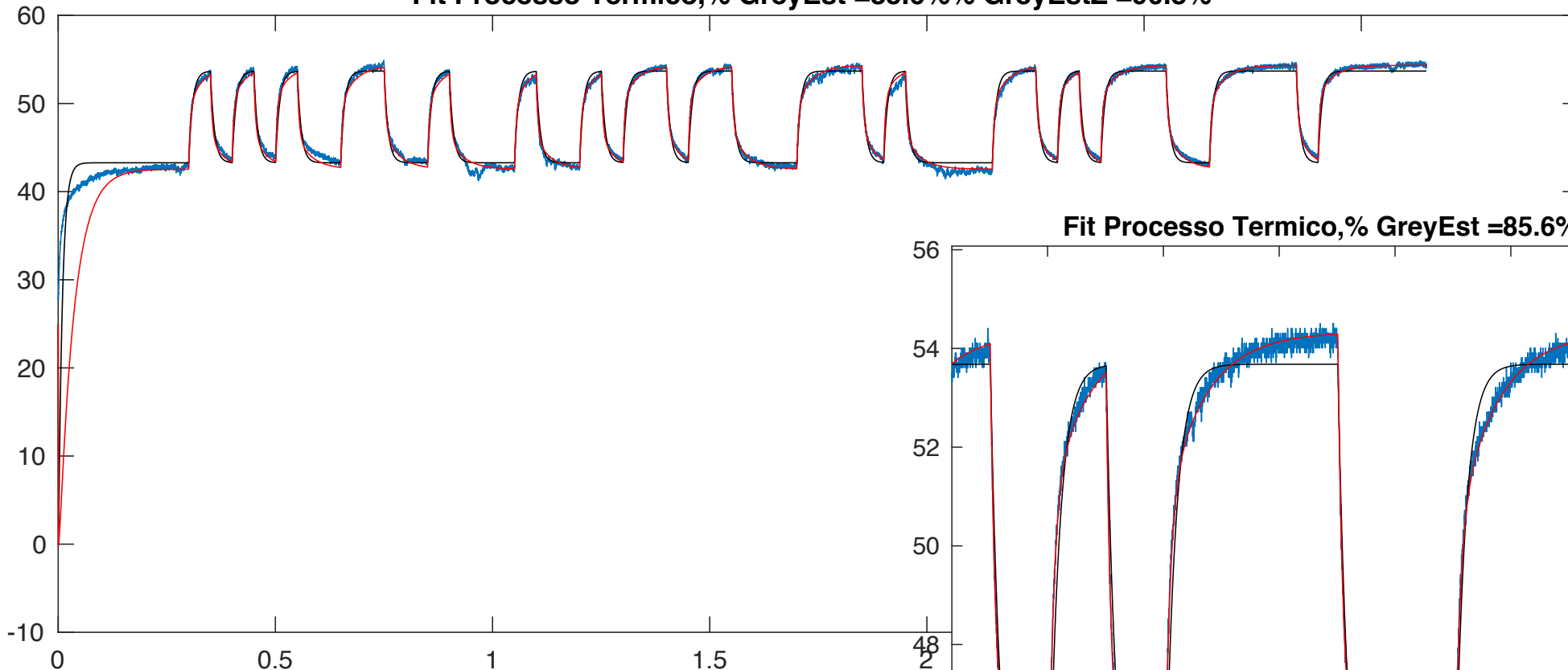
```
x0=[0 0 25 25 25 25 25]'-ym;
```

```
gn=ss(phi,gama,Hr,0,1);
```

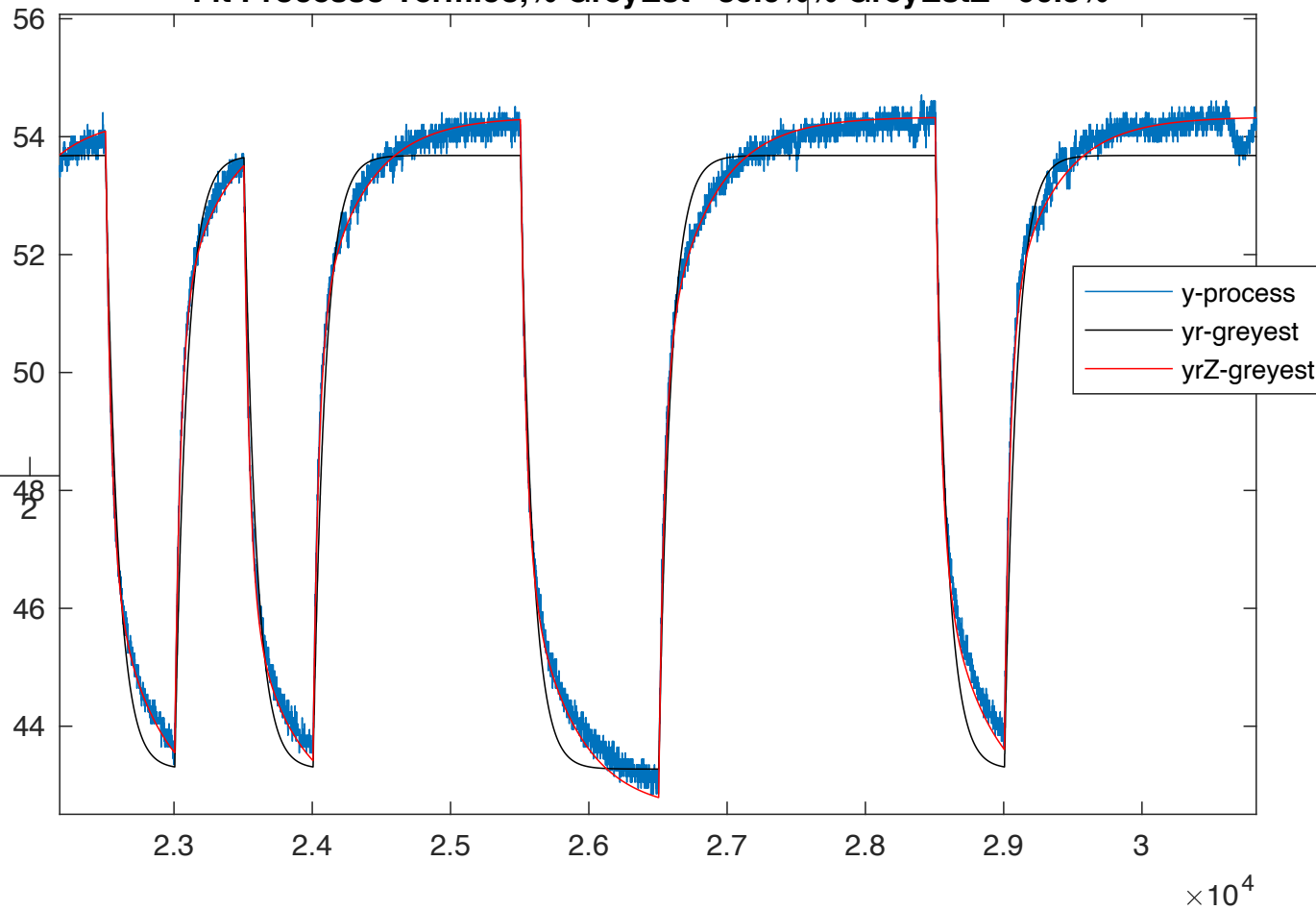
```
yr=lsim(ggr,u-um,t,x0)+ym;
```



Fit Processo Termico,% GreyEst =85.6%% GreyEstZ =90.8%



Fit Processo Termico,% GreyEst =85.6%% GreyEstZ =90.8%



Greyest- First Principles

$\times 10^4$

# Greyest- First Principles

phi (inicial) =

0.9753	0	0	0	0	0	0	0
0.0001	1.0000	0	0	0	0	0	-0.0050
0	1.0000	0	0	0	0	0	0
0	0	1.0000	0	0	0	0	0
0	0	0	1.0000	0	0	0	0
0	0	0	0	1.0000	0	0	0
0	0	0	0	0	1.0000	0	0
0	0	0	0	0	0	1.0000	0

ggr.a =

0.9763	0	0	0	0	0	0	0
-0.0000	1.0000	0	0	0	0	0	-0.0029
0	1.0000	0	0	0	0	0	0
0	0	1.0000	0	0	0	0	0
0	0	0	1.0000	0	0	0	0
0	0	0	0	1.0000	0	0	0
0	0	0	0	0	1.0000	0	0
0	0	0	0	0	0	1.0000	0

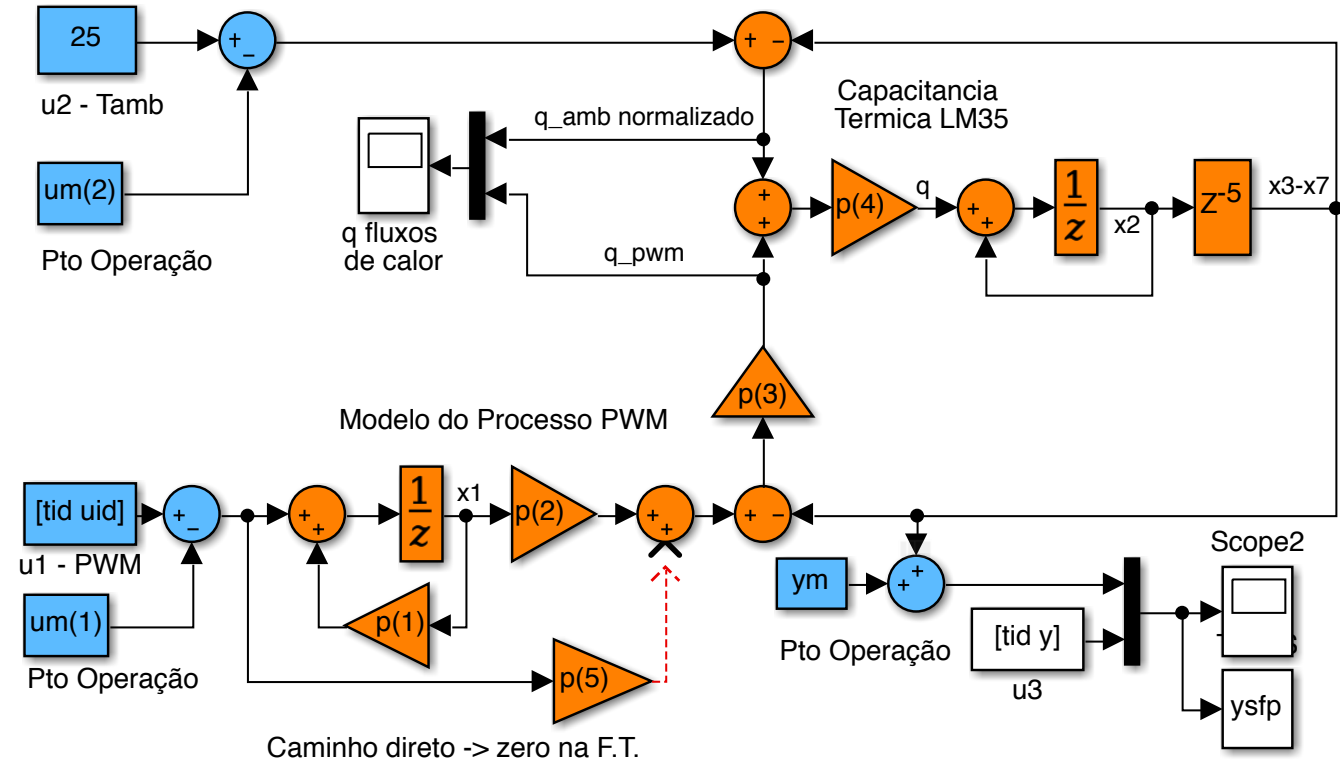
Gama(inicial) =

0.0714	1.0000	1.0000	0
1.0000	0.1000	0	0.0008
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0
0	0	0	0

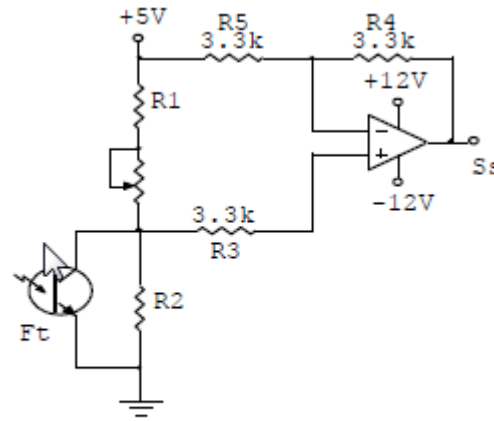
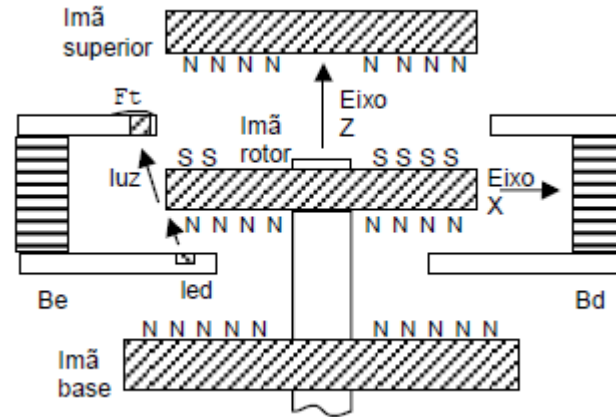
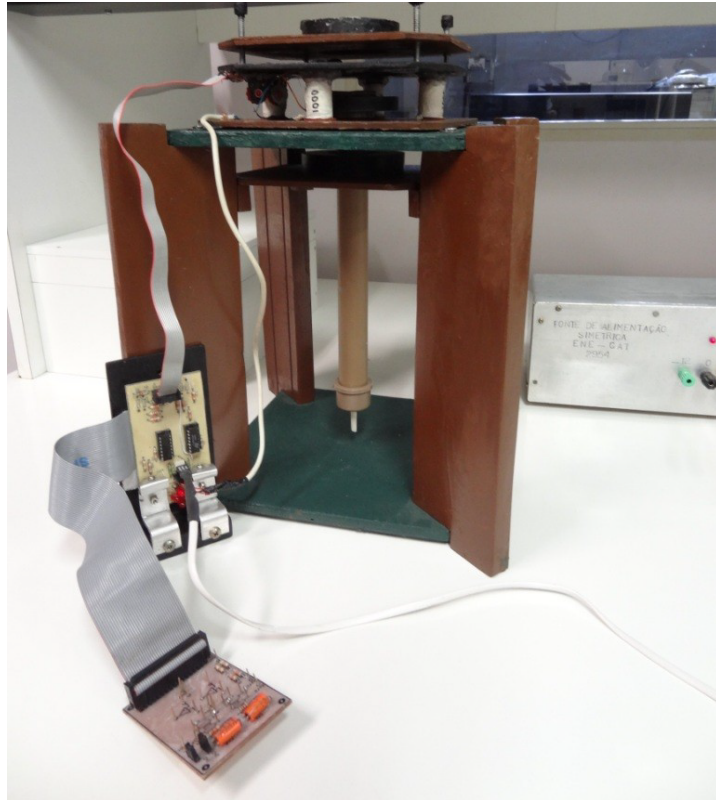
ggrz.b =

p =

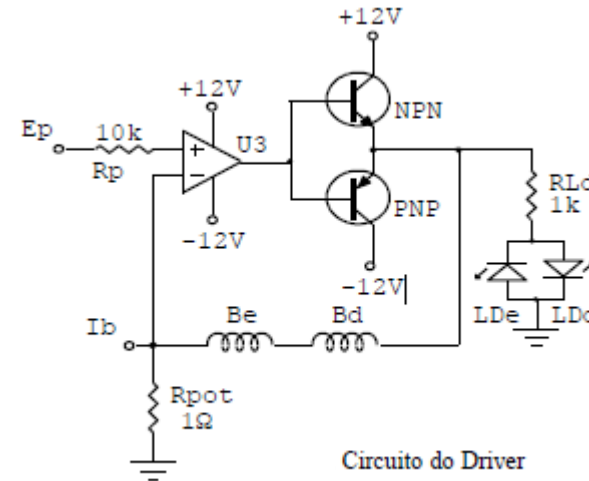
-0.9743
0.2252
2.4810
0.0031
1.0000



# Ex. 2 Levitação Magnética



Circuito sensor de posição



Circuito do Driver



# Ex. 2 Levitação Magnética

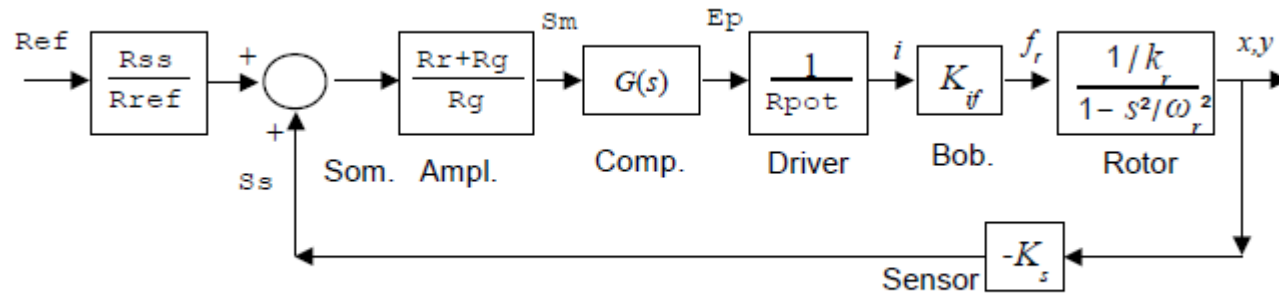
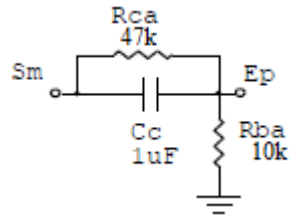


Diagrama de blocos em um dos canais

$$\frac{X(s)}{F_x(s)} = \frac{1/k_r}{1 - \frac{s^2}{\omega_r^2}}$$

Processo



Compensador em avanço

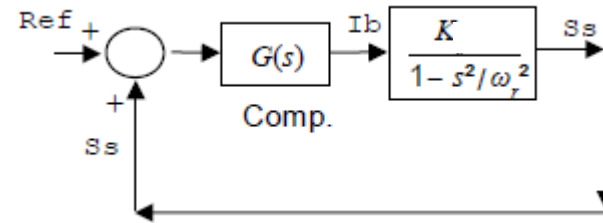


Diagrama de blocos simplificado

(K=Ganho de malha)

$$D(s) = \frac{s+z}{s+p} = \frac{s+21}{s+121}$$

$$G_{mf}(s) = \frac{-K\omega_r^2(s+z)}{s^3 + ps^2 + (K-1)\omega_r^2 s + (K-1)\omega_r^2}$$

## Ex. 2 idss

```
ze{i}=iddata(y-mean(y),u-mean(u),T);

gtf{i}=tfest(ze{i},3,1);          % Id. Caixa Preta – estrutura: 3 pólos 1 zero

% idss (fixa pólo em comp. Avnaço em -121) - 4 par. livres.
par=[-2e5;-7500;-1.3e4;-220]; % valores iniciais ~gtf

A = [0 1 0; 0 0 1; par(1) par(2) -121];
B = [0; 0; 1];
C = [par(3) par(4) 0];
D = zeros(1,1);
gs=idss(A,B,C,D);

gs.Structure.a.Free=[0 0 0; 0 0 0;1 1 0];
gs.Structure.b.Free=[0; 0; 0];
gs.Structure.c.Free=[1 1 0];
gs.Structure.d.Free=0;
set(gs,'Ts',0); % modelo contínuo

%gss = ssest(ze,gss,ssestOptions('Display','on'));
gss{i}=ssest(ze{i},gs);
```

## Ex. 2 idgrey

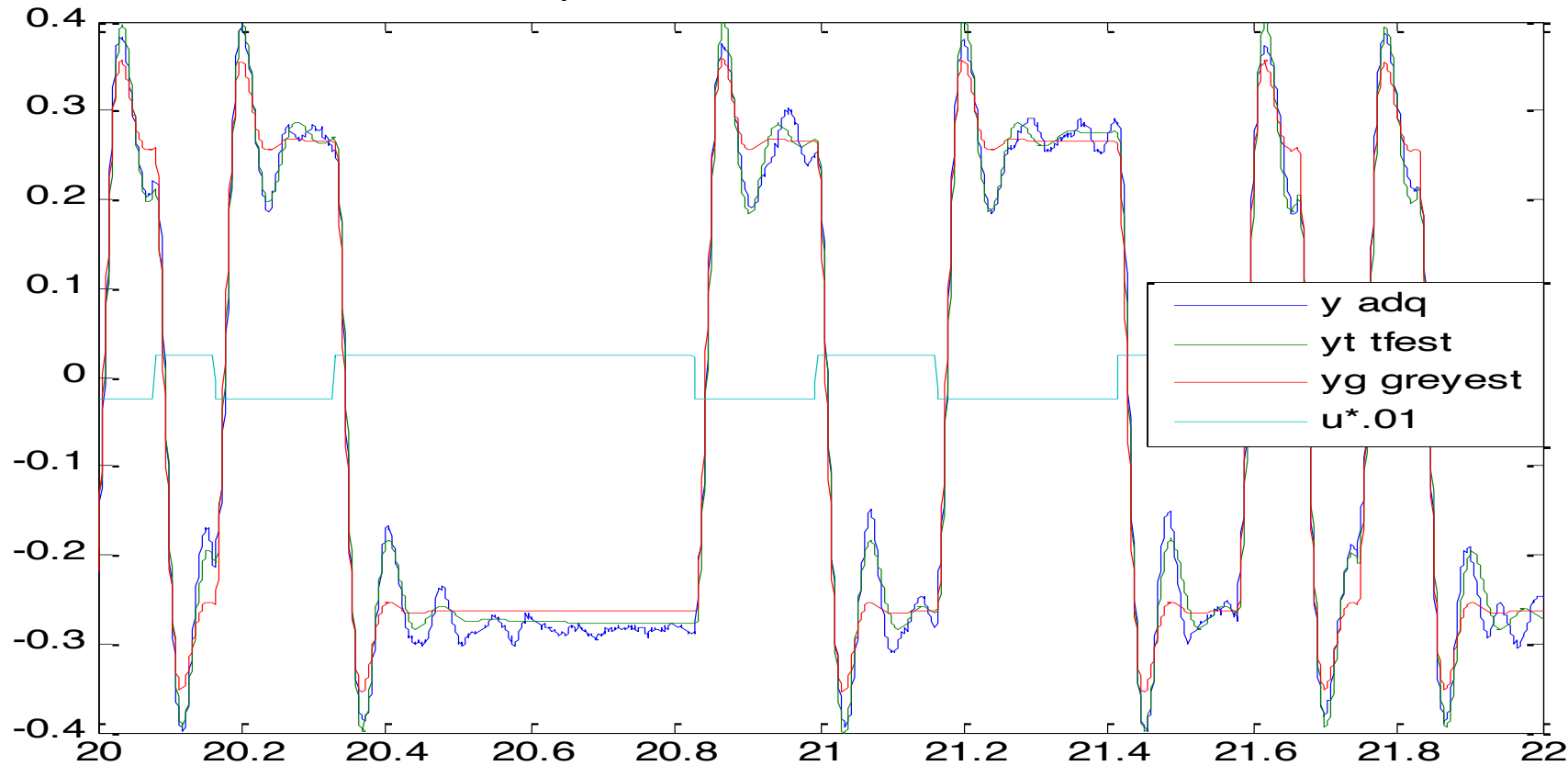
```
par=[-2e5;-7500;-1.3e4];  
aux=21; % posição do zero - fixa!  
Ts=0; % modelo contínuo  
gr=idgrey('myLevi',par,'c',aux,Ts)  
  
ggr{i}=greyest(ze{i},gr)
```

```
function [A,B,C,D,K,x0] = myLevi(par,Ts,aux)  
A = [0 1 0; 0 0 1; par(1) par(2) -121];  
B = [0; 0; 1];  
C = [par(3)*aux par(3) 0];  
D = zeros(1,1);  
K = zeros(3,1);  
x0 =[0; 0; 0];
```

```
% Forma Canonica Controlável  
%   b1 s + b0  
% -----  
% s^3 + a2 s^2 + a1 s + a0  
%  
%A=[0 1 0;  
%   0 0 1;  
%  -a0 -a1 -a2]  
%B=[0;0;1];  
%C=[b0 b1 0]
```

# Ex. 2 Levitação Magnética

i=3 arq.=dados\6-19+9-52-38\_1 ms.txt



$$tf(s) = \frac{-726.9(s+17.11)}{s^3 + 67.46s^2 + 8560s + 1.13e05}$$

$$gr(s) = \frac{-1019.6(s+21)}{s^3 + 121s^2 + 1.164e04s + 2.05e05}$$

# Ex. Predial

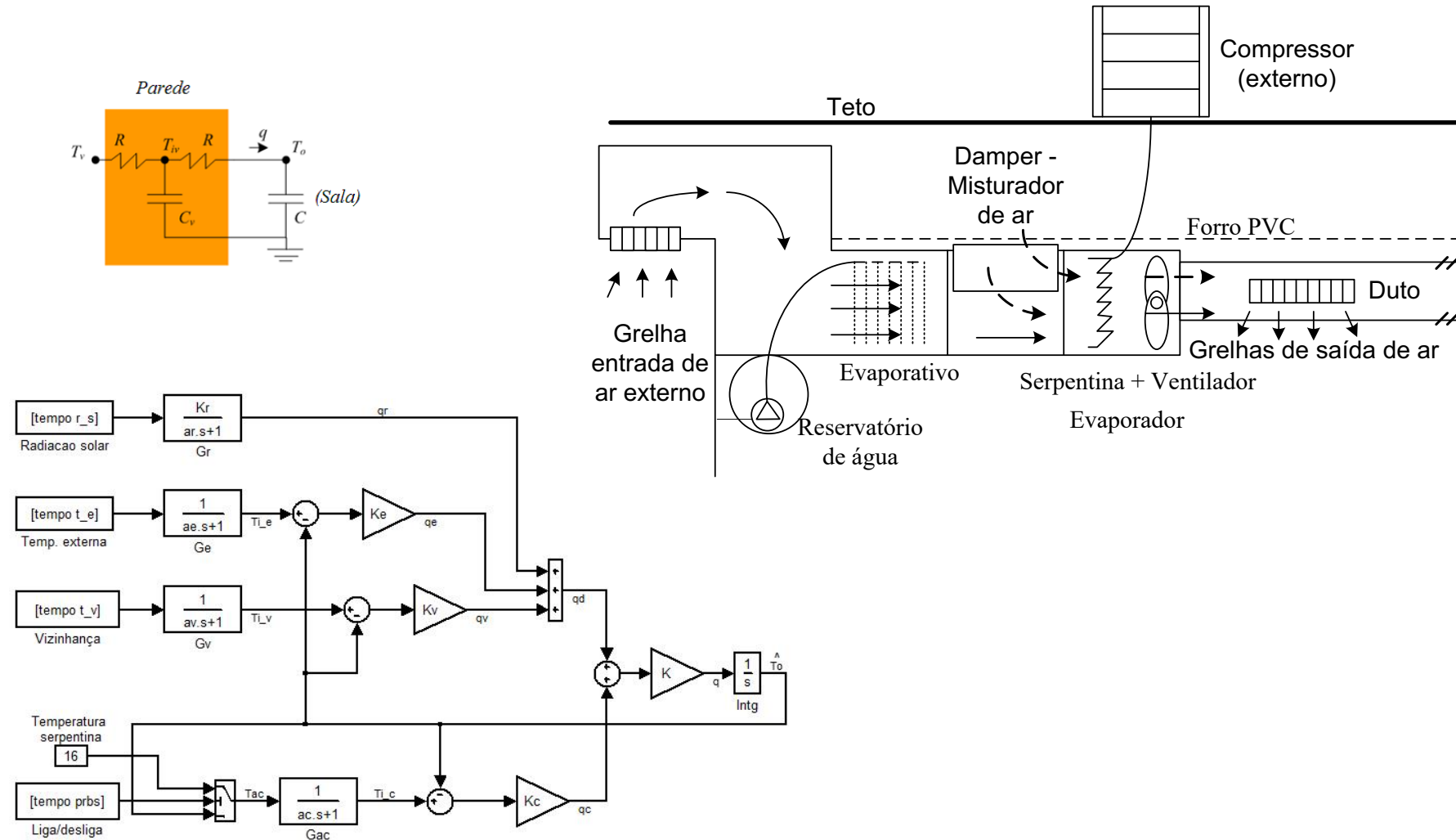


damper

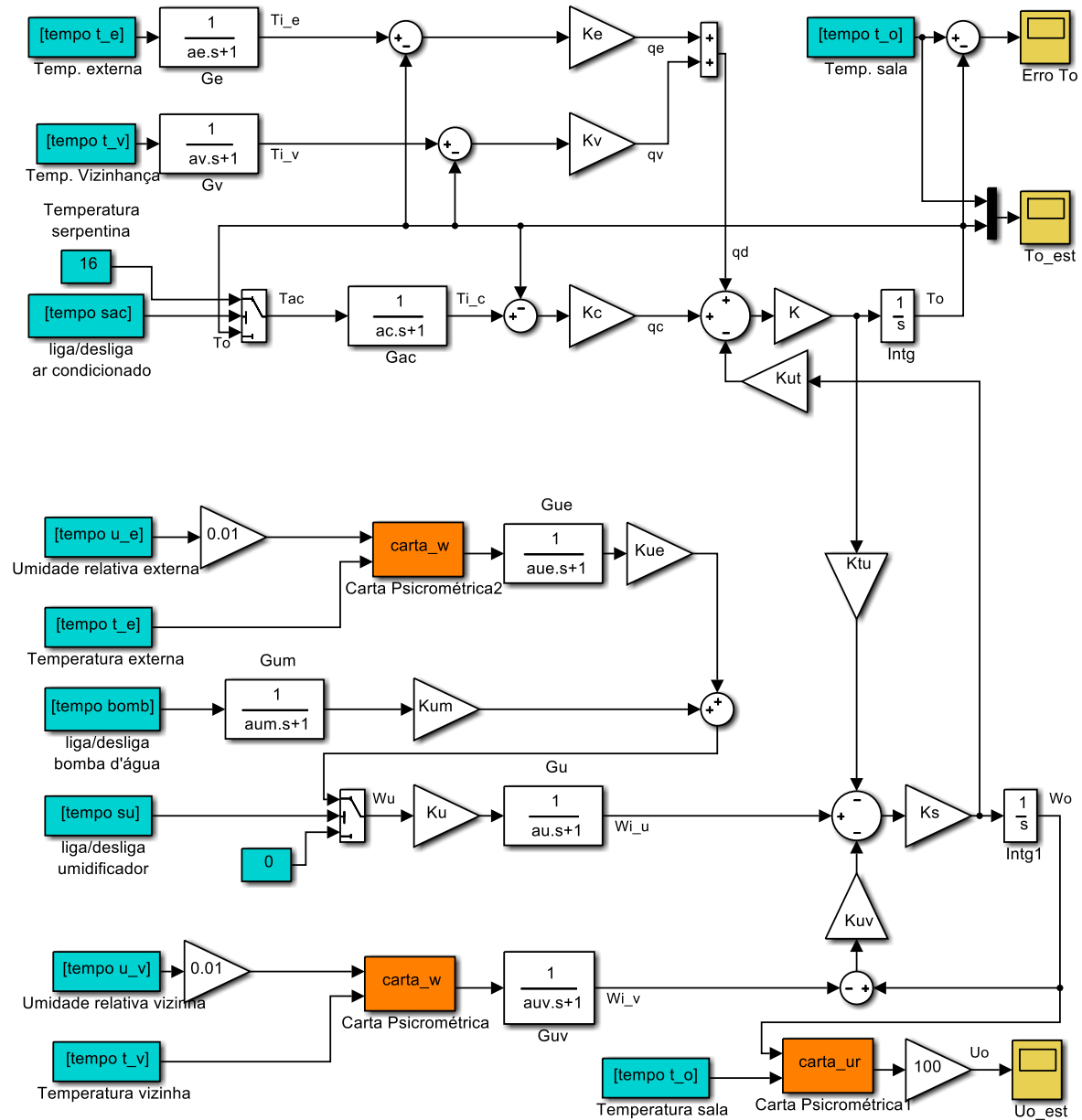
split

reservoir

# Climatização Híbrida - Temperatura



# T & UR



# idgrey

```
function [A,B,C,D,K,x0] = myfunc5(par,Ts,aux)
```

```
A = [par(1) par(2) par(3) par(4) par(5) par(6) par(7); 0 par(8) 0 0  
0 0 0; 0 0 par(9) 0 0 0 0; 0 0 0 par(10) 0 0 0; 0 0 0 0 par(11)  
par(12) par(13); 0 0 0 0 0 par(14) 0; 0 0 0 0 0 0 par(15)];
```

```
B = [par(16) par(17) par(18) par(19) par(20); (1-par(8)) 0 0 0 0; 0  
(1-par(9)) 0 0 0; 0 0 (1-par(10)) 0 0; 0 0 0 par(21) par(22); 0 0 0  
par(23) 0; 0 0 0 0 (1-par(15))];
```

```
C = [1 0 0 0 0 0 0; 0 0 0 0 1 0 0];
```

```
D = zeros(2,5);
```

```
K = zeros(7,2);
```

```
x0 =[par(24);par(25);par(26);par(27);par(28);par(29);par(30)];
```



# Matrizes – Princípios Fundamentais

$$A = \begin{bmatrix} a_{11} & \frac{K_e K}{\alpha} & \frac{K_v K}{\alpha} & \frac{K_c K}{\alpha} & a_{15} & \frac{-KK_{ut}K_s}{\alpha} & -a_{15} \\ 0 & -1/a_e & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1/a_v & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1/a_c & 0 & 0 & 0 \\ a_{51} & a_{52} & a_{53} & a_{54} & a_{55} & K_s/\alpha & -a_{55} \\ 0 & 0 & 0 & 0 & 0 & -1/a_u & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1/a_{uv} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1/a_e & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1/a_v & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1/a_c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & K_u/a_u & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/a_{uv} & 0 \end{bmatrix}$$

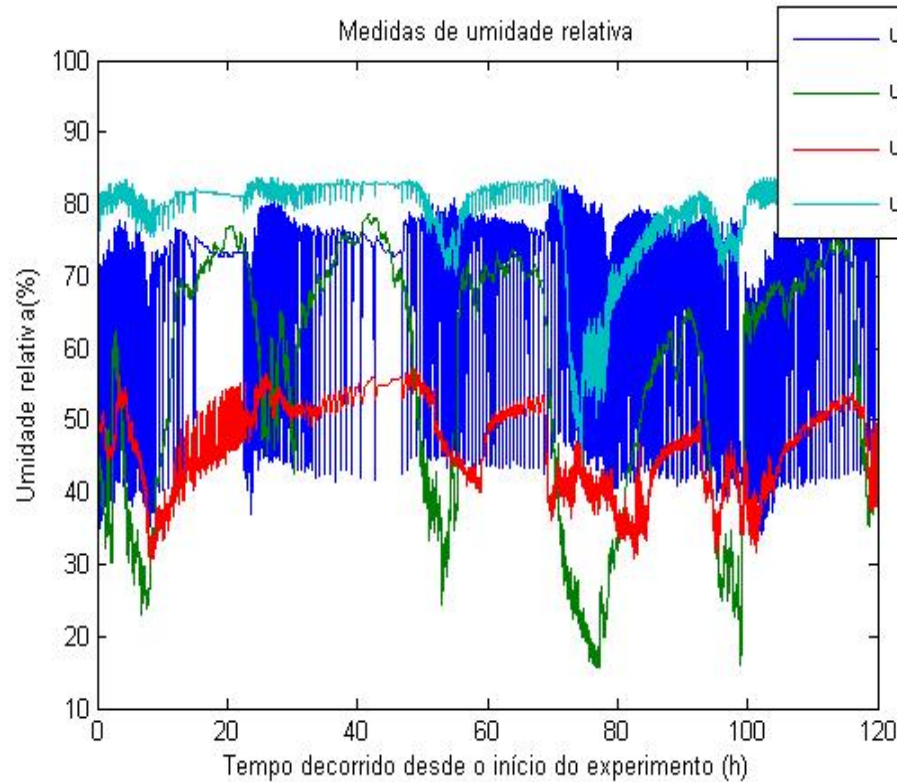
$$a_{11} = \frac{-K(K_e + K_v + K_c)}{\alpha}; a_{15} = \frac{KK_{ut}K_sK_{uv}}{\alpha};$$

$$a_{51} = \frac{KK_sK_{tu}(K_e + K_v + K_c)}{\alpha}; a_{52} = \frac{-K_eKK_sK_{tu}}{\alpha};$$

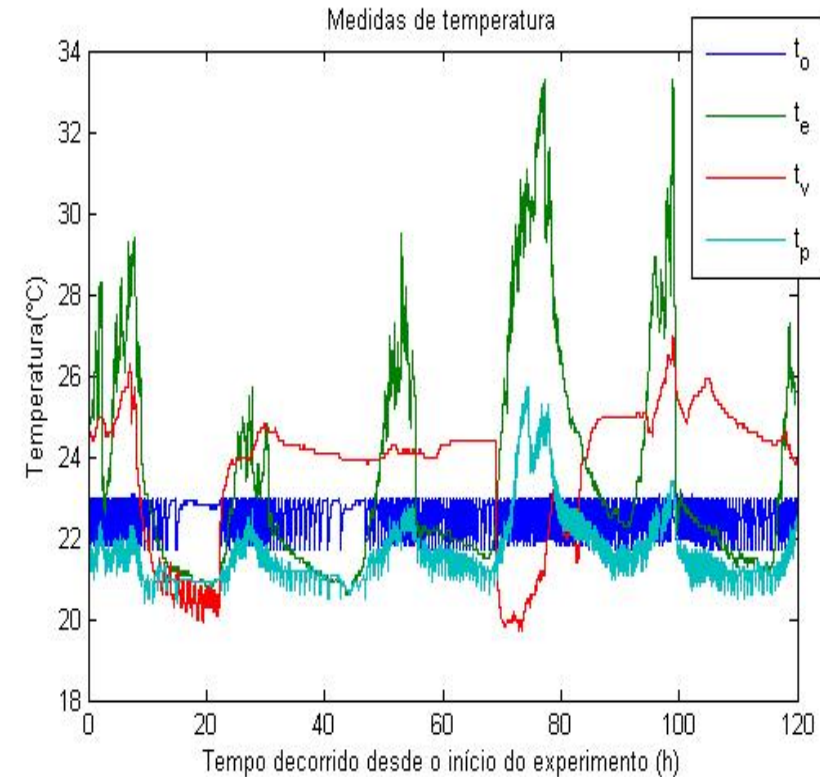
$$a_{53} = \frac{-K_vKK_sK_{tu}}{\alpha}; a_{54} = \frac{-K_cKK_sK_{tu}}{\alpha}; a_{55} = \frac{-K_sK_{uv}}{\alpha}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

# Predial - Sinais

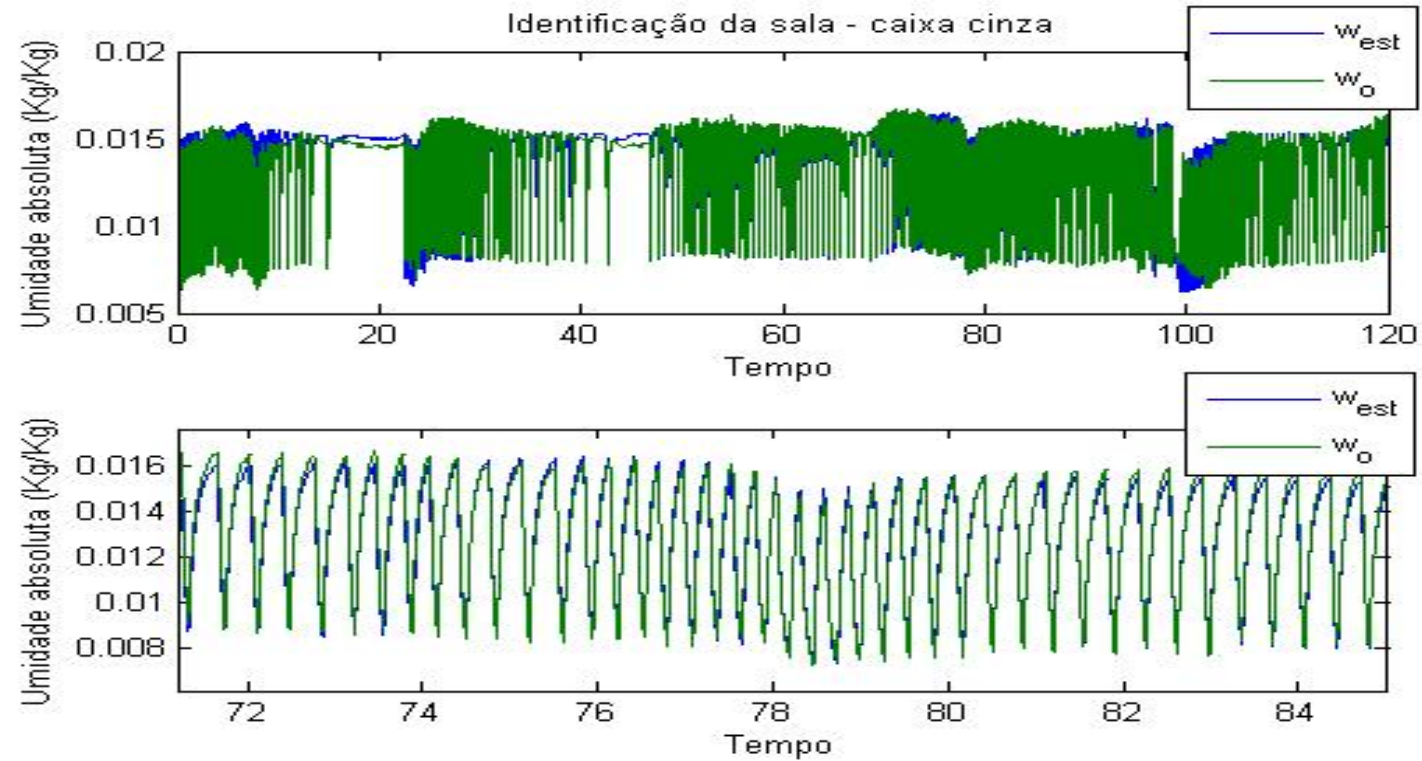


Umidade relativa da sala de reuniões ( $u_o$ ), do ambiente externo ( $u_e$ ), sala vizinha ( $u_v$ ) e pós-evaporativo ( $u_p$ ).



Temperatura da sala de reuniões ( $t_o$ ), do ambiente externo ( $t_e$ ), vizinhança ( $t_v$ ) e pós-evaporativo ( $t_p$ ).

# Resultado da Identificação



<i>i</i>	<i>u</i>	<i>uv</i>	<i>ue</i>	<i>um</i>	<i>s</i>
<b>ai</b>	168,5	164,9	80,82	171,7	-
<b>Ki</b>	2,19	3,77	0,151	-0,003	0,029

# Redução de Ordem

## Utilização de Coeficientes de Agrupamento

- Modelo ARMAX

$$y(k) = \sum_{i=1}^{n_y} a_i(T_s) y[(k-i)T_s] + \sum_{i=1}^{n_u} b_i(T_s) u[(k-i)T_s] + \sum_{i=1}^{n_\xi} c_i(T_s) \xi[(k-i)T_s] + \xi(k).$$

# Candidatos a Modelo

$$y(t) = a_1^1(T_s)y(t - T_s) + b_1^1(T_s)u(t - T_s) + c_1^1(T_s)\tilde{\xi}(t - T_s) + \tilde{\xi}(t),$$

$$y(t) = a_1^2(T_s)y(t - T_s) + a_2^2(T_s)y(t - 2T_s) + b_1^2(T_s)u(t - T_s) \\ + b_2^2(T_s)u(t - 2T_s) + c_1^2(T_s)\xi(t - T_s) + \xi(t),$$

$$y(t) = a_1^3(T_s)y(t - T_s) + a_2^3(T_s)y(t - 2T_s) + a_3^3(T_s)y(t - 3T_s) \\ + b_1^3(T_s)u(t - T_s) + b_2^3(T_s)u(t - 2T_s) + b_3^3(T_s)u(t - 3T_s) \\ + c_1^3(T_s)\xi(t - T_s) + \xi(t),$$

Considerando:

pequenas períodos de amostragem

$$y(t - T_s) \approx y(t - 2T_s) \approx y(t - 3T_s)$$

$$y(t) = a_1^1(T_s)y(t - T_s) + b_1^1(T_s)u(t - T_s) + c_1^1(T_s)\tilde{\xi}(t - T_s) + \tilde{\xi}(k)$$

$$y(t) \approx (a_1^2(T_s) + a_2^2(T_s))y(t - T_s) + (b_1^2(T_s) + b_2^2(T_s))u(t - T_s) + c_1^2(T_s)\xi(t - T_s) + \xi(k)$$

$$y(t) \approx (a_1^3(T_s) + a_2^3(T_s) + a_3^3(T_s))y(t - T_s) + (b_1^3(T_s) + b_2^3(T_s) + b_3^3(T_s))u(t - T_s) + c_1^3(T_s)\xi(t - T_s) + \xi(k).$$

# Estimação dos Parâmetros

$$\begin{bmatrix} y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} y(1) & u(1) & \tilde{\xi}(1) \\ y(2) & u(2) & \tilde{\xi}(2) \\ \vdots & \vdots & \vdots \\ y(N-1) & u(N-1) & \tilde{\xi}(N-1) \end{bmatrix} \begin{bmatrix} \Sigma_y^{1,1}(T_s) \\ \Sigma_u^{1,1}(T_s) \\ c_1^1(T_s) \end{bmatrix} + \begin{bmatrix} \tilde{\xi}(2) \\ \tilde{\xi}(3) \\ \vdots \\ \tilde{\xi}(N) \end{bmatrix}$$

$$\begin{bmatrix} y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix} \approx \begin{bmatrix} y(1) & u(1) & \xi(1) \\ y(2) & u(2) & \xi(2) \\ \vdots & \vdots & \vdots \\ y(N-1) & u(N-1) & \xi(N-1) \end{bmatrix} \begin{bmatrix} \Sigma_y^{2,2}(T_s) \\ \Sigma_u^{2,2}(T_s) \\ c_1^2(T_s) \end{bmatrix} + \begin{bmatrix} \xi(2) \\ \xi(3) \\ \vdots \\ \xi(N) \end{bmatrix}$$

$$\begin{bmatrix} y(2) \\ y(3) \\ \vdots \\ y(N) \end{bmatrix} \approx \begin{bmatrix} y(1) & u(1) & \xi(1) \\ y(2) & u(2) & \xi(2) \\ \vdots & \vdots & \vdots \\ y(N-1) & u(N-1) & \xi(N-1) \end{bmatrix} \begin{bmatrix} \Sigma_y^{3,3}(T_s) \\ \Sigma_u^{3,3}(T_s) \\ c_1^3(T_s) \end{bmatrix} + \begin{bmatrix} \xi(2) \\ \xi(3) \\ \vdots \\ \xi(N) \end{bmatrix}$$

$$\Sigma_y^{1,1}(T_s) \neq \Sigma_y^{2,2}(T_s) \approx \Sigma_y^{3,3}(T_s) \approx \dots \approx \Sigma_y^{q,q}(T_s)$$

# Redução de Ordem

- Encontrar o “joelho”  
acrécimo de termo é irrelevante:

$$| \Sigma_y(T_s) - \Sigma_{yo}(T_s) | < \gamma$$