

# EXPERIÊNCIA 3

## Usando resposta em frequência para modelar a planta



PTC 3312 – Laboratório de Controle

2º semestre de 2020

Fábio Fialho

Laboratório de Automação e Controle

Departamento de Engenharia de Telecomunicações e Controle

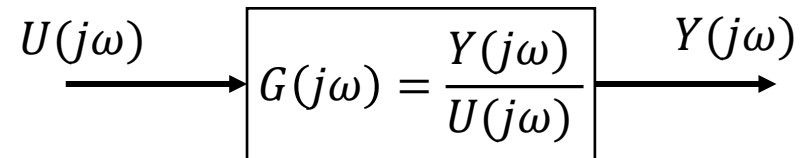
Escola Politécnica da Universidade de São Paulo

# Objetivo

- Obter um modelo matemático linear para o servomecanismo MS15 por meio de sua **resposta em frequência**.

# Método

- Considerando um sistema linear invariante no tempo:



$$G(j\omega) = |G(\omega)|e^{j\phi} \text{ sendo } \phi = \arctg \frac{\text{Im}[G(j\omega)]}{\text{Re}[G(j\omega)]}$$

- A resposta em frequência se relaciona com a função de transferência  $G(s)$  do sistema por meio da expressão

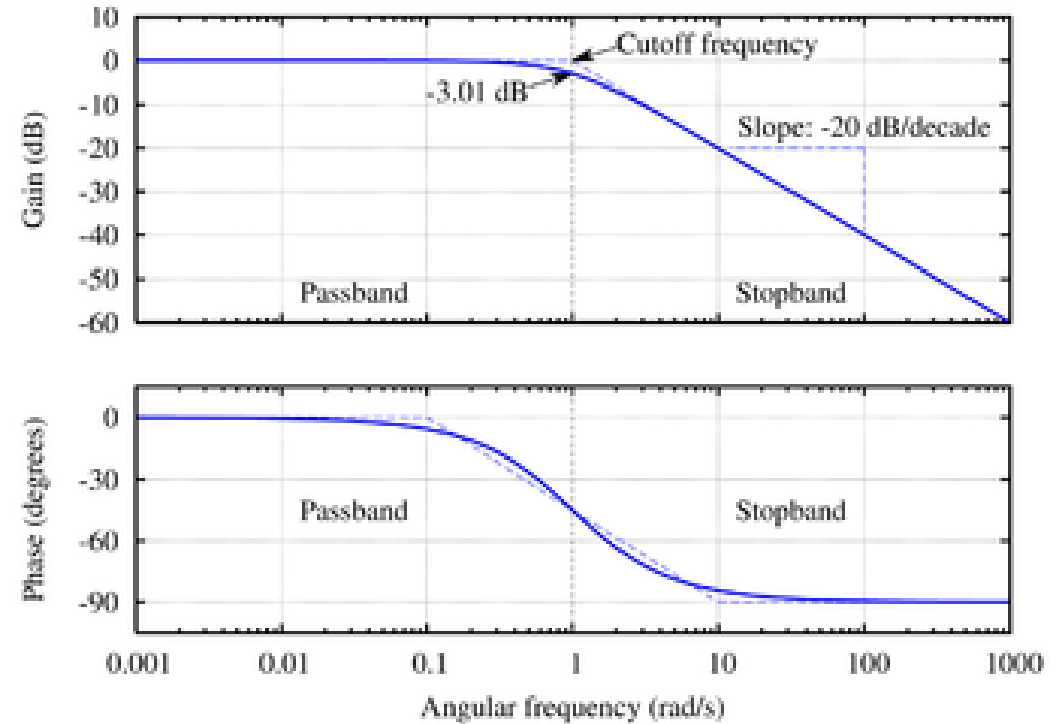
$$G(j\omega) = G(s)|_{s=j\omega}$$

- Ou seja, dado que se tenha a resposta em frequência, é possível se obter a função de transferência do sistema

# Método

- Será necessário obter a resposta em frequência
- e deduzir a função de transferência

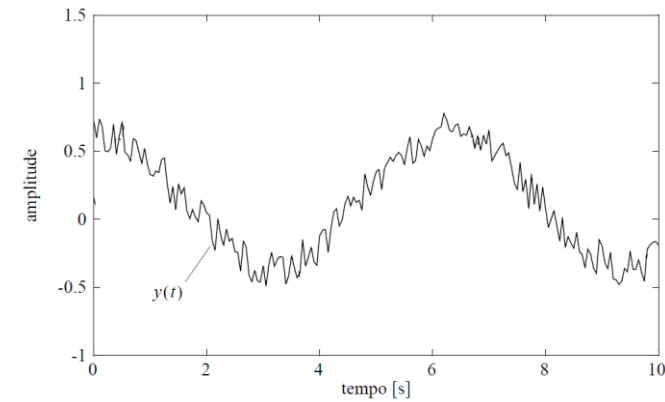
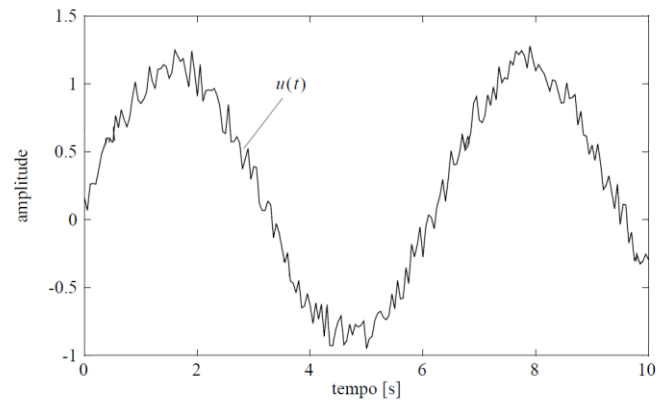
Diagrama de Bode



$$U(s) = V_m(s) \xrightarrow{[V]} \boxed{\frac{KK_t}{1 + sT}} \xrightarrow{V_t(s)} [V]$$

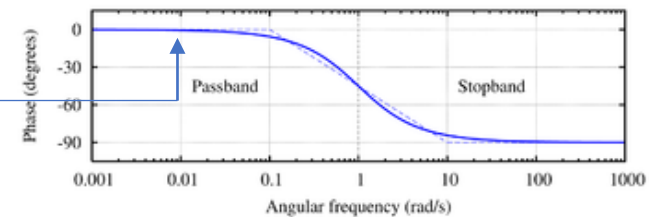
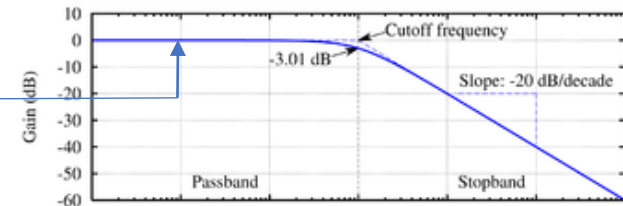
# Obtenção da resposta em frequência

$$u(t) = A_i \text{sen}(\omega t + \phi_i) \xrightarrow{u(t)} \text{[Black Box]} \xrightarrow{y(t)} y(t) = A_o \text{sen}(\omega t + \phi_o)$$



Ganho  $G(\omega) = \frac{A_o}{A_i}$

Fase  $\Phi(\omega) = \phi_o - \phi_i$



# Obtenção da resposta em frequência

- Resta o problema de identificar os parâmetros  $A_o$ ,  $\phi_i$  e  $\phi_o$ .

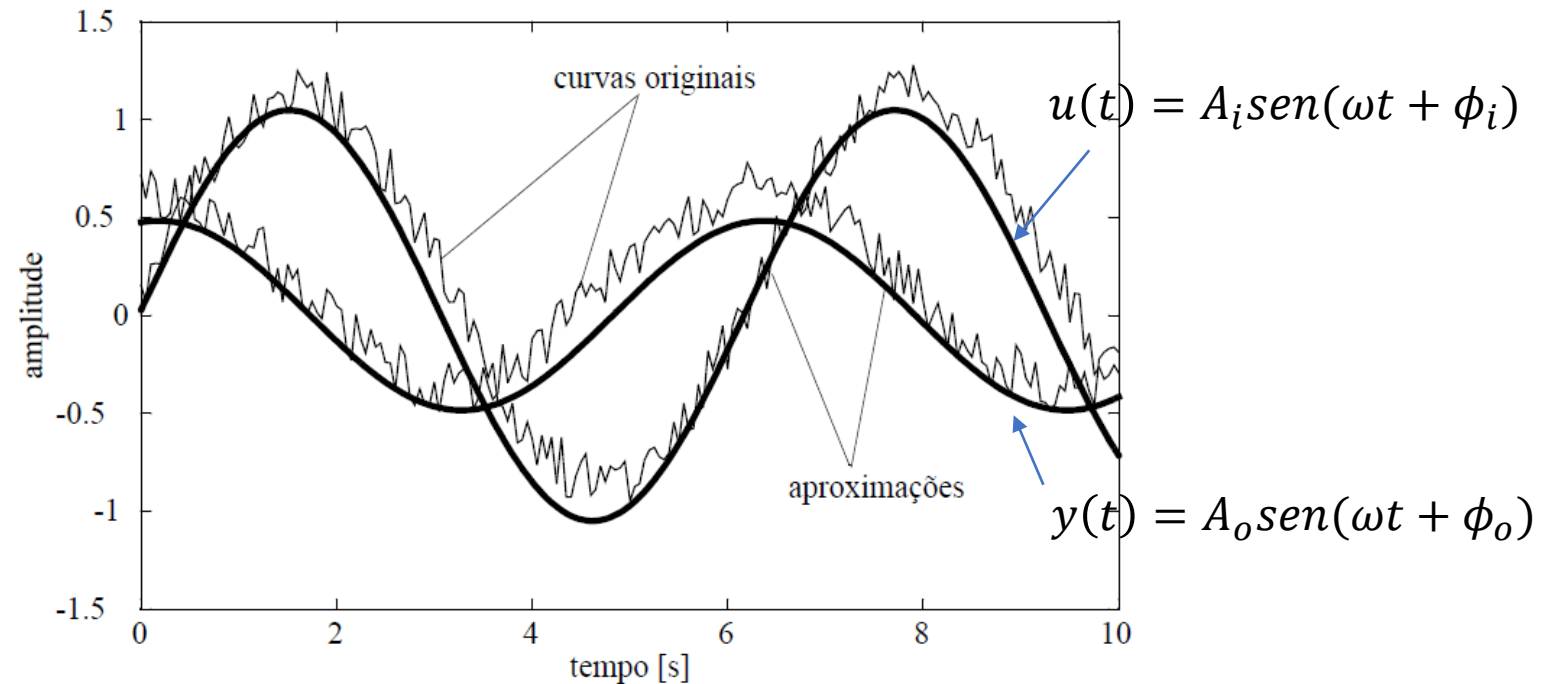


Figura 3.3 Sinais originais e aproximações senoidais

# Obtenção da resposta em frequência

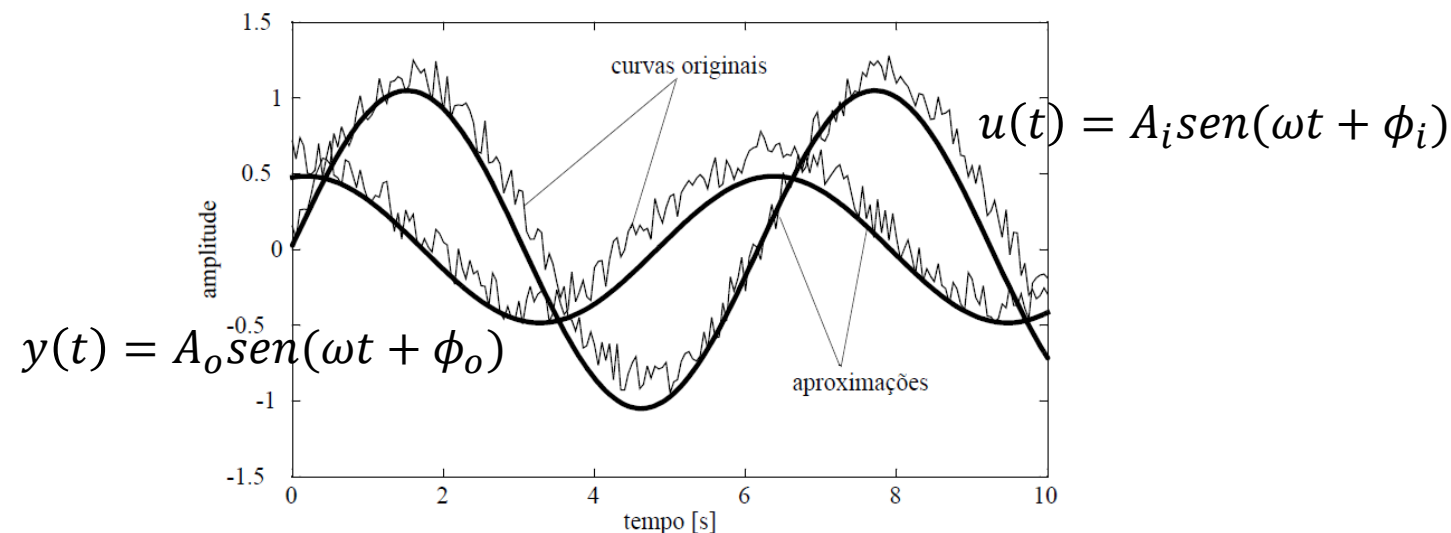


Figura 3.3 Sinais originais e aproximações senoidais

- Para se obter os parâmetros é necessário ajustar a função senoidal ao sinal adquirido, que devido a uma série de fatores tais como ruído, não linearidade do sistema, etc. pode não ser perfeitamente senoidal.
- Para realizar o ajuste, um algoritmo de regressão linear é utilizado (veja o Apêndice A).

# Método de obtenção do diagrama de bode (Função senofit)

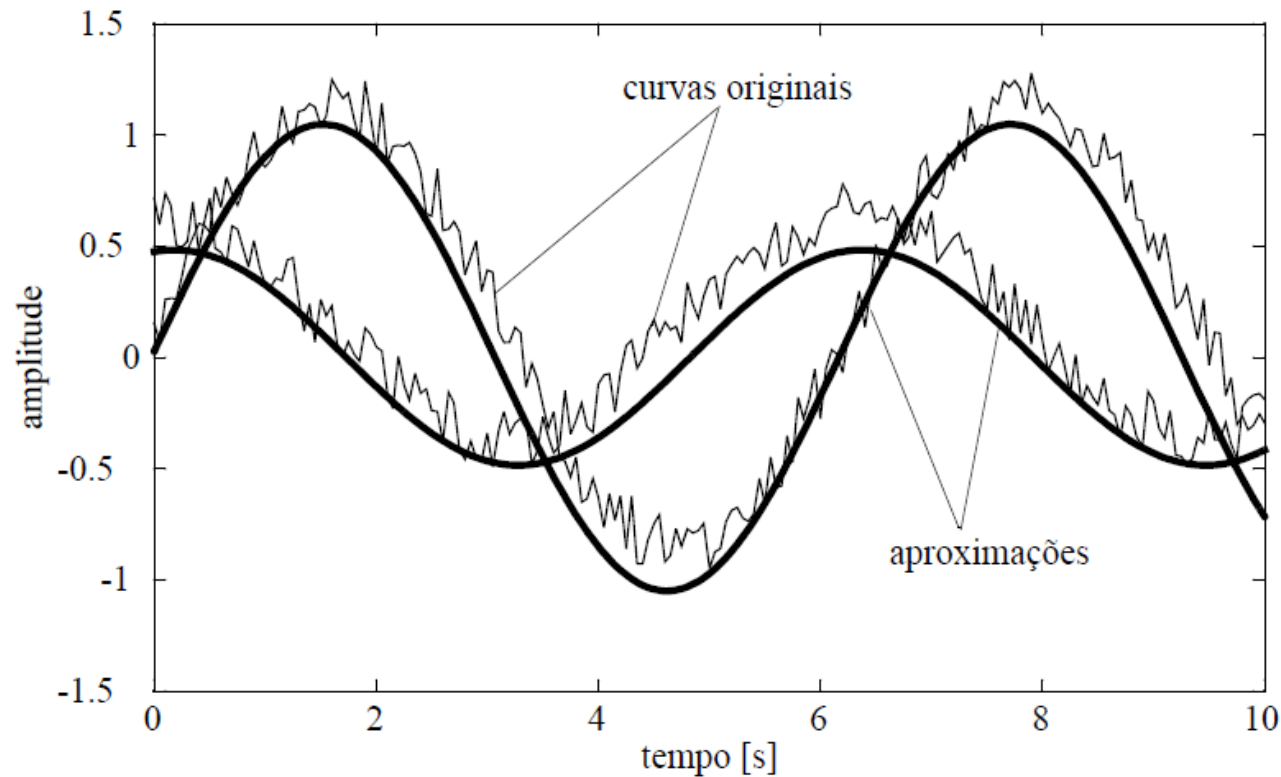


Figura 3.3 Sinais originais e aproximações senoidais

senofit

senofit\_ic

senofit\_erquad

$$u(t) = A_u \text{sen}(\omega_u t + \phi_u)$$

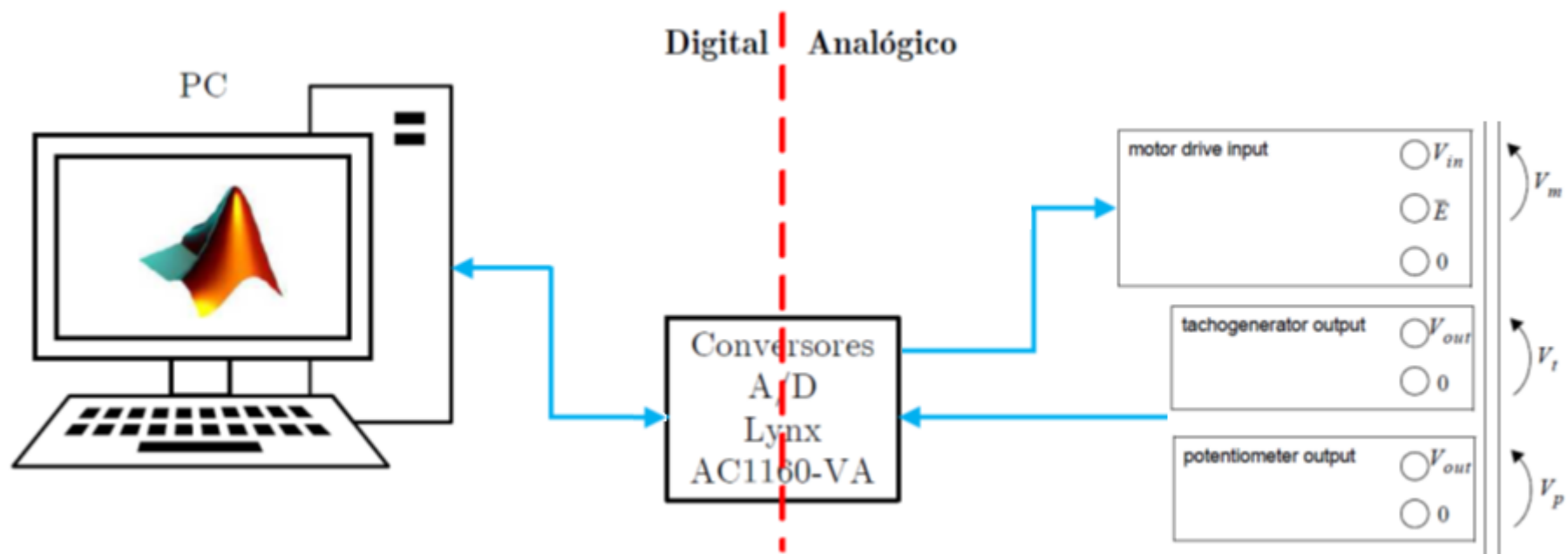
$$y(t) = A_y \text{sen}(\omega_y t + \phi_y)$$

$$G(\omega) = \frac{A_y}{A_u}$$

$$\Phi(\omega) = \phi_y - \phi_u$$



# Matlab/Simulink



- Vamos novamente utilizar nossa montagem usual
- Coletar os dados e tratá-los via Matlab

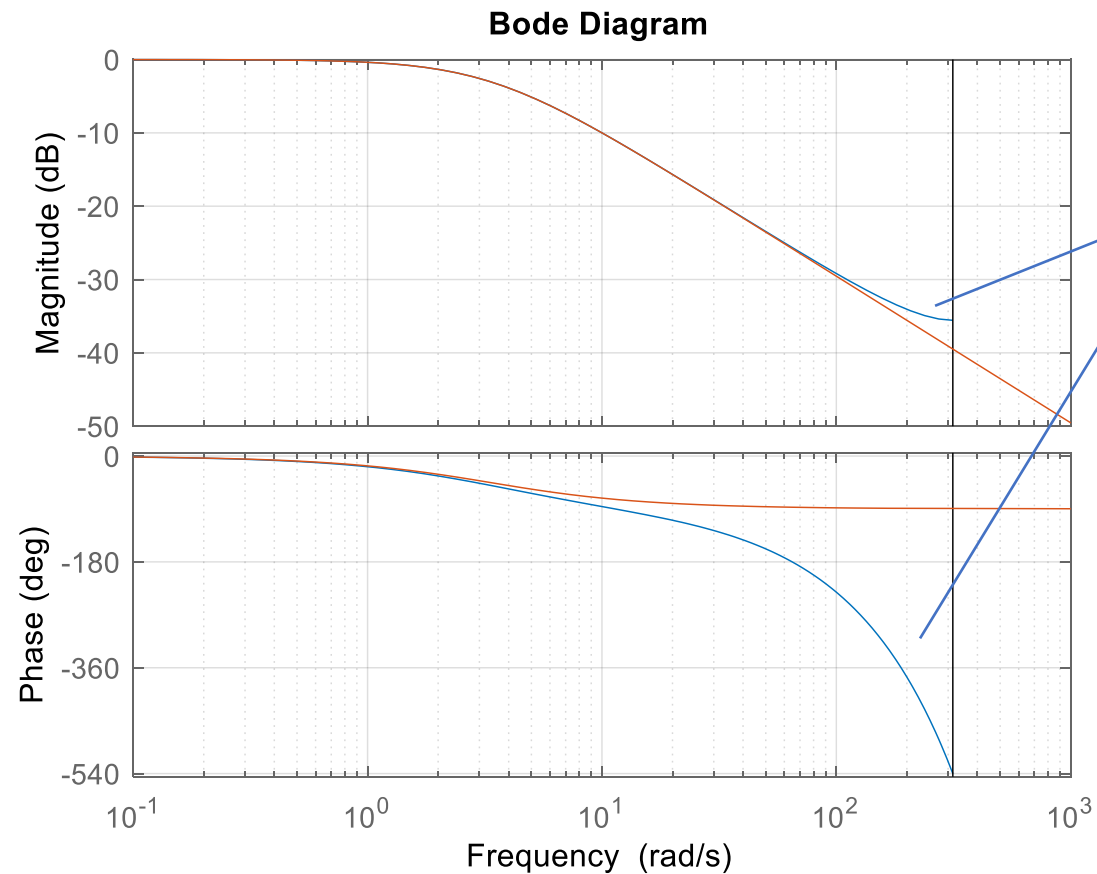
Para agilizar o processo é disponibilizado o programa respfreq.

The screenshot shows the 'respfreq' software interface. It is divided into several sections:

- Sequência de processamento:** A vertical sequence of three steps:
  - (1) Input:  $u(t) = A \cdot \text{sen}(w \cdot t)$ . Fields for 'Freq.' (rad/s) and 'Ampl.' (4.0 V).
  - (2) Button: 'COLETAR DADOS'. Below it, 'Tempo estim. (trans. + r.p.s.): (a calcular)'.
  - (3) Output:  $y(t) = A \cdot \text{sen}(w \cdot t + \text{phi})$ . Fields for 'Ganho' (dB) and 'Fase' (°). Button: 'ACEITAR O PONTO'.
- Visualização de dados:** A large empty box containing the text 'Que frequências utilizar?' and a bullet point: 'Utilizamos o modelo linear da experiência 2 para traçar o diagrama de Bode na interface e guiar as escolhas das frequências'. A blue arrow points from this text to the Bode plots below.
- Gerenciamento:** Two Bode plots (Gain and Phase) with logarithmic axes. To the right, a 'Referência' section with fields for 'Kkt' (0.85) and 'T' (0.30), and buttons 'SELECIONAR PONTO' and 'EXCLUIR SELEÇÃO'. The text '(G), (F)' is also present.

$1 \text{ Hz} = 2\pi \text{ rad/s}$

# Distorção de alta frequência do Bode com dados da planta

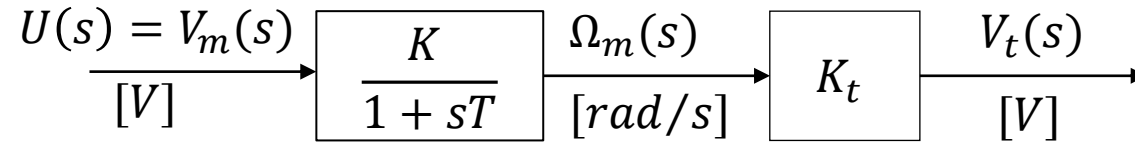


Soma de dois efeitos: atraso de amostragem e frequência de amostragem

Para simular esses dois efeitos, basta rodar os seguintes comandos:

```
sys = tf(1, [0.3 1])  
sys_at = tf(1, [0.3 1], 'iodelay', 0.01)  
sysd = c2d(sys_at, 0.01, 'zoh')  
figure, bode(sysd), hold, bode(sys)
```

# Obtenção da Função de Transferência Método



- Dois métodos possíveis:
  - Ajuste manual por assíntotas
  - Ajuste por otimização numérica

# Obtenção da Função de Transferência Método

- Ajuste manual por assíntotas

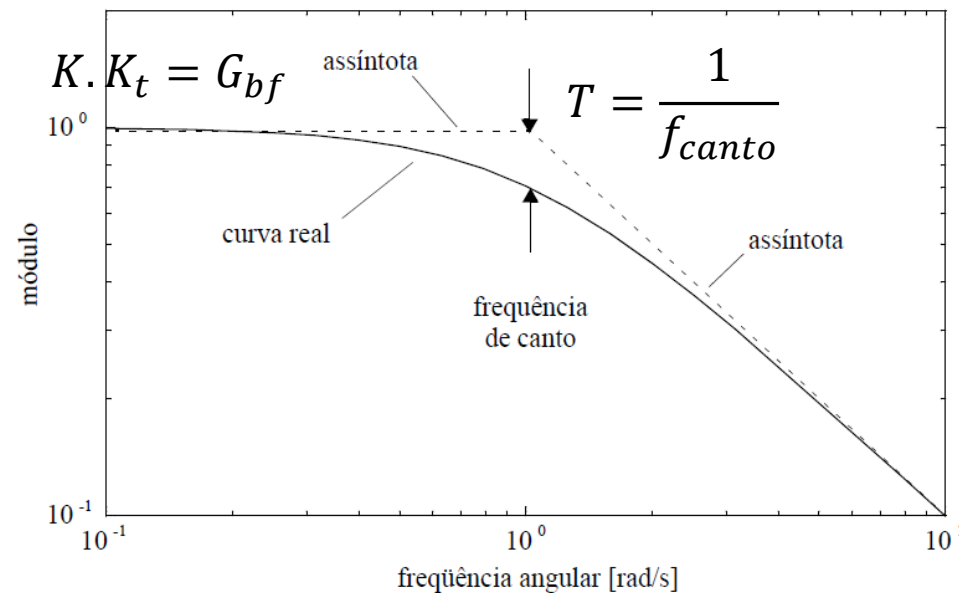
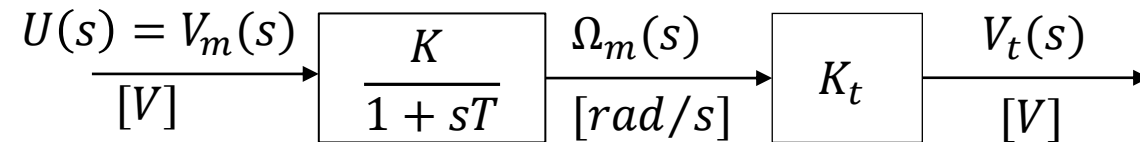
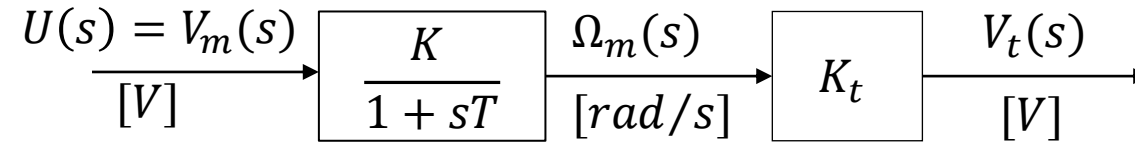


Figura 3.4 Aproximação de uma função de transferência por meio de assíntotas

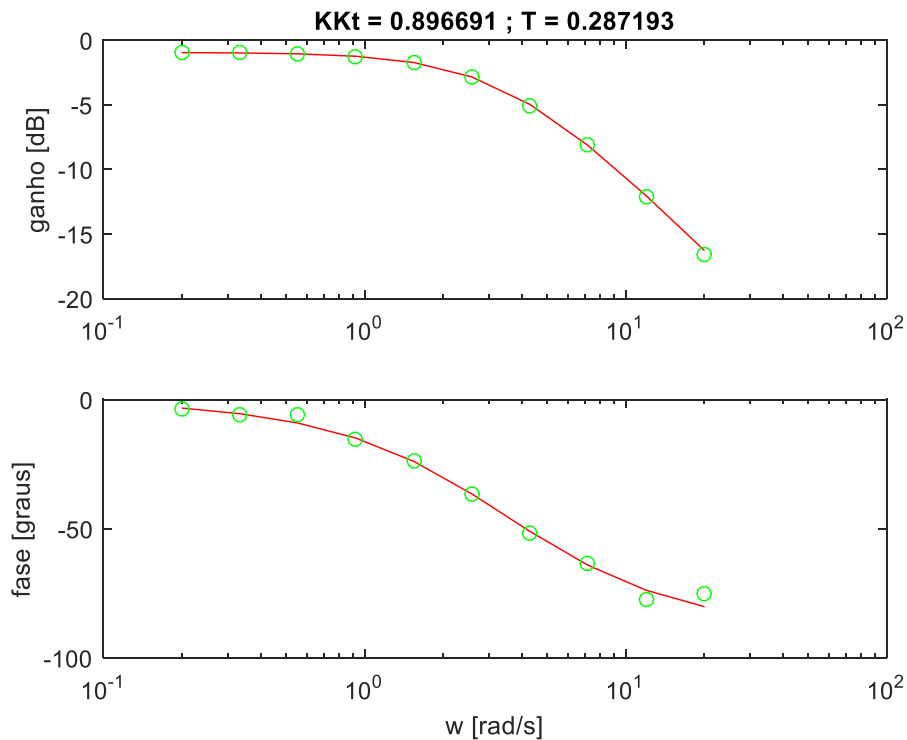
# Obtenção da Função de Transferência

## Método

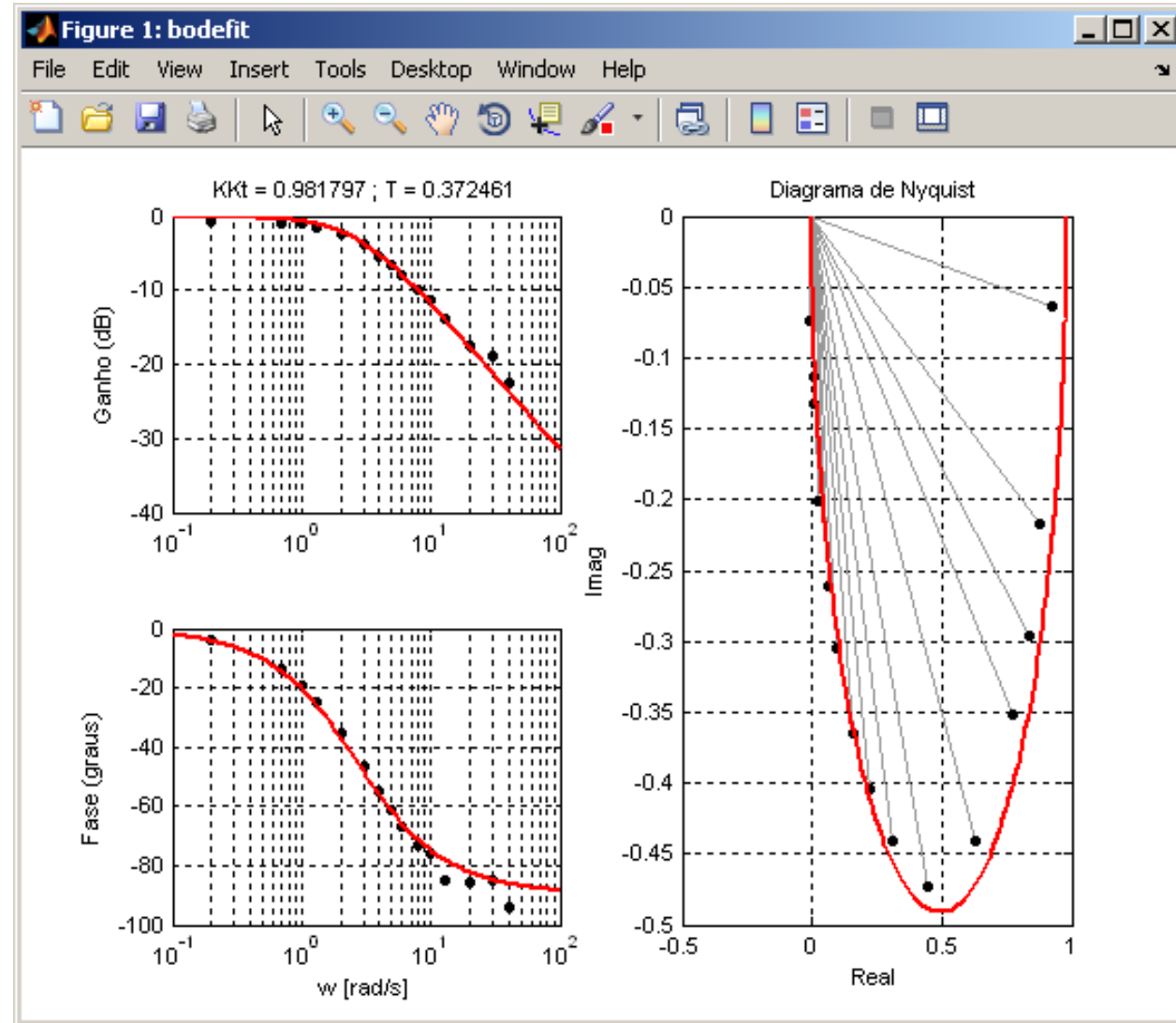


- Ajuste por otimização numérica

- Fitting do diagrama de bode



# Obtenção da Função de Transferência IHM da função Bodefit



```

function [KKt,T] = bodefit(KKto,To,wdata,Gdata,Fdata)
% BODEFIT   Ajusta uma função de transferência do tipo KKt/(1+sT) a um
%           conjunto de dados contendo a sua resposta em frequência.
%
%           Uso:   [KKt,T] = bodefit(KKto,To,wdata,Gdata,Fdata)
%
%           onde:  KKto é uma condição inicial para o ganho
%                  To é uma condição inicial para a cte. de tempo
%                  wdata, Gdata, Fdata são os pontos levantados
%
%                  KKt é o valor ajustado para o ganho
%                  T é o valor ajustado para a cte. de tempo
%
%           wdata (em rad/s), Gdata (em dB) e Fdata (em graus) são
%           compatíveis com as variáveis geradas por RESPFREQ.
%
%           RPM/V2016a

% Prepara os dados
wdata=wdata(:); Gdata=Gdata(:); Fdata=Fdata(:); % converte em vetores coluna
Gdata = 10.^(Gdata/20);                          % converte dB em linear
Fdata = Fdata*pi/180;                             % converte graus em rad
V = Gdata .* (cos(Fdata) + j.*sin(Fdata));        % converte G,F em fasor

% Prepara otimização
% (em caso de problemas de convergência pode se alterar as opções abaixo)
options = optimset('Display','off','TolFun',1e-3,'TolX',1e-3);
xo = [KKto To];                                    % condição inicial
% Otimização
% Digite " >> doc fminsearch " no Matlab para saber mais sobre o algoritmo
xf = fminsearch(@(x) erquad(x,wdata,Gdata,Fdata,V),xo,options);
KKt = xf(1); T = xf(2);                            % os resultados finais

end

```

esta é a parte  
principal do  
código





```

function erq = erquad(KKtT,wdata,Gdata,Fdata,V)
% Calcula a raiz quadrada do erro quadrático entre a função de
% transferência candidata e os pontos levantados
% chamada pelo comando FMINSEARCH de BODEFIT.

KKt=KKtT(1); T=KKtT(2);           % desempacota o vetor
sis=tf(KKt,[T 1]);                % cria uma função de transferência

[GG,FF]=bode(sis,wdata);          % GG,FF são equivalentes a Gdata,Fdata
GG=squeeze(GG);                   % acerta dimensões
FF=squeeze(FF)*pi/180;            % acerta dimensões
VV=GG.*cos(FF) + j*GG.*sin(FF);   % VV é equivalente a V

erq=norm(VV-V);                   % raiz quadrada do erro quadrático

% Exibe gráficos para acompanhamento da otimização
%
% .....
% (trecho omitido)
% .....

end

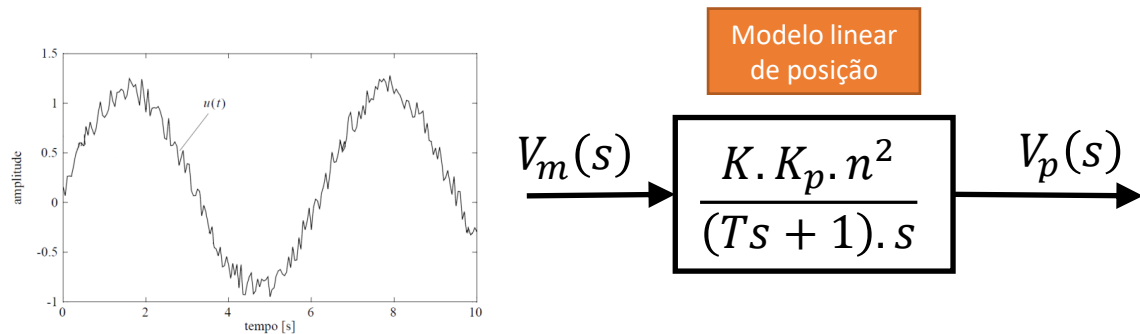
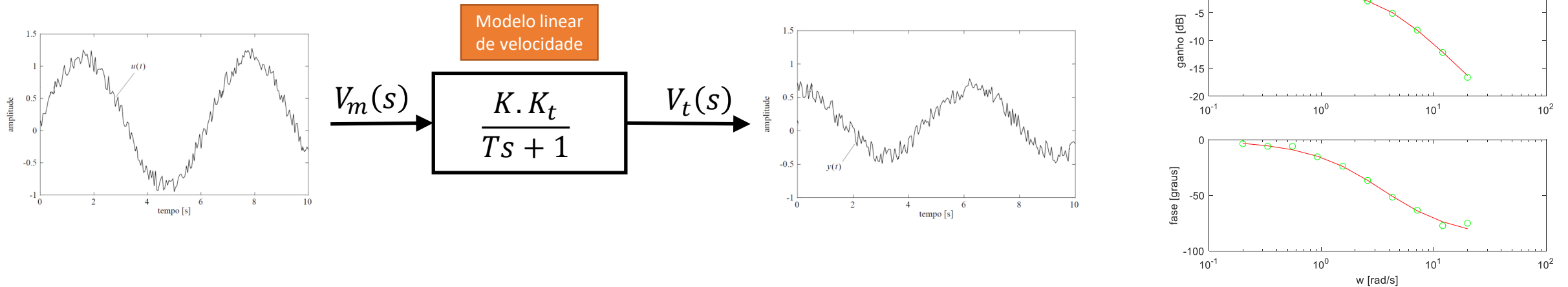
```

← a função erquad define o critério para o problema de otimização, e pode ser facilmente alterada para resolver problemas diversos.

← este é o cálculo do erro quadrático.

# Atividades

c) Não é possível levantar a resposta em frequência (pelo menos como definida aqui) tomando como saída a tensão no potenciômetro  $V_p(t)$ . Explique o porquê.



?

?

E ainda, qual o comportamento do sinal que, de fato, medimos na planta, com o potenciômetro? É possível fazer algo com ele?

# PROFESSORES DE LAB. DE CONTROLE

Diego Colón [diego@lac.usp.br](mailto:diego@lac.usp.br)

Fabio Fialho [fabio.fialho@usp.br](mailto:fabio.fialho@usp.br)

Felipe Pait [pait@lac.usp.br](mailto:pait@lac.usp.br)

Fuad Kassab Junior [fuad@lac.usp.br](mailto:fuad@lac.usp.br)

Ricardo Marques [rpm@lac.usp.br](mailto:rpm@lac.usp.br)



USP

FIM DA APRESENTAÇÃO