



**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  
**DEPARTAMENTO DE ENGENHARIA ELÉTRICA-ELETRÔNICA**  
**CURSO: ENGENHARIA ELETRÔNICA E DE TELECOMUNICAÇÃO**  
**DISCIPLINA: LABORATÓRIO DE ANÁLISE DE SISTEMAS III**

# **APOSTILA DE LABORATÓRIO DE ANÁLISE DE SISTEMAS III**

**Professor: ANIVALDO MATIAS DE SOUZA**  
**Colaboração: FABRÍCIO NORONHA GOMES PEREIRA**

# Índice

INTRODUÇÃO.....	3
<b>1 - CARACTERÍSTICAS DE SISTEMAS DE CONTROLE COM RETROAÇÃO.....</b>	<b>7</b>
<b>Exemplo 1.1</b> Sistema de Leitura do Acionador de Disco.....	8
<b>TABELA 1.1 - Valores Típicos de Parâmetros para a Leitura do Acionador de Disco .....</b>	<b>9</b>
<b>2 - SIMULAÇÃO DE SISTEMAS USANDO O MATLAB .....</b>	<b>11</b>
<b>3 - ANÁLISE DE MODELOS COM VARIÁVEIS DE ESTADO USANDO O MATLAB.....</b>	<b>15</b>
<b>4 - EXEMPLO DE PROJETO SEQUENCIAL: SISTEMA DE LEITURA DE ACIONADOR DE DISCO .....</b>	<b>20</b>
<b>TABELA 4.1</b> Resposta ao Sistema Mostrado na Fig. 4.5 (a) .....	24
<b>5 - DESEMPENHO DO SISTEMA USANDO O MATLAB.....</b>	<b>25</b>
<b>6 - EXEMPLO DE PROJETO SEQUENCIAL: SISTEMA DE LEITURA DE ACIONADOR DE DISCO .....</b>	<b>29</b>
<b>TABELA 6.1</b> Desempenho do Sistema Acionador de Disco Comparado com as Especificações.....	32
<b>7 - MÉTODOS DE RESPOSTA DE FREQUÊNCIA USANDO MATLAB.....</b>	<b>33</b>
<b>8 - MÉTODOS DE IDENTIFICAÇÃO DETERMINÍSTICOS .....</b>	<b>38</b>
8.1 - Introdução .....	38
8.2 - Alguns Casos Simples .....	38
8.2.1 - Sistemas de primeira ordem .....	38
8.2.2 - Sistemas de segunda ordem pouco amortecidos.....	39
<b>Exemplo 8.2.1</b> Estimção de $\zeta$ e $\omega_n$ .....	41
8.3 - O Método de Sundaesan.....	41
8.3.1 - O caso sobreamortecido .....	42
<b>Exemplo 8.3.1</b> Método Sobreamortecido.....	45
8.3.2 - O caso subamortecido .....	47
<b>Exemplo 8.3.2</b> Método Subamortecido.....	50
8.4 - Identificação em Malha Fechada.....	52
<b>Exemplo 8.4.1</b> Identificação em Malha Fechada .....	55
<b>9 - BIBLIOGRAFIA .....</b>	<b>57</b>

## INTRODUÇÃO

O texto apresentado abaixo tem por objetivo fornecer conceitos e informações preliminares e foi traduzido e adaptado do livro Celier, François E. “Continuous System Modeling”. As citações pertinentes à outros autores foram identificadas e as referências bibliográficas completas encontram-se ao final do capítulo.

**O termo sistema considerando-se aspectos genéricos da língua portuguesa [Dicionário Aurélio, 1970] significa:**

Conjunto de elementos materiais ou ideais entre os quais se possa encontrar ou definir uma relação ou ainda disposição das partes ou dos elementos de um todo coordenados entre si e que funcionam como uma estrutura organizada.

Dentre todos os sistemas existentes poderia - se considerar como o maior deles o próprio universo. Quando é possível isolar uma parte do universo e delimitar claramente o que pertence a esta parte e o que não pertence define-se um novo sistema. Uma definição mais objetiva e aplicada ao contexto de engenharia [Pena e Aguirre, 1997 ] seria:

**Sistema:** conjunto de elementos interrelacionados que operam como uma unidade com o objetivo de executar funções, tarefas ou operações específicas.

Uma propriedade de um sistema é o fato do mesmo poder ser controlado e observado. As interações de um sistema com o ambiente recaem naturalmente em duas categorias. As variáveis que são geradas pelo ambiente e que influenciam o comportamento do sistema são denominadas “entradas” e as variáveis que são determinadas pelo sistema e que podem influenciar o comportamento do ambiente são chamadas “saídas”.

Um sistema é considerado **estático** quando a saída num determinado instante depende apenas da entrada no mesmo instante, o que equivale a dizer que não existem realimentações. Esta definição é válida tanto para sistemas monovariáveis (somente uma entrada e uma saída) como também para sistemas multivariáveis (possuem mais de uma entrada e/ ou saída).

O sistema é considerado **dinâmico** (existe realimentação) quando em um determinado instante de tempo as saídas de um sistema dependem da entrada naquele mesmo instante e também de dados colhidos em instantes de tempo anteriores (das entradas ou saídas).

**Controlabilidade:** Um sistema é dito controlável no instante  $t_0$  se for possível, por meio de um vetor de controle não-restrito, transferir o sistema de qualquer estado inicial  $x(t_0)$  para qualquer outro estado num intervalo de tempo finito.

**Observabilidade:** Um sistema é dito ser observável no instante  $t_0$  se, com o sistema num estado  $x(t_0)$  qualquer, for possível determinar este estado a partir da observação da saída durante um intervalo de tempo finito

**Distúrbio:** Um distúrbio ou perturbação é caracterizado por um sinal que tende a afetar de modo adverso o valor da variável de saída de um sistema. Se um distúrbio for gerado internamente no sistema, ele é dito ser **interno**; ao passo que um distúrbio **externo** é produzido fora do sistema e se comporta como um sinal de entrada no sistema.

Para se estudar um sistema, na maioria das aplicações, é necessário se conhecer pelo menos algumas de suas entradas e observar seu comportamento através das saídas resultantes. Esta constatação leva à uma outra definição para o termo sistema como sendo uma fonte em potencial de dados. Através deste novo conceito é possível definir o termo experimento.

**Experimento:** processo de extração de dados de um determinado sistema através da excitação de suas entradas.

Realizar um experimento em um sistema significa aplicar um conjunto de condições externas às entradas acessíveis e observar a reação do sistema às estas entradas registrando ou armazenando a trajetória de comportamento das saídas acessíveis ao longo do tempo. Portanto, para se realizar um experimento em um sistema é necessário fazer uso da propriedade básica de controlabilidade e observabilidade.

Um dos maiores desafios de experimentos envolvendo sistemas reais é o fato de usualmente sofrerem influências de um grande número de entradas inacessíveis (também chamadas distúrbios). Uma outra limitação que pode ser facilmente constatada é que nem todas as saídas úteis são passíveis de medição por dificuldades físicas impostas pelo próprio sistema, pelo fato de corresponderem a estados internos ou devido a inadequação dos medidores disponíveis. Uma das principais motivações para se realizar uma simulação é, portanto, o fato de todas as entradas e saídas serem acessíveis.

Dado um sistema desconhecido, o processo de obtenção do conhecimento a respeito deste sistema baseia-se, principalmente, na realização de experimentos. Entretanto, no começo deste processo, este conhecimento encontra-se completamente desestruturado. Através do entendimento e da identificação dos elementos que constituem "causa" e dos elementos que constituem "efeito", sob uma perspectiva espacial e temporal, é possível organizar o conhecimento. A modelagem de um sistema consiste neste processo de organização. Assim, após definir os termos sistema e experimento é possível conceituar um modelo.

Um modelo  $M$  para um sistema  $S$  e um experimento  $E$  é alguma coisa através da qual  $E$  pode ser aplicado com o objetivo de responder as perguntas a respeito do sistema  $S$ .

É importante observar que esta definição não condiciona a aplicação do termo modelo somente em termos matemáticos. Um modelo pode abranger um conhecimento ou habilidade específica como o ato de dirigir um carro (neste caso é chamado modelo mental) e não apresenta nenhum formalismo. Um modelo também pode ser expresso de forma

gráfica através de tabelas numéricas ou gráficos de resposta. Objetivamente o termo modelo pode ser definido [Pena e Aguirre, 1997] :

**Modelo:** representação mental (abstrata) ou física (concreta) de observações do mundo real traduzida de forma gráfica ou matemática de acordo com um certo padrão.

O uso que se pretende fazer do modelo determina o grau de formalismo matemático requerido. O modelo deve conter as informações essenciais sobre o sistema, em função da aplicação do mesmo. Modelos que se propõem a serem universais são inevitavelmente impraticáveis ou antieconômicos. Assim, é necessário encontrar um ponto de equilíbrio entre a simplicidade de um modelo e sua capacidade de representar o sistema modelado de maneira mais próxima possível à realidade [Carvalho, 1998].

A complexidade do modelo dependerá do grau de entendimento a respeito dos fenômenos envolvidos. Obviamente, a complexidade de um modelo aumenta na medida em que aumenta o número de fenômenos que se pretende explicar através dele. Nas primeiras etapas do desenvolvimento de um modelo, é difícil prever se o acréscimo de um novo termo permitirá ao modelo incorporar um novo fenômeno ou se este acréscimo somente provocará um aumento desnecessário da complexidade do modelo. É, portanto, um passo importante a determinação das variáveis que são realmente significantes.

De uma maneira geral, modelos complexos e multivariáveis tem aplicação restrita, sendo utilizados principalmente em pesquisas. Para aplicações práticas são desejáveis modelos simplificados, envolvendo somente poucas variáveis. É importante que se utilize de algum conhecimento a priori ou alguma outra técnica apropriada para que se elimine os parâmetros que não possuem importância significativa, de acordo com a aplicação desejada, de modo a se obter um modelo simples e eficiente.

Existem basicamente duas vertentes na derivação de modelos matemáticos. Modelos baseados na física do processo [Doebelin, 1980] e modelos baseados nas relações entrada-saída [Box e Jenkins, 1976].

Modelos baseados na física do processo (também chamados white-box), são desenvolvidos com base nas leis da natureza que regem o comportamento do sistema em estudo. Os parâmetros que descrevem a estrutura do modelo têm, tipicamente, significados físicos próprios [Ljung, 1994]. A utilização desta abordagem permite a derivação de modelos que descrevam a dinâmica interna do sistema. Entretanto, a opção por este método pode ser bastante complicada quando o sistema a ser modelado é grande e complexo.

Modelos baseados nas relações entrada-saída (chamados black-box) são famílias de modelos nos quais os parâmetros não têm significado físico. Estes modelos são, obrigatoriamente, desenvolvidos através da experimentação e coleta de dados de entrada e saída do processo. Neste caso, objetiva-se encontrar uma estrutura que promova um bom ajuste entre os valores calculados pelo modelo e os dados observados no sistema, através de um conjunto de modelos candidatos [Ljung, 1994].

Entre estas duas vertentes, teve origem uma classe de modelos denominados modelos grey-box. Neste tipo de modelo, é utilizado algum conhecimento físico a respeito do processo, durante a etapa de identificação. Porém, este nível de conhecimento a priori não é suficientemente grande para a derivação de um modelo formal parametrizado fisicamente [Lindskog e Ljung, 1997].

Quando um experimento é realizado utilizando-se um modelo surge o termo simulação:

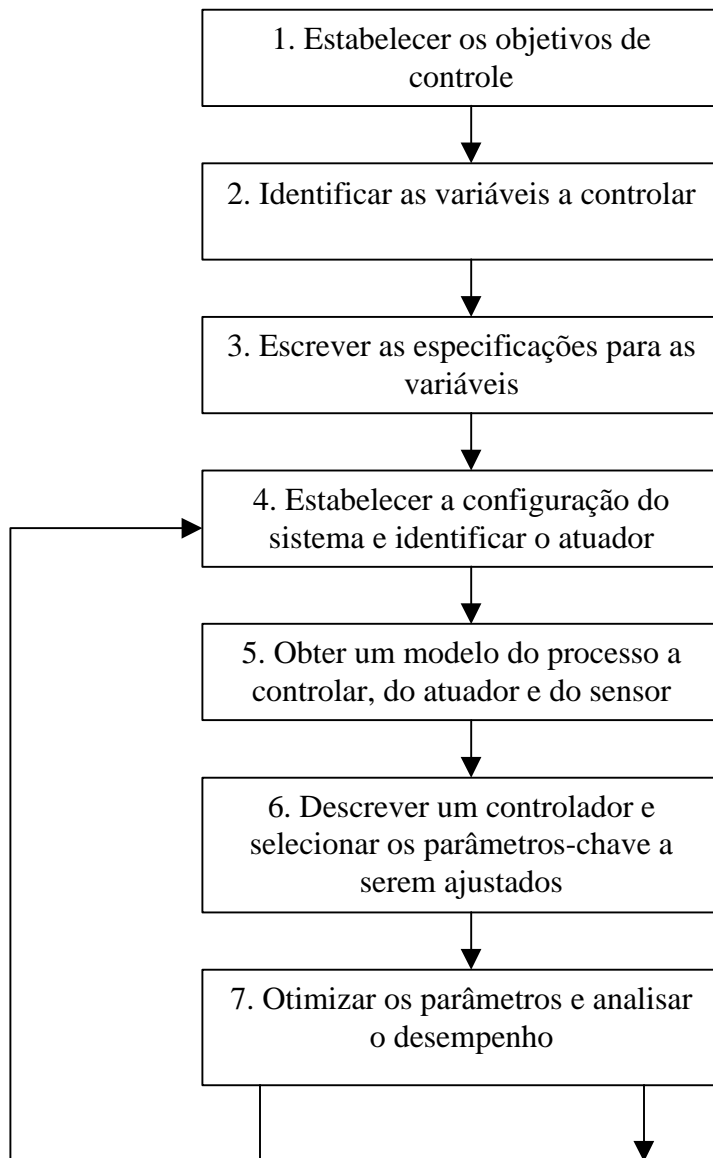
A simulação é um experimento realizado em um modelo

É importante observar que um modelo de um sistema pode ser válido para um experimento e inválido para outro. Tratando-se da identificação de modelos matemáticos, dois fatos devem ser considerados. Primeiramente, os dados observados não são integralmente reproduzíveis devido a ruídos que afetam as medições. Outro aspecto importante é que, fatalmente, os dados observados não representarão todas as situações em que o modelo poderá ser utilizado. Ou seja, considerando-se um sistema dinâmico, é uma tarefa extremamente difícil, ou mesmo impossível, a obtenção de dados que reflitam todos os estados ou situações características do processo a ser modelado [Ljung, 1994]. Desta forma, um modelo, por melhor que seja, não reproduz exatamente o comportamento do sistema original. São considerados bons modelos aqueles que representam o sistema em estudo com uma precisão considerável, de acordo com o fim a que se destinam.

Partindo-se do princípio de que o modelo inicial de um sistema pode ser considerado uma tradução matemática de hipóteses verbais, a partir de um método científico, está implícita a possibilidade de erros. Assim, a modelagem de sistemas reais é um processo iterativo no qual um modelo é formulado, comparado ao sistema real e revisado, se necessário. Este procedimento é repetido até que se atinja um grau de exatidão pré-determinado entre os dados simulados pelo modelo e os dados observados na planta.

# 1 - CARACTERÍSTICAS DE SISTEMAS DE CONTROLE COM RETROAÇÃO

O projeto de um sistema de controle genérico segue as etapas representadas abaixo:



Se o desempenho não atender às especificações, repetir a interação da configuração e do atuador.

Se o desempenho atender às especificações, finalizar então o projeto.

Fig. 1.1

### Exemplo 1.1 Sistema de Leitura do Acionador de Disco

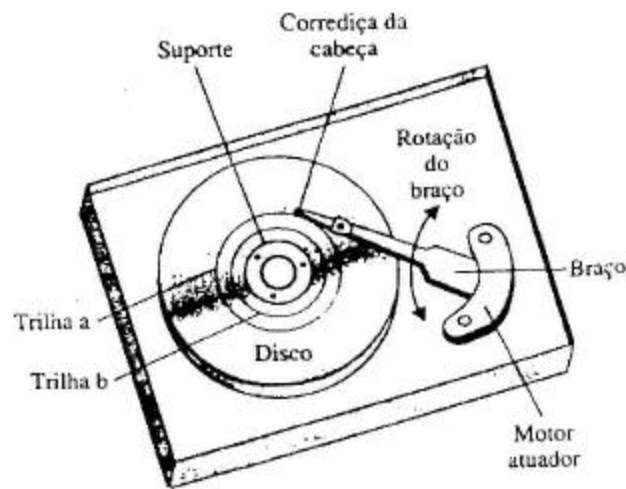
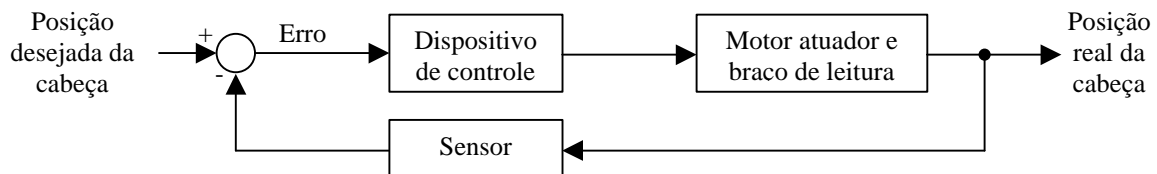


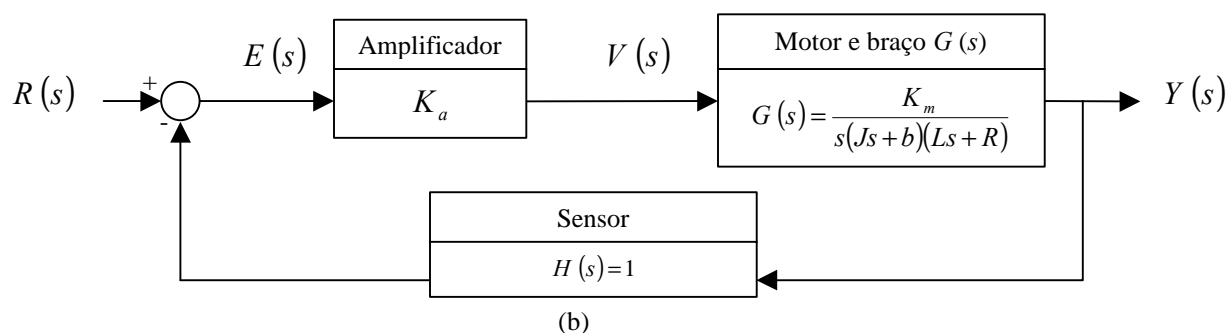
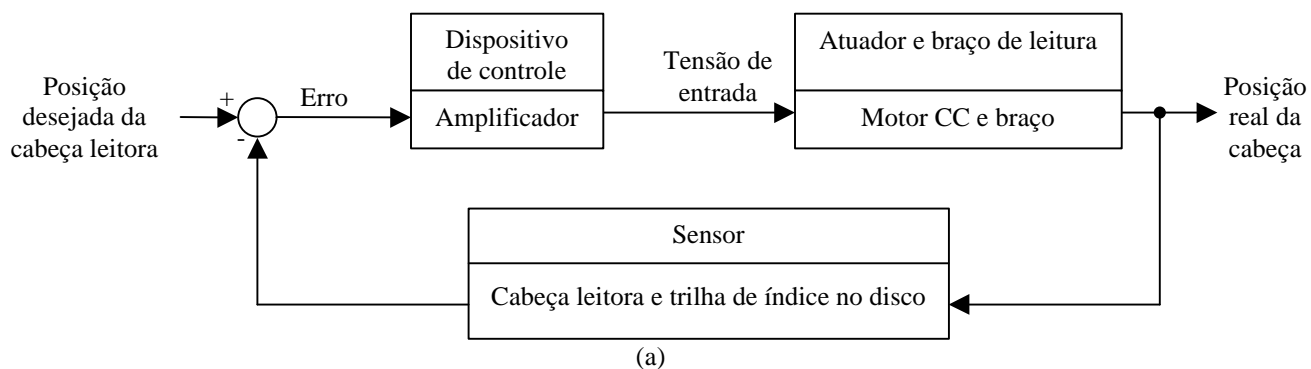
Fig. 1.2

- 1- Identificar os objetivos de controle  
Posicionar a cabeça de leitura para ler os dados armazenados em uma das trilhas do disco
- 2- Identificar as variáveis a controlar  
Posição da cabeça leitora
- 3- Escrever as especificações para as variáveis  
Precisão do posicionamento de  $1 \mu\text{m}$
- 4- Estabelecer a configuração de sistema preliminar





5- Modelo do processo a controlar, do atuador e do sensor

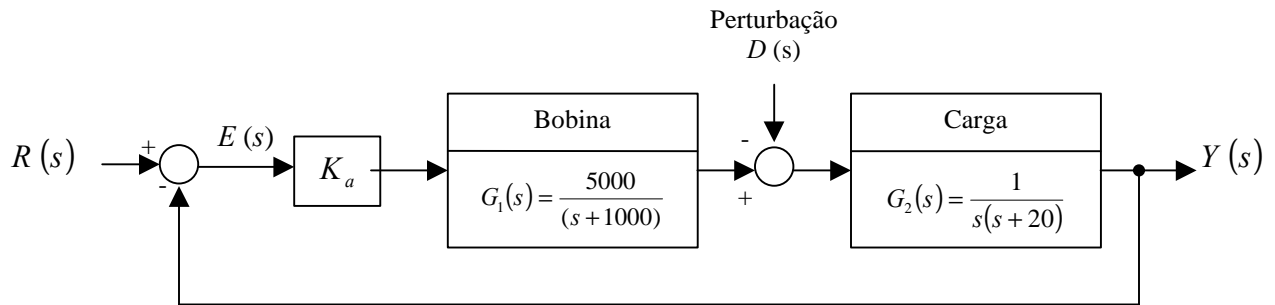
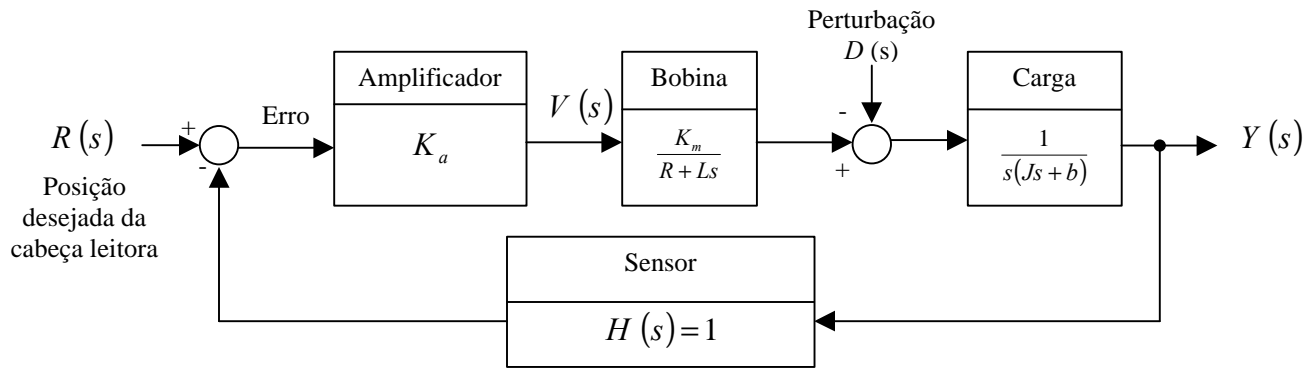


**TABELA 1.1 - Valores Típicos de Parâmetros para a Leitura do Acionador de Disco**

Parâmetro	Símbolo	Valor típico
Inércia do braço e da cabeça leitora	$J$	$1 \text{ N} \cdot \text{m} \cdot \text{s}^2/\text{rad}$
Coefficiente de atrito viscoso	$b$	$20 \text{ kg/m/s}$
Ganho do amplificador	$K_a$	$10 - 1000$
Resistência de armadura	$R$	$1\Omega$
Constante de torque do motor	$K_m$	$5 \text{ N} \cdot \text{m/A}$
Indutância da armadura	$L$	$1 \text{ mH}$

6- Descrever o controlador e selecionar os parâmetros a serem ajustados

7- Otimizar os parâmetros e analisar o desempenho



## 2 - SIMULAÇÃO DE SISTEMAS USANDO O MATLAB

A aplicação de muitas das ferramentas de análise e de projeto de sistemas de controle clássicos e modernos é baseada em modelos matemáticos. O MATLAB pode ser usado com sistemas descritos sob a forma de função de transferência.

Esta seção será iniciada mostrando-se como usar o MATLAB para auxiliar a análise de um modelo matemático típico mola-massa-amortecedor de um sistema mecânico. Usando um conjunto de instruções em MATLAB (script), será desenvolvida uma aptidão de análise interativa para analisar os efeitos da frequência natural e do amortecimento sobre a resposta livre do deslocamento da massa. Esta análise utilizará o fato de que se dispõe de uma solução analítica que descreve a resposta livre no domínio do tempo relativa ao deslocamento da massa.

Mais tarde, serão discutidas funções de transferência e diagramas de blocos. Em particular, o interesse reside em como o MATLAB pode auxiliar na manipulação de polinômios, no cálculo de pólos e zeros da função de transferência, no cálculo das funções de transferência a malha fechada, na redução de diagramas de blocos e no cálculo da resposta de um sistema a uma excitação em degrau unitário.

As funções cobertas nesta seção são `roots`, `roots1`, `series`, `parallel`, `feedback`, `cloop`, `poly`, `conv`, `polyval`, `printsys`, `minreal`, `pzmap`, e `step`.

**Sistema Mola-Massa-Amortecedor.** Um sistema mecânico mola-massa-amortecedor está mostrado na Fig. 2.1. O movimento da massa, designado por  $y(t)$ , é descrito pela equação diferencial

$$M\ddot{y}(t) + b\dot{y}(t) + ky(t) = r(t)$$

A resposta dinâmica livre,  $y(t)$ , do sistema mecânico mola-massa-amortecedor é

$$y(t) = \frac{y(0)}{\sqrt{1-z^2}} e^{-z\omega_n t} \text{sen}(\omega_n \sqrt{1-z^2} t + \varphi)$$

onde  $\varphi = \cos^{-1} z$ . O deslocamento inicial é  $y(0)$ . A resposta transitória do sistema é **subamortecida** quando  $z < 1$ , **superamortecida** quando  $z > 1$ , e **criticamente amortecida** quando  $z = 1$ . Pode-se usar o MATLAB para visualizar a resposta livre no domínio do tempo do deslocamento da massa a partir de um deslocamento inicial  $y(0)$ . Considerem-se os casos superamortecido e subamortecido:

- Caso 1:  $y(0)=0,15\text{m}$ ,  $\omega_n = \sqrt{2} \text{ rad/sec}$ ,  $z_1 = \frac{3}{3\sqrt{2}}$   $\left( \frac{k}{M} = 2, \frac{b}{M} = 3 \right)$
- Caso 2:  $y(0)=0,15\text{m}$ ,  $\omega_n = \sqrt{2} \text{ rad/sec}$ ,  $z_2 = \frac{1}{2\sqrt{2}}$   $\left( \frac{k}{M} = 2, \frac{b}{M} = 1 \right)$

```
>>y0=0.15; wn=sqrt(2)
>>zeta1=3/(2*sqrt(2)); zeta2=1/(2*sqrt(2));
>>t=[0:0.1:10],
>>unforced
```

$\omega_n$

$\zeta_1$  e  $\zeta_2$

unforced.m

```
%Calcula a Resposta Livre a uma Condição Inicial
%
t1=acos(zeta1)*ones(1,length(t));
t2=acos(zeta2)*ones(1,length(t));
c1=(y0/sqrt(1-zeta1^2));c2=(y0/sqrt(1-zeta2^2));
y1=c1*exp(-zeta1*wn*t).*sin(wn*sqrt(1-zeta1^2)*t+t1);
y2=c2*exp(-zeta2*wn*t).*sin(wn*sqrt(1-zeta2^2)*t+t2);
%
bu=c2*exp(-zeta2*wn*t);bl=-bu;
%
plot(t,y1,'-',t,y2,'-',t,bu,'--',t,bl,'--'), grid
xlabel('Tempo[s]'), ylabel('Deslocamento y(t) [m]')
text(0.2,0.85,['zeta1, superamortecido =',num2str(zeta1),...
'],'sc')
text(0.2,0.80,['zeta2, subamortecido =',num2str(zeta2),...
'],'sc')
```

$\cos^{-1} \zeta$

$y(0)/\sqrt{1-\zeta^2}$

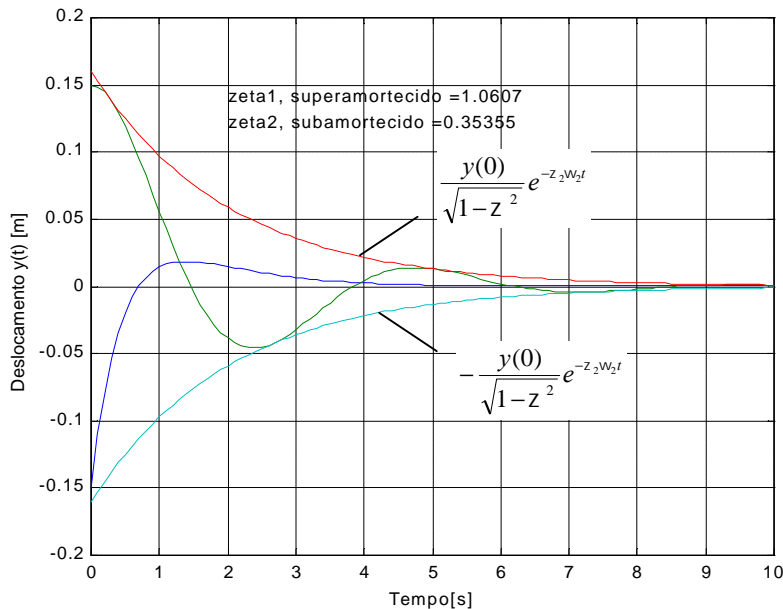
Envoltória  $e^{-\zeta\omega_n t}$

**Fig. 2.1** Conjunto de instruções para analisar o sistema mola-massa-amortecedor.

Os comandos em MATLAB para gerar o gráfico da resposta livre estão mostrados na Fig.1. Ao início do uso do MATLAB, os valores das variáveis e parâmetros  $y(0)$ ,  $\omega_n$ ,  $t$ ,  $Z_1$  e  $Z_2$  são digitados no espaço de trabalho ao nível de comando. Em seguida, executa-se o programa sistema\_livre.m para gerar os gráficos desejados. Isto cria uma possibilidade de análise interativa que permite analisar os efeitos da frequência natural e do amortecimento na resposta livre do deslocamento da massa. Pode-se investigar os efeitos da frequência natural e do amortecimento sobre a resposta temporal entrando-se com novos valores de  $\omega_n$ ,  $Z_1$  e  $Z_2$  no sinal de pedido de comando (prompt) e rodando novamente o programa sistema\_livre.m. O gráfico da resposta temporal está mostrado na Fig. 2.2. Observe-se que o programa rotula o gráfico automaticamente com os valores dos coeficientes de amortecimento. Isto evita confusão ao se executarem muitas simulações interativas. O valor da frequência natural poderia ser assinalado no gráfico. A utilização de programas constitui um aspecto importante do desenvolvimento de uma capacidade efetiva de análise e projeto interativos no MATLAB.

Para o problema mola-massa-amortecedor, a solução livre da equação diferencial estava prontamente disponível. Em geral, ao simular sistemas de controle com retroação a malha fechada sujeitos a diversas entradas e condições iniciais, é difícil tentar obter analiticamente uma solução. Nestes casos pode-se usar o MATLAB para calcular as soluções numericamente e exibir a solução graficamente.

O MATLAB pode ser usado para analisar sistemas descritos por meio de funções de transferência. Como a função de transferência é uma relação entre dois polinômios, começa-se investigando como o MATLAB manipula polinômios, recordando que trabalhar com funções de transferência significa que tanto o polinômio do numerador quanto o polinômio do denominador devem ser especificados.



**Fig. 2.2** Resposta livre do sistema mola-massa-amortecedor.

No MATLAB, os polinômios são representados por vetores linha contendo os coeficientes do polinômio em ordem decrescente. Por exemplo, o polinômio

$$p(s) = s^3 + 3s^2 + 4$$

é introduzido como está mostrado na Fig. 2.3. Observe-se que mesmo com o coeficiente de  $s$  sendo igual a zero, ele é incluído na definição de entrada de  $p(s)$ .

Se  $\mathbf{p}$  for um vetor linha contendo os coeficientes de  $p(s)$  em ordem decrescente, então  $\text{roots}(\mathbf{p})$  é um vetor coluna contendo as raízes do polinômio. Reciprocamente, se  $\mathbf{r}$  for um vetor coluna contendo as raízes do polinômio, então  $\text{poly}(\mathbf{r})$  é um vetor linha com os coeficientes do polinômio em ordem decrescente. Pode-se calcular as raízes do polinômio  $p(s) = s^3 + 3s^2 + 4$  com a função  $\text{roots}$  como está mostrado na Fig. 2.3. A função  $\text{roots}$  também calcula as raízes de um polinômio mas fornece um resultado mais exato quando o polinômio possui raízes repetidas. Na Fig. 2.3, é mostrado também como remontar o polinômio com a função  $\text{poly}$ . A multiplicação de polinômios é efetuada com a função  $\text{conv}$ . Suponha-se que se deseje expandir o polinômio  $n(s)$ , onde

$$n(s) = (3s^2 + 2s + 1)(s + 4).$$

Os comandos MATLAB associados usando a função `conv` estão mostrados na Fig. 2.4. Assim o polinômio expandido é

$$n(s) = (3s^2 + 2s + 1)(s + 4).$$

A função `polyval` é usada para calcular o valor de um polinômio para um dado valor da variável. O polinômio  $n(s)$  tem o valor  $n(-5) = -66$ , conforme está mostrado na Fig. 2.4.

No próximo exemplo será obtido um gráfico com as localizações dos pólos e zeros no plano complexo. Isto será realizado usando a função `pzmap`, mostrada na Fig. 2.5. No gráfico de pólos e zeros, os zeros serão representados por um “o” e os pólos por um “X”. Se a função `pzmap` for chamada sem os argumentos da esquerda da igualdade, o gráfico será gerado automaticamente.

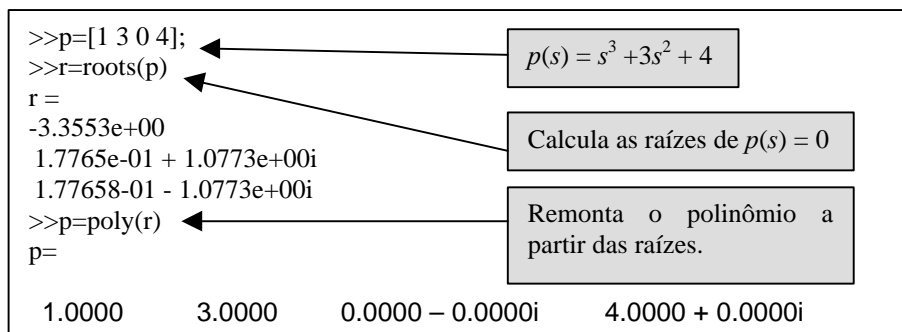


Fig. 2.3 Entrando com o polinômio  $p(s) = s^3 + 3s^2 + 4$  e calculando suas raízes.

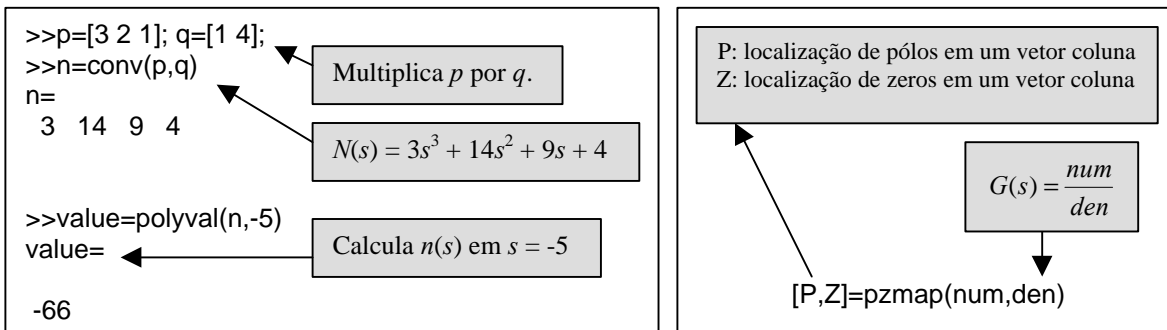


Fig. 2.4 Usando `conv` e `polyval` para multiplicar os polinômios.  $(3s^2 + 2s + 1)(s + 4)$ .

### 3 - ANÁLISE DE MODELOS COM VARIÁVEIS DE ESTADO USANDO O MATLAB

O método no domínio do tempo utiliza uma **representação no espaço de estados** de um modelo de sistema, dado por

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{b}u \quad \text{e} \quad y = \mathbf{C}\mathbf{x} + \mathbf{D}u \quad (3.1)$$

O vetor  $\mathbf{x}$  é o estado do sistema,  $\mathbf{A}$  é a matriz de sistema  $n \times n$  constante,  $\mathbf{B}$  é a matriz de entrada  $n \times m$  constante,  $\mathbf{C}$  é a matriz de saída  $p \times n$  constante e  $\mathbf{D}$  é uma matriz constante  $p \times m$ . O número de entradas,  $m$ , e o número de saídas,  $p$ , são consideradas uma única, pois estão sendo considerados somente problemas de única entrada, única saída (SISO). Por conseguinte,  $y$  e  $u$  não são variáveis representadas em negrito (matrizes).

Os principais elementos da representação no espaço de estados na Eq. (3.1) são o vetor de estado  $\mathbf{x}$  e as matrizes constantes ( $\mathbf{A}$ ,  $\mathbf{B}$ ,  $\mathbf{C}$ ,  $\mathbf{D}$ ). Como a principal unidade computacional no MATLAB é a matriz, a representação no espaço de estados presta-se intrinsecamente bem ao ambiente do MATLAB. De fato, o MATLAB inclui tantos aspectos de métodos no espaço de estados que não seria possível discuti-los todos aqui. Três novas funções cobertas nesta seção são `tf2ss`, `ss2tf`, e `lsim`. Será considerado também o uso da função `expm` para calcular a matriz de transição de estados.

Dada uma função de transferência, pode-se obter uma representação no espaço de estados equivalente e vice-versa. O MATLAB tem duas funções que convertem sistemas a partir da função de transferência para o espaço de estados e de volta para a função de transferência. A função `tf2ss` converte uma representação em função de transferência para uma representação no espaço de estados; a função `ss2tf` converte uma representação no espaço de estados para uma função de transferência. Estas funções são mostradas na Fig. 3.1.

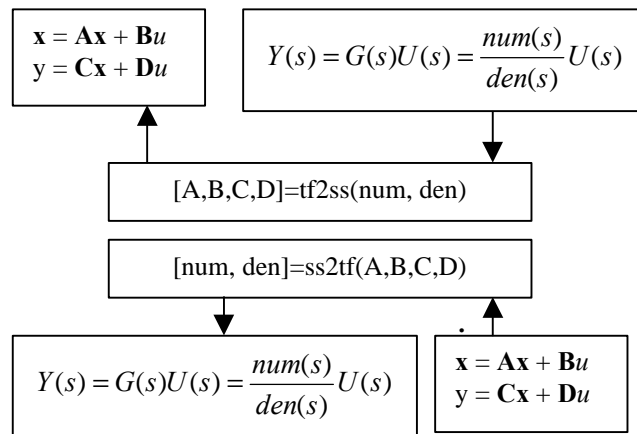


Fig. 3.1 Conversão de modelo de sistema linear.

Por exemplo, considere-se o sistema de terceira ordem

$$T(s) = \frac{Y(s)}{R(s)} = \frac{2s^2 + 8s + 6}{s^3 + 8s^2 + 16s + 6} \quad (3.2)$$

Pode-se obter uma representação no espaço de estados com variáveis de fase usando a função `tf2ss`, como mostrado na Fig. 3.2. A representação no espaço de estados da Eq. (3.2) é dada pela Eq. (3.1), onde

$$\mathbf{A} = \begin{bmatrix} -8 & -16 & -6 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix},$$

e

$$\mathbf{C} = [2 \ 8 \ 6], \quad \mathbf{D} = [0].$$

Note-se que o formato no espaço de estados chamado por `tf2ss` utiliza as variáveis de estado definidas como apresentado na Fig. 3.3, que mostra  $x_1$  no integrador da extremidade esquerda.

A resposta no domínio do tempo do sistema da Eq. (3.1) é dada pela solução da equação diferencial vetorial

$$x(t) = \exp(At)x(0) + \int_0^t \exp[A(t-t)]Bu(t)dt \quad (3.3)$$

A função exponencial matricial na Eq. (3.3) é a matriz de transição de estados,  $\Phi(t)$ , onde

$$\Phi(t) = \exp(At),$$

e

$$e^{At} = \exp(At) = I + At + \frac{A^2 t^2}{2!} + \dots + \frac{A^k t^k}{k!} + \dots$$

Pode-se usar a função `expm` para calcular a matriz de transição para um dado instante de tempo, como ilustrado na Fig. 3.4. A função `expm(A)` calcula a exponencial matricial. Em contraste, a função `exp(A)` calcula  $e^{a_{ij}}$  para cada um dos elementos  $a_{ij} \in \mathbf{A}$ .



convert.m

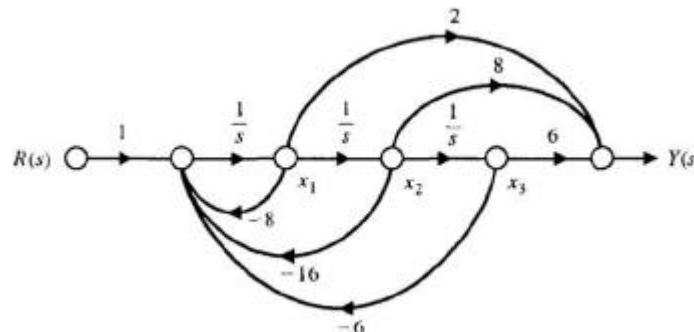
```
% Converte G(s) =  
(2s^2+8s+6)/(s^3+8s^2+16s+6)  
% em uma representação no espaço de  
estados  
%  
num=[2 8 6];den=[1 8 16 6];  
[A,B,C,D]=tf2ss(num,den);  
printsys(A,B,C,D);
```

(a)

```
»convert  
a =  
      x1      x2      x3  
x1  -8.00000 -16.00000 -6.00000  
x2   1.00000  0         0  
x3   0         1.00000  0  
  
b =  
      u1  
x1   1.00000  
x2   0  
x3   0  
  
c =  
      x1      x2      x3  
y1   2.00000  8.00000  6.00000
```

(b)

**Fig. 3.2** Conversão da Eq. (3.2) para uma representação no espaço de estados com variáveis de fase. (a) Script MATLAB, (b) saída impressa.



**Fig. 3.3** Modelo em diagrama de fluxo de sinal com  $x$ , sendo definida como a variável de estado mais à esquerda.

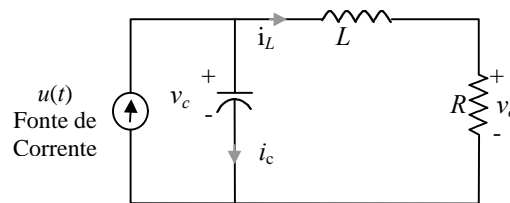
$$\dot{x} = \begin{bmatrix} 0 & -1/c \\ 1/L & -R/L \end{bmatrix} x + \begin{bmatrix} 1/c \\ 0 \end{bmatrix} u(t)$$

$$y = [0 \quad R]x \quad (3.4)$$

onde,  $R = 3$ ,  $L = 1$  e  $c = 1/2$

Por exemplo, seja considerar o circuito *RLC* da abaixo descrito pela representação da Eq. (3.4) com

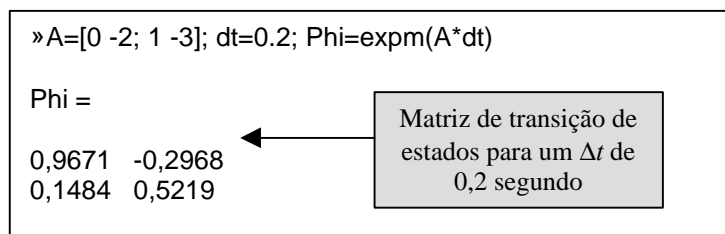
$$\mathbf{A} = \begin{bmatrix} 0 & -2 \\ 1 & -3 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 2 \\ 0 \end{bmatrix}, \quad \mathbf{C} = [1 \quad 0], \quad \mathbf{D} = 0.$$



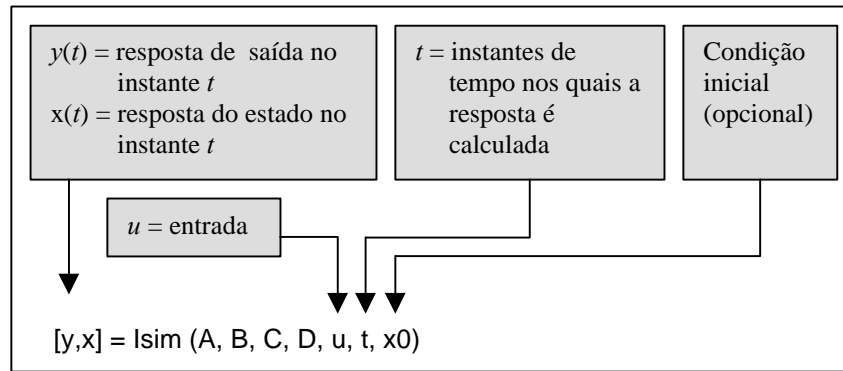
As condições iniciais são  $x_1(0) = x_2(0) = 1$  e a entrada  $u(t) = 0$ . Em  $t = 0,2$ , a matriz de transição de estados é dada na Fig. 3.4. O estado em  $t = 0,2$  é dado pelos métodos de transição de estado para ser

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_{t=0,2} = \begin{bmatrix} 0,9671 & -0,2968 \\ 0,1484 & 0,5219 \end{bmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}_{t=0} = \begin{pmatrix} 0,6703 \\ 0,6703 \end{pmatrix}$$

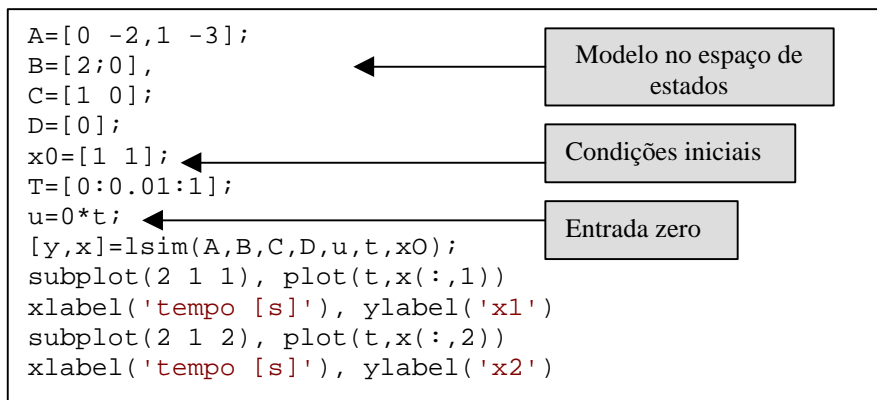
A resposta no domínio do tempo do sistema da Eq. (3.1) pode também ser obtida usando a função `lsim`. A função `lsim` pode receber como entrada condições iniciais não-nulas bem como uma função de entrada, como mostrado na Fig. 3.5. Usando a função `lsim`, pode-se calcular a resposta do circuito *RLC* como mostrado na Fig. 3.6.



**Fig. 3.4** Calculando a matriz de transição de estados para um tempo dado,  $\Delta t = dt$ .



**Fig. 3.5** A função **lsim** para calcular a saída e a resposta de estado.



**Fig. 3.6** Calculando a resposta no domínio do tempo para condições iniciais não-nulas e entrada zero usando **lsim**.

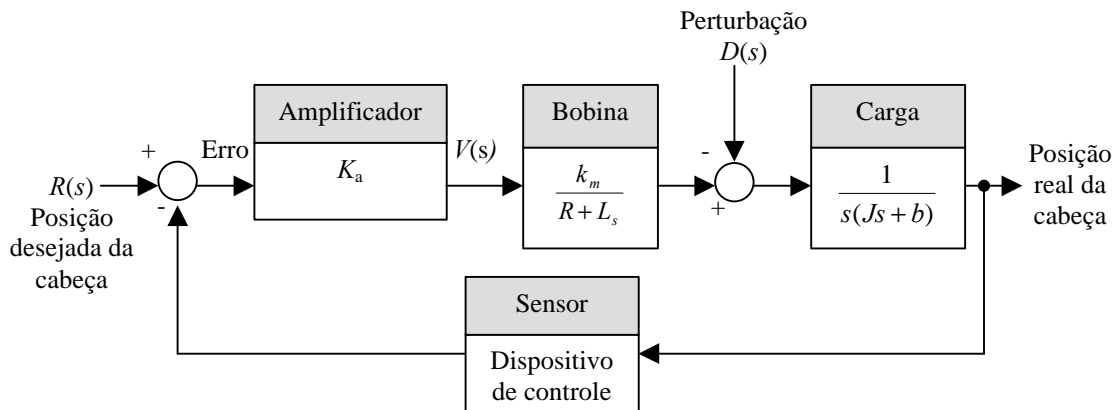
Prevê-se que o estado em  $t = 0,2$  com a função **lsim** como sendo  $x_1(0,2) = x_2(0,2) = 0,6703$ . Se for possível comparar os resultados obtidos pela função **lsim** com os obtidos multiplicando-se o vetor de condição inicial de estado pela matriz de transição de estados, serão encontrados resultados exatamente idênticos.

## 4 - EXEMPLO DE PROJETO SEQUENCIAL: SISTEMA DE LEITURA DE ACIONADOR DE DISCO

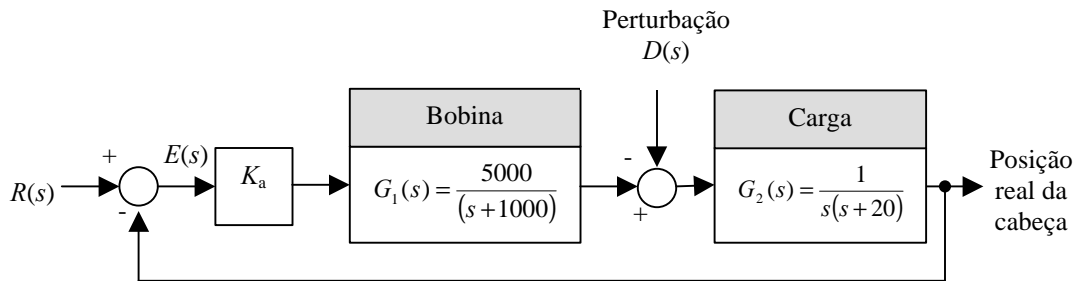
O projeto de um sistema de acionador de disco é um exercício de compromisso e de otimização. O acionador de disco deve posicionar, com precisão, a cabeça de leitura ao mesmo tempo que deve ser capaz de reduzir os efeitos devidos à variação de parâmetros e aos choques e vibrações externos. O braço mecânico e a conexão flexível poderão apresentar ressonância em frequências que podem ser provocadas por excitações tais como choques mecânicos sobre um computador portátil. As perturbações na operação do acionador de disco incluem choques mecânicos, desgaste ou desalinhamento nos mancais do eixo, bem como alterações nos parâmetros devidas a mudanças ocorridas com os componentes. Nesta seção será examinado o desempenho do sistema de acionador de disco sob a ação de perturbações e de mudanças nos parâmetros do sistema. Adicionalmente será examinado o erro de estado estacionário do sistema a um comando em degrau e a resposta transitória à medida que o ganho do amplificador,  $K_a$ , é ajustado. Assim, nesta seção estão sendo realizados os passos 6 e 7 do procedimento de projeto da Fig. 1.1.

Considere-se o sistema mostrado na Fig. 4.1. Este sistema a malha fechada usa como controlador um amplificador com ganho variável. Usando os parâmetros especificados na TABELA 1.1, obtêm-se as funções de transferência mostradas na Fig. 4.2. Primeiro serão determinados os estados estacionários para uma entrada em degrau unitário,  $R(s) = 1/s$ , quando  $D(s) = 0$ . Então, para  $H(s) = 1$ , será obtido,

$$E(s) = \frac{1}{1 + K_a G_1(s) G_2(s)} R(s).$$



**Fig. 4.1** Sistema de controle para cabeça de leitura de acionador de disco.



**Fig. 4.2** Sistema de controle de cabeça de acionador de disco com os parâmetros típicos da TABELA 1.1.

Logo

$$\lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} s \left[ \frac{1}{1 + K_a G_1(s) G_2(s)} \right] \frac{1}{s} \quad (4.1)$$

Então, o erro de estado estacionário é  $e(\infty) = 0$  para uma entrada em degrau. Este desempenho é obtido a despeito de alterações nos parâmetros do sistema.

Seja agora determinar o desempenho transitório do sistema à medida que  $K_a$  é ajustado. A função de transferência a malha fechada (com  $D(s) = 0$ ) é

$$\begin{aligned} T(s) &= \frac{Y(s)}{R(s)} = \frac{K_a G_1(s) G_2(s)}{1 + G_1(s) G_2(s)} \\ &= \frac{5000 K_a}{s^3 + 1020s^2 + 20000s + 5000 K_a} \end{aligned} \quad (4.2)$$

Usando os comandos do script MATLAB mostrado na Fig. 4.3(a), obtêm-se as respostas do sistema para  $K_a = 10$  e  $K_a = 80$ , mostradas na Fig. 4.3(b). Obviamente o sistema é mais rápido na resposta ao comando de entrada quando  $K_a = 80$ , mas a resposta é inaceitavelmente oscilatória.

Seja determinar agora o efeito da perturbação  $D(s) = 1/s$  quando  $R(s) = 0$ . Deseja-se reduzir o efeito da perturbação a um nível insignificante. Usando o sistema da Fig. 4.2, obtém-se a resposta  $Y(s)$  para a entrada  $D(s)$  quando  $K_a = 80$  como

$$Y(s) = \frac{G_2(s)}{1 + k_a G_1(s) G_2(s)} D(s) \quad (4.3)$$

Usando-se o script MATLAB mostrado na Fig. 4.4(a), obtém-se a resposta do sistema quando  $K_a = 80$  e  $D(s) = 1/s$ , como está mostrado na Fig. 4.4(b). A fim de reduzir ainda mais o efeito da perturbação, seria necessário aumentar o valor de  $K_a$  acima de 80. Contudo, a resposta a um comando em degrau  $r(t) = 1, t > 0$  seria inaceitavelmente oscilatória. No próximo capítulo se tentará determinar o melhor valor para  $K_a$  considerando o requisito especificado de se ter uma resposta rápida, porém não-oscilatória.

As razões fundamentais para se usar retroação, a despeito do custo e da complexidade adicional, são as seguintes:

1. Diminuição da sensibilidade do sistema com relação a variações nos parâmetros do processo  $G(s)$ .
2. Facilidade de controlar e ajustar a resposta transitória do sistema.
3. Melhoria na rejeição a perturbações e ruídos no interior do sistema.
4. Melhoria na redução do erro de estado estacionário do sistema.

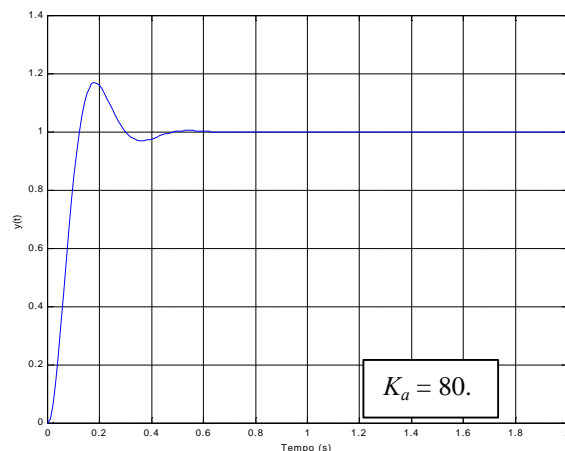
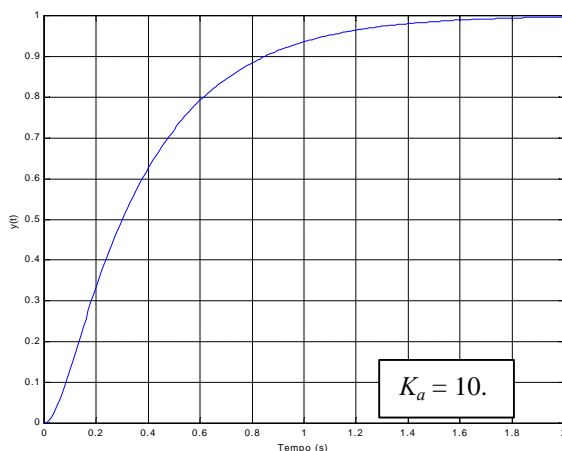
```

Ka=10;
nf=[5000];
df=[1 1000];
ng=[1];
dg=[1 20 0];
[num,den]=series(Ka*nf,df,ng,dg);
[n,d]=cloop(num,den);
t=[0:0.01:2];
y=step(n,d,t);
plot(t,y), grid
ylabel('y(t)'), xlabel('Tempo (s)')

```

Seleciona  $K_a$

(a)



(b)

**Fig. 4.3** Resposta a malha fechada, (a) Script MATLAB. (b) Resposta a um degrau para  $K_a = 10$  e  $K_a = 80$ .

Os benefícios da retroação podem ser ilustrados considerando-se o sistema mostrado na Fig. 4.5(a). Este sistema pode ser considerado para diversos valores de ganho,  $K$ . A TABELA 4.1 resume os resultados do sistema quando operado como um sistema a malha aberta (desconectando-se o percurso de retroação) e para diversos valores de ganho,  $K_a$  com a retroação ligada. Fica evidente que o tempo de subida e a sensibilidade do sistema são reduzidos quando o ganho é aumentado. Além disto, o sistema com retroação demonstra excelente redução do erro de estado estacionário à medida que se aumenta o ganho. Finalmente, a Fig. 4.5(b) mostra a resposta para uma perturbação em degrau unitário (para  $R(s) = 0$ ) e mostra como um ganho maior reduzirá o efeito da perturbação.

Os sistemas de controle de retroação possuem muitas características benéficas, e não é surpresa encontrar uma enorme gama de sistemas com retroação em diversos campos como indústria, governo e natureza.

```

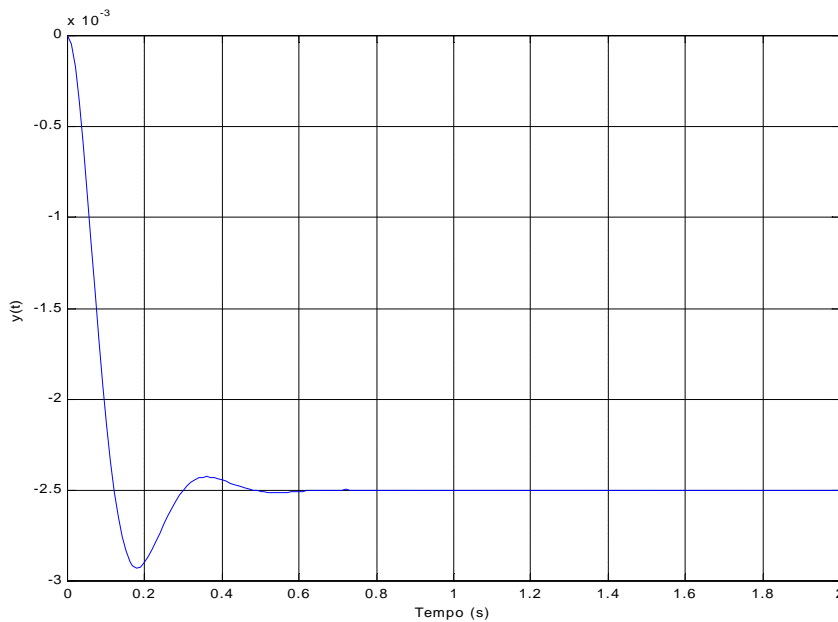
Ka=80;
nf=[5000];
df=[1 1000];
ng=[1];
dg=[1 20 0];
[num,den]=feedback(ng,dg,Ka*nf,df);
num=-num;
t=[0:0.01:2];
y=step(num,den,t);
plot(t,y), grid
ylabel('y(t)'), xlabel('Tempo (s)')

```

Seleciona  $K_a$

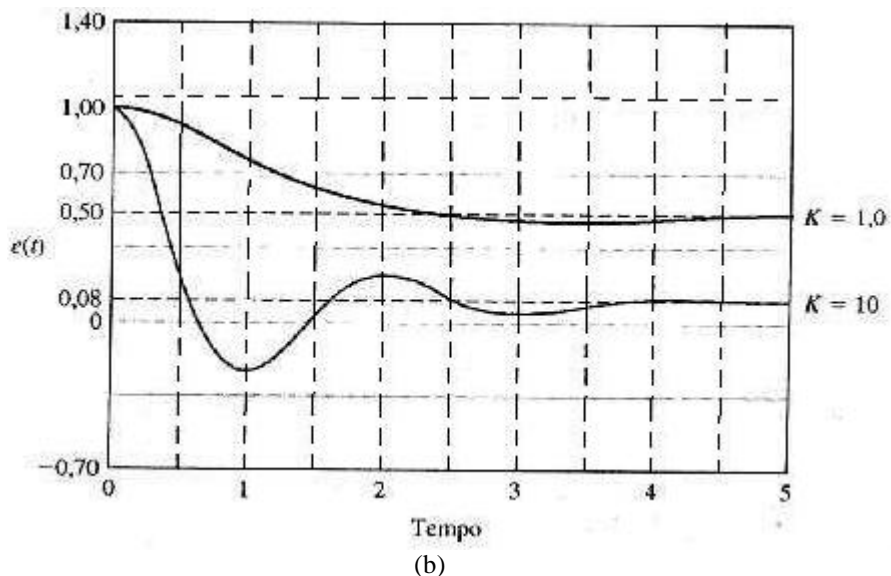
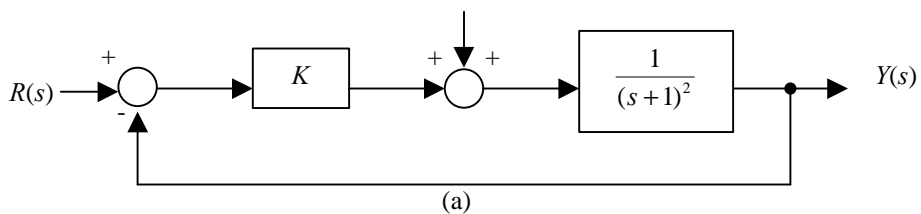
A perturbação entra no somador com um sinal negativo

(a)



(b)

**Fig. 4.4** Resposta a uma perturbação em degrau, (a) Script MATLAB. (b) Resposta a uma perturbação em degrau unitário para  $K_a = 80$ .



**Fig. 4.5** (a) Sistema de controle com retroação constituído por malha única. (b) O erro em resposta a uma perturbação em degrau unitário para  $R(s) = 0$ .

**TABELA 4.1** Resposta ao Sistema Mostrado na Fig. 4.5 (a)

	Malha Aberta	Malha Fechada		
	K = 1	K = 1	K = 8	K = 10
Tempo de subida (segundos) (10% a 90% do valor final)	3,35	1,52	0,45	0,38
Ultrapassagem percentual de (%)	0	4,31	33	40
Valor final de $y(t)$ devido a perturbação $D(s) = 1/s$	1,0	0,50	0,11	0,09
Variação percentual do erro de estado estacionário para entrada em degrau unitário	0	50%	11%	9%
Variação percentual no erro estacionário devido a 10% de decréscimo no valor de K	10%	5,3%	1,2%	0,9%

\*Resposta somente quando  $K = 1$  exatamente.



## 5 - DESEMPENHO DO SISTEMA USANDO O MATLAB

Nesta seção serão investigadas as especificações de desempenho no domínio do tempo fornecidas em termos de resposta transitória a um dado sinal de entrada e os erros de acompanhamento resultantes em estado estacionário. Conclui-se com uma discussão da simplificação de sistemas lineares. As funções do MATLAB introduzidas nesta seção são `impulse` e `lsim`. Estas funções são usadas para simular sistemas lineares.

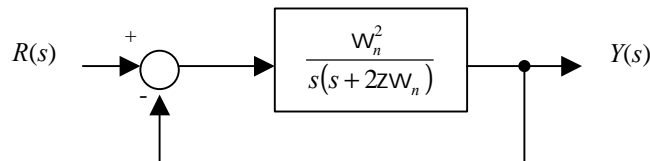
**Especificações no domínio do tempo.** As especificações de desempenho no domínio do tempo são fornecidas geralmente em termos da resposta transitória de um sistema a um dado sinal de entrada. Como os sinais de entrada são geralmente desconhecidos, é usado um sinal de teste padronizado como entrada. Considere-se o sistema de Segunda ordem mostrado na Fig. 5.1. A saída a malha fechada é

$$Y(s) = \frac{W_n^2}{s^2 + 2ZW_n s + W_n^2} R(s) \quad (5.1)$$

Já foi discutido o uso da função `step` para calcular a resposta do sistema ao degrau. Agora será considerado outro sinal de teste importante: o impulso. A resposta ao impulso é derivada em relação ao tempo da resposta ao degrau. Calcula-se a resposta ao impulso como função `impulse` mostrada na Fig. 5.2.

Pode-se obter uma resposta transitória de um sistema de segunda ordem a uma entrada degrau com a função `step`, como está mostrado na Fig. 5.3. Usando a função `impulse`, pode-se obter uma resposta transitória de um sistema de segunda ordem a uma entrada em impulso. A resposta de um sistema de Segunda ordem para uma função impulso de entrada é mostrada na Fig. 5.4. No script, fez-se  $\omega_n = 1$ , que é equivalente a calcular a resposta ao degrau *versus*  $\omega_n t$ . Isto fornece um gráfico mais geral para qualquer  $\omega_n > 0$ .

Em muitos casos, pode ser necessário simular a resposta do sistema a uma entrada arbitrária porém conhecida. Nestes casos, usa-se a função `lsim`. A função `lsim` é mostrada na Fig. 5.5.



**Fig. 5.1** Sistema com retroação de Segunda ordem de malha única.

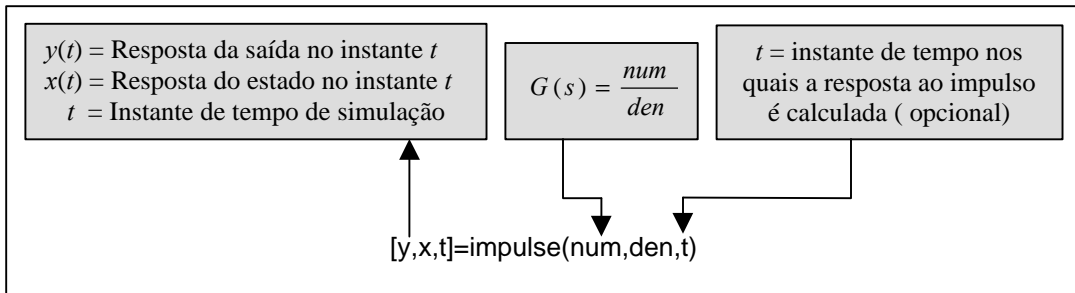
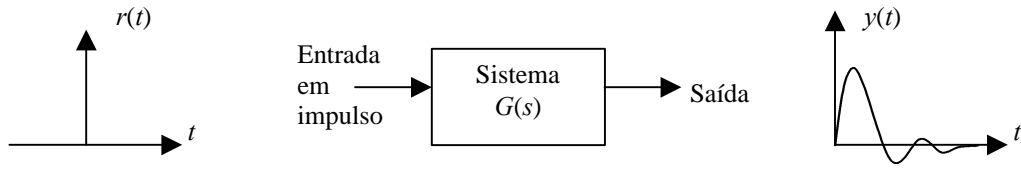
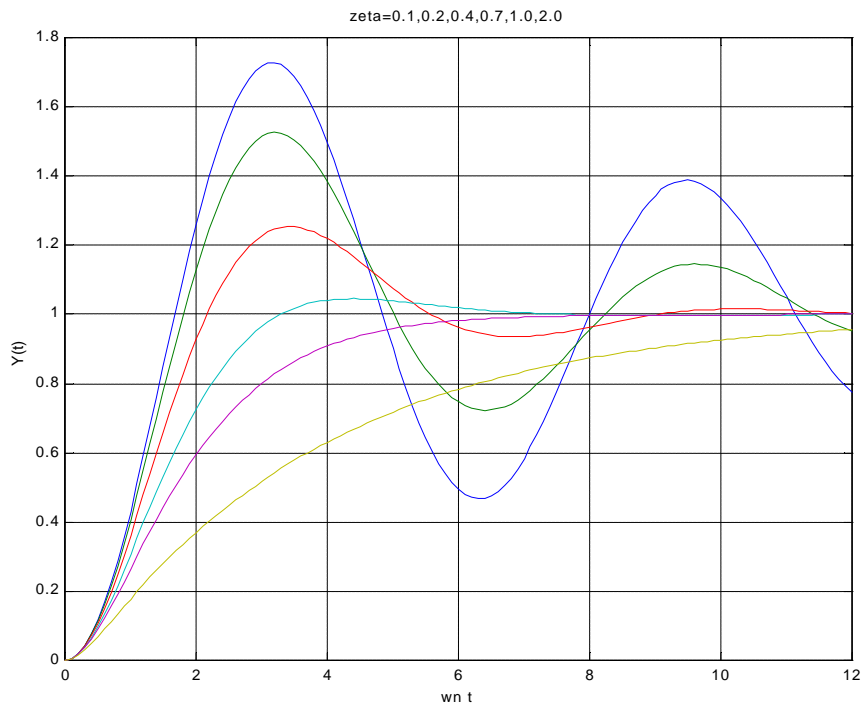


Fig. 5.2 A função **impz**.



(a)

```

% Calcula a resposta de um sistema de segunda ordem ao degrau
% Duplica a Figura 5.5(a)
%
t=[0:0.1:12];
num=[1];
zeta1=0.1; den1=[1 2*zeta1 1];
zeta2=0.2; den2=[1 2*zeta2 1];
zeta3=0.4; den3=[1 2*zeta3 1];
zeta4=0.7; den4=[1 2*zeta4 1];
zeta5=1.0; den5=[1 2*zeta5 1];
zeta6=2.0; den6=[1 2*zeta6 1];
%
[y1,x,t]=step(num,den1,t);
[y2,x,t]=step(num,den2,t);
[y3,x,t]=step(num,den3,t);
[y4,x,t]=step(num,den4,t);
[y5,x,t]=step(num,den5,t);
[y6,x,t]=step(num,den6,t);
%
plot(t,y1,t,y2,t,y3,t,y4,t,y5,t,y6)
xlabel('wn t'),ylabel('Y(t)')
title('zeta=0.1,0.2,0.4,0.7,1.0,2.0'),grid

```

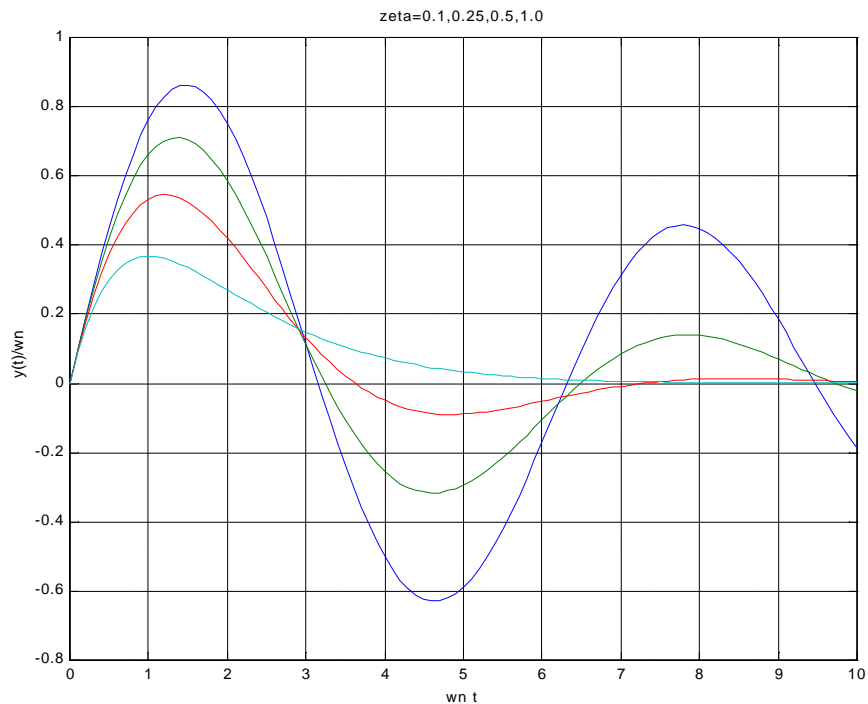
Configura o polinômio do denominador para cada zeta diferente.

Calcula resposta ao degrau.

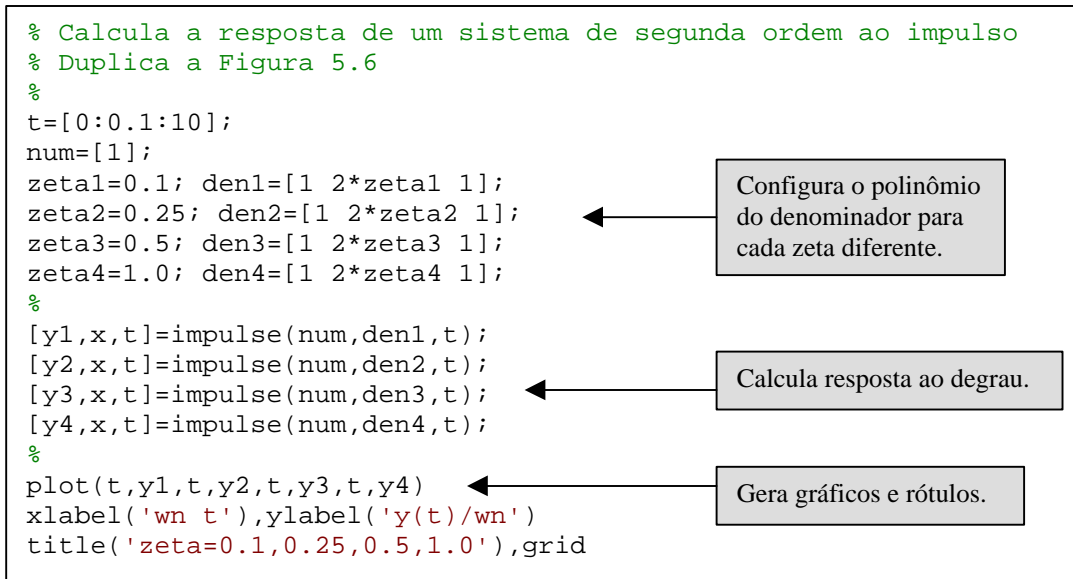
Gera gráficos e rótulos.

(b)

**Fig. 5.3** (a) Resposta de um sistema de Segunda ordem a uma entrada em degrau. (b) Script em MATLAB.

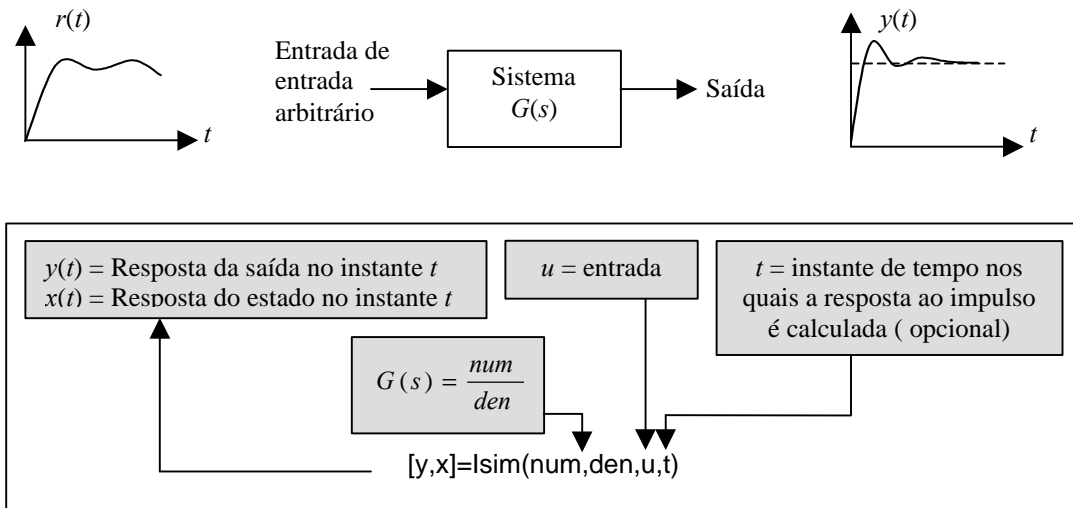


(a)



(b)

**Fig. 5.4** (a) Resposta de um sistema de Segunda ordem a uma função de entrada em impulso. (b) Script em MATLAB.

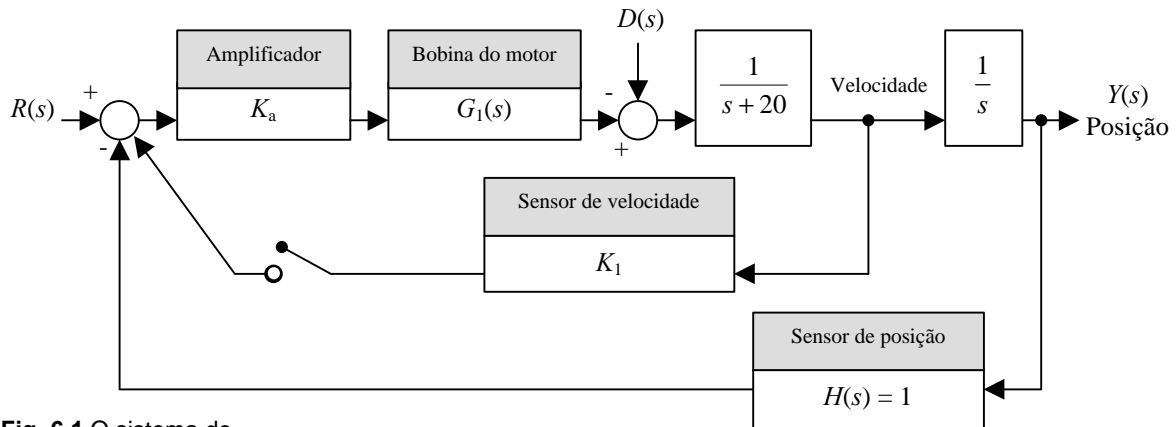


**Fig. 5.5** A função lsim.

## 6 - EXEMPLO DE PROJETO SEQUENCIAL: SISTEMA DE LEITURA DE ACIONADOR DE DISCO

Nesta seção será examinada a estabilidade do sistema de cabeça de leitura à medida que  $K_a$  é ajustado e em seguida o sistema será configurado como Passo 4 do procedimento de projeto (ver Fig. 1.1).

Seja considerar o sistema mostrado na Fig. 6.1. Este é o mesmo sistema com um modelo de motor e carga considerado no Cap.5, exceto que se adicionou um sensor para retroação da velocidade, como está mostrado na Fig. 6.1.



**Fig. 6.1** O sistema de acionamento a malha fechada da cabeça leitora de disco com uma retroação opcional de velocidade.

Inicialmente se considera o caso em que a chave está aberta. Então, a função de transferência de malha fechada é

$$\frac{Y(s)}{R(s)} = \frac{K_a G_1(s) G_2(s)}{1 + K_a G_1(s) G_2(s)},$$

onde

$$G_1(s) = \frac{5000}{s+1000}$$

e

$$G_2(s) = \frac{1}{s(s+20)}.$$

A equação característica é

$$s(s+20)(s+1000) + 5000K_a = 0,$$

ou

$$s^3 + 1020s^2 + 20000s + 5000K_a = 0,$$

Usa-se o arranjo tabular de Routh

$$\begin{array}{r|rr} s^3 & 1 & 20000 \\ s^2 & 1020 & 5000K_a \\ s^1 & b_1 & \\ s^0 & 5000K_a & \end{array}$$

onde

$$b_1 = \frac{(20000)1020 - 5000K_a}{1020}.$$

O caso  $b_1 = 0$  resulta em estabilidade marginal quando  $K_a = 4080$ . Usando-se a equação auxiliar, tem-se

$$1020s^2 + 5000(4080) = 0,$$

ou as raízes sobre o eixo  $j\omega$  são  $s = \pm j141,1$ . Para que o sistema seja estável,  $K_a < 4080$ .

Seja agora a adição de retroação de velocidade pelo fechamento da chave no sistema da Fig. 6.1. A função de transferência a malha fechada para o sistema é então

$$\frac{Y(s)}{R(s)} = \frac{K_a G_1(s)G_2(s)}{1 + [K_a G_1(s)G_2(s)](1 + K_1s)},$$

uma vez que o fator de retroação é igual a  $(1 + K_1s)$ , como está mostrado na Fig. 6.2.

A equação característica é então

$$1 + [K_a G_1(s)G_2(s)](1 + K_1s) = 0,$$

ou

$$s(s + 20)(s + 1000) + 5000K_a(1 + K_1s) = 0.$$

Tem-se, por conseguinte,

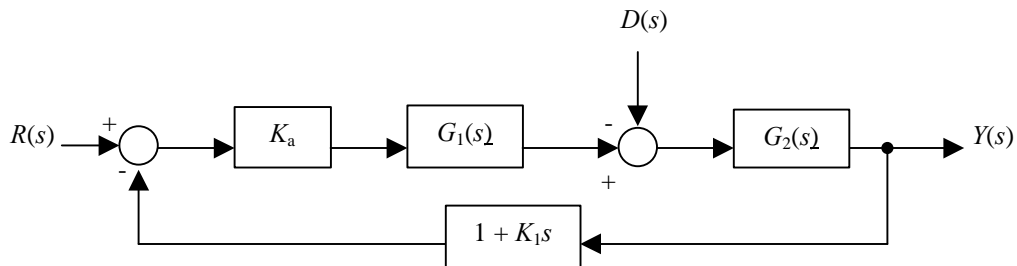
$$s^3 + 1020s^2 + [20000 + 5000K_a K_1]s + 5000K_a = 0.$$

Então, o arranjo tabular de Routh é

$$\begin{array}{c|cc}
 s^3 & 1 & (20000 + 5000K_aK_1) \\
 s^2 & 1020 & 5000K_a \\
 s^1 & b_1 & \\
 s^0 & 5000K_a & 
 \end{array}$$

onde

$$b_1 = \frac{(20000 + 5000K_aK_1) - 5000K_a}{1020}$$



**Fig. 6.2** Sistema equivalente com a chave da retroação de velocidade fechada.

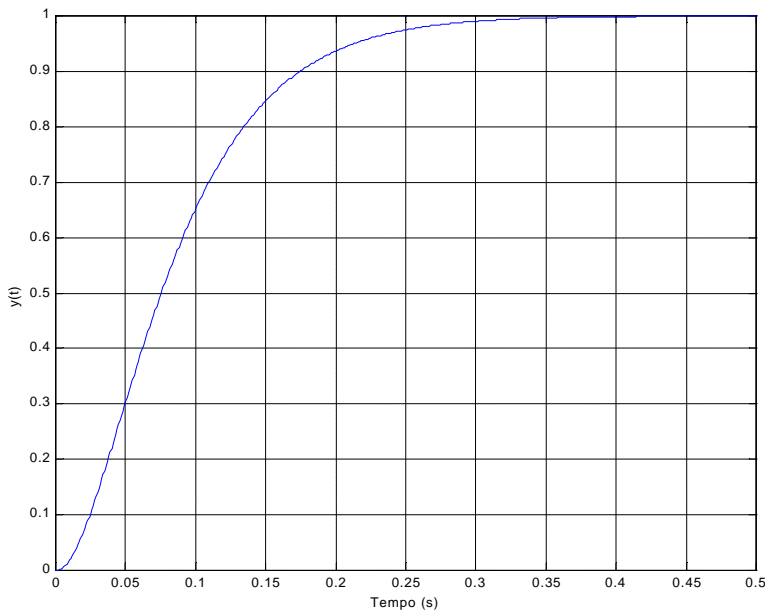
```

Ka=100;
K1=0.05;
ng1=[5000];dg1=[1 1000];
ng2=[1];dg2=[1 20 0];
nc=[K1 1];dc=[0 1];
[n,d]=series(Ka*ng1,dg1,ng2,dg2);
[num,den]=feedback(n,d,nc,dc);
t=[0:0.001:0.5];
y=step(num,den,t);
plot(t,y),grid
xlabel('Tempo (s)'),ylabel('y(t)')

```

Seleciona o ganho de retroação de velocidade  $k_1$  e o ganho do amplificador  $K_a$

(a)



(b)

**Fig. 6.3** Resposta do sistema com retroação de velocidade. (a) Script MATLAB. (b) Resposta com  $K_a = 100$  e  $K_1 = 0,05$ .

Para garantir estabilidade, é necessário seleccionar o par  $(K_a, K_1)$  tal que  $b_1 > 0$ , onde  $K_a > 0$ . Quando  $K_1 = 0,05$  e  $K_a = 100$ , pode-se determinar a resposta usando o MATLAB como está mostrado na Fig. 6.3. O tempo de assentamento (critério de 2%) é aproximadamente de 260 ms e a ultrapassagem percentual é igual a zero. O desempenho do sistema está resumido na TABELA 6.1. As especificações de desempenho estão praticamente satisfeitas e algumas interações sobre  $K_1$  são necessárias para se obter o tempo de assentamento desejado de 250 ms.

**TABELA 6.1 Desempenho do Sistema Acionador de Disco Comparado com as Especificações**

Medida de Desempenho	Valor Desejado	Resposta Real
Ultrapassagem percentual	Inferior a 5%	0%
Tempo de assentamento	Inferior a 250 ms	260 ms
Resposta máxima a uma perturbação unitária	$K_a$	10 – 1000



## 7 - MÉTODOS DE RESPOSTA DE FREQUÊNCIA USANDO MATLAB

Esta seção começa com uma apresentação dos diagramas de Bode e em seguida discute a conexão entre a resposta de frequência às especificações no domínio do tempo. A seção termina com um exemplo ilustrativo de projeto de um sistema de controle no domínio da frequência.

As funções do MATLAB envolvidas são `bode` e `logspace`. A função `bode` é usada para gerar os diagramas de Bode e a função `logspace` produz um vetor com valores de frequência espaçados em escala logarítmica utilizado pela função `bode`.

**Diagramas de Bode.** Seja a função de transferência

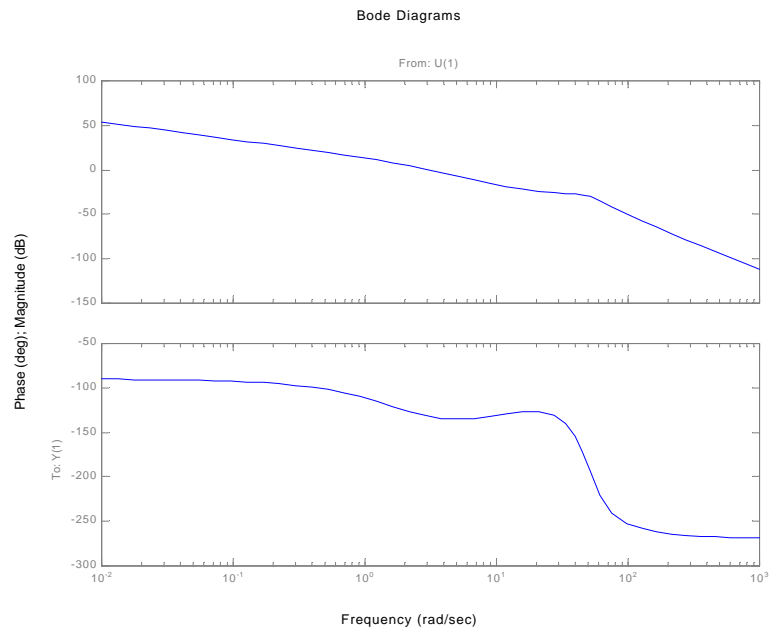
$$G(s) = \frac{5(1 + 0,1s)}{s(1 + 0,5s)(1 + (0,6/50)s + (1/50^2)s^2)} \quad (7.1)$$

Os diagramas de Bode correspondentes à Eq. (7.1) estão mostrados na Fig., 7.1. Os diagramas consistem no ganho logarítmico em dB *versus*  $\omega$  em um gráfico na fase  $f(\omega)$  *versus*  $\omega$  em um segundo gráfico. Como nos gráficos do lugar das raízes, será tentador contar exclusivamente com o MATLAB para obter diagramas de Bode. O MATLAB deve ser tratado como uma das ferramentas de um conjunto que pode ser usada para projetar e analisar sistemas de controle. É essencial desenvolver a capacidade de obter manualmente diagramas de Bode aproximados. Não existe substituto para uma compreensão clara da teoria subjacente.

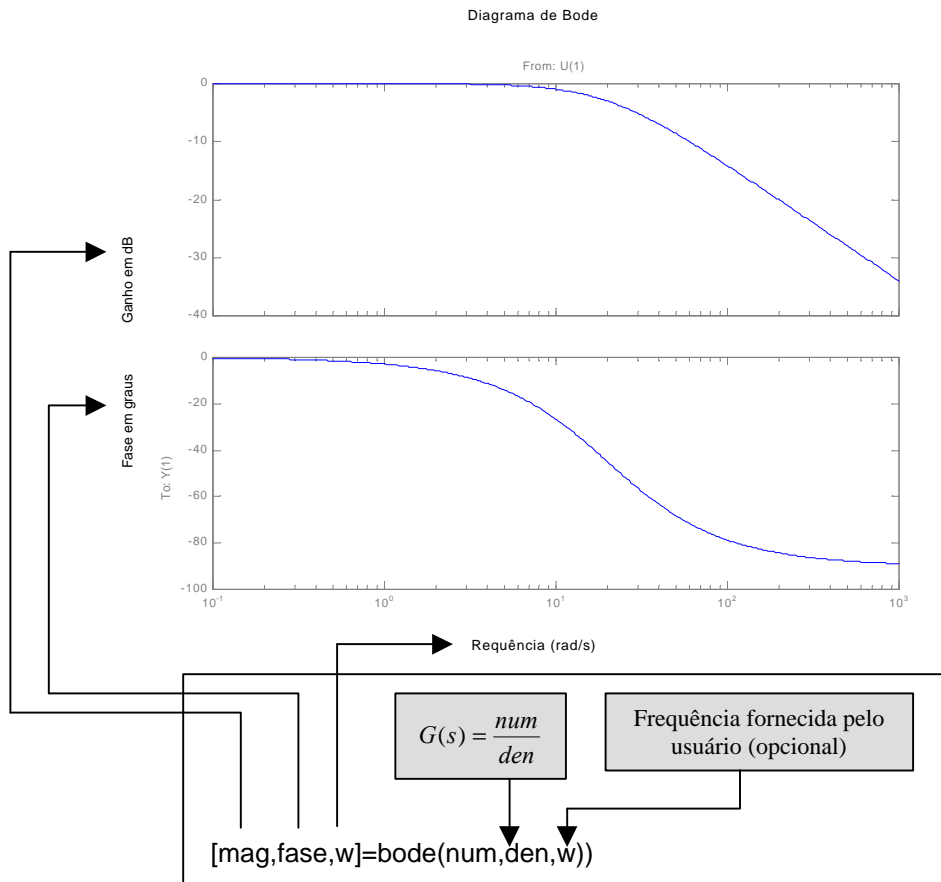
Na Fig. 7.2 estão mostrados os diagramas de Bode obtidos com a função `bode`. Os diagramas de Bode são gerados automaticamente ao se chamar a função `bode` sem agrupamentos na esquerda. Em caso contrário, as características de magnitude e de fase são colocadas na área de trabalho por meio de variáveis *mag* e *fase*. Os diagramas de Bode são obtidos com a função `plot` usando *mag*, *fase* e  $\omega$ . O vetor  $\omega$  contém os valores de frequência em rad/s para os quais os diagramas de Bode serão calculados. Se  $\omega$  não for especificado, o MATLAB escolherá automaticamente os valores de frequência colocando mais pontos nas regiões onde a resposta de frequência estiver mudando rapidamente. Se for desejado especificar explicitamente as frequências, é conveniente gerar o vetor  $\omega$  usando a função `logspace`. A função `logspace` está mostrada na Fig. 7.3.

Os diagramas de Bode da Fig. 7.1 foram gerados usando o script mostrado na Fig. 7.3. A função `bode` selecionou automaticamente a faixa de frequências de  $\omega = 0,1$  a 1000 rad/s. Esta faixa é possível de ser selecionada pelo usuário utilizando a função `logspace`. A função `bode` pode ser utilizada com modelos em variáveis de estado, como está mostrado na Fig. 7.4.

Tendo em mente o objetivo de projetar sistemas de controle que satisfaçam certas especificações de desempenho dadas no domínio do tempo, deve-se estabelecer uma conexão entre a resposta de frequência e a resposta transitória de um sistema em função do tempo. A relação entre as especificações dadas no domínio do tempo e as dadas no domínio da frequência depende da aproximação do sistema por um sistema de Segunda ordem cujos pólos sejam as **raízes dominantes** do sistema.



**Fig. 7.1** Diagramas de Bode associados à Eq. (7.1).



**Fig. 7.2** Uso da função **bode**, dada  $G(s)$ .

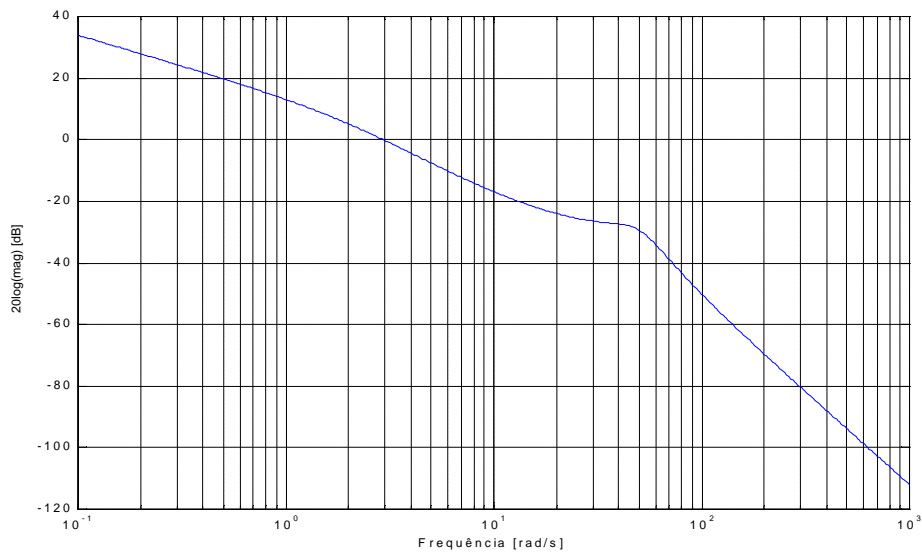
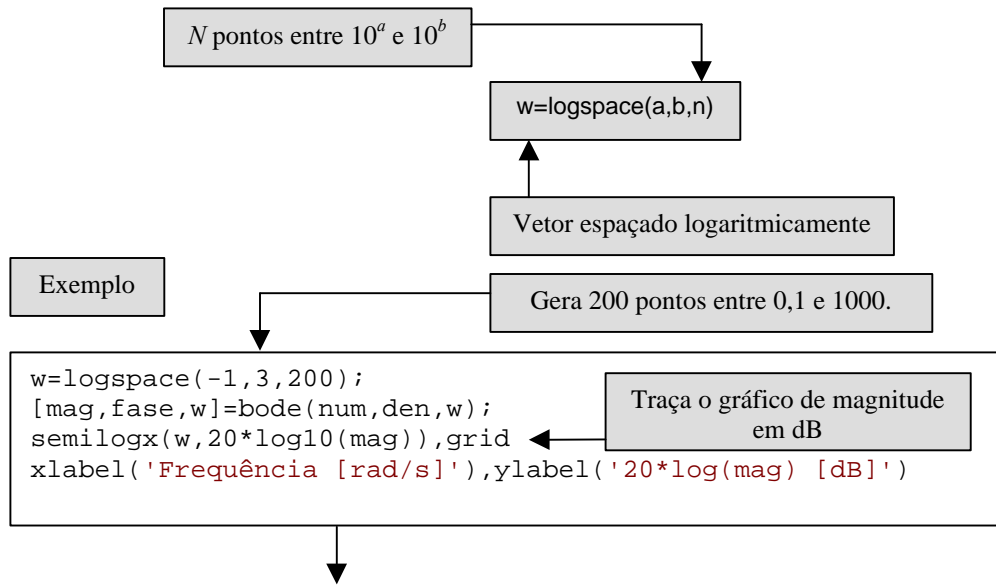


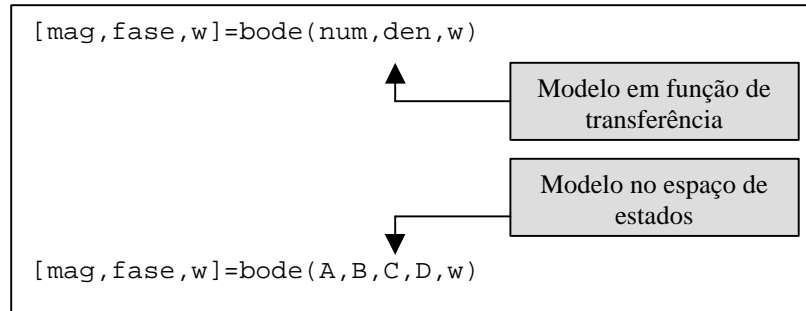
Fig. 7.3 A função `logspace`.

```
%Script dos gráficos de Bode relativos à Fig. 7.1

num=5*[0.1 1];
f1=[1 0]; f2=[0.5 1]; f3=[1/2500 .6/50 1];
den=conv(f1,conv(f2,f3));

bode(num,den)
```

Fig. 7.4 Script para os diagramas de bode da Fig. 7.1.

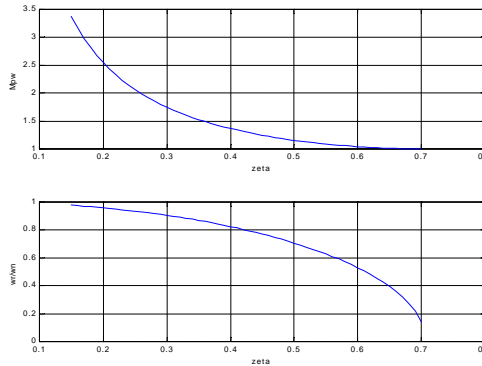


**Fig. 7.5** A função **bode** com um modelo em variáveis de estado.

Seja o sistema de Segunda ordem mostrado na Fig. 5.1. A função de transferência a malha fechada é

$$t(s) = \frac{W_n^2}{s^2 + 2z W_n s + W_n^2} \quad (7.2)$$

A característica de magnitude dos diagramas de Bode associada a função de transferência a malha fechada da Eq. (7.1) está mostrada na Fig. 7.6. A relação entre a frequência de ressonância,  $\omega_r$ , o valor máximo da resposta de frequência,  $M_{p\omega}$  o coeficiente de amortecimento,  $z$ , e a frequência natural,  $W_n$ , é mostrada na Fig. 7.6. A informação da Fig. 7.6 será bastante útil ao se projetar sistemas de controle no domínio da frequência que satisfaçam especificações no domínio do tempo.



(a)

relation.m

```

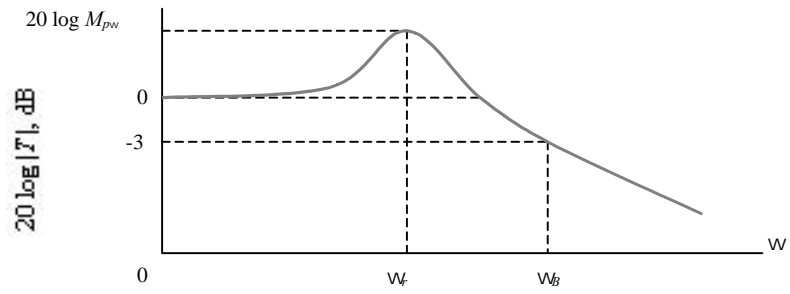
zeta=[0.15:0.01:0.7];
wr_over_wn=sqrt(ones(1,length(zeta))-2*zeta.^2);
Mp=(2*zeta.*sqrt(ones(1,length(zeta))-zeta.^2)).^(-1);

subplot(2,1,1),plot(zeta,Mp),grid
xlabel('zeta'),ylabel('Mp')
subplot(2,1,2),plot(zeta,wr_over_wn),grid
xlabel('zeta'),ylabel('wr/wn')

```

(b)

**Fig. 7.5** (a) Relação entre ( $M_{p\omega}$ ,  $\omega_r$ ) e ( $z$ ,  $W_n$ ) para um sistema de segunda ordem. (b) Script em MATLAB.



**Fig. 7.6** Característica de magnitude do sistema de segunda ordem.

## 8 - MÉTODOS DE IDENTIFICAÇÃO DETERMINÍSTICOS

### 8.1 - Introdução

Métodos determinísticos são aqueles que não dão nenhum tratamento especial ao ruído presente nos dados, ainda que se aceite o fato de que os dados estejam contaminados. Consequentemente, os métodos determinísticos são pouco imunes a ruído e só apresentam bons resultados quando a relação sinal/ruído é suficientemente alta. Métodos que levam em consideração o ruído presente nos dados, por outro lado, são denominados estocásticos e não serão estudados neste capítulo.

Métodos determinísticos processam os dados como se não tivessem ruído nenhum, como se os dados fossem puramente determinísticos. Por outro lado, os métodos estocásticos utilizam recursos adequados para levar em consideração o ruído e, dessa forma, reduzir os seus efeitos sobre o modelo identificado.

Outras duas categorias são a dos métodos paramétricos e não paramétricos. Essa divisão, na realidade, diz respeito à *forma do modelo* identificado e não necessariamente ao algoritmo ou método usado. Apesar disso, é comum falar de *métodos* paramétricos e não paramétricos nos casos em que os *modelos* resultantes são paramétricos e não paramétricos. Consequentemente, métodos de identificação não paramétricos resultam em respostas ao impulso e respostas em frequência, representações gráficas que não têm nenhum parâmetro explícito.

Portanto, a denominação determinístico-estocástico diz respeito ao *algoritmo*, ou seja, à forma com que o ruído é tratado no algoritmo de identificação. Por outro lado, a denominação paramétrico e não paramétrico se refere ao tipo de *modelo* resultante. Assim sendo, existem métodos determinísticos tanto para identificação paramétrica quanto para não paramétrica. Por outro lado, é possível obter um modelo não paramétrico (por exemplo, a resposta em frequência do processo) usando tanto métodos determinísticos quanto estocásticos. A gama de combinações entre métodos é grande, e é possível categorizar as técnicas de várias formas diferentes. Optou-se por abordar os métodos determinísticos no presente capítulo.

### 8.2 - Alguns Casos Simples

Nesta seção serão mencionados dois casos simples para a identificação determinística de sistemas. Os casos abordados são: sistemas de primeira ordem e sistemas de segunda ordem muito pouco amortecidos. A identificação de sistemas de segunda ordem amortecidos e superamortecidos será tratada na próxima seção.

Em todos os casos abordados nesta seção supõe-se que os sistemas não tenham atraso puro de tempo. Se este não for o caso, então o atraso puro deverá ser estimado separadamente e adicionado ao modelo.

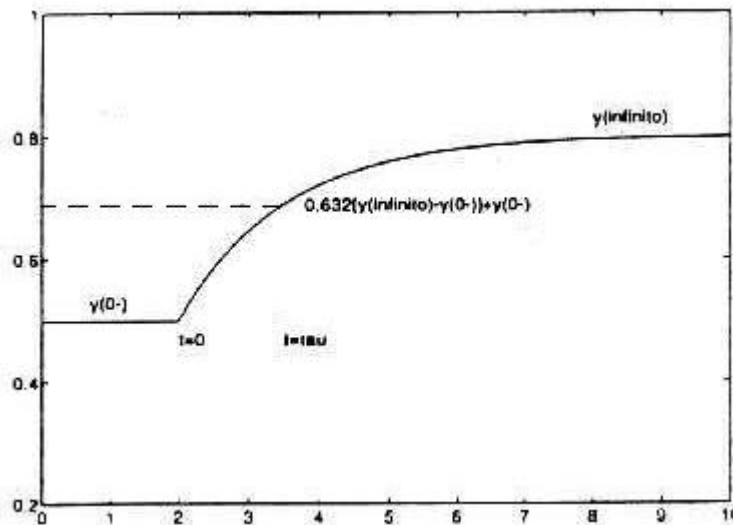
#### 8.2.1 - Sistemas de primeira ordem

Se o sistema a ser identificado apresentar uma resposta típica de sistemas de

primeira ordem, então um possível modelo para esse sistema é da forma

$$\frac{Y(s)}{U(s)} = \frac{K}{t s + 1}, \quad (8.1)$$

sendo que  $K$  é o ganho do sistema e  $t$  é a sua constante de tempo. Se a resposta a um degrau de amplitude  $A$  foi registrada e se o nível de ruído nos dados for suficientemente baixo, é possível determinar com relativa facilidade os parâmetros  $K$  e  $t$ . O ganho é dado por  $K = (y(\infty) - y(0^-))/A$ , sendo que  $y(\infty)$  é o valor em regime permanente de  $y(t)$ . A constante de tempo, por outro lado, pode ser determinada a partir da resposta ao degrau. No período de tempo equivalente a uma constante de tempo,  $t$ ,  $y(t)$  sai do valor que tinha antes da aplicação do degrau e chega a 63,2% da variação total devida ao degrau, ou seja,  $y(t) = 0,632(y(\infty) - y(0^-)) + y(0^-)$ , onde supõe-se que o degrau foi aplicado em  $t = 0$ . Esses conceitos estão ilustrados na Fig. (8.1).



**Fig. 8.1** Resposta ao degrau de um sistema de primeira ordem com constante de tempo  $t$ . O degrau foi aplicado no instante  $t = 0$ .

Se o sistema tiver atraso puro de tempo, o desenvolvimento acima ainda pode ser seguido, mas, nesse caso,  $t = 0$  será o instante em que o sistema começa a responder à entrada. O atraso puro de tempo será igual ao período decorrido entre a aplicação do degrau e o início da resposta do sistema à entrada. No próximo capítulo será visto como estimar o atraso puro de tempo de forma menos restritiva.

### 8.2.2 - Sistemas de segunda ordem pouco amortecidos

Se o sistema a ser identificado apresentar uma resposta típica de sistemas de segunda ordem subamortecido, um possível modelo é a função de transferência do tipo

$$\frac{Y(s)}{U(s)} = \frac{K\omega_n^2}{s^2 + 2Z\omega_n s + \omega_n^2}, \quad (8.2)$$

sendo que  $\omega_n$  é a frequência natural do sistema e  $\zeta$  é o seu quociente de amortecimento.

A seguir, descreve-se um método para estimar  $\omega_n$  e  $\zeta$  para uma função de transferência do tipo (8.2), para o caso do sistema ser pouco amortecido ( $\zeta^2 \ll 1$ ), a partir da sua resposta ao degrau (Phillips & Parr, 1991). A resposta temporal de (8.2) ao degrau unitário é

$$y(t) = 1 - \frac{1}{b} e^{-z\omega_n t} \text{sen}(b\omega_n t + f), \quad (8.3)$$

sendo que  $b = \sqrt{1-z^2}$  e  $f = \tan^{-1} b/z$ . A frequência amortecida é  $\omega_h = \omega_n b$  e o número de oscilações em um segundo, ou seja, a frequência do sinal amortecido, em hertz, é  $\omega_h/2\pi$ . Conforme pode ser visto na Eq. (8.3), o envelope ou a componente modulante do sinal amortecido tem uma constante de tempo igual a  $t = 1/z\omega_n$ s. Portanto, o número de ciclos do sinal amortecido no período de uma constante de tempo é

$$\frac{\omega/2\pi}{z\omega_n} = \frac{\omega_n b}{2\pi} \frac{1}{z\omega_n} = \frac{\sqrt{1-z^2}}{2\pi z}. \quad (8.4)$$

Supondo agora que, após  $N$  constantes de tempo, as oscilações não são mais perceptíveis, então o número de ciclos visíveis na resposta do sistema a uma entrada em degrau é dado por

$$\frac{\sqrt{1-z^2}}{2\pi z}. \quad (8.5)$$

Sabe-se que, para funções de transferência do tipo (8.2), a resposta ao degrau se mantém dentro da faixa de  $\pm 2\%$  do valor em regime permanente após quatro constantes de tempo, ou seja, nesse caso  $N = 4$ . Assumindo que o sistema é pouco amortecido de forma a se ter  $\zeta^2 \ll 1$ , o número de ciclos visíveis pode ser aproximado por

$$\text{Número de ciclos visíveis} = \frac{0,6}{z}. \quad (8.6)$$

A exatidão da equação acima, evidentemente, depende de quão boa é a aproximação  $\sqrt{1-z^2} \approx 1$ . A frequência  $\omega$  pode ser estimada diretamente do gráfico e da escala de tempo. Como foi assumido que  $\sqrt{1-z^2} \approx 1$ , então  $\omega_h \approx \omega$ . Finalmente, o ganho  $K$  é obtido a partir do gráfico da forma usual, isto é,  $K = \Delta y/\Delta u$  determinado em regime permanente.



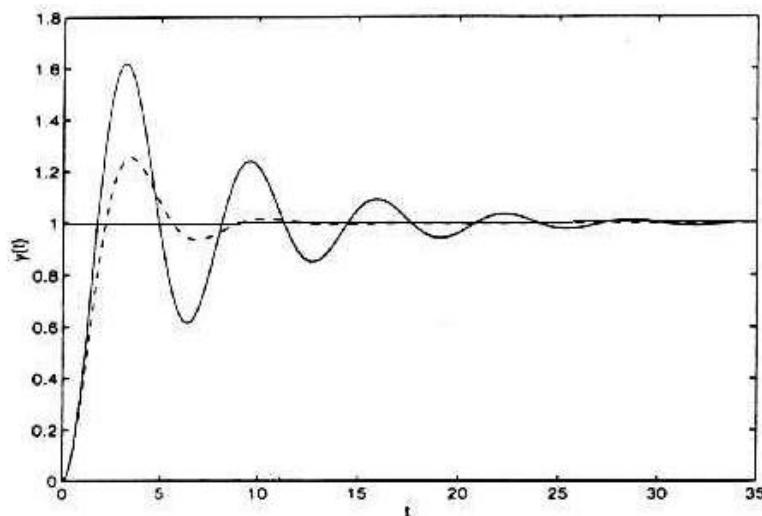
### Exemplo 8.2.1 Estimação de $\zeta$ e $\omega_n$ .

A Fig. 8.2 mostra as respostas ao degrau unitário de dois sistemas de segunda ordem do tipo (8.2) com  $K = 1$ ,  $\omega_n = 1$  e com os seguintes valores distintos do quociente de amortecimento  $z = 0,15$  e  $z = 0,4$ .

Aplicando o procedimento acima e usando (8.6), para a resposta menos amortecida chega-se, sem maiores dificuldades, a

$$z \approx \frac{0,6}{4,5} = 0,13.$$

É interessante notar que se houvesse um pouco de ruído nos dados, seria natural contar apenas quatro ciclos completos, o que resultaria em 0,15.



**Fig. 8.2** Respostas ao degrau unitário da função de transferência (8.2) com  $K=1$ ,  $\omega_n=1$  e ( )  $z=0,15$  e (--)  $z=0,4$ .

Por outro lado, observando a figura, nota-se que aproximadamente quatro ciclos ocorrem nos primeiros 25 segundos. O “período médio”, então, pode ser estimado como  $T = 25/4 = 6,25$ , e

$$\omega_n = 2\pi/T \approx 1,0,$$

conforme esperado. Por outro lado, é mais difícil contar ciclos para a resposta mais amortecida. Mesmo na presença de pouco ruído apenas um ciclo seria visível o que resultaria em  $z = 0,6$  que é 50% maior que o valor usado.

### 8.3 - O Método de Sundaesan

Nesta seção apresentaremos uma técnica de identificação conhecida como o *método de Sundaesan* (Deshpande & Ash, 1981). Este método pressupõe que o sistema em questão

pode ser aproximado satisfatoriamente por funções de transferência de segunda ordem do tipo

$$G(s) = \frac{e^{-t_d s}}{(t_1 s + 1)(t_2 s + 1)} \quad (8.7)$$

ou

$$G(s) = \frac{e^{-t_d s} W_n^2}{s^2 + 2zW_n s + W_n^2}, \quad (8.8)$$

sendo que  $t_d$  é o atraso puro de tempo (ou tempo morto ou, ainda, atraso de transporte),  $t_1$  e  $t_2$  são constantes de tempo fixas,  $z$  é o quociente de amortecimento e  $W_n$  é a frequência natural do sistema. Modelos do tipo (8.7) são usados para descrever sistemas dinâmicos lineares de segunda ordem e sobreamortecidos, ou seja, sistemas que não oscilam quando excitados por um degrau. Por outro lado, modelos do tipo (8.8) representam sistemas de segunda ordem subamortecidos que, por ter pólos complexos conjugados, normalmente apresentam oscilações nas suas respostas ao degrau e ao impulso.

O objetivo do método é determinar os parâmetros  $t_d$ ,  $t_1$ ,  $t_2$  ou  $t_d$ ,  $W_n$ ,  $z$  da respectiva função de transferência, a partir de uma resposta ao degrau.

Deve ser notado que as funções de transferência acima têm ganho unitário. Na maioria dos casos práticos será necessário determinar o ganho do processo (que provavelmente será diferente de um) e acrescentá-lo ao modelo. O ganho é a razão entre a variação do sinal de saída pela amplitude do degrau aplicado, conforme visto anteriormente. A seguir descreve-se o procedimento para determinação dos parâmetros.

### 8.3.1 - O caso sobreamortecido

$$G(s) = \frac{e^{-t_d s}}{(t_1 s + 1)(t_2 s + 1)}$$

Uma resposta ao degrau típica de sistemas sobreamortecidos é mostrada na Fig. 8.3.

O primeiro momento da função resposta  $y(t)$  é a área hachurada na Fig. 8.3 e é dado por

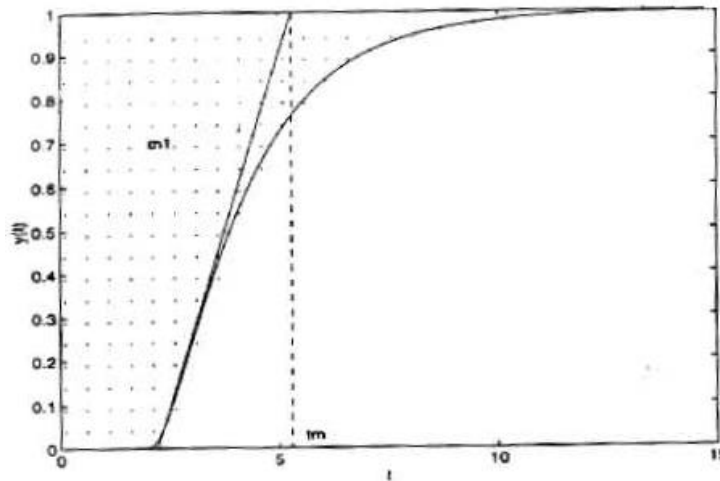
$$m_1 = \int_0^{\infty} (1 - y(t)) dt. \quad (8.9)$$

A função de transferência  $G(s)$  e  $m_1$  estão relacionados pela seguinte definição

$$m_1 = -\left. \frac{dG(s)}{ds} \right|_{s=0} = t_d + t_1 + t_2 . \quad (8.10)$$

A resposta de (8.7) ao degrau é

$$y(t) = \left[ 1 - \frac{t_1}{t_1 - t_2} e^{-\frac{t-t_d}{t_1}} + \frac{t_2}{t_1 - t_2} e^{-\frac{t-t_d}{t_2}} \right] u(t - t_d) , \quad (8.11)$$



**Fig. 8.3** Resposta ao degrau típica de um sistema sobreamortecido com tempo morto.

sendo que  $u(t - t_d)$  é um degrau unitário começando no instante  $t_d$ . Igualando-se a derivada de segunda ordem de (8.11) a zero, obtém-se o ponto de inflexão

$$t_i = t_d + a \ln h , \quad (8.12)$$

sendo que  $h = t_1/t_2$  e  $a = t_1 t_2 / (t_1 - t_2)$ . A inclinação da reta tangente no ponto de inflexão é

$$M_i = \frac{h^{1/(1-h)}}{a(h-1)} . \quad (8.13)$$

Esta reta tangente intercepta o valor final de  $y(t)$  num ponto correspondente ao instante  $t_m$  dado por

$$t_m = t_d + a \left[ \ln h + \frac{h^2 - 1}{h} \right] . \quad (8.14)$$

Combinando as equações (8.10), (8.13) e (8.14) obtém-se

$$M_i(t_m - m_1) = \frac{h^{1/(1-h)}}{h-1} \ln h. \quad (8.15)$$

Note que se nesta equação  $h$  for substituído por  $1/h$ , a equação não se altera. Além disso, ela pode ser reescrita da seguinte forma

$$l = ce^{-c}, \quad (8.16)$$

sendo que  $l = (t_m - m_1)M_i$  e

$$c = \frac{\ln h}{h-1}. \quad (8.16)$$

O máximo valor de  $l$  é  $e^{-1}$  e ocorre quando o sistema é criticamente amortecido ( $h = 1$ ). O outro extremo é quando  $h = 0$ , e neste caso o modelo se reduz a um de primeira ordem. A Fig. 8.4 mostra a solução da Eq. (8.16).

Portanto, o procedimento pode ser resumido da seguinte forma:

1. Determinar o ganho em regime permanente dividindo a variação do sinal de saída pela amplitude do degrau aplicado à entrada;
2. Determinar a área sombreada na Fig. 8.3. Este valor é  $m_1$ ;
3. Determinar a inclinação da tangente no ponto de inflexão de  $y(t)$ . Este valor é  $M_i$ ;
4. Determinar  $t_m$  que é a interseção da tangente com o valor em regime permanente de  $y(t)$ ;
5. Determinar  $l$  a partir da Eq. (8.17);
6. Determinar  $h$  a partir do gráfico da Fig. 8.4, que corresponde à solução da Eq. (8.16) substituindo-se  $c$  pela Eq. (8.17);
7. Finalmente, determinar  $t_1$ ,  $t_2$  e  $t_d$  como se segue

$$\begin{aligned} t_1 &= \frac{h^{\frac{h}{1-h}}}{M_i}, \\ t_2 &= \frac{h^{\frac{h}{1-h}}}{M_i}, \\ t_d &= m_1 - t_1 - t_2. \end{aligned} \quad (8.18)$$

A Fig. 8.5 (a) mostra um exemplo de método sobreamortecido e (b) os gráficos gerados.

### Exemplo 8.3.1 Método Sobreamortecido

```
%-----  
% Pontifícia Universidade Católica de Minas Gerais  
% Núcleo de Modelagem Análise e Simulação  
% Disciplina: Análise de Sistemas III  
% Método de Sundaresan  
% Sistema Sobreamortecido  
%-----  
  
clear all  
close all  
  
% Função de transferência Original  
n=1;  
d=[1 3.5 3.5 1];  
  
% Resposta ao Degrau  
[y,x,t]=step(n,d);  
  
m1=0; %condição inicial de integração  
deltat=t(2)-t(1); %variação do vetor tempo  
  
% Observação:  
% Se a subtração for realizada fora do loop, o vetor  
% de tempo deve ser normalizado para que a área seja = 1  
  
% Cálculo da área m1=1-y(t)  
for i=1:(size(y,1)),  
    m1=m1+(1-y(i,1))*deltat;  
end  
m1  
  
% Cálculo do ponto de inflexão  
for i=2:length(y)-1,  
    dif(i)=y(i+1)-y(i);  
    if dif(i)>dif(i-1),  
        Y0=y(i);  
        T0=t(i);  
        deltay=dif(i);  
    end  
end  
  
% Considerando o ganho do sistema = 1  
Y=1;  
  
% Inclinação da reta tangente ao ponto de inflexão  
Mi=(deltay/deltat);  
  
% Projeção no eixo de tempo do ponto no qual a reta tangente  
% intercepta uma linha correspondente ao valor do ganho  
tm=((Y-Y0)/Mi)+T0;  
  
% Cálculo do parâmetro lambda  
lambda=(tm-m1)*Mi;
```

```

% O parâmetro eta é estimado de acordo com o gráfico
resultante
%da resolução da equação:
%
% lambda=chi*exp(-chi),
%
%onde lambda=chi*exp(-chi) e chi=ln(eta)/(eta-1)
eta=[0.01:0.01:1];

%chi=log(eta)/(eta-1)
a=log(eta);
chi=a./(eta-1); %divisão ponto a ponto

% Observação: Foi acrescentado uma constante de valor insigni-
%ficante eps para evitar a divisão por zero e posteriormente
%retirada devido a uma distorção causada no resultado

lambda=chi.*exp(-chi); %multiplicação ponto a ponto
figure(1);
plot(lambda,eta);
title('Gráfico eta x lambda');
xlabel('lambda');ylabel('eta');grid;

% Parâmetro eta estimado "etaest"
etaest=0.55;

% Cálculo das constantes de tempo e do atraso teta
taul=(etaest^(etaest/(1-etaest)))/Mi;
tau2=(etaest^(1/(1-etaest)))/Mi;
teta=m1-taul-tau2;

% Modelo obtido
n1=[1];d1=[taul 1];
n2=[1];d2=[tau2 1];
[num,den]=series(n1,d1,n2,d2);

% Aproximação de Padé (para representar o atraso de tempo)
g=3; %grau do polinômio
[np,dp]=pade(teta,g);

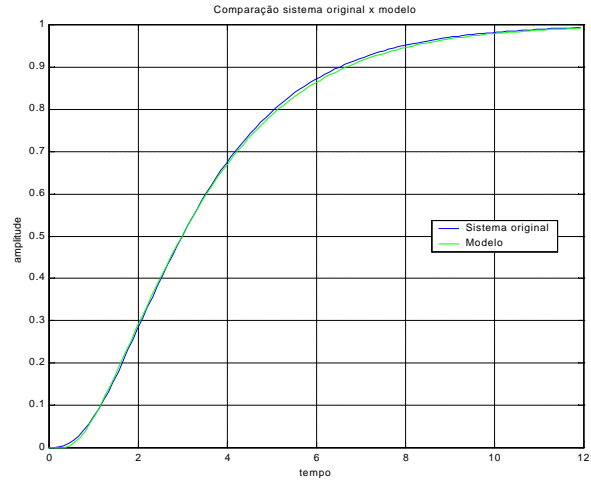
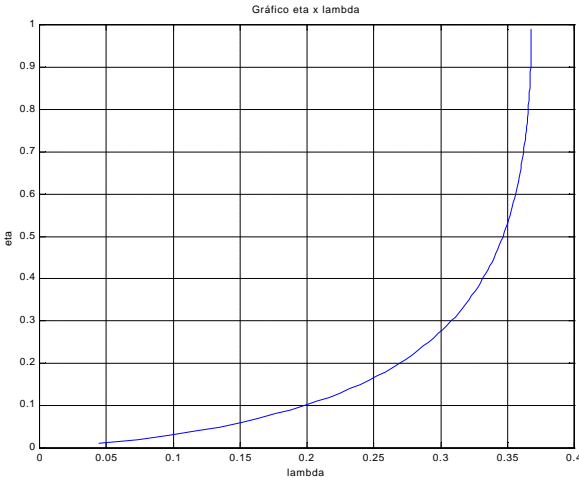
% Modelo obtido + Aproximação de Padé
[n,d]=series(num,den,np,dp);

% Resposta ao degrau do sistema original
figure(2);
plot(t,y);
hold;

% Resposta ao degrau do modelo obtido
[ys,x,t]=step(n,d);
plot(t,ys,'g');
title('Comparação sistema original x modelo');
legend('Sistema original','Modelo',0);
xlabel('tempo');ylabel('amplitude');grid;

```

(a)



(b)

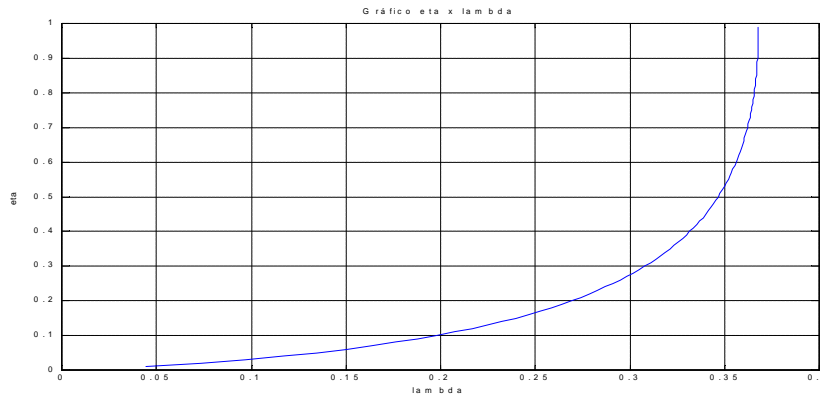
**Fig. 8.5** Exemplo do método sobre-amortecido.

### 8.3.2 - O caso subamortecido

$$G(s) = \frac{e^{-t_d s} \omega_n^2}{s^2 + 2z\omega_n s + \omega_n^2}$$

A Fig. 8.6 mostra uma resposta ao degrau típica de sistemas de segunda ordem subamortecidos. A equação que descreve esta resposta temporal é

$$y(t) = u(t - t_d) \left\{ 1 - z\omega_n e^{t-t_d} \left[ \frac{z}{\sqrt{1-z^2}} \operatorname{sen}(\omega_n(t-t_d)\sqrt{1-z^2}) + \cos(\omega_n(t-t_d)\sqrt{1-z^2}) \right] \right\}. \quad (8.19)$$



**Fig. 8.4** Solução da Eq. (8.16) substituindo-se  $\chi$  pela Eq. (8.17).

Seguindo um procedimento semelhante ao anterior, as seguintes equações podem ser obtidas

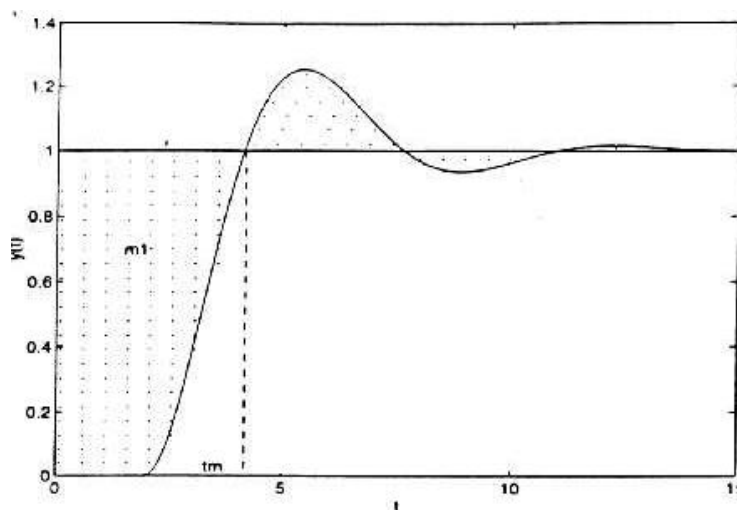
$$l = (t_m - m_1)M_i = \frac{\cos^{-1} Z}{\sqrt{1-Z^2}} e^{\frac{-Z \cos^{-1} Z}{\sqrt{1-Z^2}}}, \quad (8.20)$$

$$W_n = \frac{\cos^{-1} Z}{\sqrt{1-Z^2}} \frac{1}{t_m - m_1}, \quad (8.21)$$

$$t_d = m_1 - \frac{2Z}{W_n}. \quad (8.22)$$

As equações acima são utilizadas na determinação dos parâmetros de (8.8). Primeiramente deve-se determinar  $m_1$ . Para o caso subamortecido é importante notar que, no cálculo de  $m_1$ , as áreas acima de  $y(t) = 1$  são consideradas negativas e as áreas abaixo, positivas. Portanto, os valores de  $m_1$  e de  $t_m$  podem ser obtidos a partir do gráfico ou dos dados medidos.

A Eq. (8.20) mostra a relação entre o quociente de amortecimento  $Z$  e o parâmetro  $l$ . Note que  $l$  é determinado a partir do conhecimento de  $m_1$  e  $t_m$ , obtidos a partir do gráfico da resposta em degrau e a partir de  $M_i$  que é a inclinação da curva no ponto de inflexão. Como seria muito difícil obter analiticamente  $Z$  a partir de  $l$  na Eq. (8.20), isto pode ser feito graficamente através da Fig. 8.7, que mostra a relação entre tais parâmetros.



**Fig. 8.6** Resposta ao degrau típica de um sistema subamortecido com tempo morto.

Finalmente, as equações (8.21) e (8.22) podem ser utilizadas para determinar os demais parâmetros da função de transferência. Portanto, o procedimento completo para o caso subamortecido é



1. Determinar o ganho em regime permanente dividindo a variação do sinal de saída pela amplitude do degrau aplicado à entrada;
2. Determinar a área sombreada na Fig. 8.5, a área acima do valor de  $y(\infty)$  deve ser subtraída. Este valor é  $m_1$ ;
3. Determinar a inclinação da tangente no ponto de inflexão de  $y(t)$ . Este valor é  $M_i$ ;
4. Determinar  $t_m$  que é a interseção da tangente com o valor em regime permanente de  $y(t)$ ;
5. Determinar  $l$  a partir da Eq. (8.20);
6. Determinar  $z$  a partir do gráfico da Fig. 8.6, que corresponde à solução da Eq. (8.20);
7. Determinar  $w_h$  e  $t_d$  a partir das equações (8.21) e (8.22), respectivamente.

A Fig. 8.7 (a) mostra um exemplo de método subamortecido e (b) o gráfico gerado.

### Exemplo 8.3.2 Método Subamortecido

```
%-----  
% Pontifícia Universidade Católica de Minas Gerais  
% Núcleo de Modelagem Análise e Simulação  
% Disciplina: Análise de Sistemas III  
% Método de Sundaresan  
% Sistema Subamortecido  
%-----  
  
clear all  
close all  
  
% Função de transferência Original  
%  $G(s) = \frac{(s+2)}{(2s^3 + 4s^2 + 4s + 2)} e^{-0.5 \text{ teta}}$   
n1=[1 2];  
d1=[2 4 4 2];  
  
% Aproximação de Padé (Para representar o atraso de tempo)  
T=5;  
n=3;  
[n2,d2]=pade(T,n);  
[num1,den1]=series(n1,d1,n2,d2);  
[y,x,t]=step(num1,den1);  
  
m1=0; %condição inicial de integração  
deltat=t(2)-t(1); %variação do vetor tempo  
  
% Observação: Se a subtração for realizada fora do loop, o  
%vetor de tempo deve ser normalizado para que a área seja = 1  
  
% Cálculo da área m1  
for i=1:(size(y,1)),  
    m1=m1+(1-y(i,1))*deltat;  
end  
m1  
  
% Existem vários pontos de inflexão?  
%for i=1:length(y)  
%    dif(i)=y(i+1)-y(i)  
%    if dif>0  
%        cont=cont+1  
%    end  
%end  
  
% Cálculo do ponto de inflexão  
for i=2:length(y)-1,  
    dif(i)=y(i+1)-y(i);  
    if dif(i)>dif(i-1),  
        Y0=y(i);  
        T0=t(i);  
        deltay=dif(i);  
    end  
end  
  
% Considerando o ganho do sistema = 1  
Y=1;
```

```

% Inclinação da reta tangente ao ponto de inflexão
Mi=(deltay/deltat);

% Projeção no eixo de tempo do ponto no qual a reta tangente
% intercepta uma linha correspondente ao valor do ganho
tm=((Y-Y0)/Mi)+T0;

% Cálculo do parâmetro lambda
lambda=(tm-m1)*Mi;
ksi=0.72
g=acos(ksi);
gg=(1/(tm-m1));
%wn=(g/(1-ksi^2)^0.5)*gg;
wn=(g/sqrt(1-ksi^2))*gg
teta=m1-((2*ksi)/wn)

% Função de transferência obtida
n1=[1];
a=1/wn^2;
b=(2*ksi)/wn;
d1=[a b 1];

% Aproximação de Padé (para representar o atraso no tempo)
np=3;
[n2,d2]=pade(teta,np);

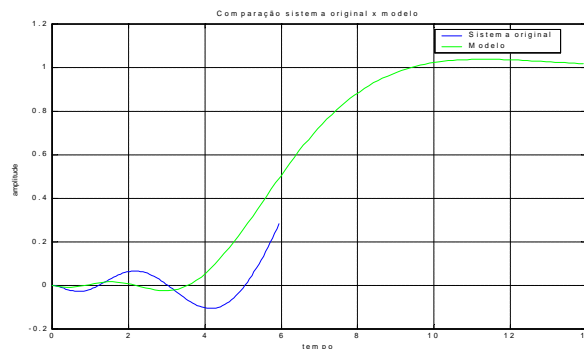
% Modelo obtido + Aproximação de Padé
[num2,den2]=series(n1,d1,n2,d2);

% Resposta ao degrau do sistema original
figure(1);
plot(t,y);
hold on;

% Resposta ao degrau do modelo obtido
[ys,x,t]=step(num2,den2);
plot(t,ys,'g');
title('Comparação sistema original x modelo');
legend('Sistema original','Modelo',0);
xlabel('tempo');ylabel('amplitude');grid;hold off

```

(a)

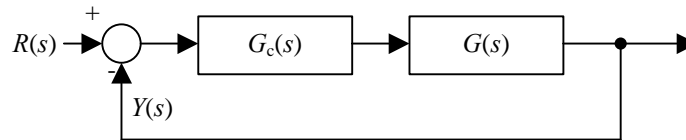


(b)

**Fig. 8.6** Exemplo do método subamortecido.

## 8.4 - Identificação em Malha Fechada

O método descrito aqui foi originalmente proposto por Yawana e Seborg (1982). Considere o seguinte diagrama de blocos mostrado na Fig. 8.7.

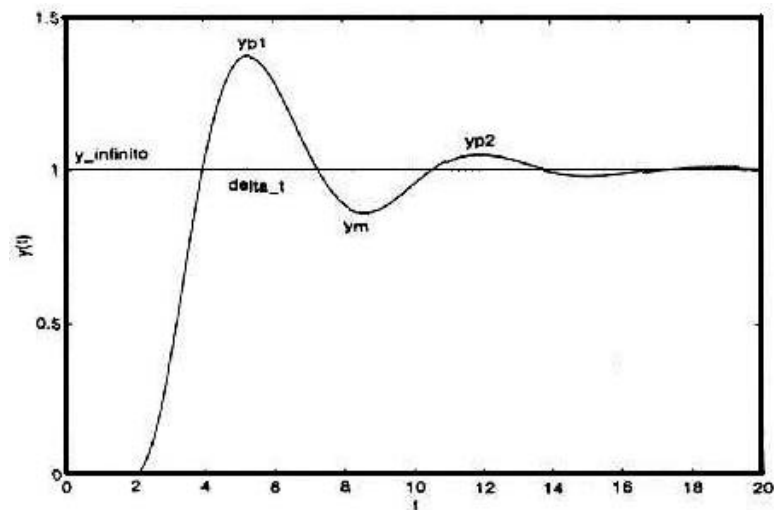


**Fig. 8.7** Diagrama de blocos de um sistema de controle em malha fechada.

No presente método, assume-se que onde considera-se que o controlador é puramente proporcional, ou seja,  $G_c(s) = K_c$  e que a função de transferência do processo tem a seguinte forma

$$G(s) = \frac{Ke^{-t_d s}}{ts + 1}, \quad (8.23)$$

sendo que  $K$ ,  $t_d$  e  $t$  são respectivamente o ganho, o atraso puro de tempo e a constante de tempo do processo. Funções de transferência com esta estrutura são comumente utilizadas para descrever um grande número de processos industriais reais.



**Fig. 8.8** Resposta ao degrau típica necessária para o método proposto por Yawana e Seborg (1982). Nesta figura  $y_{\infty} = y(\infty) = 1$ .

A resposta típica a um degrau aplicado à referência  $r(t)$  é mostrada na Fig. 8.8, que mostra a resposta ao degrau da função de transferência em malha fechada

$$\frac{Y(s)}{R(s)} = \frac{K_f e^{-t_d s}}{t s + [K_f e^{-t_d s} + 1]} \quad (8.24)$$

sendo que  $K_f = K_c K$ . O método descrito a seguir pode ser usado para estimar os parâmetros do processo se a resposta ao degrau em malha fechada for subamortecida.

O uso da expressão (8.24) na estimação de seus parâmetros não é de todo simples, pois o tempo morto é o expoente do número  $e$  e, assim, caracteriza uma não linearidade. A fim de contornar esse problema, pode-se utilizar a seguinte aproximação de Padé.

$$e^{-t_d s} = \frac{1 - 0,5t_d s}{1 + 0,5t_d s}. \quad (8.25)$$

Portanto, substituindo (8.25) em (8.24) obtém-se, após algumas manipulações,

$$\frac{Y(s)}{R(s)} = \frac{\bar{K}(1 - 0,5t_d s)}{t^2 s^2 + 2z t s + 1} \quad (8.26)$$

onde

$$\bar{K} = \frac{K_f}{K_f + 1}, \quad (8.27)$$

$$t = \left[ \frac{t_d t}{2(K_f + 1)} \right]^{0,5}, \quad (8.28)$$

$$z = \frac{t + 0,5t_d(1 - K_f)}{\sqrt{2t_d t(K_f + 1)}}. \quad (8.29)$$

Os parâmetros do processo podem ser obtidos das seguintes expressões

$$K = \frac{y_\infty}{K_c(A - y_\infty)}, \quad (8.30)$$

sendo que  $y_\infty = y(\infty)$  é o valor de  $y(t)$  em regime permanente,  $A$  é a amplitude do degrau aplicado à referência do sistema de controle e

$$t = \frac{\Delta t}{p} \left[ z \sqrt{K_f + 1} + \sqrt{z^2 (K_f + 1) + K_f} \right] \sqrt{(1 - z^2)(K_f + 1)}, \quad (8.31)$$

$$t = \frac{\Delta t \sqrt{(1 - z^2)(K_f + 1)}}{p \left[ z \sqrt{K_f + 1} + \sqrt{z^2 (K_f + 1) + K_f} \right]}, \quad (8.32)$$

sendo que  $z$  pode ser avaliada de duas formas diferentes

$$z = \frac{-\ln \left[ \frac{y_\infty - y_m}{y_{p1} - y_\infty} \right]}{\sqrt{p^2 + \left( \ln \left[ \frac{y_\infty - y_m}{y_{p1} - y_\infty} \right] \right)^2}}, \quad (8.33)$$

ou

$$z = \frac{-\ln \left[ \frac{y_2 - y_\infty}{y_{p1} - y_\infty} \right]}{\sqrt{4p^2 + \left( \ln \left[ \frac{y_2 - y_\infty}{y_{p1} - y_\infty} \right] \right)^2}}. \quad (8.34)$$

Na prática, é comum usar o valor médio de (8.33) e (8.34).

As cinco variáveis independentes que aparecem nas equações (8.30) – (8.34) podem ser obtidas diretamente a partir da resposta ao degrau em malha fechada, como mostrado na Fig. 8.8.

Alguns processos reais levam muito tempo para convergir ao valor final  $y_\infty$ . Em outros casos, não é possível esperar até que o processo atinja  $y_\infty$  por questões operacionais, tais como mudança do ponto de operação, distúrbio de carga etc. Nesses casos é possível estimar o valor em regime permanente como se segue

$$y_\infty = \frac{y_{ps2} y_{p1} - y_m^2}{y_{p2} + y_{p1} - 2y_m}, \quad (8.35)$$

sendo que  $y_{p1}$ ,  $y_{p2}$ , e  $y_m$  são mostrados na Fig. 8.8.

### Exemplo 8.4.1 Identificação em Malha Fechada

```
% Modelo de Identificação em malha fechada

clear all;
close all;

% Resposta ao degrau
nf=[1];
df=[1 40 5 1];
[np,dp]=pade(0.1,3);
[n,d]=series(nf,df,np,dp);
[y,x,t]=step(n,d);
plot(t,y),grid on,hold on;

%Entrada dos parâmetro avaliados no gráfico
yp1=input(' yp1= '); %yp1=1.2596
yp2=input(' yp2= '); %yp2=1.0176
ym=input(' ym= '); %ym=0.9323
deltat=input(' Deltat= '); %deltat=35.2871-13.7111=21.576
kc=input('ganho do controlador kc = ');
A=input('Amplitude do degrau aplicado = '); %A=1

yinf=(yp2*yp1-ym^2)/(yp2+yp1-2*ym),pause;

zeta= -log((yp2-yinf)/(yp1-yinf))/sqrt(4*pi^2+(log((yp2-yinf)/(yp1-yinf)))^2),pause;

k=yinf/(kc*(A-yinf)),pause;

kf=kc*k,pause;

td=(deltat*sqrt((1-zeta^2)*(kf+1)))/(pi*(zeta*sqrt(kf+1)+sqrt(zeta^2*(kf+1)+kf))),pause;

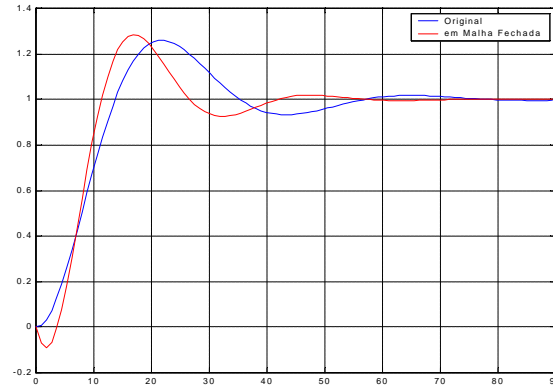
tau=(deltat/pi)*(zeta*sqrt(kf+1)+sqrt(zeta^2*(kf+1)+kf))*sqrt((1-zeta^2)*(kf+1)),pause;

taubarra=((td*tau)/(2*(kf+1)))^0.5,pause;

kbarra=kf/(kf+1),pause;

ni=[-0.5*kbarra*td kbarra];
di= [taubarra^2 2*zeta*taubarra 1];
yi=step(ni,di,t);
plot(t,yi,'r'),legend('Original','em Malha Fechada',1);
```

(a)



(b)

**Fig. 8.6** Exemplo de Identificação em malha fechada.



## 9 - BIBLIOGRAFIA

Cellier, F.E. (1990); "Continuous System Modeling"; Springer-Verlag.

Carvalho, A. L. (1998); "Modelagem de um Reator Anaeróbico de Manta de Lodo Utilizando Redes Neurais Artificiais"; Dissertação de Mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais.

Doebelin, E. O. (1980); "System modeling and response - Theoretical and experimental approaches". John Wiley & Sons, Inc.

Lindskog, P. e Ljung, L. (1997); "Tools for Semi-Physical Modeling "; IEEE.

Ljung, L. (1994); "System Identification in a MIC perspective"; Modeling, Identification and Control, vol.15, n° 3, 153-159.

Dorf, Richard C. e Bishop, Robert H. (1998); "Sistemas de Controle Moderno"; LTC.

Pena, R e Aguirre, L. A. (1997); "Notas de Aula, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais".