

UNIVERSIDADE FEDERAL DE MINAS GERAIS  
CENTRO DE PESQUISA E DESENVOLVIMENTO EM ENGENHARIA ELÉTRICA  
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

# Uso de Técnicas de Detecção de Estrutura na Identificação de Modelos Dinâmicos Contínuos Não Lineares Polinomiais

Ubiratan Santos Freitas

Dissertação submetida à banca examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Minas Gerais como parte dos requisitos necessários à obtenção do grau de Mestre em Engenharia Elétrica.

Orientador: Luís Antônio Aguirre

BELO HORIZONTE, OUTUBRO DE 2001

Não há religião nem ciência acima da Beleza. Eu construiria uma cidade à beira do mar, e numa ilha do porto erigiria uma estátua não à Liberdade, mas à Beleza. Pois foi ao redor da Liberdade que os homens travaram suas batalhas. Por oposição, ante a face da Beleza, todos os homens estendem as mãos uns aos outros como irmãos.

*Gibran Khalil Gibran*

# Resumo

A identificação de sistemas permite obter modelos matemáticos para sistemas reais, a partir de dados coletados.

O presente trabalho trata do desenvolvimento de um algoritmo de identificação de modelos dinâmicos não lineares polinomiais de tempo contínuo. O ponto principal do algoritmo proposto é o emprego de técnicas de detecção de estrutura. O uso de tais técnicas permite identificar modelos mais simples e, ao mesmo tempo, mais adequados para os sistemas em estudo .

Por técnicas de detecção de estrutura entendem-se os algoritmos e processos usados para escolher uma estrutura particular dentro de uma determinada representação. A representação é o tipo de modelo usado na identificação. Por exemplo, pode-se utilizar como representação redes neurais, funções de base radial e funções polinomiais, entre outras. Por outro lado, a estrutura é a configuração particular do modelo, dada uma determinada representação.

No caso de redes neurais, por exemplo, a detecção de estrutura seria responsável por escolher os neurônios e as conexões (sinapses) mais adequadas para um determinado problema. No presente trabalho, as técnicas de detecção de estrutura são utilizadas para escolher quais os termos de uma função polinomial devem ser incluídos no modelo identificado.

As estratégias de detecção de estrutura empregadas neste trabalho são o critério ERR e o uso de informação disponível *a priori* sobre o sistema a ser identificado. Tais técnicas foram utilizadas na identificação de modelos de tempo discreto, até o presente momento. Neste trabalho, elas são aplicadas na identificação de modelos contínuos, aparentemente pela primeira vez.

O algoritmo proposto é descrito detalhadamente e comparado com a alternativa discreta mais semelhante: os modelos NARX polinomiais. O procedimento de identificação é, então, testado com a identificação de sistemas não lineares.

Primeiramente, são usados no teste dois sistemas caóticos: o sistema de Lorenz e o circuito de Chua. Na identificação do primeiro, são usados dados provenientes de simulação enquanto na identificação do último são usados dados reais coletados de uma implementação do circuito.

Num segundo momento, o algoritmo é usado para identificar uma câmara de ensaios de satélites, um sistema térmico presente no Laboratório de Integração e Testes do Instituto de Pesquisas Espaciais. O resultado é comparado com a identificação de modelos discretos.

Verificou-se que o algoritmo de identificação proposto foi capaz de identificar os sistemas com sucesso, embora com algumas ressalvas.

# Abstract

System identification studies techniques allow one to obtain mathematical models for systems using sampled data.

This work is about the development of an algorithm for identification of continuous-time dynamical nonlinear polynomial models. The main feature of the proposed algorithm is the use of structure detection techniques. This allows identification of simpler and better models.

Structure detection techniques are algorithms and routines that are used to choose a particular structure from a given representation. The model representation is the type of model employed in the identification procedure. For instance, one may use several representations such as neural networks, radial basis functions, polynomial functions and many others. On the other hand, model structure is the particular model configuration within the chosen representation.

In the case of neural networks, for example, structure detection could be used to choose which neurons and connections should be used in the final model. In this work, we use structure selection techniques in order to choose which terms of a polynomial function should be included in the identified model.

The structure detection strategies employed here are the ERR criterion and the use of *a priori* information about the original system. So far, such techniques have been employed in the identification of discrete-time models. In this work, the aforementioned structure selection techniques are applied to continuous-time polynomial models, apparently for the first time.

The proposed algorithm is described and compared with the closest discrete alternative: NARX polynomial models. The identification procedure is then tested using simulated and real data from nonlinear systems.

First, two chaotic systems were used: the Lorenz's system and the Chua's circuit. Simulated data were used in the identification of the former and real data, collected from a prototype, in the identification of the latter.

In addition, continuous and discrete models were identified for a satellite test chamber, a thermal system of the Laboratório de Integração e Testes

of the Instituto Nacional de Pesquisas Espaciais. The identified continuous models were compared to discrete counterparts.

As discussed in detail, the proposed algorithm was able to successfully identify models for such systems.

# Agradecimentos

Um trabalho como este nunca é fruto do esforço de uma pessoa só. Diversas pessoas contribuíram para sua realização, seja diretamente, com conselhos e observações técnicas, seja indiretamente, dando apoio, conforto e amizade.

O primeiro agradecimento vai para meus pais, por todo o amor, a dedicação, e os sofrimentos passados para que eu tivesse condições de chegar aqui. Particularmente, sou grato a meu pai pelas conversas e discussões que foram, em parte, responsáveis pela escolha de minha carreira.

Gostaria de agradecer ao meu orientador Luís Aguirre pelos conselhos insubstituíveis, ensinamentos e, principalmente, pela paciência de me aguentar por tanto tempo.

Um agradecimento todo especial vai para minha namorada Juliana Paulino pelo amor, carinho e presença. Obrigado também pelo auxílio com as correções do texto.

Gostaria de agradecer também aos colegas do grupo MACSIN, em particular a Álvaro Polati de Souza e a Leonardo Tôrres pelas excelentes discussões. Agradeço também a todos os colegas do CPDEE por proporcionar um ótimo ambiente de trabalho.

Também gostaria de agradecer aos integrantes do grupo Guano Speleo pelos maravilhosos momentos passados tanto abaixo quanto acima da terra.

Agradeço também a todas as pessoas que, de uma forma ou de outra, me ajudaram nessa jornada e que devido à imperfeição da memória humana não foram aqui homenageadas. Que não se sintam esquecidas.

Finalmente, agradeço ao CNPq e à CAPES pelo apoio financeiro.

# Sumário

Resumo	ii
Abstract	iv
Agradecimentos	vi
Sumário	vii
Lista de Símbolos	ix
Lista de Figuras	xii
Lista de Tabelas	xiv
<b>1 Introdução</b>	<b>1</b>
1.1 Motivação . . . . .	2
1.2 Objetivo . . . . .	4
1.3 Apresentação do trabalho . . . . .	4
<b>2 Identificação de modelos discretos</b>	<b>5</b>
2.1 Identificação de sistemas . . . . .	5
2.2 Identificação de modelos NARMAX polinomiais . . . . .	9
2.2.1 Estimação de parâmetros . . . . .	11
2.2.2 Detecção de estrutura . . . . .	13
2.3 Conclusão . . . . .	18
<b>3 Identificação de modelos contínuos</b>	<b>19</b>
3.1 Identificação de modelos contínuos — visão geral . . . . .	19
3.1.1 Comparação com a identificação de modelos discretos . . . . .	21
3.2 Coleta de dados . . . . .	23
3.3 Cálculo das derivadas . . . . .	25
3.4 Identificação de modelos contínuos . . . . .	28

3.4.1	Estimação de parâmetros . . . . .	29
3.4.2	Detecção de estrutura . . . . .	31
3.4.3	Validação . . . . .	35
3.5	Conclusão . . . . .	37
<b>4</b>	<b>Identificação de sistemas caóticos</b>	<b>38</b>
4.1	Sistemas caóticos e identificação . . . . .	38
4.2	O sistema de Lorenz . . . . .	39
4.2.1	Descrição do sistema . . . . .	40
4.2.2	Identificação . . . . .	40
4.3	O circuito de Chua . . . . .	43
4.3.1	Descrição do sistema . . . . .	44
4.3.2	Identificação . . . . .	45
4.4	Conclusão . . . . .	51
<b>5</b>	<b>Identificação de uma Câmara de Termo-Vácuo</b>	<b>52</b>
5.1	A Câmara de testes de termo-vácuo . . . . .	52
5.2	Identificação utilizando modelos contínuos . . . . .	54
5.2.1	Coleta dos dados . . . . .	54
5.2.2	O procedimento de identificação . . . . .	56
5.2.3	Resultados e análise . . . . .	57
5.3	Identificação utilizando os modelos discretos . . . . .	61
5.4	Conclusão . . . . .	63
<b>6</b>	<b>Discussão dos resultados e conclusões</b>	<b>64</b>
6.1	Discussão . . . . .	64
6.2	Conclusões . . . . .	66
6.3	Propostas de continuidade . . . . .	66
<b>A</b>	<b>Guia de Rotinas</b>	<b>68</b>
A.1	Introdução . . . . .	68
A.2	Coleta de dados e cálculo de derivadas . . . . .	69
A.3	O processo de identificação . . . . .	70
A.3.1	Modelos monovariáveis — a rotina <i>contín.m</i> . . . . .	70
	<b>Referências Bibliográficas</b>	<b>81</b>

# Lista de Símbolos

- $a_j$   $j$ -ésimo elemento de  $\varphi(t)$ , ver equação (3.15).
- $d$  Grau de não linearidade.
- $e$  Vetor de resíduos da modelagem contínua. Apesar de ser resíduo, não é erro de predição.
- $f(\cdot)$  Função estática possivelmente não linear.
- $\hat{f}(\cdot)$  Função  $f(\cdot)$  estimada.
- $g(\cdot)$  Função polinomial.
- $\hat{g}(\cdot)$  Função polinomial estimada.
- $g$  Vetor de parâmetros do modelo ortogonal.
- $h(\cdot)$  Função no espaço de imersão.
- $\hat{h}(\cdot)$  Função  $h(\cdot)$  estimada.
- $k$  Índice de tempo discreto.
- $N$  Número de amostras.
- $n_u$  Número de atrasos no sinal  $u$  para modelo discretos.
- $n_y$  Número de atrasos no sinal  $y$  para modelos discretos. Determina a ordem dinâmica do modelo.
- $p$  Número de termos.
- $q$  Dimensão dos vetores  $\phi(k)$  e  $\varphi(t)$ .
- $T$  Matriz de regressores do modelo discreto.
- $u(k)$  Sinal de entrada discreto.

$\mathbf{u}$	Vetor de entrada.
$W$	Matriz de regressores ortogonais.
$\mathbf{w}_i$	Regressor ortogonal.
$X$	Matriz de regressores do modelo contínuo.
$\mathbf{x}$	Vetor de estados no espaço original.
$y(k)$	Sinal de saída discreto.
$\mathbf{y}$	Vetor de estados no espaço de imersão.
$\mathbf{y}_d$	Vetor de saída discreto.
$Y_j$	Termo independente da unidade básica de identificação.
$\mathbf{Y}$	Vetor de termos independentes da unidade básica de identificação.
$\alpha_j$	$j$ -ésimo elemento de $\phi(k)$ , ver equação (2.6).
$\beta_{ij}$	Expoentes inteiros.
$\gamma$	Grau do polinômio usado no cálculo de derivadas.
$\eta_i$	Termo ou regressor contínuo.
$\theta_i$	Coefficiente ou parâmetro do termo $i$ .
$\boldsymbol{\theta}$	Vetor de parâmetros.
$\hat{\boldsymbol{\theta}}$	Vetor de parâmetros estimados.
$\xi(k)$	Erro de predição de um passo à frente ou resíduo.
$\boldsymbol{\xi}$	Vetor de resíduos.
$\tau_i$	Termo ou regressor discreto.
$\phi(k)$	Vetor de entrada discreto, ver equação (2.2).
$\varphi(t)$	Vetor de entrada contínuo, ver equação (3.13).
CTV	Câmara de termo-vácuo.
EDO	Equações diferenciais ordinárias.

- ERR *Error Reduction Ratio*. Taxa de redução de erro. Um critério de detecção de estrutura, p. 16.
- MLP *Multilayer Perceptron*. Tipo de rede neural artificial.
- NARMAX *Nonlinear AutoRegressive Moving Average with eXogenous inputs*. Tipo de modelo discreto, p. 9.
- NARX *Nonlinear AutoRegressive with eXogenous inputs*. Tipo de modelo discreto, p. 9.
- QR Técnica de fatoração de matrizes .
- RBF *Radial Basis Function*. Tipo de rede neural artificial.
- SISO *Single Input Single Output*. Diz-se de modelos que possuem uma entrada e uma saída.

# Lista de Figuras

2.1	Aproximação polinomial da função seno usando um polinômio de grau 7. São mostrados os dados usados no ajuste (cruzes), o gráfico do polinômio (linha contínua) e a função seno real (linha tracejada). . . . .	14
2.2	Aproximação polinomial da função seno usando um polinômio de grau 17. São mostrados os dados usados no ajuste (cruzes), o gráfico do polinômio (linha contínua) e a função seno real (linha tracejada). . . . .	15
4.1	Atrator do sistema de Lorenz (4.1). . . . .	41
4.2	Lorenz variável $x$ . Modelo (linha tracejada) e dados originais (linha cheia). . . . .	42
4.3	Erro de predição da variável $x$ . . . . .	42
4.4	Atrator do modelo identificado. . . . .	43
4.5	Diagrama do circuito de Chua. . . . .	44
4.6	Curva corrente por tensão medida no diodo de Chua utilizado. As maiores divisões são de 0,5V. . . . .	45
4.7	Projeção plana do atrator do circuito de Chua. $y = \dot{x}$ . . . . .	46
4.8	Projeções planas do atrator do circuito de Chua e dos atratores dos modelos identificados. . . . .	48
4.9	Mapas de primeiro retorno na seção de Poincaré do atrator do circuito de Chua e dos atratores dos modelos identificados. . . . .	49
4.10	Projeção do atrator (a) e o correspondente mapa de primeiro retorno (b) gerado pelo modelo simétrico de 20 termos. . . . .	50
5.1	Diagrama da CTV. As setas indicam o fluxo de nitrogênio. . . . .	54
5.2	Dados do ensaio. A entrada é a linha cheia e a saída é a linha tracejada. A linha vertical mostra o momento da interrupção do teste. . . . .	55
5.3	Derivada do sinal $y$ calculada numericamente. A linha vertical mostra o momento da interrupção do teste. . . . .	56

5.4	Validação do modelo de 4 termos. Modelo: linha tracejada, sistema: linha cheia. . . . .	59
5.5	Validação do modelo de 12 termos. Modelo: linha tracejada, sistema: linha cheia. . . . .	60
5.6	Validação do modelo discreto de 4 termos. Modelo: linha tracejada, sistema: linha cheia. . . . .	62

# Lista de Tabelas

4.1	Dinâmica dos modelos obtidos com e sem detecção de estrutura. . . . .	47
5.1	Relação entre o número de termos do modelo identificado e o erro de predição. A simulação do modelo de 15 termos identificado usando 800 centros não convergiu. . . . .	58
5.2	Relação entre o número de termos do modelo identificado e o erro de predição para os modelos discretos. . . . .	62
A.1	Exemplos de representações de termos . . . . .	74

# Capítulo 1

## Introdução

Esta Dissertação trata da identificação de modelos dinâmicos contínuos não lineares polinomiais. Em particular, especial atenção será dada ao problema de detecção de estrutura nesse tipo de modelos.

O que são modelos? Modelos são, segundo Ferreira (1986), conjuntos de hipóteses sobre a estrutura ou o comportamento de um sistema físico pelas quais se procuram explicar ou prever, dentro de uma teoria científica, as propriedades do sistema.

Os modelos são ferramentas importantes. Eles são usados para prever o comportamento de sistemas e como ferramentas de análise e projeto. Os modelos são, enfim, um meio para melhor compreender a natureza. Um tipo de modelo comumente encontrado é o modelo matemático.

Um modelo matemático de um sistema real é um análogo matemático que representa algumas das características observadas em tal sistema (Aguirre, 2000a). Entre os diversos tipos de modelos matemáticos, os modelos dinâmicos (Ogata, 1993) ocupam uma área de destaque. Esse tipo de modelo é usado para representar sistemas cujo estado atual depende de seus estados em outros instantes de tempo. Exemplos de sistemas que podem ser descritos por modelos dinâmicos incluem a temperatura de uma panela cheia d'água, a posição de uma bola num campo de futebol e as tensões e correntes de um circuito elétrico.

Os modelos dinâmicos podem ser de dois tipos: de tempo contínuo e de tempo discreto. Um modelo dinâmico de tempo contínuo é formado por equações diferenciais. Um modelo discreto é formado por equações de diferenças.

Outra grande classificação dos modelos dinâmicos se refere à linearidade. Modelos lineares são aqueles em que se aplica o princípio da superposição. Se tal princípio não é válido para um determinado modelo, este é dito não linear. Todos os sistemas reais são, a rigor, não lineares. Por outro lado, os

modelos lineares são mais fáceis de analisar.

## 1.1 Motivação

O primeiro passo para se utilizar um modelo é obtê-lo. Isso nem sempre é fácil. Muitas vezes é possível usar de conhecimento já disponível para derivar um modelo dinâmico para um determinado sistema. Por exemplo, a partir da mecânica clássica é possível escrever modelos para a maior parte dos sistemas constituídos de corpos em movimento que são encontrados no dia-a-dia.

Outras vezes, porém, encontram-se sistemas para os quais se deseja obter modelos mas não existe uma teoria que permita derivá-los. Às vezes, nem se sabe se um modelo dinâmico determinístico pode existir. Pode-se citar como exemplo desses sistemas a evolução do preço de uma determinada mercadoria ao longo do tempo.

Pode ocorrer também que exista uma teoria da qual seria possível extrair um modelo, mas ele seria tão complexo que não teria utilidade. Esse é o caso de se tentar modelar o fenômeno da turbulência a partir das equações de movimento de cada molécula do fluido.

Quando a construção direta de modelos não é viável mas o fenômeno em estudo possui grandezas mensuráveis existe, a possibilidade da busca por modelos empíricos. É nesse meio que se insere o presente trabalho.

Esse procedimento de busca por modelos empíricos recebe o nome de identificação de sistemas (Box e Jenkins, 1970; Ljung, 1987; Aguirre, 2000a). A identificação de sistemas pode ser aplicada sempre que se procura obter um modelo para um sistema que possui grandezas mensuráveis e para o qual, por qualquer motivo, não tenha sido possível derivar um modelo.

Existem várias alternativas para se obter modelos a partir de dados reais. Tradicionalmente, os métodos de identificação de modelos paramétricos identificavam, em geral, modelos discretos lineares (Ljung, 1987). Os modelos lineares são mais simples de se identificar e, principalmente, são fáceis de analisar, pois existe um grande número de ferramentas disponíveis para a análise de sistemas lineares (Ogata, 1993; Chen, 1993).

Porém, a maioria dos sistemas encontrados na natureza são não lineares. Dessa forma, o uso de uma representação linear para modelá-los é limitada. Modelos lineares não são capazes de reproduzir uma série de comportamentos dinâmicos observados na prática.

Nesse contexto, surgiram diversas maneiras de se identificar modelos dinâmicos globais. Diversas representações foram empregadas como, por exemplo, modelos NARMAX polinomiais (Billings et al., 1989), modelos contínuos (Cremers e Hübler, 1987; Giona et al., 1991; Baake et al., 1992;

Rowlands e Sprott, 1992; Brown et al., 1994; Gouesbet e Letellier, 1994; Lainscsek et al., 1998; Perona et al., 2000), funções de base radial (Broomhead e Lowe, 1988), entre outros. Entre as representações contínuas, existem diversas abordagens de estimação de parâmetros, como minimização de erro de predição (Baake et al., 1992; Perona et al., 2000; Timmer et al., 2000) e via cálculo de derivadas (Cremers e Hübler, 1987; Gouesbet e Letellier, 1994).

Por outro lado, o campo de identificação de modelos não lineares, produziu grande número de trabalhos sobre detecção de estrutura (Billings et al., 1989; Kadtke et al., 1993; Aguirre et al., 1997; Mendes e Billings, 1998; Celka et al., 1999; Henrique et al., 2000; Reed, 1993). As técnicas de detecção de estrutura permitem escolher, dentro de uma representação, a base de funções mais adequada para representar um determinado sistema. Por exemplo, no caso de se utilizar a representação na forma de redes neurais, técnicas de detecção de estrutura poderiam ser usadas para escolher os neurônios e conexões mais adequados. Já no caso da representação polinomial, usada neste trabalho, tais técnicas são usadas para escolher os termos, ou monômios, que devem ser incluídos no modelo.

A necessidade de se fazer tal escolha de forma apropriada vem, em primeiro lugar, da própria não linearidade das representações. Uma representação não linear tem uma ampla gama de estruturas possível. Um exemplo típico é o caso das redes neurais artificiais (Elsner, 1992). Uma rede neural artificial tipo *perceptron* multicamadas pode facilmente ter uma estrutura muito complexa. Assim, recentemente, tem-se procurado alternativas para reduzir o tamanho da rede e, assim, melhorar a qualidade da identificação. Um exemplo são as técnicas de *prunning* (Reed, 1993; Henrique et al., 2000).

A principal motivação para o uso de técnicas de detecção de estrutura é, no entanto, dinâmica. Modelos com uma estrutura exageradamente complexa costumam exibir comportamentos dinâmicos espúrios (Aguirre e Billings, 1995).

Apesar disso, pouco material existe na literatura a respeito de técnicas de detecção de estrutura aplicadas à identificação de sistemas contínuos. Em Brown et al. (1997), os autores removem certos termos do conjunto de termos candidatos de forma a garantir simetrias específicas dos modelos. Em Bezruchko e Smirnov (2000), os autores consideram o caso específico de sistemas excitados harmonicamente.

Talvez o trabalho que mais se aproxima dessa Dissertação seja Swain e Billings (1998). Nesse artigo, os autores empregam o critério ERR na identificação de modelos contínuos polinomiais. A filosofia do artigo é obter uma função de resposta em frequência generalizada não linear a partir da identificação de modelos discretos NARMAX usando os dados coletados no sistema. Então, os modelos contínuos são identificados a partir dessa res-

posta em frequência generalizada empregando a detecção de estrutura para determinar quais termos devem ser incluídos. Tal procedimento é relativamente complexo, pois envolve duas etapas de identificação, e é limitado por questões de ordem prática a modelos de terceira ordem. Por outro lado, tem a grande vantagem de não precisar de cálculo numérico de derivadas de sinais amostrados.

Nesta Dissertação, pretende-se utilizar técnicas de detecção de estrutura na identificação de modelos dinâmicos contínuos polinomiais. A identificação dos modelos é feita diretamente com os dados coletados, embora precise do cálculo de derivadas. Os modelos identificados segundo esse procedimento podem ser, a princípio, de qualquer ordem.

## 1.2 Objetivo

O objetivo desse trabalho é adaptar e empregar técnicas de detecção de estrutura de modelos discretos disponíveis na literatura na identificação de modelos contínuos polinomiais.

O produto desse trabalho deverá ser um novo algoritmo para identificar esse tipo de modelo.

## 1.3 Apresentação do trabalho

Essa Dissertação está dividida da seguinte forma. O Capítulo 2 traz uma breve revisão da identificação de sistemas dinâmicos não lineares discretos. O Capítulo 3 apresenta o algoritmo de identificação de modelos contínuos proposto. O Capítulo 4 mostra um exemplo de aplicação no qual sistemas caóticos conhecidos são identificados usando o procedimento apresentado. O Capítulo 5 é outro estudo de caso. Nesse Capítulo, uma câmara de ensaios de satélite é identificada, ilustrando uma aplicação mais próxima do cotidiano da engenharia. Por fim, no Capítulo 6 são apresentadas as conclusões do trabalho e propostas de continuação.

# Capítulo 2

## Identificação de modelos discretos

Antes de apresentar a descrição do procedimento de identificação de modelos contínuos, é importante fazer uma breve revisão da identificação de modelos discretos, particularmente modelos NARMAX polinomiais. Tal revisão é interessante porque o procedimento de identificação dos modelos contínuos apresentado neste trabalho possui muitas semelhanças com essa técnica e a analogia torna-se, então, clara. Na verdade, as técnicas de detecção de estrutura dos modelos contínuos usadas nesse trabalho foram adequadas da identificação de modelos NARMAX polinomiais.

Esse Capítulo está dividido da seguinte forma. A Seção 2.1 contém uma breve revisão da identificação de modelos discretos. A Seção 2.2 descreve sucintamente a identificação de modelos NARMAX polinomiais. A Seção 2.3 apresenta os comentários finais do Capítulo. O material coberto neste Capítulo pode ser encontrado com mais detalhes em outras dissertações tais como Rodrigues (1996) e Cassini (1999). Uma revisão do uso de modelos racionais pode ser encontrada em Corrêa (1997, 2001). Estas representações foram apresentadas de forma tutorial em Aguirre (2000a).

### 2.1 Identificação de sistemas

A identificação de sistemas é o procedimento que permite obter, a partir de dados experimentais, modelos matemáticos para sistemas quaisquer. O produto da identificação é, pois, um modelo matemático que representa (bem ou mal) um determinado sistema. A essa forma de obtenção de modelos se opõe a modelagem conceitual que procura obter modelos a partir da descrição matemática dos fenômenos inerentes ao sistema (Garcia, 1997).

A identificação de sistemas é normalmente preferida nos casos em que a modelagem conceitual, também chamada modelagem pela física do processo, é muito complexa ou até mesmo impossível devido à inexistência de conhecimento a respeito do sistema em questão. Isso ocorre porque a identificação se baseia nas relações de causa e efeito presentes nos dados coletados no sistema e não na descrição de fenômenos.

Porém, além dos dados, a identificação necessita que se escolha uma forma de representação para os modelos que se desejam obter.

Nesse ponto, o usuário de identificação deve optar entre diversas possibilidades. Por exemplo, ele pode escolher entre modelos dinâmicos ou estáticos, de tempo discreto ou de tempo contínuo e entre modelos lineares e não lineares, entre outros.

Dentre os fatores que influenciam essa escolha pode-se citar a familiaridade do usuário com uma determinada representação matemática, algum conhecimento que se tenha *a priori* sobre o sistema a identificar (Aguirre, 2000b) e a aplicação a que se destina o modelo.

Esse processo de escolha pode ocorrer em várias etapas. Por exemplo, uma vez determinado que serão utilizados modelos dinâmicos, de tempo discreto e não lineares, uma outra escolha ainda se faz necessária. Existe uma ampla gama de representações entre as quais o usuário das técnicas de identificação pode optar, como redes MLP (*Multilayer Perceptron*) (Elsner, 1992), redes RBF's (*Radial Basis Function*) (Broomhead e Lowe, 1988), lógica nebulosa (Takagi e Sugeno, 1985) redes *Neo Fuzzy Neuron* (Yamakawa et al., 1992; Caminhas e Gomide, 2000), modelos NARMAX polinomiais (Leontaritis e Billings, 1985a,b) ou racionais (Billings e Chen, 1989), entre muitos outros. Cada uma dessas representações tem características distintas em relação à complexidade, flexibilidade, facilidade de treinamento, capacidade de generalização, custo computacional e outros quesitos. A escolha da melhor representação para um sistema em particular é, de maneira geral, um problema ainda em aberto, porém progressos têm sido feitos nessa área (Aguirre, 2000b). Essas representações, no entanto, apresentam um ponto em comum: elas são todas paramétricas. Ou seja, essas representações possuem parâmetros numéricos que podem ser ajustados. São justamente esses parâmetros que conferem a essas estruturas a sua flexibilidade tão fundamental na identificação de sistemas.

Independentemente da representação escolhida, o procedimento de identificação é o mesmo e pode ser resumido nas seguintes etapas (Ljung, 1987; Aguirre, 2000a):

- Ensaio e coleta de dados;
- Pré-processamento dos dados;

- Escolha de estrutura;
- Estimação dos parâmetros;
- Validação.

Na primeira etapa, um ensaio no sistema de interesse é projetado e executado. Os dados são, então, coletados. Já nesse ponto ocorre influência da representação escolhida. Um dos principais parâmetros da coleta de dados é o tempo de amostragem  $T_s$ . Quando o modelo a ser identificado é discreto, a escolha desse parâmetro deve ser feita de modo a gerar um espaço de reconstrução adequado<sup>1</sup>. Existem na literatura diversas estratégias para a escolha de  $T_s$  quando a representação é discreta (Aguirre, 1995; Rosenstein et al., 1994). A escolha desse parâmetro é geralmente diferente na identificação de modelos contínuos.

No pré-processamento dos dados, vários procedimentos podem ser adotados. Os dados podem ser decimados (ou interpolados) para obter um  $T_s$  adequado, filtrados para diminuir o efeito de ruído, podem ser aplicados testes para detecção de não linearidades (Haber, 1985; Billings e Voon, 1986; Aguirre, 2000a), derivadas temporais podem ser calculadas (Oliveira, 1998; Peixoto, 2000; Gibson et al., 1992), entre outros.

Na escolha ou *detecção de estrutura* um conjunto de funções base é escolhido dentro das possibilidades de cada representação. Existem técnicas de detecção de estrutura para diversas representações como modelos NARMAX polinomiais (Billings et al., 1989), racionais (Billings e Chen, 1989; Corrêa, 1997), redes neurais (Reed, 1993), entre outras.

O passo seguinte é a estimação dos parâmetros do modelo. Essa etapa é basicamente a solução de um problema de otimização cujas variáveis são os parâmetros do modelo e a função custo a otimizar é alguma função dos dados de identificação e do modelo identificado.

Na etapa de validação, a qualidade do modelo é verificada de alguma forma. Entre as técnicas de validação pode-se citar a validação estatística (Billings e Voon, 1986) e a validação dinâmica (Aguirre e Billings, 1994).

Seja, por exemplo, o caso de se identificar um modelo dinâmico discreto não linear com uma entrada e uma saída. Após a amostragem, o conjunto de dados de identificação será composto de duas séries temporais: a entrada  $u(k)$  e a saída  $y(k)$ , para  $k = 1, 2, \dots, N$ . Uma maneira geral de representar o sistema a ser identificado é

$$y(k) = f[y(k-1), y(k-2), \dots, y(k-n_y), u(k-1), u(k-2), \dots, u(k-n_u)] , \quad (2.1)$$

---

<sup>1</sup>Mais detalhes a respeito no Capítulo 3.

sendo  $n_y$  o número de atrasos em  $y$  ou ordem dinâmica do modelo,  $n_u$  o número de atrasos em  $u$  e  $f(\cdot)$  uma função não linear com  $q = n_y + n_u$  entradas. Nessa equação não foram incluídos termos de ruído<sup>2</sup>. A essência do processo de identificação nesse caso é aproximar  $f(\cdot)$  por uma outra função  $\hat{f}(\cdot)$ , usando uma representação qualquer como redes neurais MLP ou redes RBF's, por exemplo (Aguirre, 2000a). Essa aproximação é feita baseada nos dados coletados.

Um ponto interessante é que  $f(\cdot)$ , na equação (2.1), é uma função estática, ou seja, seu valor depende apenas do valor do seu vetor de entrada

$$\phi(k) = \begin{bmatrix} y(k-1) \\ y(k-2) \\ \vdots \\ y(k-n_y) \\ u(k-1) \\ u(k-2) \\ \vdots \\ u(k-n_u) \end{bmatrix} \quad (2.2)$$

e não de valores anteriores desse vetor. A dinâmica é conseguida através da forma de construção do vetor  $\phi(k)$ , que utiliza valores atrasados de  $y$  e de  $u$ .

Outro ponto importante é que se pode conseguir modelos de ordem maior que um sem ter que amostrar mais de uma variável de estado.

Para se obter a função aproximada  $\hat{f}(\cdot)$ , é definido o erro de predição de um passo à frente ou resíduo

$$\xi(k) = y(k) - \hat{f}[\phi(k)] \quad (2.3)$$

que é avaliado em  $N$  pontos gerando o vetor de erros de resíduos  $\boldsymbol{\xi} = [\xi(1) \ \xi(2) \ \cdots \ \xi(N)]^T$ . Uma função de custo qualquer  $J(\boldsymbol{\xi})$  é então definida sobre esse vetor. A função de custo normalmente utilizada é o somatório dos erros quadráticos

$$J(\boldsymbol{\xi}) = \boldsymbol{\xi}^T \boldsymbol{\xi} . \quad (2.4)$$

Essa função depende de  $\hat{f}(\cdot)$ . Assim, pode-se usar uma técnica otimização para procurar a função ótima  $\hat{f}^*(\cdot)$ . Um ponto importante é que a função ótima depende da função de custo escolhida e essa escolha não é única.

Esse procedimento é o mesmo para todas as representações citadas anteriormente e constitui a parte principal da identificação. Porém, existe uma

---

<sup>2</sup>A inclusão de termos de ruído, em geral, tem o objetivo de melhorar as características estatísticas do processo de estimação de parâmetros. Em Aguirre (2000a) podem ser vistas algumas formas de utilizar termos de ruído na identificação de modelos discretos.

diferença importante entre as diversas representações. Algumas representações são lineares em relação aos seus parâmetros e outras não. Isso acarreta diferenças significativas na fase de otimização. Enquanto as representações que são lineares nos parâmetros permitem a utilização de ferramentas de otimização mais simples e eficazes como o método dos mínimos quadrados, as outras exigem ferramentas de otimização não lineares. O fato do problema de otimização ser não linear acarreta uma série de complicações, como existência de mínimos locais e elevado custo computacional.

A seguir, será exemplificado o procedimento de identificação. Para tal, serão utilizados como representação os modelos NARMAX polinomiais.

## 2.2 Identificação de modelos NARMAX polinomiais

Os modelos NARMAX<sup>3</sup> polinomiais são modelos de tempo discreto não lineares (Leontaritis e Billings, 1985a,b; Rodrigues, 1996; Aguirre, 2000a). Porém tais modelos são lineares nos parâmetros.

Os modelos NARMAX polinomiais aproximam a função  $f(\cdot)$  da equação (2.1) por uma função polinomial, daí o nome. Essa função tem a forma

$$g[\phi(k)] = \sum_{i=1}^p \theta_i \tau_i(k) , \quad (2.5)$$

sendo  $\theta_i$  os parâmetros ou coeficientes,  $p$  o número de termos e  $\tau_i$  os *termos* ou *regressores*. Esses termos têm a forma

$$\tau_i(k) = \prod_{j=1}^q \alpha_j^{\beta_{ij}}(k) , \quad i = 1, \dots, p , \quad (2.6)$$

sendo  $q$  o número de elementos de  $\phi(k)$  e  $\alpha_j(k)$  o  $j$ -ésimo elemento de  $\phi(k)$  (ver equação 2.2). Assim, os  $\alpha_j(k)$  são, na verdade, os sinais  $y(k)$  e  $u(k)$  atrasados de diversas quantidades diferentes. Os expoentes  $\beta_{ij}$  são números naturais. A quantidade  $d = \sum_{j=1}^q \beta_{ij}$  é denominada *grau* do termo  $i$ .

---

<sup>3</sup>A sigla NARMAX vem do inglês *Nonlinear AutoRegressive Moving Average with exogenous inputs* sendo que *Moving Average* se refere à modelagem de ruído tipo média móvel feita durante a estimação para se evitar polarização do estimador. Nesse trabalho, a identificação de modelos NARMAX polinomiais é citada apenas para comparação com a técnica contínua e a modelagem de ruído não é mostrada aqui. Portanto, a sigla mais correta para os modelos aqui descritos seria NARX. Porém o termo mais comum é NARMAX e esse termo foi usado aqui como representante dessa ampla classe de modelos.

É importante observar duas características desses modelos. Primeiro, os termos são formados a partir de grandezas conhecidas, nada mais que os sinais medidos  $y$  e  $u$  e seus deslocamentos no tempo, conforme mostra a equação (2.2). Segundo, como já foi dito, os modelos NARMAX polinomiais são lineares nos parâmetros.

O primeiro passo da identificação é o ensaio de coleta de dados (Ljung, 1987). O ensaio deve ser feito de forma que os dados de identificação tenham informação suficiente sobre o sistema. No caso da identificação de sistemas lineares, é desejado que o sinal de entrada usado no teste tenha um espectro de frequência mais amplo do que a banda passante do sistema a ser identificado, de forma a excitar adequadamente a dinâmica do sistema. No caso de sistemas não lineares, também deve-se garantir que os sinais de excitação possuam uma larga faixa de amplitudes de forma a excitar as não linearidades do sistema (Aguirre, 2000a; Rodrigues, 1996; Leontaritis e Billings, 1987).

O tempo de amostragem deve ser escolhido segundo as considerações da Seção 2.1.

A etapa seguinte é o pré-processamento dos dados. Normalmente nessa etapa são realizados testes de detecção de não linearidades (Haber, 1985; Billings e Voon, 1986; Aguirre, 2000a), testes de correlação para determinar as entradas e saídas (Aguirre, 2000a), análise em componentes principais para eliminar redundância entre sinais (Yared, 2001), entre outras possibilidades.

A seguir, vem a identificação propriamente dita, que compreende as etapas de detecção de estrutura e estimação de parâmetros.

Obter um modelo NARMAX polinomial por identificação significa determinar todas as constantes desconhecidas nas equações (2.5) e (2.6) da melhor forma possível, tendo em vista os dados de identificação. Isso quer dizer determinar não somente os coeficientes  $\theta_i$ , mas também o número de termos  $p$  e todos os expoentes  $\beta_{ij}$ .

Claramente pode-se notar que determinar os parâmetros  $\theta_i$  é muito diferente do que determinar  $\beta_{ij}$  e  $p$ . Além da questão de linearidade, há também o problema dos últimos serem variáveis inteiras. Esses dois problemas, a determinação dos  $\theta_i$  e a determinação de  $\beta_{ij}$  e de  $p$ , são tão distintos que na literatura são tratados por nomes diferentes. O primeiro é conhecido como estimação de parâmetros e o segundo, como detecção de estrutura.

Tais procedimentos, devido a sua importância, serão explicados separadamente nas Seções 2.2.1 e 2.2.2.

Finalmente, após essas etapas, vem a parte da validação dos modelos identificados (Aguirre, 2000a). É na validação que a qualidade dos modelos é aferida. Existem várias formas de se fazer a validação, dependendo do objetivo ao qual se destina o modelo. Uma alternativa é simular o modelo

identificado e comparar o resultado da simulação com o sinal coletado do sistema original. Também há técnicas de validação que funcionam pela análise dos resíduos<sup>4</sup> de identificação. São chamadas de validação estatística (Billings e Voon, 1986). Essas técnicas procuram detectar alguma característica determinística que porventura tenha escapado ao processo de modelagem.

Em contrapartida da validação estatística existe a validação dinâmica (Aguirre e Billings, 1994). Ela consiste em calcular grandezas (ou outras propriedades) invariantes características do sistema em questão e compará-las com aquelas calculadas para os modelos identificados. De qualquer forma, a validação de modelos não lineares ainda é um problema em aberto (Aguirre, 2000a).

### 2.2.1 Estimação de parâmetros

Nessa seção é abordado o problema da estimação de parâmetros dos modelos NARMAX polinomiais, ou seja, a estimação dos  $\theta_i$  na equação (2.5). Para tal, é pressuposto que a estrutura, isto é, os  $\beta_{ij}$  e o  $p$ , já é conhecida.

Para que se possa estimar os parâmetros é necessário que dados de entrada  $u$  e saída  $y$  do sistema original tenham sido coletados. Obviamente, se o sistema não apresentar entradas, o sinal  $u$  não é considerado na identificação.

O procedimento, como mostrado aqui, se aplica à identificação de modelos monovariáveis ou SISO (*Single Input Single Output*, uma entrada uma saída).

Suponha que o sistema a ser identificado possa ser representado por

$$y(k) = g[\phi(k)] = \sum_{i=1}^p \theta_i \tau_i(k) . \quad (2.7)$$

A equação (2.7) pode ser escrita para diferentes instantes de amostragem gerando

$$\begin{cases} y(1) = \sum_{i=1}^p \theta_i \tau_i(1) \\ y(2) = \sum_{i=1}^p \theta_i \tau_i(2) \\ \vdots \\ y(N) = \sum_{i=1}^p \theta_i \tau_i(N) \end{cases} \quad (2.8)$$

que em forma matricial fica

$$\begin{bmatrix} y(1) \\ y(2) \\ \vdots \\ y(N) \end{bmatrix} = \begin{bmatrix} \tau_1(1) & \tau_2(1) & \cdots & \tau_p(1) \\ \tau_1(2) & \tau_2(2) & \cdots & \tau_p(2) \\ \vdots & \vdots & \ddots & \vdots \\ \tau_1(N) & \tau_2(N) & \cdots & \tau_p(N) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_p \end{bmatrix} \quad (2.9)$$

<sup>4</sup>O resíduos serão explicados adiante.

ou

$$\mathbf{y}_d = T\boldsymbol{\theta} . \quad (2.10)$$

Repare<sup>5</sup> que, apesar de  $\tau_i(k)$  ter o mesmo argumento de  $y(k)$ , as colunas de  $T$  contém sempre sinais atrasados em relação a  $\mathbf{y}_d$ . Isso ocorre porque, na definição de  $\phi(k)$ , os sinais  $y(k)$  e  $u(k)$  estão atrasados de pelo menos um instante de amostragem.

Utilizando a equação (2.3) tem-se o erro de predição de um passo à frente dos modelos NARMAX polinomiais

$$\boldsymbol{\xi} = \mathbf{y}_d - T\hat{\boldsymbol{\theta}} . \quad (2.11)$$

O princípio aqui utilizado foi o de gerar um grande número de restrições pela amostragem continuada do sistema ao longo do tempo. Cada linha de (2.8) é a equação (2.7) escrita para um instante de amostragem diferente. Uma consideração implícita na passagem de (2.7) para (2.8) é que o sistema é invariante no tempo. Essa suposição também é feita durante a identificação de modelos usando outras representações, como redes neurais por exemplo, e é decorrente do artifício de gerar restrições pela amostragem ao longo do tempo. Porém, nem todos os procedimentos de identificação exigem invariância no tempo. Algoritmos como o mínimos quadrados recursivo com fator de esquecimento (Ljung e Söderström, 1983; Cassini, 1999; Aguirre, 2000a) levam em conta a variação temporal do sistema.

Outro ponto importante é que a passagem de (2.8) para (2.9) só foi possível porque  $g[\phi(k)]$  é linear em relação aos parâmetros  $\theta_i$ . Qualquer representação linear nos parâmetros permite essa separação. As RBF's também são lineares em relação a alguns de seus parâmetros (Broomhead e Lowe, 1988).

A estimação de  $\boldsymbol{\theta}$  está relacionada à definição de uma função de custo, conforme visto na Seção 2.1. O algoritmo empregado nesse trabalho utiliza a função definida em (2.4), daí o seu nome *Estimador de Mínimos Quadrados*. Essa função custo apresenta apenas um ponto de mínimo se a representação for linear nos parâmetros, como é o caso. Assim, um dos problemas mais complicados que existe em otimização — a presença de mínimos locais — não ocorre aqui. Além do mais, esse ponto mínimo pode ser determinado analiticamente através da equação normal, levando a

$$\hat{\boldsymbol{\theta}} = [T^T T]^{-1} T^T \mathbf{y}_d , \quad (2.12)$$

---

<sup>5</sup>Na equação (2.10) foi usado o índice  $d$  em  $\mathbf{y}_d$  para diferenciar esse vetor do vetor de estados  $\mathbf{y}$  que será visto adiante.

se  $T^T T$  for não singular. Por essa equação é possível estimar os parâmetros de um modelo NARMAX polinomial<sup>6</sup>, dispondo da estrutura e dos dados. Vale a pena lembrar que a equação (2.12) não é a única forma de se encontrar  $\hat{\theta}$  segundo o critério dos mínimos quadrados. Existem diversas outras formas de se fazer isso, com propriedades numéricas e computacionais diferentes (Aguirre, 2000a).

Esse procedimento de estimação de parâmetros pode apresentar problemas de polarização (*bias*) (Aguirre, 2000a), ou seja, a esperança matemática dos valores estimados dos parâmetros pode ser diferente dos valores “reais”. Isso ocorre se o ruído no sistema for não branco. Tal problema é normalmente atacado utilizando modelagem de ruído. A filosofia nessa abordagem é incluir no modelo a parte determinística (não branca) do ruído. Se a representação escolhida para o modelo de ruído é do tipo média móvel, os modelos identificados são do tipo NARMAX propriamente ditos. Nesse caso, os parâmetros devem ser estimados por outros estimadores que não o mínimos quadrados. Uma alternativa é o estimador de mínimos quadrados estendido.

## 2.2.2 Detecção de estrutura

Antes de se fazer a pergunta “*como* fazer a detecção de estrutura?” deve-se responder outra: “*por que* fazer a detecção de estrutura?”

A princípio, poderia se pensar que a detecção de estrutura não é de forma alguma um problema. Por exemplo, um modelo NARMAX polinomial aproxima a função  $f(\cdot)$  em (2.1) por uma função polinomial. Intuitivamente, poderia-se pensar que basta incluir os termos em ordem crescente de grau e tantos termos quantos se queira, pois quanto maior o número de termos melhor seria a aproximação, tal como acontece na aproximação de uma função por séries de Taylor. Isso seria verdade se ocorresse uma situação completamente ideal, na qual não existissem incertezas de nenhuma natureza. Como, na prática, o ruído está sempre presente, essa estratégia de incluir termos no modelo indiscriminadamente pode piorar a identificação, ao invés de melhorá-la.

Como forma de ilustrar esse problema é mostrada a seguir a aproximação polinomial da função seno.

Nesse exemplo, a função seno foi amostrada em 20 pontos igualmente espaçados no intervalo  $[-\pi, \pi]$  e a essas amostras foi acrescentado ruído aditivo gaussiano branco de desvio padrão 0,01. Depois foi feita a aproximação polinomial desses dados utilizando a função *polyfit* do GNU Octave. Duas

---

<sup>6</sup>A rigor, essa equação só pode ser usada diretamente na estimação de coeficientes de modelos que não têm termos de ruído (termos de média móvel).

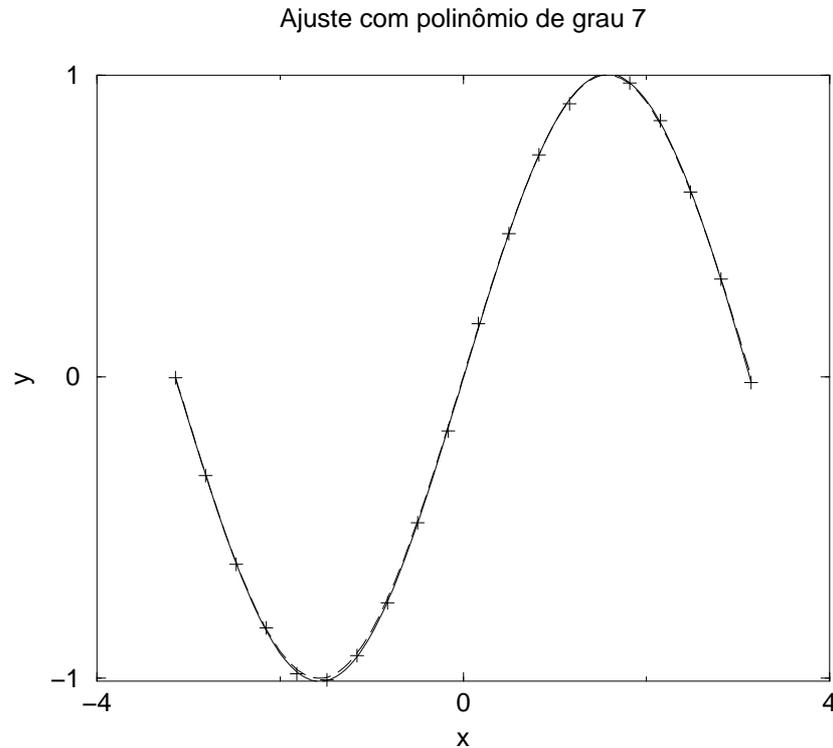


Figura 2.1: Aproximação polinomial da função seno usando um polinômio de grau 7. São mostrados os dados usados no ajuste (cruzes), o gráfico do polinômio (linha contínua) e a função seno real (linha tracejada).

aproximações diferentes foram feitas. Na primeira, foi usado um polinômio de grau 7 e na segunda, um de grau 17. O resultado da primeira aproximação se encontra na Figura 2.1 e o da segunda na Figura 2.2

Pode-se notar que a aproximação com um polinômio de grau 7 ficou bastante razoável. Seria de se supor que, baseado na expansão em série de potências da função seno, um aumento no grau do polinômio melhoraria a aproximação. Porém, dificilmente alguém poderia dizer que o gráfico mostrado na Figura 2.2 é uma melhor aproximação para a função seno que o da Figura 2.1. No entanto, em um quesito, o polinômio de grau 17 se saiu melhor que o de grau 7. O erro da aproximação *nos pontos usados para calculá-la* foi menor ( $4,08 \times 10^{-5}$  contra  $7,51 \times 10^{-4}$ , soma dos erros quadráticos). O que causou o problema foi o excesso de flexibilidade do polinômio de maior grau aliado à presença de ruído.

Isso é um bom exemplo do problema da sobreparametrização, ou seja, excesso de flexibilidade da estrutura utilizada para representar o sistema.

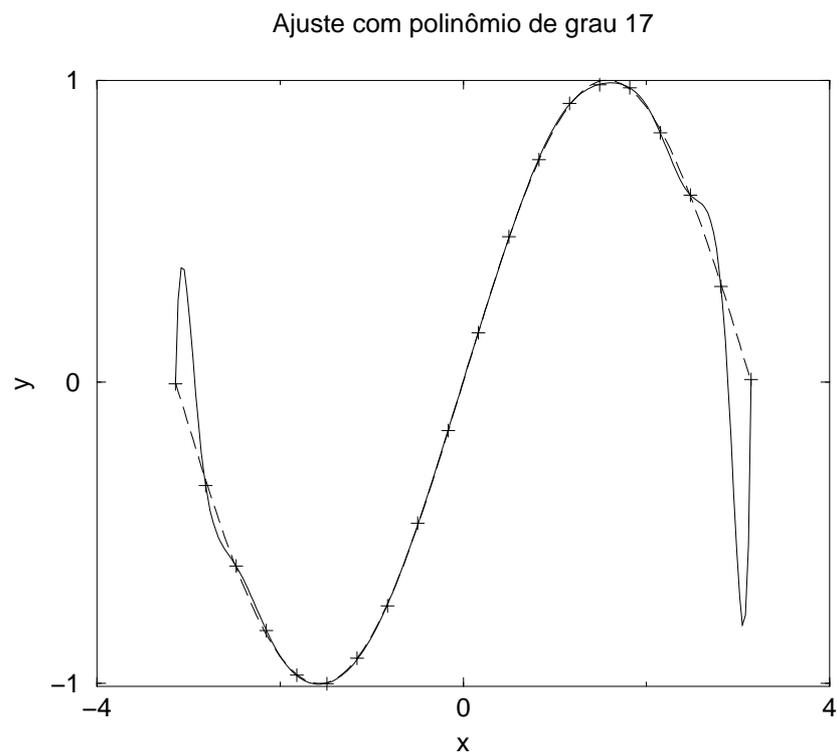


Figura 2.2: Aproximação polinomial da função seno usando um polinômio de grau 17. São mostrados os dados usados no ajuste (cruzes), o gráfico do polinômio (linha contínua) e a função seno real (linha tracejada).

Esse exemplo foi feito usando uma função monovariável e estática. Num caso dinâmico, tal problema é mais difícil de detectar e corrigir. A sobrepametrização ocorre com frequência na identificação de sistemas dinâmicos não lineares resultando em regimes dinâmicos espúrios (Aguirre e Billings, 1995).

Do ponto de vista da identificação de modelos polinomiais, o problema da detecção de estrutura se resume à pergunta: “Quais (e quantos) termos incluir num modelo quando se identifica um determinado sistema?”

Embora o número possível de termos na representação polinomial seja muito elevado, apenas uma pequena parte deles é suficiente para aproximar a dinâmica do processo na maioria dos casos (Rodrigues et al., 1996). O problema é justamente escolher, dentre um universo de termos candidatos, alguns mais “adequados”. Isso é feito na detecção de estrutura.

Várias técnicas podem ser usadas para fazer a detecção de estrutura, como por exemplo algoritmos genéticos (Fonseca et al., 1993; Mao e Billings, 1997) e *zeroing-and-refitting* (Kadtke et al., 1993).

Em particular, a taxa de redução de erro (ERR) (Chen et al., 1989) é bastante utilizada para esse propósito na identificação de modelos NARMAX polinomiais. O emprego dessa e outras técnicas na identificação de modelos contínuos é o tema central do presente trabalho. Assim, a taxa de redução de erro será descrita aqui apenas brevemente, ficando a descrição detalhada para a parte que trata da identificação dos modelos contínuos.

A motivação por trás do critério do ERR é a busca por uma forma de quantificar a importância individual de cada termo na representação do vetor  $\mathbf{y}_d$  em (2.10).

A taxa de redução de erro mede a fração da soma dos erros quadráticos que pode ser explicada por cada termo. Ela é, portanto, uma medida estatística. A soma dos erros quadráticos é, não por acaso, também utilizada como função de custo para a estimação dos parâmetros. Esse conceito fica mais claro através de um exemplo.

Reescrevendo (2.11) chega-se a

$$\mathbf{y}_d = T\hat{\boldsymbol{\theta}} + \boldsymbol{\xi} , \quad (2.13)$$

sendo que  $\hat{\boldsymbol{\theta}}$  foi suposto como tendo sido calculado de forma a minimizar  $J(\boldsymbol{\xi})$  em (2.4). O maior valor que  $J(\boldsymbol{\xi})$  pode assumir nesse caso é  $J(\mathbf{y}_d) = \mathbf{y}_d^T \mathbf{y}_d$  quando  $\hat{\boldsymbol{\theta}}$  for o vetor nulo ou quando o modelo não tiver nenhum termo.

Se a um modelo for acrescentado um termo que não estava presente antes, uma fração do sinal  $\mathbf{y}_d$  passa a ser “explicada” por esse termo e a soma dos erros quadráticos  $\boldsymbol{\xi}^T \boldsymbol{\xi}$  diminui. Essa diminuição, normalizada por  $J(\mathbf{y})$ , que é o maior valor que  $J(\boldsymbol{\xi})$  pode assumir, é o ERR para o termo incluído.

Um termo que diminua muito os erros de modelagem é, a princípio, melhor para a qualidade do modelo que um outro que não os reduza tanto assim. É importante ressaltar que o cálculo do ERR exige a estimação dos parâmetros do modelo.

O ERR é portanto uma medida da “importância” individual de cada termo na modelagem. Essa medida permite ordenar um conjunto de termos candidatos a serem incluídos num modelo em ordem decrescente de “importância”. Essa é a base da detecção de estrutura utilizando o critério ERR.

Esse critério, usado diretamente, tem uma limitação significativa. Apesar de ser (ou tentar ser) uma medida da “importância” individual de cada termo, ele não é completamente individual. O valor do ERR para cada termo depende dos termos já presentes no modelo, ou seja, depende da ordem na qual o valor do ERR é calculado para cada termo.

Para resolver em parte esse problema, normalmente é utilizado um algoritmo de busca não ótimo introduzido em Billings et al. (1989) chamado regressão direta. Basicamente, nesse algoritmo, o ERR de cada termo candidato é recalculado toda vez que um termo é incluído no modelo. Nesse trabalho, a regressão direta é sempre empregada junto com o critério ERR.

Em Chen et al. (1989) é apresentada uma forma eficiente para se calcular o ERR quando a estimação de parâmetros é feita por um algoritmo ortogonal.

Apesar de ser uma ferramenta importante na identificação de sistemas, o ERR não resolve completamente o problema. Ainda é necessário determinar quantos termos incluir no modelo e o ERR não responde a essa pergunta. Para isso pode-se utilizar um critério de informação como o de Akaike (Akaike, 1974). Uma outra alternativa é fazer uma busca no espaço do número de termos  $p$ .

O critério ERR não se limita a modelos NARMAX polinomiais. Ele pode ser empregado em qualquer estrutura que seja linear nos parâmetros, ou seja, qualquer estrutura da forma da equação (2.10).

O ERR requer para a sua utilização apenas os dados de entrada e saída (ou apenas as grandezas necessárias para formar  $\phi(k)$  em (2.2)). Quando, no processo de identificação, estão disponíveis outras informações além dos dados de entrada e saída, elas podem ser utilizadas na detecção de estrutura. Por exemplo, o conhecimento do número e simetria de pontos fixos pode ajudar a excluir termos candidatos espúrios (Aguirre e Mendes, 1996; Corrêa, 2001). Essa questão será vista detalhadamente na parte que trata dos modelos contínuos.

## 2.3 Conclusão

Neste Capítulo foi descrito um procedimento genérico de identificação de modelos discretos e, em particular, um algoritmo de identificação de modelos NARX polinomiais. Tal algoritmo faz uso do critério ERR como técnica de detecção de estrutura. O uso de detecção de estrutura é fundamental para o bom desempenho da identificação de modelos com essa representação.

# Capítulo 3

## Identificação de modelos contínuos

O presente Capítulo trata da identificação de modelos dinâmicos contínuos não lineares e do uso de técnicas de detecção de estrutura durante esse processo de identificação.

Nesse Capítulo, ferramentas de detecção de estruturas empregadas na identificação de modelos discretos, como o critério ERR apresentado no Capítulo 2, são usadas para criar um novo algoritmo de identificação de modelos contínuos.

O processo de identificação requer a utilização de diferentes algoritmos. A compreensão desses algoritmos, por sua vez, requer familiaridade com a representação matemática utilizada para descrever os sistemas a serem identificados.

O objetivo deste capítulo é, pois, apresentar ao leitor os algoritmos utilizados no procedimento de identificação, bem como demais detalhes a respeito do processo.

A Seção 3.1 contém uma visão geral do processo de identificação dos modelos contínuos e algumas considerações importantes sobre esse assunto. A Seção 3.2 descreve a coleta de dados, a Seção 3.3 detalha o cálculo de derivadas e a Seção 3.4 descreve o processo de identificação propriamente dito. Por fim, a Seção 3.5 fornece comentários finais do Capítulo.

### 3.1 Identificação de modelos contínuos — visão geral

Antes de proceder à descrição do procedimento de identificação é importante que algumas considerações sejam feitas a respeito das pressuposições sobre o

sistema a ser identificado.

Os modelos identificados neste trabalho são equações diferenciais ordinárias (EDO's). Assim, o método supõe que o sistema possa ser representado por equações da forma

$$\frac{d\mathbf{x}}{dt} \equiv \dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) , \quad (3.1)$$

sendo que  $\mathbf{x}(t) \in R^n$  é o vetor de estados,  $\mathbf{u}(t) \in R^r$  é o vetor de entradas e  $f(\cdot): R^{n+r} \rightarrow R^n$  é uma função possivelmente não linear<sup>1</sup>.

Se isso não for verdade, é provável que a identificação não seja bem sucedida. Por exemplo, se o sistema a ser identificado não for determinístico, ele com certeza não poderá ser representado pela equação (3.1). Um outro caso em que essa representação não é muito apropriada é quando o sistema em estudo é intrinsecamente de tempo discreto, como por exemplo uma bomba hidráulica de deslocamento positivo.

Uma necessidade importante do algoritmo usado para fazer a identificação é que  $f(\cdot)$  deve ser invariante no tempo, pelo mesmo motivo mencionado na Seção 2.2. É importante notar que a exigência de invariância é uma característica do algoritmo aqui apresentado e é decorrente tanto do método de mínimos quadrados convencional usado para a estimação de parâmetros quanto do cálculo do ERR usado na detecção de estrutura. Pode-se usar uma variante do método de mínimos quadrados chamada mínimos quadrados recursivo com fator de esquecimento (Aguirre, 2000a) para contornar a exigência de invariância na estimação de parâmetros. No entanto, o cálculo do ERR ainda exige invariância do sistema original.

Outro requisito é que um ou mais componentes de  $\mathbf{x}(t)$  têm que ser amostrados de forma eficaz, bem como todas as entradas. Se alguma entrada não for coletada, ela será tratada pelo algoritmo como uma perturbação. Se essa entrada for muito importante, isso pode comprometer todo o processo de identificação. Esse problema é comparável à identificação de um forno elétrico sem que se tenha coletado alguma grandeza referente à potência elétrica consumida. Dificilmente um bom modelo poderá ser obtido dessa forma. Na Seção 3.2 são fornecidos mais detalhes a respeito do processo de amostragem.

Estando essas condições verificadas procede-se à identificação. O ponto de partida são os dados coletados. Se nem todas as componentes de  $\mathbf{x}$  foram coletadas, o espaço original da equação (3.1) não pode ser reconstruído diretamente. Porém um espaço equivalente pode ser obtido através do uso

---

<sup>1</sup>Na apresentação do problema discreto, a função  $f(\cdot)$  era considerada como tendo apenas uma saída. No caso de funções com mais de uma saída, como é o caso aqui, a extensão é trivial. Basta considerar uma função de mais de uma saída como sendo composta de várias funções de uma só saída.

de derivadas dos sinais coletados. Essas derivadas têm de ser calculadas numericamente. A equação (3.1) fica então da forma

$$\dot{\mathbf{y}} = h(\mathbf{y}, \mathbf{u}) , \quad (3.2)$$

sendo  $\mathbf{y}$  a imersão de  $\mathbf{x}$  nesse novo espaço e  $h(\cdot) : R^{m+r} \rightarrow R^m$  uma função qualquer. Deve-se notar que a ordem desse novo sistema não é necessariamente a mesma daquela do sistema original.

A identificação propriamente dita consiste em encontrar uma função  $\hat{h}(\cdot)$  que aproxima  $h(\cdot)$ , levando em conta os dados coletados. Para isso, os vetores  $\dot{\mathbf{y}}$ ,  $\mathbf{y}$  e  $\mathbf{u}$  devem ser conhecidos para vários instantes de tempo. É importante notar que as derivadas das trajetórias  $\dot{\mathbf{y}}$  são necessárias. Assim, mesmo que todas as coordenadas originais  $\mathbf{x}$  tenham sido observadas, o cálculo numérico de derivadas ainda tem que ser feito. Mais detalhes sobre esse assunto são dados na Seção 3.2.

A função  $\hat{h}(\cdot)$  aplicada à mesma trajetória da equação (3.2) gera um vetor tangente a essa trajetória estimado  $\hat{\dot{\mathbf{y}}}$

$$\hat{\dot{\mathbf{y}}} = \hat{h}(\mathbf{y}, \mathbf{u}) . \quad (3.3)$$

O erro de estimação desse vetor tangente  $\boldsymbol{\xi} = \dot{\mathbf{y}} - \hat{\dot{\mathbf{y}}}$  é então utilizado na definição de uma função de custo  $J(\boldsymbol{\xi})$  qualquer. Essa função depende da função  $\hat{h}(\cdot)$ . Assim uma técnica de otimização irrestrita é utilizada para encontrar uma função  $\hat{h}^*(\cdot)$  ótima que minimiza  $J(\cdot)$ . Essa é a essência do processo de identificação.

### 3.1.1 Comparação com a identificação de modelos discretos

A identificação de modelos contínuos e a identificação de modelos discretos possuem várias características em comum, bem como diferenças significativas. A compreensão dessas diferenças e similaridades ajuda a contextualizar melhor a identificação de modelos contínuos dentro da área de identificação de sistemas.

Uma primeira semelhança entre as duas abordagens é a etapa do ajuste das funções. Do ponto de vista do processo de ajuste, os dois casos são completamente idênticos. Isso ocorre porque as funções ajustadas são estáticas. Assim elas não carregam na sua estrutura informação sobre a dinâmica do sistema a ser modelado. Nos dois casos, a dinâmica é obtida através da forma como os vetores de entrada e saída são construídos e não pela função em si. A forma de definição dos vetores de entrada e saída da função a ser ajustada é, portanto, a principal diferença dessas duas representações.

Os modelos discretos utilizam deslocamentos no tempo para obter as diferentes coordenadas dos vetores de entrada. Ou seja, as coordenadas do vetor de entrada  $\phi(k)$  em (2.2) são obtidas por deslocamentos dos vetores de dados coletados  $\mathbf{y}$  e  $\mathbf{u}$ . Essa operação de deslocamento não altera os dados sob nenhum aspecto. É dessa forma que se obtém o espaço de reconstrução dos modelos discretos. Tal espaço possui um parâmetro importante a ser determinado: o tempo de amostragem. Esse parâmetro é fundamental na qualidade da reconstrução (Casdagli et al., 1992; Aguirre, 1995).

Por sua vez, os modelos contínuos necessitam do cálculo de derivadas temporais dos sinais amostrados para gerar o espaço de reconstrução<sup>2</sup>. Assim, o espaço de reconstrução não tem uma dependência explícita do tempo de amostragem. Por outro lado, existe uma dificuldade maior nesse tipo de reconstrução do que na reconstrução por coordenadas de atraso. O cálculo das derivadas tem que ser feito numericamente. Dessa forma, no processo de obtenção dos vetores de entrada e saída que serão utilizados em (3.2) para estimar  $\hat{h}(\cdot)$  é introduzido um erro além dos ruídos de medição já presentes nos dados. Esse erro é decorrente do processo de estimação das derivadas dos dados coletados e ele é tão mais significativo quanto maior forem as ordens das derivadas a serem estimadas ou quanto pior for a relação sinal ruído dos dados. Como será mostrado na Seção 3.2, essas ordens podem ser elevadas.

Esse ruído extra acrescentado aos dados é uma dificuldade a mais do processo de identificação de modelos contínuos em relação à identificação de modelos discretos e é inerente à maneira como a identificação é feita. É importante comentar que os procedimentos — discreto e contínuo — aqui apresentados, não são as únicas formas de se fazer identificação. Por exemplo, em Timmer et al. (2000) é mostrada uma forma de obter modelos contínuos não lineares que não depende do cálculo de derivadas. Esse procedimento será comparado de forma crítica no Capítulo 6.

Uma outra semelhança importante entre os dois procedimentos é que se pode utilizar nos modelos contínuos quaisquer das representações não lineares que são normalmente utilizadas nos modelos discretos. Além disso, o procedimento de ajuste das funções é o mesmo tanto para modelos discretos quanto para modelos contínuos. Dessa forma é possível utilizar representações como redes neurais artificiais, RBF's, lógica nebulosa, modelos polinomiais, dentre outras. Isso é uma consequência direta do fato da função ajustada não ser dinâmica.

Uma outra diferença entre a representação discreta e a contínua, porém de caráter menos importante, tem a ver com a maneira pela qual os modelos identificados são simulados. Os modelos discretos são facilmente simulados

---

<sup>2</sup>Mais detalhes a respeito adiante.

através de laços tipo *for* de implementação trivial. Os modelos contínuos, por sua vez, precisam de uma ferramenta de integração numérica para poderem ser simulados já que, na maioria dos casos, não há solução fechada para problemas de valor inicial envolvendo EDO's não lineares. O usuário da identificação de modelos contínuos fica então obrigado a usar rotinas de integração de EDO's como o *Runge-Kutta*, por exemplo. Isso torna o processo de simulação bem mais complicado e demorado.

Finalmente, existe uma diferença com relação à modelagem de ruído e análise de resíduos. Tais procedimentos são usados na modelagem de sistemas discretos para contornar o problema de polarização e fazer a validação estatística respectivamente. No caso contínuo, tais procedimentos não podem ser empregados, pelo menos não de forma direta.

A seguir são dados os detalhes do processo de identificação de modelos contínuos.

## 3.2 Coleta de dados

A coleta de dados constitui o ponto de partida do processo de identificação. Esses dados são grandezas que devem ser amostradas do sistema real e que quantificam variáveis de interesse do sistema.

Uma possível forma de se fazer a coleta é amostrar todas as coordenadas de  $\mathbf{x}(t) = [x_1(t) \ x_2(t) \ \dots \ x_n(t)]^T$ , de forma a gerar  $n$  séries temporais  $\mathbf{x}(kT_s) = [x_1(kT_s) \ x_2(kT_s) \ \dots \ x_n(kT_s)]^T$ , sendo que  $k = 1, 2, \dots, N$ ,  $T_s$  é o período de amostragem e  $n$  é a ordem do sistema.  $N$  é o número de amostras por variável que foi coletado do sistema.

Dessa forma, não é necessário o uso do espaço alternativo da equação (3.2), porém as derivadas  $\dot{\mathbf{x}}$  ainda precisam ser calculadas numericamente.

Essa abordagem exige o conhecimento prévio da ordem e das variáveis de estado do sistema, bem como a sua disponibilidade para coleta. Isso nem sempre ocorre. Porém, se for esse o caso, essa abordagem é preferível do ponto de vista da identificação, pois facilita o processo e aumenta a qualidade dos modelos identificados.

Tal melhora acontece por dois motivos. Primeiramente, ao amostrar todas as variáveis do sistema, mais informação sobre o mesmo fica disponível para o algoritmo de identificação do que se menos variáveis fossem coletadas. Em segundo lugar, apenas derivadas de primeira ordem precisam ser calculadas.

Infelizmente, essa situação ideal é rara. Às vezes, a ordem do sistema a ser identificado não é conhecida e, tipicamente, uma ou mais variáveis de estado não estão disponíveis para coleta.

Nesse caso, quem está fazendo a identificação deve coletar tantas variáveis

quanto for possível. Deve-se também estimar a ordem do sistema ou escolher uma ordem para o modelo a ser identificado.

As coordenadas que faltam podem, então, ser substituídas por derivadas das coordenadas amostradas, calculadas numericamente. Esse processo é chamado de *reconstrução de espaço de estados* e é definido como a criação de um espaço de estados determinístico multidimensional a partir de uma série temporal de dimensão menor (Gibson et al., 1992). Dessa forma está sendo criado um espaço de imersão para as trajetórias diferente do espaço de estados original, mas que é topologicamente equivalente (Takens, 1980) e capaz de preservar características do sistema original como autovalores de pontos fixos, desde que a dimensão do espaço de reconstrução seja grande o bastante (Takens, 1980). Se o sistema original tem dimensão  $n$ , um espaço de reconstrução de dimensões  $2n + 1$  é suficiente para uma reconstrução bem sucedida.

Por exemplo, suponha-se que se deseja um modelo de terceira ordem e somente duas variáveis,  $x_1$  e  $x_2$ , são observáveis no sistema original. Pode-se derivar uma das variáveis amostradas e usar essa derivada como uma outra coordenada, como mostrado na equação (3.4).

$$\begin{aligned} \frac{dx_1}{dt} &= x_1' , \\ \frac{dx_2}{dt} &= h_1(x_1, x_2, x_1', \mathbf{u}) , \\ \frac{dx_1'}{dt} &= h_2(x_1, x_2, x_1', \mathbf{u}) . \end{aligned} \tag{3.4}$$

Comparando a equação (3.4) com a equação (3.2) tem-se  $\mathbf{y} = [x_1 \ x_2 \ x_1']^T$  e  $g(\mathbf{y}, \mathbf{u}) = [y_3 \ h_1(\cdot) \ h_2(\cdot)]^T$ .

Um caso extremo e bastante comum ocorre quando apenas uma grandeza  $x_i(t)$  (qualquer uma) é medida, gerando a série  $x_i(kT_s)$ . Essa grandeza é chamada “observável” (Gouesbet e Letellier, 1994) e esse caso, de identificação monovariável. Os casos anteriores são chamados identificação multivariável. A escolha da variável observável é crítica no processo de modelagem, conforme discutido em Letellier et al. (1998).

Na identificação monovariável apenas a coordenada  $x_i(t)$  de  $\mathbf{x}(t)$  é medi-

da. Então o seguinte conjunto alternativo de coordenadas

$$\begin{aligned} y_1 &= x_i , \\ \frac{dy_1}{dt} &= y_2 , \\ &\vdots \\ \frac{dy_{m-1}}{dt} &= y_m , \\ \frac{dy_m}{dt} &= g(y_1, y_2, \dots, y_m, \mathbf{u}) \end{aligned} \tag{3.5}$$

é utilizado para representar o sistema. Nota-se que a ordem do modelo identificado,  $m$ , não é necessariamente igual a ordem do sistema original  $n$ .

Além da relativamente baixa quantidade de informação existente em apenas uma variável, a abordagem monovariável ainda tem um outro problema. As derivadas de (3.5) têm que ser calculadas numericamente. A determinação precisa de derivadas de alta ordem de sinais reais é um difícil problema mesmo em séries temporais com uma alta relação sinal-ruído.

No entanto, quer seja a identificação mono ou multivariável, é importante que os dados coletados contenham informações sobre os diversos regimes dinâmicos apresentados pelo sistema (Rodrigues et al., 1996). Para isso, é necessário que os estados estejam evoluindo através de um grande intervalo de amplitudes e com um amplo espectro de frequências. É também imprescindível que *todas* as entradas  $\mathbf{u}$  sejam coletadas.

Devido ao fato de que serão calculadas as derivadas dos dados, a frequência de amostragem deve ser bem maior do que a requerida pelo teorema da amostragem. Por exemplo, no caso de sistemas com oscilações caóticas, em várias situações, percebeu-se que o período de amostragem  $T_s$  deve ser tal que se tenha aproximadamente 100 amostras por pseudo-período. O pseudo-período é o tempo correspondente a uma volta em torno de um ponto fixo.

### 3.3 Cálculo das derivadas

A seguir, será discutida uma forma de calcular as derivadas  $\dot{\mathbf{x}}(t)$  ou  $\dot{\mathbf{y}}(t)$  numericamente. Outras formas têm sido publicadas recentemente na literatura (Lainscsek et al., 1998). O procedimento usado nesse trabalho é o seguinte.

Seja uma componente amostrada de  $\mathbf{x}(t)$  dada por<sup>3</sup>

$$x_i(kT_s) \equiv x_i(k); \quad k = 1, 2, \dots, N , \tag{3.6}$$

---

<sup>3</sup>No caso da identificação multivariável esse procedimento é repetido para todas as coordenadas amostradas de  $\mathbf{x}(t)$ . Já no caso monovariável, é feito apenas na única coordenada amostrada.

sendo que  $T_s$  é o período de amostragem e  $N$  é o número de observações.

Em seguida, escolhe-se um intervalo de largura  $L$  no início da série temporal, ou seja,  $x_i(k)$  com  $1 \leq k \leq L$ . Nesse intervalo aproxima-se  $x_i(k)$  por um polinômio algébrico  $P(k)$  tal que <sup>4</sup>

$$x_i(k) \approx P(k) = a_0 + a_1 k^1 + a_2 k^2 + \cdots + a_\gamma k^\gamma, \quad (3.7)$$

sendo  $\gamma$  o grau de  $P(k)$ .

Escrevendo (3.7) para  $k = 1, \dots, L$  tem-se

$$\begin{aligned} x_i(1) &= a_0 + a_1 1^1 + \cdots + a_\gamma 1^\gamma \\ x_i(2) &= a_0 + a_1 2^1 + \cdots + a_\gamma 2^\gamma \\ &\vdots \\ x_i(L) &= a_0 + a_1 L^1 + \cdots + a_\gamma L^\gamma \end{aligned} \quad (3.8)$$

que, na forma matricial, pode ser representado como sendo:

$$\mathbf{x}_i = \lambda \mathbf{a}. \quad (3.9)$$

Deve-se escolher  $L$  tal que  $L > \gamma + 1$  para que um algoritmo de mínimos quadrados possa ser usado em (3.8) para calcular os  $a_j$ . A escolha de  $L$  também depende dos dados<sup>5</sup>. Depois de alguns testes, percebeu-se que  $L = 3$  ou  $L = 5$  é uma boa escolha em muitos sistemas, porém a escolha de  $L$  sempre requer alguma experimentação.

Existem várias formas de se resolver (3.9) e obter os coeficientes estimados  $\hat{\mathbf{a}}$  (Golub e Van Loan, 1989). A maneira mais simples é através da solução da equação normal, o que leva a

$$\hat{\mathbf{a}} = [\lambda^T \lambda]^{-1} \lambda^T \mathbf{x}_i. \quad (3.10)$$

Se a matriz  $\lambda \in R^{L \times \gamma+1}$  é mal-condicionada, a solução do problema através de (3.10) pode-se tornar inviável. Uma alternativa é a utilização de algoritmos de mínimos quadrados ortogonais (Golub e Van Loan, 1989).

Neste trabalho, a aproximação polinomial foi feita no MATLAB ou no GNU Octave, utilizando a função *polyfit*. Essa função monta a matriz  $\lambda$  e resolve o problema de mínimos quadrados utilizando um algoritmo ortogonal (a fatoração *QR*).

---

<sup>4</sup>Esse polinômio não tem nada a ver com o polinômio dinâmico  $\hat{h}(\cdot)$  que será visto a seguir.

<sup>5</sup>Depende, na verdade, da relação entre  $T_s$  e o pseudo-período do sistema. Quanto menor for  $T_s$  em relação a este, maior deve ser  $L$  e vice versa. Deve-se também aumentar  $L$  se houver mais ruído nos dados.

As derivadas de  $x_i$  podem ser computadas derivando  $P(k)$  analiticamente e calculando essas últimas no centro  $k_c$  do intervalo (logo  $L$  deve ser ímpar). No caso da identificação multivariável em que todas as coordenadas foram amostradas apenas a primeira derivada é necessária e, portanto,  $\gamma = 1$ . No caso monovariável,  $d$  deve ser escolhido tal que  $\gamma \geq m$ , sendo  $m$  a ordem do modelo a ser identificado. Aqui é necessária alguma experimentação para determinar o valor mais adequado de  $\gamma$  para um certo conjunto de dados, porém  $\gamma$  não deve exceder  $m$  por mais que uma ou duas unidades.

A derivada de  $x_i(k)$  no ponto  $k_c$  é então dada por:

$$\dot{x}_i(k_c) \approx \left. \frac{dP(k)}{dk} \right|_{k=k_c}. \quad (3.11)$$

A seguir, desloca-se o intervalo inteiro de um período de amostragem, de modo que o próximo centro de intervalo seja em  $k_c + 1$ , e repete-se o processo até que toda a série tenha sido “varrida”, exceto por  $(L - 1)/2$  pontos no início e no final da mesma.

A equação (3.11) supõe que o período de amostragem é unitário. Se isso não for verdade, é necessário dividir a derivada pelo período de amostragem para obter o valor correto (dividir a derivada segunda pelo quadrado do período de amostragem; a derivada terceira pelo cubo e assim por diante).

O cálculo das derivadas feito dessa maneira é menos sensível à presença de ruído nos dados. Em Oliveira (1998) essa filosofia também foi empregada com resultados satisfatórios. Porém, dependendo do nível de ruído, é necessária ainda uma pré-filtragem dos dados, que pode ser feita com um filtro digital. A frequência de corte desse filtro deve ser algumas vezes maior que a frequência da oscilação mais rápida do sistema (de 5 a 10 vezes).

Nesse trabalho, os filtros digitais usados foram calculados com a função *butter* do MATLAB e a filtragem foi feita usando a função *filtfilt*.

É importante ressaltar que esse procedimento é uma das maneiras de se calcular derivadas de sinais reais. Em Lainscsek et al. (1998) as derivadas são calculadas por decomposição em valores singulares de uma matriz cujas colunas são formadas pelo sinal atrasado várias vezes. Em Gibson et al. (1992) as derivadas são calculadas por um filtro linear baseado no que os autores chamam de polinômio de Legendre discretos. Tal método foi usado com sucesso em Letellier et al. (1995) para identificar um modelo contínuo para um experimento eletroquímico caótico.

### 3.4 Identificação de modelos contínuos

Conforme visto na Seção 3.1, a identificação de modelos contínuos é muito semelhante à identificação de modelos discretos em certos aspectos. A identificação propriamente dita consiste no ajuste de uma função aos dados e isso é feito da mesma forma tanto para modelos contínuos quanto para modelos discretos, ficando a principal diferença na forma como os dados são formatados nesse ajuste de função.

A identificação de modelos contínuos pode ser tanto monovariável quanto multivariável, dependendo se foram coletadas uma ou mais variáveis de estado. No entanto, sempre existe uma “unidade básica”, idêntica nos dois casos, sobre a qual são feitos os mesmos procedimentos para se fazer a identificação. Essa “unidade básica” da identificação é sempre da forma

$$Y_j = h_j[\varphi(t)] , \quad (3.12)$$

sendo  $Y_j$  uma derivada temporal de uma variável de estado e  $h_j(\cdot)$  uma função que se deseja ajustar.  $\varphi(t)$  é o vetor de entrada desta função. Ele é formado pelas coordenadas alternativas de (3.2) e o vetor de entradas  $\mathbf{u}$  (compare com  $\phi(k)$  em (2.2)):

$$\varphi(t) = \begin{bmatrix} y_1(t) \\ y_2(t) \\ \vdots \\ y_m(t) \\ \mathbf{u}(t) \end{bmatrix} . \quad (3.13)$$

A grandeza  $Y_j$  pode ser uma derivada de qualquer ordem, dependendo do tipo do modelo identificado. Por exemplo, no caso monovariável, ela vale  $\dot{y}_m$ , que é uma derivada de ordem  $m + 1$ . No caso multivariável em que todas as variáveis de estado foram coletadas,  $Y_j$  é uma derivada temporal de primeira ordem de uma das variáveis de estado observadas.

O número de funções da forma de (3.12) a serem ajustadas é igual ao número de variáveis de estado coletadas no sistema. Essa é a única diferença entre a identificação monovariável e a multivariável no que se refere ao procedimento de ajuste. Quando apenas uma variável foi amostrada (caso monovariável), apenas uma função deve ser ajustada. Na equação (3.4) pode-se ver o caso em que foram observadas duas variáveis de estado e duas funções,  $h_1(\cdot)$  e  $h_2(\cdot)$ , precisam ser ajustadas. Essas funções têm sempre  $q$  entradas e uma saída, ou seja, são da forma de  $f(\cdot)$  em (2.1), porém com entradas diferentes. Assim, dado  $Y_j$  e  $\varphi(t)$ , o problema de ajuste de função é idêntico àquele mostrado na Seção 2.1.

Diversas representações poderiam ser utilizadas para aproximar  $h_j(\cdot)$ . Nesse trabalho, optou-se por utilizar uma aproximação polinomial. Essa mesma representação foi empregada em Gouesbet e Letellier (1994), porém sem usar nenhum tipo de detecção de estrutura. Nesse artigo, os autores utilizam uma representação polinomial para a função a ser ajustada, cálculo numérico das derivadas e estimação de parâmetros via mínimos quadrados. Exatamente a mesma abordagem dessa Dissertação. Porém, a escolha dos termos não tem nenhuma relação com o sistema a ser identificado. Os termos são incluídos em ordem crescente de grau.

A diferença do presente trabalho em relação a Gouesbet e Letellier (1994) é, portanto, a utilização de técnicas de detecção de estrutura.

A representação polinomial usada nesse trabalho na identificação de modelos contínuos é a mesma daquela utilizada na identificação de modelos NARMAX polinomiais. Assim, a notação empregada aqui é a mesma da Seção 2.2, a menos da substituição de  $k$  por  $t$  para indicar tempo contínuo e de algumas outras diferenças.

A função usada na aproximação tem a forma

$$g[\varphi(t)] = \sum_{i=1}^p \theta_i \eta_i(t) , \quad (3.14)$$

sendo  $\theta_i$  os parâmetros,  $p$  o número de termos e  $\eta_i$  os termos. Esses termos, por sua vez, têm a forma

$$\eta_i(k) = \prod_{j=1}^q a_j^{\beta_{ij}}(t) , \quad (3.15)$$

sendo  $q$  o número de elementos de  $\varphi(t)$  e  $a_j(t)$  o  $j$ -ésimo elemento de  $\varphi(t)$ . Os expoentes  $\beta_{ij}$  são números naturais. A quantidade  $d = \sum_{j=1}^q \beta_{ij}$  é o grau do termo  $i$ . Repare que as equações (3.14) e (3.15) são análogas às equações (2.5) e (2.6). A diferença está nos  $a_j$ . Enquanto no modelo discreto os  $\alpha_j$  são sinais deslocados no tempo, os  $a_j$  são derivadas temporais de sinais, todos calculados no mesmo instante, conforme a equação (3.13).

Novamente, o ajuste da função  $g(\cdot)$  pode ser dividido em duas etapas: a estimação de parâmetros e a detecção de estrutura.

Estas etapas estão detalhadas a seguir.

### 3.4.1 Estimação de parâmetros

A estimação de parâmetros dos modelos contínuos é feita aqui de forma idêntica à estimação de parâmetros dos modelos NARMAX polinomiais.

Antes da estimação, os dados de entrada e saída (ou somente de saída se o sistema for autônomo) precisam ser coletados. Derivadas desses dados devem ser, então, calculadas numericamente. A ordem dessas derivadas é determinada pela ordem do modelo e pelo seu tipo, mono ou multivariável.

Feito isso, prossegue-se à estimação. De novo, utiliza-se suposição que o sistema original é invariante no tempo para gerar restrições diferentes pela amostragem continuada ao longo do tempo.

Substituindo (3.14) em (3.12) e escrevendo para  $N$  instantes de amostragem tem-se

$$\begin{cases} Y(1) = \sum_{i=1}^p \theta_i \eta_i(1) \\ Y(2) = \sum_{i=1}^p \theta_i \eta_i(2) \\ \vdots \\ Y(N) = \sum_{i=1}^p \theta_i \eta_i(N) \end{cases}, \quad (3.16)$$

sendo que o subscrito  $j$  em  $Y$  foi omitido para simplificar a notação. Porém, deve-se ter sempre em mente que esse processo deve ser repetido para todas as “unidades básicas” (da forma de (3.12)) presentes no modelo.

Esses instantes de amostragem usados na identificação são chamados *centros*<sup>6</sup>. A equação (3.16) em forma matricial fica

$$\begin{bmatrix} Y(1) \\ Y(2) \\ \vdots \\ Y(N) \end{bmatrix} = \begin{bmatrix} \eta_1(1) & \eta_2(1) & \cdots & \eta_p(1) \\ \eta_1(2) & \eta_2(2) & \cdots & \eta_p(2) \\ \vdots & \vdots & \ddots & \vdots \\ \eta_1(N) & \eta_2(N) & \cdots & \eta_p(N) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_p \end{bmatrix} \quad (3.17)$$

ou

$$\mathbf{Y} = \mathbf{X}\boldsymbol{\theta}. \quad (3.18)$$

A matriz  $X$  é chamada de matriz de regressores.

Repare que, ao contrário dos modelos NARMAX polinomiais, as colunas de  $X$  referem-se ao mesmo instante de tempo que  $\mathbf{Y}$ , ou seja, cada linha de  $X$  e o correspondente elemento de  $\mathbf{Y}$  possuem valores referentes ao mesmo instante de tempo. Isto é uma consequência da definição de  $\varphi(t)$ , que utiliza derivadas em vez de deslocamentos no tempo para criar diferentes coordenadas.

O erro de identificação é definido de forma semelhante à equação (2.11):

$$\mathbf{e} = \mathbf{Y} - \mathbf{X}\hat{\boldsymbol{\theta}}. \quad (3.19)$$

---

<sup>6</sup>Esses centros não são a mesma coisa que os centros das RBF's. Esses últimos são na verdade parâmetros a serem determinados enquanto os primeiros são instantes de amostragem. Porém, devido à semelhança entre eles recebem os mesmos nomes.

Apesar do erro definido na equação (3.19) ter a mesma forma do erro definido na equação (2.11), essas duas quantidades são conceitualmente diferentes. O erro usado na identificação dos modelos NARMAX polinomiais é chamado erro de *predição de um passo à frente*, pois ele quantifica o erro de predição um passo à frente do sinal de saída  $y(k)$ . O erro definido na equação (3.19) *não* é de predição. Ele é o erro de modelagem do sinal  $Y(t)$ . Esse sinal é uma derivada de um sinal amostrado, calculada numericamente. A ordem dessa derivada pode, inclusive, ser elevada. Portanto, o que se minimiza no processo de estimação de parâmetros na identificação dos modelos contínuos é o erro de modelagem de um sinal que foi estimado através de cálculo numérico de derivadas.

A função de custo  $J(\mathbf{e})$  utilizada nesse trabalho é o somatório dos erros quadráticos definida na equação (2.4).

Uma vez definida essa função de custo o problema de estimação de parâmetros se torna um problema de mínimos quadrados. Uma forma de resolvê-lo é resolvendo-se

$$\hat{\boldsymbol{\theta}} = [X^T X]^{-1} X^T \mathbf{Y} . \quad (3.20)$$

Porém, essa forma é sensível a mal condicionamento numérico. Nesse trabalho, é utilizada a fatoração Householder (Golub e Van Loan, 1989), que é um método de ortogonalização. Além do motivo numérico para a utilização da fatoração householder, existe um outro. O cálculo do ERR é bem mais eficiente por um algoritmo ortogonal quando é usado o esquema da regressão direta.

A estimação de parâmetros requer que a estrutura do modelo esteja determinada. Para isso devem-se usar técnicas de detecção de estrutura.

### 3.4.2 Detecção de estrutura

O objetivo da detecção de estrutura é determinar quais termos incluir no modelo de forma a obter uma boa representação para o sistema original.

Dois tipos de detecção de estrutura são utilizados nesse trabalho. O primeiro é baseado no critério ERR. Nessa abordagem não é utilizada nenhuma informação do sistema original a não ser os dados amostrados. Esse tipo de abordagem é chamada de caixa preta. O contrário, a abordagem caixa branca, ocorre quando o modelo vem inteiramente da física do processo. Entre esses dois extremos existe o amplo domínio, ainda pouco explorado, da modelagem caixa cinza (Corrêa, 2001).

Na abordagem caixa preta, os modelos são obtidos baseado nos dados coletados. Porém, uma eventual informação a respeito da natureza do sistema

a ser identificado é utilizada na confecção do modelo. Esse é o segundo tipo de detecção empregado aqui.

### Detecção de estrutura baseada no ERR

O ERR, conforme explicado na Seção 2.2.2, é uma medida da importância individual de cada termo. Com isso é possível ordenar os termos candidatos em ordem decrescente de importância e obter um modelo parcimônio capaz de representar o sistema.

Partindo da equação (3.18) escreve-se a equação auxiliar

$$\mathbf{Y} = W\hat{\mathbf{g}} + \mathbf{e} , \quad (3.21)$$

sendo  $W$  uma matriz cujas colunas  $\mathbf{w}_i$  (termos) são ortogonais sobre os dados e  $\hat{\mathbf{g}}$  é o vetor de parâmetros estimados por mínimos quadrados no espaço da matriz  $W$ . O vetor  $\mathbf{e}$  é o vetor de resíduos.

Dessa forma, a quantidade  $\mathbf{Y}^T\mathbf{Y}$  é dada por

$$\begin{aligned} \mathbf{Y}^T\mathbf{Y} &= (W\hat{\mathbf{g}} + \mathbf{e})^T(W\hat{\mathbf{g}} + \mathbf{e}) \\ &= \hat{\mathbf{g}}^T W^T W \hat{\mathbf{g}} + \hat{\mathbf{g}}^T W^T \mathbf{e} + \mathbf{e}^T W \hat{\mathbf{g}} + \mathbf{e}^T \mathbf{e} \\ &= \sum_{i=1}^p \hat{g}_i^2 \mathbf{w}_i^T \mathbf{w}_i + \mathbf{e}^T \mathbf{e} . \end{aligned} \quad (3.22)$$

No último passo de (3.22), foi usada a propriedade do estimador de mínimos quadrados gerar resíduos ortogonais aos regressores  $\mathbf{w}_i$  e a ortogonalidade de  $W$ .

Pela equação (3.22), pode-se notar que a soma dos valores quadráticos de  $\mathbf{Y}$  pode ser decomposta em uma parte que envolve os regressores ortogonais e uma parte que não é explicada pelo modelo, referente aos resíduos. A parte que envolve os regressores é, por sua vez, o resultado da soma da contribuição individual de cada regressor ortogonal. Uma outra forma de interpretar essa equação é

$$\mathbf{e}^T \mathbf{e} = \mathbf{Y}^T \mathbf{Y} - \sum_{i=1}^p \hat{g}_i^2 \mathbf{w}_i^T \mathbf{w}_i . \quad (3.23)$$

Aqui fica claro que os resíduos são uma função dos regressores e que a inclusão de um determinado regressor leva a uma diminuição no somatório dos valores quadráticos dos resíduos.

Assim, pode-se definir a taxa de redução de erro como

$$[\text{ERR}]_i = \frac{\hat{g}_i^2 \mathbf{w}_i^T \mathbf{w}_i}{\mathbf{Y}^T \mathbf{Y}} , \quad (3.24)$$

sendo  $[\text{ERR}]_i$  a taxa de redução de erro do  $i$ -ésimo regressor (Chen et al., 1989). Os parâmetros estimados são dados por

$$\hat{g}_i = \frac{\mathbf{w}_i^T \mathbf{Y}}{\mathbf{w}_i^T \mathbf{w}_i} . \quad (3.25)$$

Na Seção 2.2.2 foi mencionado que o valor da taxa de redução de erro de um termo em particular depende dos outros termos no modelo. Isso não é evidente na equação (3.24) porque ela está escrita numa base ortogonal, diferente daquela que contém os termos originais. Porém, para se obter a matriz  $W$  a partir da matriz de regressores original  $X$  é necessário utilizar um procedimento de ortogonalização e este, necessariamente, depende de todas as colunas de  $X$ .

Para reduzir esse efeito é utilizado o procedimento de regressão direta. Esse algoritmo é descrito a seguir.

Inicialmente, tem-se um conjunto de termos candidatos que forma a matriz de regressores candidatos  $X_c$ . O objetivo é obter uma matriz  $X$  com os melhores  $p$  regressores, segundo o critério ERR.

1. No início, a matriz  $X$  não possui nenhum regressor, assim como  $W$ . Fazer  $i = 1$ .
2. Ortogonalizar a primeira coluna de  $X_c$  em relação à matriz  $W$  gerando o regressor  $\mathbf{w}_i$ . Calcular o ERR de  $\mathbf{w}_i$  usando (ver (3.24) e (3.25))

$$\text{ERR} = \frac{\mathbf{w}_i^T \mathbf{Y}}{\mathbf{w}_i^T \mathbf{w}_i \mathbf{Y}^T \mathbf{Y}} . \quad (3.26)$$

*Retirar*  $\mathbf{w}_i$  de  $W$  e ortogonalizar a segunda coluna de  $X_c$  gerando um novo regressor  $\mathbf{w}_i$ . Calcular o ERR para esse novo regressor. Repetir esse processo para as demais colunas de  $X_c$ .

3. Escolher a coluna de  $X_c$  que, no passo anterior, obteve o maior ERR. Acrescentar essa coluna a  $X$ . Ela será o regressor  $\mathbf{x}_i$  do modelo a ser identificado. *Retirar* essa coluna de  $X_c$ . Ortogonalizar essa coluna em relação a  $W$  gerando  $\mathbf{w}_i$ . Fazer  $i = i + 1$ .
4. Repetir os passos 2 e 3 até  $i = p$ .

Ao final desse procedimento, ter-se-á uma matriz  $X$  com os  $p$  melhores termos dentre os termos candidatos, segundo o ERR. Para se obter o modelo final resta somente calcular os coeficientes conforme explicado na Seção 3.4.1.

Todo este processo requer a escolha de um conjunto de termos candidatos. Uma forma simples de se fazer isso e que é empregada nesse trabalho é a

limitação do grau máximo que os termos candidatos podem ter. Uma vez determinado esse grau, escolhem-se todos os termos possíveis que apresentam grau menor ou igual a esse limite.

Também é possível usar alguma informação disponível sobre o sistema para determinar parte do conjunto de termos candidatos, como é mostrado a seguir.

### Detecção de estrutura baseada em informação *a priori*

Algumas vezes, além dos dados de identificação, outras informações a respeito do sistema a ser identificado estão disponíveis. Estas informações poderão ser ou não usadas na identificação, dependendo da natureza das mesmas e da representação utilizada na identificação. Os modelos contínuos polinomiais monovariáveis utilizados nesse trabalho são particularmente adequados para incorporação de informação a respeito de pontos fixos.

Pontos fixos ou pontos de equilíbrio são os estados para os quais a função  $g(\cdot)$  em (3.2) se anula. Se o sistema for autônomo, os pontos fixos serão pontos no espaço de estados. Se o sistema apresentar entradas, os pontos fixos passam a ser funções também das entradas.

Os pontos fixos de modelos contínuos polinomiais autônomos monovariáveis são facilmente determinados.

Considere a equação (3.5) na situação sem entradas. Para encontrar os pontos fixos desse modelo deve-se fazer  $\dot{\mathbf{y}} = g(\mathbf{y}) = \mathbf{0}$  em (3.2). Isso implica que  $y_i = 0$  para  $i = 2, \dots, m$ . Também implica que

$$g(y_1, 0, 0, \dots, 0) = 0 . \quad (3.27)$$

Como  $g(\cdot)$  é uma função polinomial no modelo identificado, isso leva a

$$\sum_{j=0}^{d_m} \theta_{i(j)} y_1^j = 0 , \quad (3.28)$$

sendo os  $i(j)$  os índices dos termos que não apresentam nenhuma das coordenadas  $y_i$ ,  $\forall i \neq 1$ , elevadas a expoentes não nulos e o  $d_m$  o maior expoente de  $y_1$  nestes termos. Assim, as coordenadas dos pontos fixos no espaço reconstruído  $\mathbf{y}_{\text{pf}}$  são da forma

$$\mathbf{y}_{\text{pf}} = \begin{bmatrix} y_i^* \\ 0 \\ \vdots \\ 0 \end{bmatrix} , \quad (3.29)$$

sendo  $y_i^*$  as raízes do polinômio definido em (3.28).

Dessa forma, é possível forçar o modelo identificado a ter uma certa estrutura de pontos fixos baseado em informações *a priori*. Por exemplo, se o número de pontos fixos do sistema original for conhecido, pode-se garantir esse mesmo número de pontos fixos no modelo identificado apenas fazendo com que este tenha termos do tipo  $\theta_i y_1^{\beta_i}$  com  $\beta$  variando de zero até, no máximo, o número de pontos fixos desejado.

Pode-se também incluir informações a respeito da simetria dos pontos fixos.

As informações sobre os pontos fixos podem vir de algum conhecimento que se tenha sobre o sistema ou experimentalmente, como proposto em Aguirre e Souza (1998), onde é mostrado um algoritmo para estimação de pontos fixos a partir de séries temporais.

Além de facilitar a incorporação de informação a respeito de pontos fixos, a representação polinomial permite que se inclua informação a respeito de uma eventual simetria global do sistema. Por exemplo, se o sistema original (3.1) (autônomo) apresentar a seguinte simetria

$$f(-\mathbf{x}) = -f(\mathbf{x}) \quad (3.30)$$

então um modelo reconstruído a partir de um observável também deveria apresentar a mesma simetria.

Para garantir isso, basta incluir no conjunto de termos candidatos apenas termos de grau ímpar.

De qualquer forma, o uso de informação *a priori* na detecção de estrutura ajuda significativamente o processo de identificação, pois faz com que a estrutura já presente no conjunto de termos candidatos seja mais próxima de uma estrutura ideal, facilitando o trabalho do ERR na determinação da forma final do modelo. Sempre que houver informação extra sobre o sistema original, esta deve ser incluída, se possível, no modelo a ser identificado. Isso será ilustrado no Capítulo 4.

### 3.4.3 Validação

A validação é a etapa na qual a qualidade dos modelos identificados é aferida. Nela são descartados os modelos que não foram capazes de adequadamente representar o sistema original. É importante observar que a qualidade do modelo é um conceito um tanto relativo, pois depende do objetivo para o qual o modelo se destina. Idealmente, o modelo deveria ser capaz de se comportar exatamente como o sistema em estudo em quaisquer situações, mas isso não é viável. Assim, durante a identificação, deve-se ter sempre em

mente o uso a que se destina o modelo. Um bom modelo para predição pode não ser tão bom quando utilizado para controle e, por sua vez, um péssimo modelo para predição pode ser suficiente para a extração de informação do sistema (Aguirre e Souza, 1998).

Ainda assim, um emprego muito freqüente para os modelos é a predição, ou seja, usar o modelo para saber como o sistema se comportaria frente a determinadas condições. Nesse caso, deseja-se que o modelo tenha um comportamento temporal o mais próximo possível daquele do sistema original.

Uma das formas mais simples de se fazer a validação nesse sentido é através da comparação entre o sinal coletado do sistema real e o sinal resultante da simulação do modelo identificado. Se o modelo acompanhar o sistema, a menos de um erro pequeno, então o modelo é bom. Preferencialmente, essa comparação é feita com dados diferentes daqueles usados para identificar o modelo, pois é possível que o modelo identificado capture alguma característica particular do conjunto de dados de identificação, como ruído, por exemplo (Aguirre, 2000a).

Porém, se o sistema a ser identificado é caótico (Gleick, 1991), a comparação do sinal simulado com o sinal oriundo do modelo não é capaz de dizer se um modelo é bom ou não. Isso ocorre devido às características desse tipo de sistema.

Uma trajetória simulada de um modelo irá divergir de uma trajetória real de um sistema caótico, mesmo que elas tenham começado no mesmo ponto e independentemente da qualidade do modelo identificado. Isso torna necessária a utilização de outras técnicas de validação que não a simples comparação de séries temporais.

Uma alternativa simples é a comparação visual do atrator estranho do sistema com o atrator do modelo identificado. O atrator estranho é uma característica do sistema caótico. O atrator é o lugar geométrico, no espaço de estados, para o qual as trajetórias tendem quando o tempo tende ao infinito. Ele é uma característica de regime permanente e independente das trajetórias em si. Dessa forma, mesmo que as séries temporais sejam diferentes, é de se esperar que o atrator do modelo e o atrator do sistema sejam parecidos se o modelo for bom.

Comparar visualmente dois atratores traz uma certa subjetividade ao processo de validação. Em vez disso, pode-se comparar medidas objetivas desses atratores, como a dimensão de correlação. Pode-se também comparar grandezas invariantes, como expoentes de Lyapunov (Aguirre e Billings, 1994). Por fim, quando uma validação mais cuidadosa for necessária, pode-se comparar mapas de primeiro retorno em seções de Poincaré e populações de órbitas quase-periódicas (Letellier et al., 1995). Porém, esses procedimentos de validação são de utilização mais complexa e requerem do usuário algum

conhecimento de caracterização topológica de atratores.

De qualquer forma, apesar da validação de modelos caóticos ser mais complicada, o uso desses sistemas para verificar o desempenho da técnica de identificação é proveitoso. Obter modelos que recuperem, mesmo que ligeiramente, um atrator estranho não é tarefa fácil. Assim, a identificação de tais sistemas será estudada no Capítulo 4 como uma forma de teste do procedimento de identificação.

### 3.5 Conclusão

Neste Capítulo foi apresentado um procedimento de identificação de modelos contínuos não lineares. Esse procedimento foi comparado com um procedimento semelhante usado na identificação de sistemas de tempo discreto, evidenciando os pontos diferentes dessas duas abordagens.

A característica mais importante do algoritmo de identificação proposto é a utilização de técnicas de detecção de estrutura nesse tipo de modelo. Isso é abordado pela primeira vez nesse trabalho.

A detecção de estrutura é feita por dois métodos. Um, totalmente caixa preta, utiliza um critério estatístico chamado ERR para ordenar os termos de acordo com a sua importância. O outro se baseia em informações *a priori* para incluir ou retirar termos do modelo ou do conjunto de termos candidatos.

# Capítulo 4

## Identificação de sistemas caóticos

No Capítulo 3 o procedimento de identificação de modelos contínuos não lineares foi apresentado e descrito. No entanto, nenhum exemplo de aplicação foi mostrado.

O objetivo do presente Capítulo é, pois, fornecer um exemplo de aplicação da técnica desenvolvida no Capítulo 3. Esse exemplo, além de ilustrar a aplicação do procedimento de identificação, também serve como uma forma de teste para o algoritmo.

Os sistemas utilizados nesse Capítulo foram escolhidos de forma a ilustrar bem um caso de identificação em que os métodos apresentados são bem aplicáveis: os sistemas caóticos (Gleick, 1991). Uma das principais características de tais sistemas é a sua não linearidade. Um sistema totalmente linear não é capaz de exibir comportamento de caos determinístico. Uma outra vantagem do uso desse tipo de sistema para testar o procedimento de identificação é a complexidade do seu comportamento dinâmico, que faz com que o teste seja mais exigente do que com sistemas mais simples.

Este Capítulo está organizado da seguinte forma. Na Seção 4.1 estão alguns comentários sobre a identificação de sistemas caóticos. A Seção 4.2 mostra a identificação do *sistema de Lorenz* utilizando dados simulados. A Seção 4.3 ilustra o caso da identificação de dados reais gerados pelo *circuito de Chua*. Uma conclusão do Capítulo se encontra na Seção 4.4.

### 4.1 Sistemas caóticos e identificação

Os sistemas caóticos são um tópico de estudo relativamente recente. Um dos primeiros trabalhos na área foi Lorenz (1963). Um sistema caótico pode

apresentar um comportamento dinâmico muito complexo, aperiódico. Uma outra característica é a chamada *sensibilidade às condições iniciais*: a trajetória de um sistema caótico iniciada em um determinado ponto diverge exponencialmente de uma outra trajetória iniciada em um ponto próximo, porém distinto, do anterior.

O comportamento caótico foi observado em diversas situações práticas como observa Gleick (1991):

Agora que a ciência está atenta, o caos parece estar por toda a parte. Uma coluna ascendente de fumaça de cigarro se decompõe em anéis desordenados. Uma bandeira drapeja de um lado para outro ao vento. Uma torneira gotejante passa de um ritmo constante para outro, aleatório. O caos surge no comportamento das condições do tempo, no comportamento de um avião em vôo, no comportamento dos carros que se agrupam numa auto-estrada, no comportamento do petróleo que flui em tubos subterrâneos. [...] A percepção desse fato começou a modificar a maneira pela qual os executivos tomam decisões sobre seguros, os astrônomos vêem o sistema solar, e os teóricos de política falam sobre as tensões que provocam conflitos armados.

Os sistemas caóticos são bastante interessantes para a identificação de sistemas. Um dos motivos é que o comportamento complexo que eles apresentam é, muitas vezes, modelado por equações simples, embora não lineares. Por isso, tais sistemas podem ser usados para testar métodos de identificação. O objetivo, nesse caso, é verificar se o algoritmo é capaz de recuperar o modelo simples que existe por trás do comportamento dinâmico complexo.

Por outro lado, a identificação de sistemas caóticos traz uma dificuldade. Uma vez de posse do modelo identificado é necessário verificar a sua validade, o que necessita de bons procedimentos de validação, conforme visto no Capítulo 3.

## 4.2 O sistema de Lorenz

O primeiro sistema usado para testar o procedimento de identificação foi o *sistema de Lorenz* (Lorenz, 1963). Tal sistema foi um dos primeiros sistemas estudados que apresentava caos determinístico.

O sistema, na verdade, é um modelo teórico para um experimento de convecção de fluidos.

### 4.2.1 Descrição do sistema

Pouco se tem a dizer desse sistema, já que se trata de um modelo matemático. Ele é composto de um sistema de três equações diferenciais ordinárias autônomas, isto é, sem dependência explícita do tempo. Esse sistema é mostrado na equação (4.1).

$$\begin{cases} \dot{x} = \sigma(y - x) , \\ \dot{y} = rx - y - xz , \\ \dot{z} = -bz + xy . \end{cases} \quad (4.1)$$

Esse sistema de equações descreve, com algumas simplificações, o comportamento de uma célula de convecção de formato prismático, com o comprimento muito maior do que a largura e a altura. A parte inferior da célula se encontra a uma temperatura um pouco maior do que a parte superior. Se a diferença de temperatura for grande o suficiente, ocorre convecção. A variável  $x$  em (4.1) é proporcional à velocidade da convecção; a  $y$ , à diferença de temperatura entre a coluna ascendente e descendente de fluido e a  $z$ , à distorção do perfil de temperatura vertical em relação a um perfil linear que ocorreria se a troca de calor fosse puramente por condução (Fiedler-Ferrara e Prado, 1994). Os parâmetros  $\sigma$ ,  $b$  e  $r$  representam características geométricas do experimento, constantes do fluido e a diferença de temperatura. Para  $(r, \sigma, b) = (28, 10, \frac{8}{3})$ , o sistema exibe caos.

### 4.2.2 Identificação

As equações (4.1) foram integradas numericamente para gerar a massa de dados de identificação contendo 10.000 pontos. O passo de integração (e o período de amostragem) usado foi de 0,01s e as condições iniciais foram  $[x(0), y(0), z(0)] = [3, 3574; 5, 3824; 20, 9735]$ . Essa massa de dados pode ser representada no espaço de estados conforme visto na Figura 4.1.

O objetivo nessa etapa é apenas verificar se o algoritmo de identificação é coerente e se as técnicas de detecção de estrutura funcionam. Assim, a identificação foi a mais simples possível. Os modelos identificados são do tipo multivariável, com todas as variáveis de estado sendo utilizadas. Nenhum tipo de ruído foi acrescentado aos dados e a única técnica de detecção de estrutura usada foi o ERR.

As derivadas  $\dot{x}, \dot{y}, \dot{z}$  foram calculadas através de (4.1). As rotinas de identificação foram, então, executadas com a massa de dados e suas derivadas. Os termos candidatos foram limitados em monômios de até terceiro grau. Foram escolhidos 900 centros uniformemente distribuídos ao longo da massa de dados.

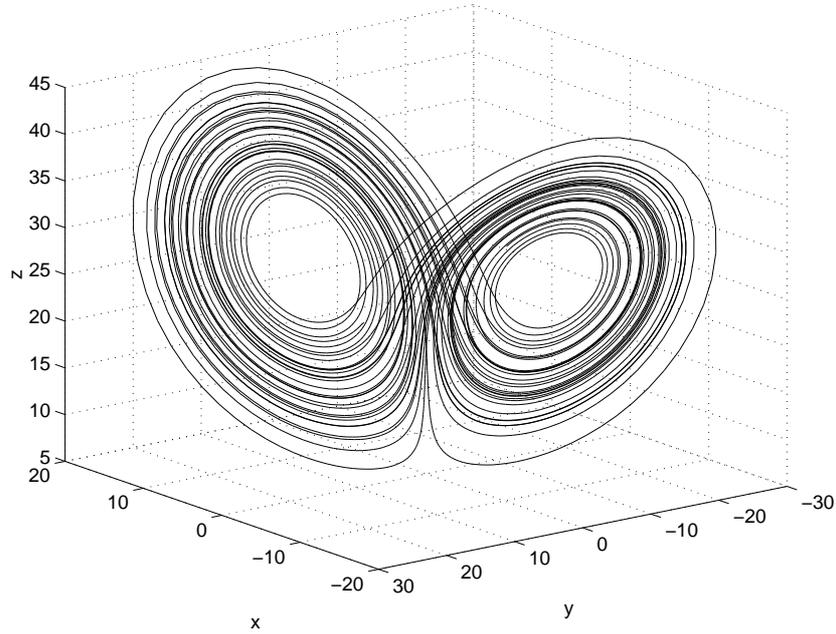


Figura 4.1: Atrator do sistema de Lorenz (4.1).

Diversos modelos foram obtidos, cada um com um número diferente de termos para cada função  $g_j[\varphi(t)]$ . Porém, um desses modelos se mostra mais interessante. Nesse modelo específico foram usados 6 termos para identificar  $g_1[\varphi(t)]$  e 4 termos para as demais funções. Esse modelo identificado pode ser visto na equação (4.2):

$$\begin{aligned}
 \dot{x} &= 4,94 \times 10^{-19}y^3 - 4,64 \times 10^{-19}xz^2 + 2,12 \times 10^{-17}yz \\
 &\quad + 1,00 \times 10^{-16}xy + 10,0y - 10,0x \\
 \dot{y} &= -8,63 \times 10^{-18}xz^2 + 28,0x - 1,00xz - 1,00y \\
 \dot{z} &= 7,77 \times 10^{-18}y^2z - 2,67z + 1,00xy - 5,32 \times 10^{-18}xyz .
 \end{aligned} \tag{4.2}$$

Observando-se esse modelo nota-se que o critério do ERR foi bastante eficaz na escolha de termos, principalmente nas duas últimas funções, tendo em vista que o conjunto de termos candidatos tem 20 monômios. Além disso, os termos que não fazem parte do sistema original tiveram coeficientes quase nulos ( $< 10^{-15}$ ). Por isso, pode-se afirmar que o modelo (4.2) recuperou, na prática, exatamente a dinâmica do sistema (4.1).

A título ilustrativo, o modelo (4.2) foi integrado partindo-se das mesmas condições iniciais que os dados originais. O resultado da simulação do modelo e os dados originais (apenas a variável  $x$ ) estão representados na Figura 4.2. Para melhor ilustrar a evolução do erro de predição, ele está mostrado na Figura 4.3.

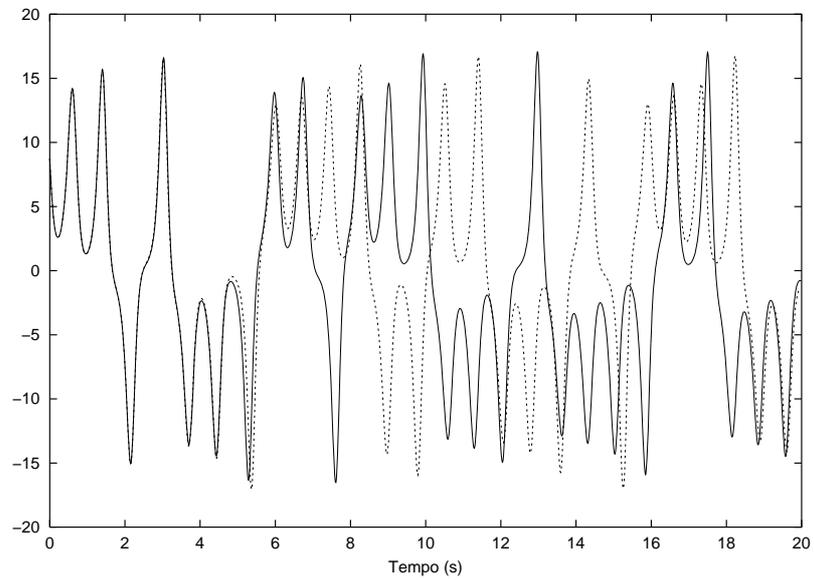


Figura 4.2: Lorenz variável  $x$ . Modelo (linha tracejada) e dados originais (linha cheia).

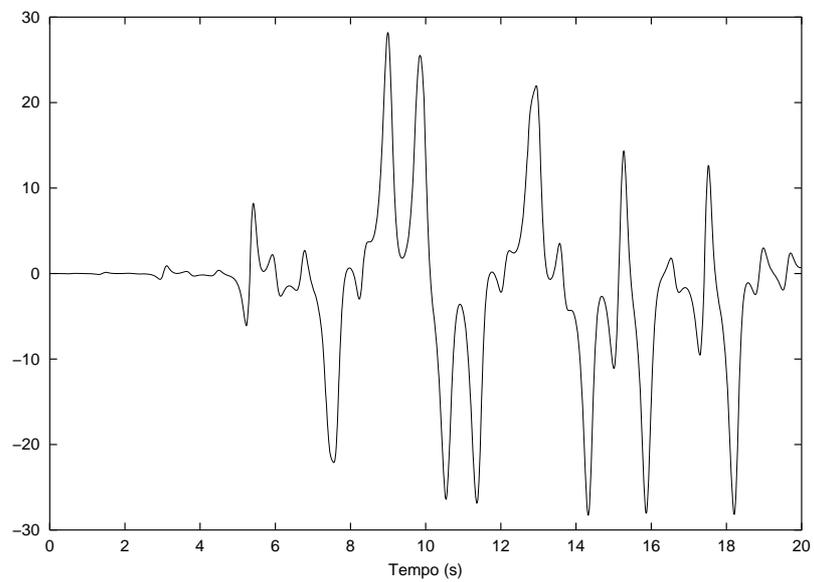


Figura 4.3: Erro de predição da variável  $x$ .

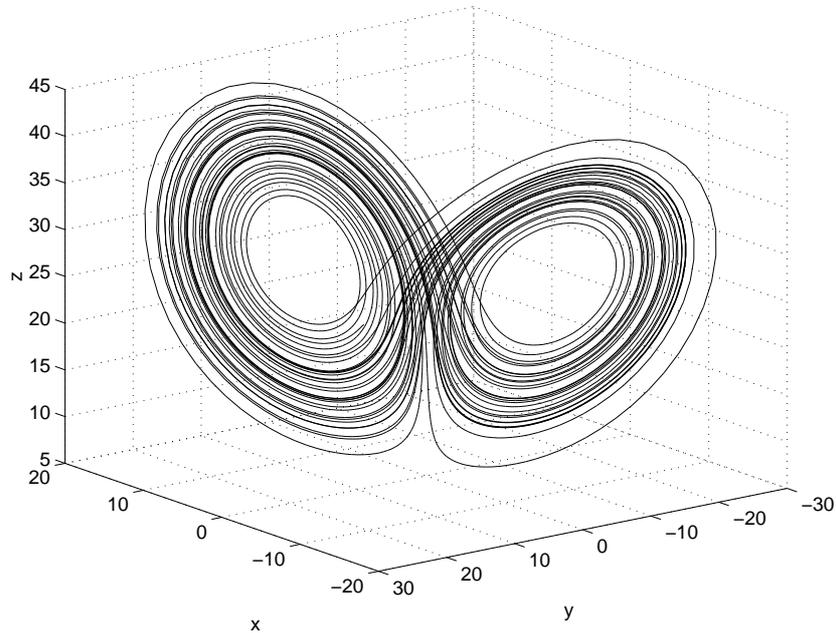


Figura 4.4: Atrator do modelo identificado.

É interessante notar que, apesar da diferença entre (4.1) e (4.2) ser quase nenhuma, ocorre uma divergência entre os dados originais e os provenientes do modelo. Isso é devido à natureza caótica do sistema que faz com que qualquer perturbação no mesmo (até mesmo variações no passo de integração) sejam suficientes para causar divergência na trajetória da solução, conforme já mencionado na Seção 4.1

Como forma de validação foi utilizada a comparação visual dos atratores. O atrator de (4.2) pode ser visto na Figura 4.4.

Nota-se que, apesar da divergência no tempo, os atratores são os mesmos. Isso mostra que o fluxo (4.1) foi muito bem captado por (4.2). Outro ponto interessante é que a maior parte dos outros modelos identificados apresentou um atrator semelhante ao da Figura 4.1.

### 4.3 O circuito de Chua

O circuito de Chua (Chua, 1992) é um oscilador não linear que pode apresentar comportamento caótico dependendo dos seus parâmetros. Esse circuito é bastante utilizado como paradigma de estudo de sistemas não lineares.

O circuito utilizado nesse trabalho é um protótipo pertencente ao grupo

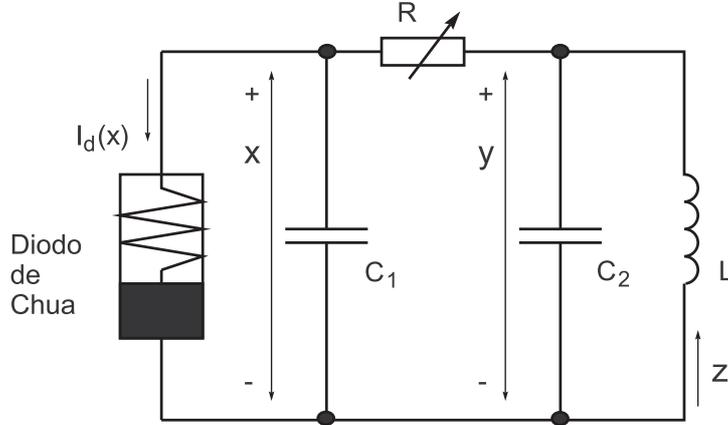


Figura 4.5: Diagrama do circuito de Chua.

MACSIN<sup>1</sup>. Tal montagem gerou muitos trabalhos, como Aguirre et al. (1997), Elgar et al. (1998), Aguirre e Tôrres (1997), entre outros.

### 4.3.1 Descrição do sistema

O diagrama do circuito pode ser visto na Figura 4.5.

O diodo de Chua é, na verdade, um resistor não linear. Esse elemento foi realizado utilizando dois amplificadores operacionais (Tôrres e Aguirre, 1995, 2000). As equações que modelam o circuito são

$$\begin{aligned} C_1 \frac{d v_1}{d t} &= \frac{(v_2 - v_1)}{R} - i_d(v_1) \\ C_2 \frac{d v_2}{d t} &= \frac{(v_1 - v_2)}{R} + i_L \\ L \frac{d i_L}{d t} &= -v_2, \end{aligned} \quad (4.3)$$

sendo  $v_i$  a tensão sobre o capacitor  $C_i$  e  $i_L$  a corrente no indutor. A corrente no diodo de Chua é dada por

$$i_d(v_1) = \begin{cases} m_0 v_1 + B_p(m_0 - m_1) & \text{para } v_1 < -B_p \\ m_1 v_1 & \text{para } |v_1| \leq B_p \\ m_0 v_1 + B_p(m_1 - m_0) & \text{para } v_1 > +B_p \end{cases} . \quad (4.4)$$

<sup>1</sup>Maiores detalhes sobre o grupo podem ser encontrados em [www.cpdee.ufmg.br/~MACSIN](http://www.cpdee.ufmg.br/~MACSIN)

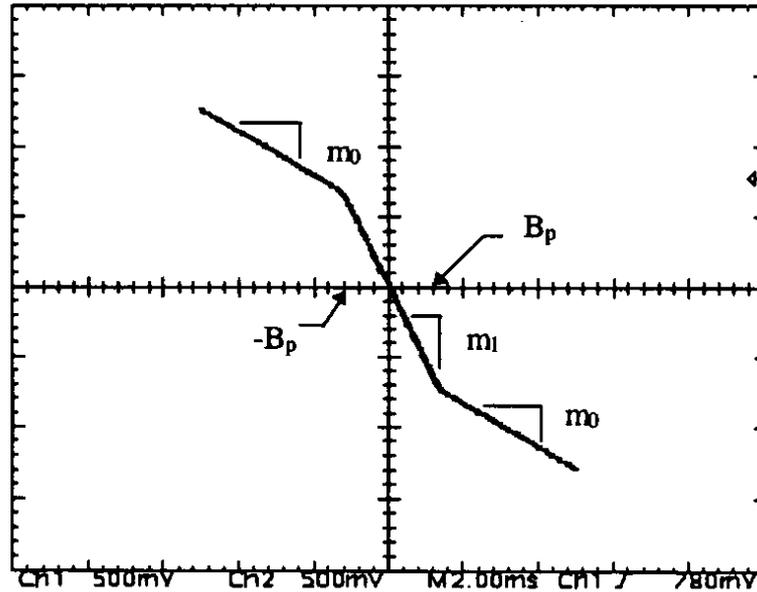


Figura 4.6: Curva corrente por tensão medida no diodo de Chua utilizado. As maiores divisões são de 0,5V.

Os valores dos componentes do circuito são:  $C_1 = 11,0 \mu\text{F}$ ,  $C_2 = 45 \mu\text{F}$ ,  $L = 22\text{H}$  e  $R$  é um *trimpot* multivoltas de  $2,0\text{k}\Omega$ . Os parâmetros do diodo de Chua medidos são:  $m_0 = -0,37 \pm 0,04\text{mS}$ ,  $m_1 = -0,68 \pm 0,04\text{mS}$  e  $B_p = 1,1 \pm 0,2\text{V}$ . A curva característica real do diodo de Chua pode ser vista na figura 4.6. O elevado valor de indutância foi obtido através de um simulador de indutor, um circuito ativo que simula o comportamento de um indutor (Tôrres e Aguirre, 2000).

A resistência do *trimpot* é o parâmetro de controle do sistema. Quando variado, o circuito exibe diferentes comportamentos dinâmicos, desde regime estacionário até caos. Com o *trimpot* em torno de  $1800\Omega$ , o sistema tem um comportamento caótico com um atrator dupla volta.

### 4.3.2 Identificação

Os dados de identificação foram coletados com o circuito já em funcionamento para evitar transitórios. O tipo de modelo escolhido para a identificação foi monovariável. A variável observada foi a  $v_1$ , a tensão sobre  $C_1$ . A frequência de amostragem foi de 300 Hz, o que resulta em aproximadamente 100 pontos por pseudo-período. A massa de dados original tem 100.000 pontos.

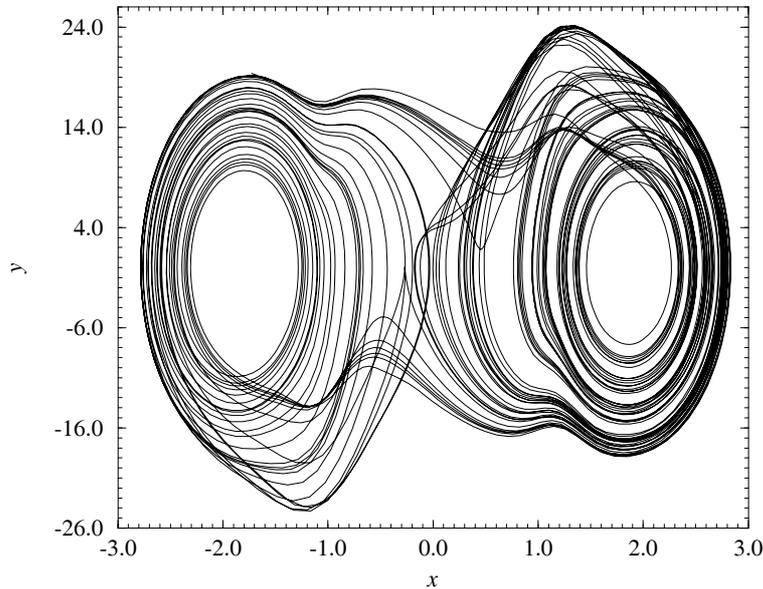


Figura 4.7: Projecção plana do atrator do circuito de Chua.  $y = \dot{x}$ .

Embora o sistema original seja de terceira ordem, foi necessário identificar modelos de quinta ordem, pelos motivos comentados na Seção 3.2.

As derivadas foram calculadas conforme descrito na Seção 3.3. Para identificar modelos de quinta ordem foi preciso computar derivadas também de quinta ordem.

Uma projeção bidimensional do atrator do circuito pode ser vista na Figura 4.7. Esse atrator foi reconstruído a partir da série temporal observada utilizando coordenadas derivativas.

Numa primeira abordagem, tentou-se obter os modelos diretamente pelo procedimento apresentado nesse trabalho. O grau de não linearidade dos termos candidatos foi limitado em 3, o que define 56 termos candidatos. O sistema original possui 3 pontos fixos, sendo um nulo e os outros dois simétricos. Essa informação foi usada para eliminar do conjunto de termos candidatos os termos constante e  $y_1^2$ . Nenhum bom modelo foi obtido dessa forma.

A seguir, tentou-se identificar modelos sem nenhuma detecção de estrutura, até mesmo incluindo os termos que haviam sido removidos para garantir a simetria dos pontos fixos. Foram identificados modelos variando o número de termos de forma crescente. A ordem de inclusão de termos foi dos de menor

Tabela 4.1: Dinâmica dos modelos obtidos com e sem detecção de estrutura.

Seleção de estrutura	Não	Sim
Intervalo do número de termos	$10 \leq p \leq 38$	$8 \leq p \leq 36$
Modelos instáveis	18	4
Ponto fixos ou ciclo limite	10	9
Caos (atrator dupla volta)	1	16

grau para os de maior. Essa ordem foi a mesma empregada em Gouesbet e Letellier (1994).

Seguindo esse procedimento, um modelo de 38 termos foi encontrado que exibia caos. A partir de então, o procedimento de identificação foi alterado um pouco. O objetivo era tentar melhorar a qualidade do modelo identificado usando técnicas de detecção de estrutura.

Os 38 termos do modelo obtido sem detecção de estrutura foram considerados o conjunto de termos candidatos para um novo processo de identificação. A primeira etapa foi eliminar novamente os termos constante e  $y_1^2$ . Foram então identificados modelos com número crescente de termos, porém, *a ordem de inclusão de termos foi determinada pelo critério ERR*. O resultado dessa identificação está resumido na Tabela 4.1, onde foi mostrado o número de modelos que exibe um determinado comportamento dinâmico. Também foi mostrado o resultado da identificação sem detecção de estrutura, por motivo de comparação.

Pode-se perceber que, apesar de não ter sido capaz de obter nenhum modelo caótico na primeira etapa, a identificação com detecção de estrutura foi capaz de obter 16 modelos com esse comportamento numa região do espaço de número de termos na qual a identificação sem detecção de estrutura obteve somente um modelo caótico.

De posse desses resultados, procurou-se descobrir a causa do mau desempenho da identificação com detecção de estrutura observado no início do trabalho. Depois de mais alguns testes, percebeu-se que, quando o 39º termo fazia parte do conjunto de termos candidatos, ele era sempre escolhido pelo ERR junto com os termos mais importantes. A presença deste termo no modelo tornava-o instável. Concluiu-se então que o processo de identificação é muito sensível ao conjunto de termos candidatos.

Os 16 modelos de comportamento caótico obtidos possuíam atratores com um bom aspecto visual, ou seja, possuíam atratores visualmente parecidos com o atrator original. Porém, esse critério é muito fraco. Com o objetivo de se fazer uma validação mais refinada, foi estimado o maior expoente de Lyapunov para o sistema original e para os modelos identificados. Três modelos

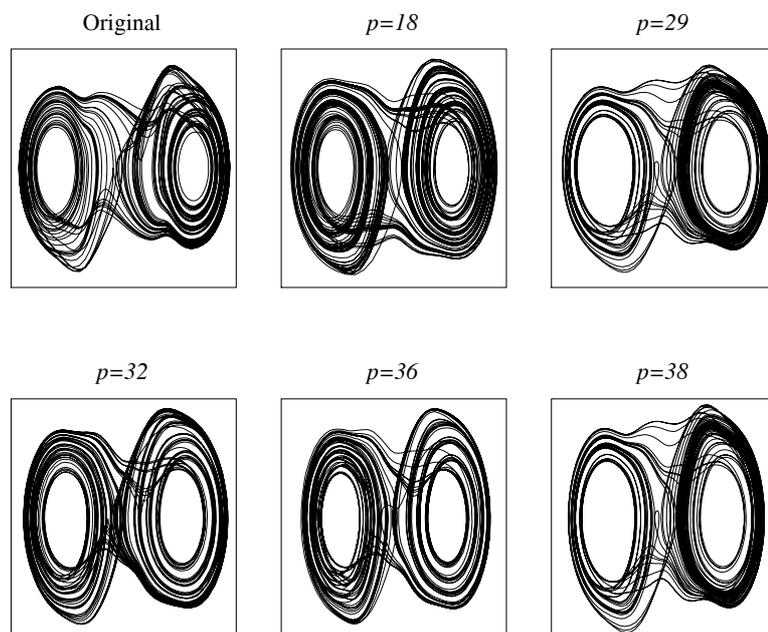


Figura 4.8: Projeções planas do atrator do circuito de Chua e dos atratores dos modelos identificados.

tiveram expoentes de Lyapunov próximos daquele do sistema original: o de 29 termos, o de 32 termos e o de 36 termos<sup>2</sup>. A esses modelos foram adicionados outros dois: o modelo de 38 termos obtido sem detecção de estrutura e um modelo de 18 termos que foi o mais simples a ter um bom (visualmente falando) atrator.

Projeções planas dos atratores desses modelos, junto com o sistema original, podem ser vistas na Figura 4.8. Pode-se notar que escolher um modelo em particular baseado somente nesses gráficos é muito difícil.

O próximo passo na validação foi a construção do mapa de primeiro retorno de uma seção de Poincaré feita sobre o atrator. Mapas de primeiro retorno foram feitos para todos os modelos da Figura 4.8 e para o sistema original. O resultado é mostrado na Figura 4.9.

Pode-se notar agora com clareza o desempenho dos modelos. O modelo que se saiu melhor, segundo esse critério de validação, foi o de 29 termos. Mesmo assim, esse modelo apresenta um mapa com estrutura estratificada,

<sup>2</sup>O modelo de 36 termos é o modelo de 38 termos sem os termos constante e  $y_1^2$ . Ou seja, ele é um modelo identificado sem nenhuma técnica de detecção de estrutura a não ser o uso de informação *a priori* sobre a simetria dos pontos fixos.

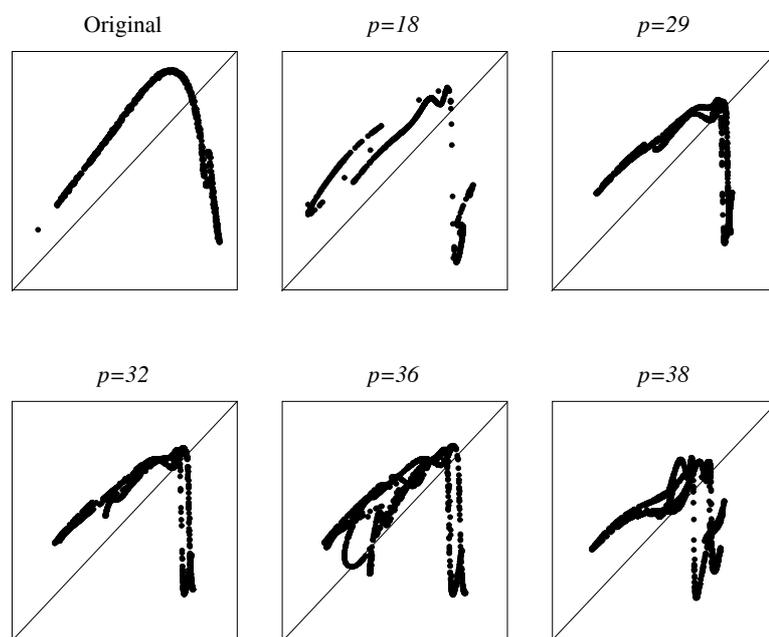


Figura 4.9: Mapas de primeiro retorno na seção de Poincaré do atrator do circuito de Chua e dos atratores dos modelos identificados.

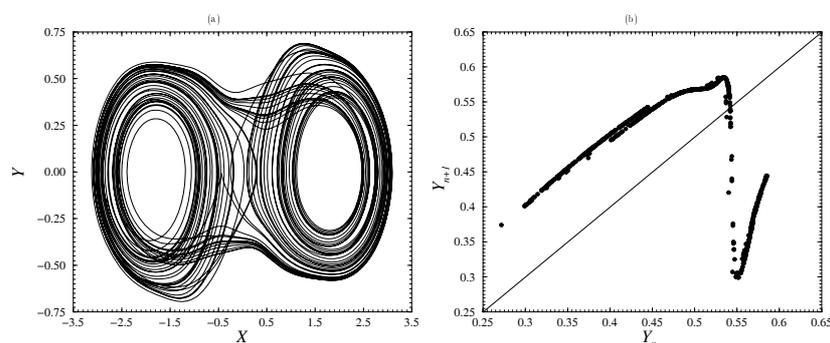


Figura 4.10: Projeção do atrator (a) e o correspondente mapa de primeiro retorno (b) gerado pelo modelo simétrico de 20 termos.

ou seja, dividida em camadas. Essa estrutura estratificada está presente em maior grau nos outros modelos e é decorrente da assimetria do campo vetorial dos modelos identificados<sup>3</sup>.

Por outro lado, o sistema original é simétrico, já que o seu mapa não apresentou estrutura estratificada. Assim, decidiu-se forçar a simetria dos modelos identificados. Para tal, partindo-se dos mesmos 38 termos candidatos já citados, repetiu-se o processo de identificação, com a diferença que os dados foram tornados simétricos artificialmente. O artifício usado para obter essa simetria foi tomar, para cada centro escolhido, o seu simétrico. Por exemplo, se um determinado vetor de entrada  $\varphi(t_i)$ , associado a um valor de saída  $Y_j(t_i)$ , foi escolhido então toma-se também  $-\varphi(t_i)$  associado a  $-Y_j(t_i)$ .

Isso foi feito e repetiu-se o processo de identificação, usando novamente o critério do ERR. Foi obtido um modelo de 20 termos. A projeção de seu atrator e o seu mapa de primeiro retorno estão mostrados na Figura 4.10.

Embora esse mapa ainda não esteja idêntico ao mapa original, houve uma melhora significativa. Aliás, a detecção de estrutura, de uma maneira geral, foi capaz de melhorar a qualidade dos modelos identificados. Porém, não se obteve nenhum modelo capaz de capturar adequadamente o mapa de primeiro retorno.

Existem diversas causas que podem ser apontadas para explicar o desempenho do método nesse exemplo. Uma delas é o tipo de não linearidade presente no sistema original. A função definida em (4.4) é do tipo linear por partes. O fato dela não ser suave pode exigir uma maior flexibilidade do processo de aproximação. Uma outra possibilidade é uma desvantagem inerente ao método: o fato da aproximação ser feita por minimização de erro sobre uma derivada de alta ordem calculada numericamente. A detecção de

<sup>3</sup>Embora o campo vetorial dos modelos seja assimétrico, os seus pontos fixos não o são.

estrutura via ERR e a estimação dos parâmetros são feitas baseadas nesse erro. Dessa forma, a estimação do modelo como um todo não garante automaticamente o seu bom desempenho dinâmico. O que é garantido é que o resíduo de identificação  $e$  seja pequeno. A norma do resíduo é uma medida da capacidade do modelo de explicar os sinais  $Y_j$  em (3.12). Tais sinais são sempre derivadas de sinais coletados calculadas numericamente. Se houver muito erro no cálculo das derivadas, obter um pequeno resíduo não significa necessariamente que o modelo identificado terá um bom comportamento dinâmico.

Vale mencionar que a obtenção de modelos discretos para esse sistema é bem mais fácil. Em Freitas (1998) e Elgar et al. (1998), por exemplo, foram obtidos modelos NARMAX polinomiais com a aplicação direta das técnicas de detecção de estrutura empregadas nesse trabalho.

Esses resultados foram publicados resumidamente em Aguirre et al. (2000) e de forma mais detalhada em Aguirre et al. (2001).

## 4.4 Conclusão

Neste Capítulo, foram identificados modelos para dois sistemas caóticos: o sistema de Lorenz e o circuito de Chua. A identificação do sistema de Lorenz usou dados simulados enquanto a identificação do circuito de Chua foi feita com dados reais. Outra diferença foi que a identificação do primeiro usou modelos multivariáveis e a identificação do último usou modelos monovariáveis.

No primeiro caso, a detecção de estrutura foi capaz de adequadamente recuperar os termos presentes no sistema original. O atrator do modelo identificado ficou muito próximo daquele do sistema original.

A identificação no segundo caso se mostrou muito difícil. Não se conseguiu obter um modelo razoável com a aplicação direta das técnicas aqui apresentadas. Porém, a partir de modelos obtidos sem detecção de estrutura foi possível derivar modelos melhores através do uso dessas técnicas, seja pelo critério ERR, seja por uso de informação *a priori*.

De qualquer forma, a detecção de estrutura foi responsável pela melhora dos modelos obtidos.

## Capítulo 5

# Identificação de uma Câmara de Termo-Vácuo

No Capítulo 4 foi vista a aplicação do procedimento de identificação a sistemas caóticos. Tais sistemas, embora de importância reconhecida, têm relativamente poucas aplicações em engenharia, no presente momento. No entanto, novas aplicações têm surgido recentemente (Tôrres, 2001).

Um emprego mais comum e freqüente da identificação é o estudo de sistemas presentes na indústria. No presente Capítulo, será mostrada a identificação de um sistema mais simples e mais próximo do cotidiano da engenharia — uma câmara de testes de termo-vácuo.

Este sistema é de média complexidade, apresenta entradas que não são desprezíveis, é não linear e envolve diversos processos físicos. É um bom exemplo do que se encontra normalmente na prática da identificação de sistemas, principalmente na indústria.

O objetivo deste Capítulo é mostrar mais um estudo de caso, ilustrando os aspectos mais práticos da identificação. Este capítulo está organizado da seguinte forma. A Seção 5.1 descreve o sistema a ser identificado. A Seção 5.2 mostra o procedimento de identificação contínuo e os resultados obtidos. A Seção 5.3 faz uma comparação com a identificação discreta e a Seção 5.4 apresenta as conclusões do Capítulo.

### 5.1 A Câmara de testes de termo-vácuo

Um satélite em órbita está exposto aos raios solares de forma intermitente e ao vácuo de forma perene. Devido a isso, ele pode atingir temperaturas muito elevadas ou muito baixas, tipicamente no intervalo  $-180^{\circ}\text{C}$  a  $150^{\circ}\text{C}$ . Para simular esse ambiente é utilizada uma câmara de termo-vácuo (CTV).

A CTV utilizada neste trabalho pertence ao Laboratório de Integração e Testes (LIT)<sup>1</sup> do Instituto Nacional de Pesquisas Espaciais (INPE) (Araújo Filho e Macau, 1999; Araújo Filho, 2000).

A CTV é uma câmara especial, de formato aproximadamente cilíndrico, na qual é colocado o espécime a ser testado. No início do teste, é produzido vácuo no interior da câmara. Quando a condição de vácuo é atingida, tem início o teste propriamente dito.

O teste consiste, basicamente, em fazer o espécime seguir uma curva de referência de temperatura. Neste teste, o espécime, normalmente um satélite artificial ou um de seus subsistemas, é mantido desligado. Tal teste é chamado de teste com carga passiva. Existem outros tipos de teste, como os testes com carga ativa e os testes nos quais o espécime é aquecido diretamente por meio de lâmpadas. Porém, somente os testes com carga passiva são tratados aqui.

Para variar a temperatura, a CTV dispõe de um sistema de circulação de nitrogênio gasoso numa camisa que circunda o corpo de prova. Essa camisa é dividida em três partes: duas circulares, correspondentes à tampa e ao fundo da câmara e uma cilíndrica central. O aquecimento é obtido através de uma resistência elétrica que aquece o gás. Já o resfriamento é conseguido por pulverização de nitrogênio líquido dentro do sistema de circulação de gás. A referência para o controle de pressão do gás é calculada em função da sua temperatura, de forma a manter a densidade do gás constante no sistema. Um diagrama da CTV pode ser visto na Figura 5.1.

Assim, a câmara possui dois sistemas de controle: um de temperatura e um de pressão.

A variável controlada pelo primeiro controlador é a temperatura do nitrogênio no sistema. O sinal de referência está relacionado ao ensaio que está sendo feito, mas como a variável controlada não é a temperatura do espécime, é necessária a presença de um operador que continuamente forneça a referência para o controlador de temperatura. Dessa forma, o controle de temperatura no espécime é feito manualmente.

O controle de temperatura atua na resistência de aquecimento e no nitrogênio líquido pulverizado. O sistema de controle de pressão atua na válvula do selo de controle.

---

<sup>1</sup>Mais informações a respeito do LIT podem ser encontradas em [www.lit.inpe.br](http://www.lit.inpe.br).

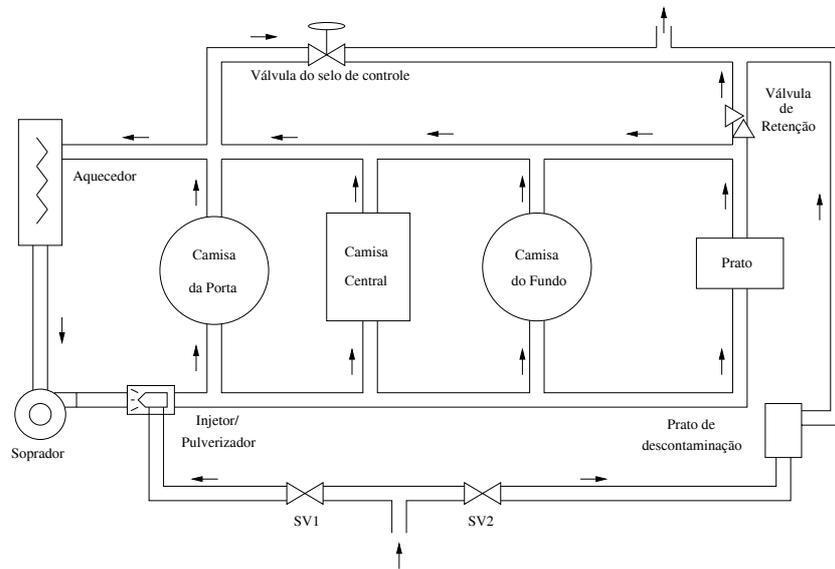


Figura 5.1: Diagrama da CTV. As setas indicam o fluxo de nitrogênio.

## 5.2 Identificação utilizando modelos contínuos

### 5.2.1 Coleta dos dados

A primeira etapa da identificação é o ensaio para a coleta de dados. Dentre as diversas câmaras presentes no LIT, a escolhida para esse ensaio foi uma das duas câmaras menores, de capacidade de 250 l.

Foram feitos dois ensaios na CTV, um com a câmara vazia e o outro com a câmara com uma carga passiva. Esses ensaios foram projetados e conduzidos pelo pessoal do LIT (Araújo Filho, 2000).

A massa de dados se constitui do sinal de referência de temperatura, sinais de temperatura coletados por vários termopares na câmara, tanto na camisa quanto no corpo de prova, a temperatura medida do gás na camisa, o sinal de controle para o aquecedor, o sinal de controle de pressão e vários sinais calculados a partir dos sinais anteriores. O período de amostragem usado foi de 10s.

No caso do ensaio com carga, a saída do sistema foi escolhida como sendo a temperatura no espécime. Para determiná-la foi calculada a média aritmética dos sinais de temperatura fornecidos pelos termopares que se encontravam no espécime. A entrada foi definida como sendo a temperatura da camisa, que foi calculada tomando a média dos sinais de provenientes dos termopares nela

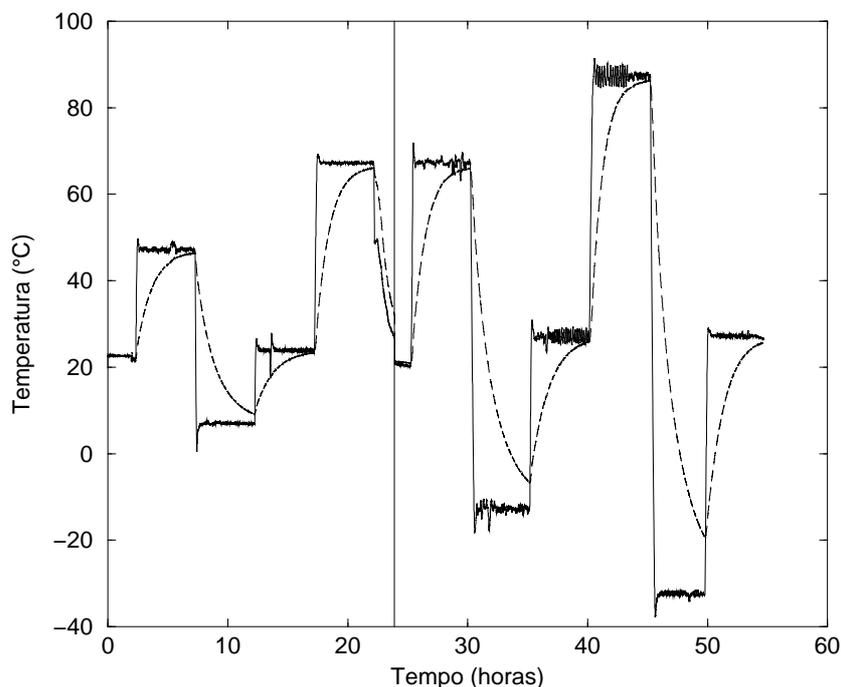


Figura 5.2: Dados do ensaio. A entrada é a linha cheia e a saída é a linha tracejada. A linha vertical mostra o momento da interrupção do teste.

localizados. A temperatura da camisa é, de certa forma, a variável controlada pelo controlador de temperatura da câmara<sup>2</sup> e a temperatura do espécime é a variável de interesse durante um ensaio. Assim, um modelo obtido para esse processo poderia ser útil no projeto de um sistema de controle automático para os ensaios. Atualmente, esse controle é feito por um operador humano.

Os dados do ensaio podem ser vistos na Figura 5.2.

O ensaio é bastante longo, durando cerca de 54 horas. Isso fez com que fosse necessária a sua interrupção por volta da vigésima quarta hora de teste. O resto do experimento foi feito várias horas depois da interrupção. No intervalo, o vácuo foi desfeito e a câmara, aberta.

O ensaio consistiu em aplicar uma seqüência de degraus como referência para o controlador de temperatura. Foram aplicados ao todo dez degraus de amplitudes diferentes ora positivos, ora negativos. O degrau aplicado logo após a interrupção do ensaio foi uma repetição do último degrau aplicado

<sup>2</sup>A rigor, a variável controlada é a temperatura do nitrogênio dentro da camisa. Porém, a diferença entre essas temperaturas é desprezível, já que a dinâmica de transmissão de calor entre o gás e a camisa é tão rápida em relação à dinâmica da temperatura do corpo de prova que a primeira pode ser considerada instantânea.

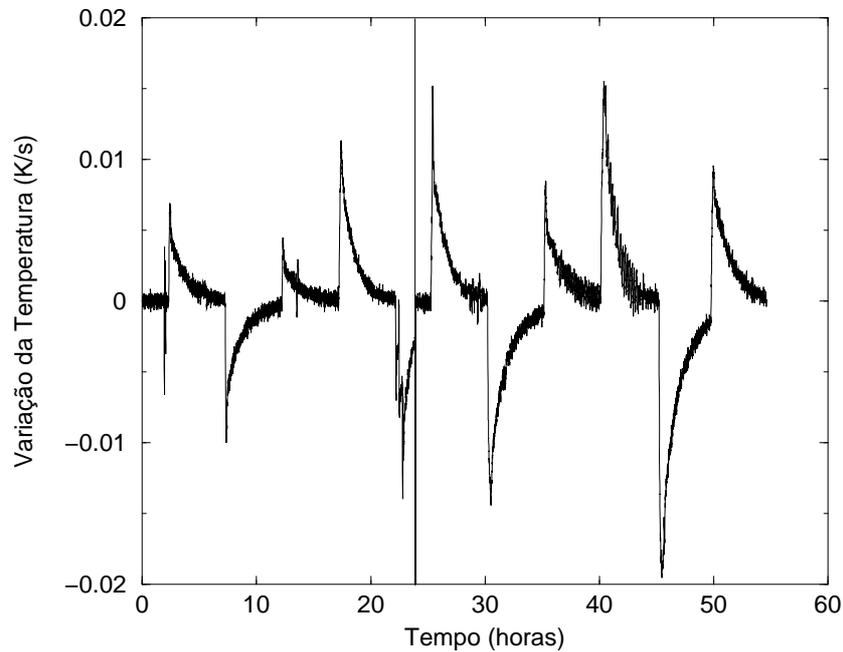


Figura 5.3: Derivada do sinal  $y$  calculada numericamente. A linha vertical mostra o momento da interrupção do teste.

antes da mesma. Isso foi uma forma de verificar a invariância no tempo do sistema.

### 5.2.2 O procedimento de identificação

Após a coleta dos dados, procedeu-se à identificação. O primeiro procedimento feito foi o cálculo da derivada do sinal de saída, de agora em diante chamado por  $y$ .

Esse cálculo foi feito conforme indicado na Seção 3.3. Optou-se por identificar modelos de primeira ordem. Assim, o polinômio usado para aproximar o sinal (vide Seção 3.3) foi de grau um, ou seja, uma função linear. O intervalo  $L$  no qual foi feita a aproximação foi de tamanho 15 amostras. A derivada calculada pode ser vista na Figura 5.3.

Os dados do ensaio foram divididos para formar duas massas. O trecho que foi realmente usado na identificação vai de 38,8 horas a 50 horas e corresponde aos dois últimos degraus aplicados. O trecho de validação vai de 2,2 horas a 16,7 horas e corresponde aos três primeiros degraus aplicados no sistema.

Os sinais de temperatura, originalmente medidos em graus Celsius, foram

transformados para Kelvin, pois essa unidade caracteriza melhor o fenômeno de radiação térmica presente no processo.

O grau máximo para os termos candidatos foi fixado em quatro, o que leva a um conjunto de 15 termos. Foi usado esse grau máximo devido ao sistema ser de natureza térmica e envolver trocas de calor por radiação. Isso completa a utilização de informação *a priori* na identificação. Essas informações determinaram apenas a ordem dos modelos (primeira), a mudança de unidades de graus Celsius para Kelvin e o grau máximo dos termos candidatos.

O número de centros  $N$  é um parâmetro importante na identificação de modelos contínuos. Porém, ainda não existe um procedimento para a sua determinação. Nesse trabalho, foram escolhidos dois valores para  $N$ , um baixo e um elevado. O valor baixo foi fixado arbitrariamente em 100 centros. O valor elevado foi escolhido de forma a evitar o mal condicionamento da matriz de regressores que ocorre para  $N$  muito grande. Após alguns testes, escolheu-se o valor 800. Os centros foram sempre tomados de forma igualmente espaçada no tempo.

A partir de então, para cada valor de  $N$ , foram estimados 15 modelos diferentes com número de termos  $p$  diferentes. Como forma de validação, esses modelos foram simulados e a sua saída foi comparada com a saída do sistema original no trecho de validação. Essa forma de validação pôde ser empregada porque o sistema em questão, ao contrário daquele do Capítulo 4, não é caótico.

A simulação dos modelos foi feita usando a função *lsode* do GNU Octave e foi uma simulação livre, ou seja, os únicos valores da saída do sistema real que entraram no processo de simulação foram as condições iniciais. Portanto, o resultado da simulação dos modelos é a *previsão* do modelo para o comportamento do sistema para a entrada utilizada.

A função usada na simulação emprega passo de integração ajustável. Porém, os dados de entrada foram amostrados com um intervalo de tempo fixo de 10s. Assim, foi necessária a implementação de um artifício para fornecer valores da entrada  $u$  para instantes de tempo arbitrários. A solução adotada foi a interpolação linear da entrada entre duas amostras sucessivas. Isso é equivalente a empregar um *hold* de primeira ordem na conversão de um sinal de digital para analógico.

### 5.2.3 Resultados e análise

Os valores dos somatórios dos erros quadráticos entre a saída do sistema e a saída dos modelos podem ser vistos na Tabela 5.1.

Pode-se notar que um elevado número de termos não significa melhor qualidade dos modelos identificados, muito pelo contrário. Modelos com

Tabela 5.1: Relação entre o número de termos do modelo identificado e o erro de predição. A simulação do modelo de 15 termos identificado usando 800 centros não convergiu.

Número de termos	Somatório dos erros quadráticos	
	100 centros	800centros
1	$4,3988 \times 10^6$	$5,0989 \times 10^6$
2	5427,3	6649,4
3	2018,9	3710,3
4	2099,7	728,24
5	3927,3	3349,4
6	5886,7	$2,4280 \times 10^4$
7	$1,5964 \times 10^4$	$1,9766 \times 10^4$
8	$2,1610 \times 10^4$	$4,8650 \times 10^4$
9	$2,9430 \times 10^4$	$8,8353 \times 10^4$
10	$5,5728 \times 10^4$	$8,8809 \times 10^4$
11	$6,0466 \times 10^4$	$8,6454 \times 10^4$
12	$1,1489 \times 10^5$	$1,0232 \times 10^5$
13	$1,0191 \times 10^5$	$1,0314 \times 10^5$
14	$8,3117 \times 10^4$	$9,1093 \times 10^4$
15	$8,3302 \times 10^4$	—

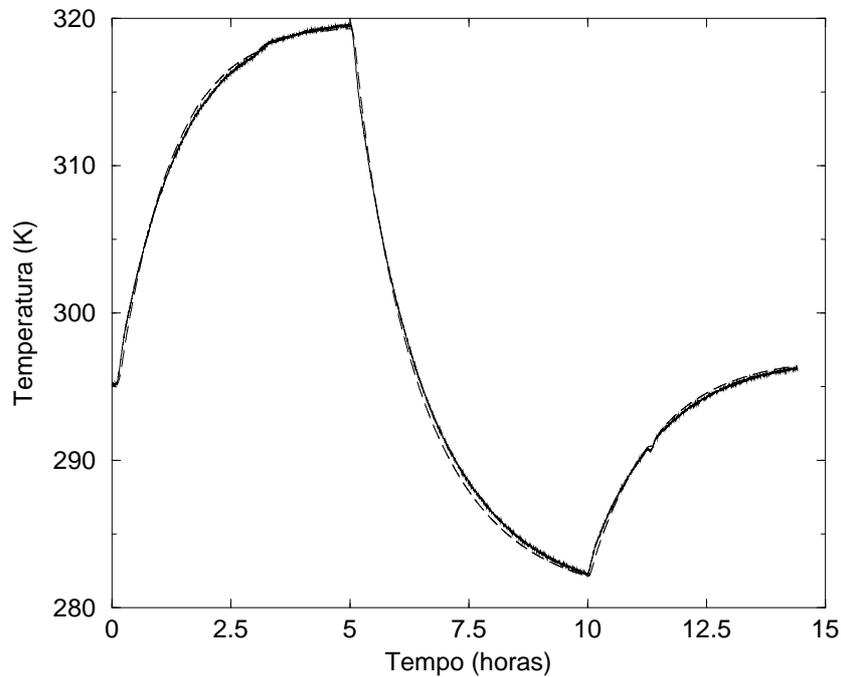


Figura 5.4: Validação do modelo de 4 termos. Modelo: linha tracejada, sistema: linha cheia.

muitos termos tiveram um desempenho pior do que modelos mais simples.

Dos modelos identificados, o que apresentou melhor predição foi o modelo de quatro termos identificado usando 800 centros. O resultado de sua validação pode ser visto na Figura 5.4. O modelo é mostrado na equação (5.1), sendo que a ordem dos termos aqui mostrados foi a determinada pelo ERR.

$$\dot{y} = 2,13 \times 10^{-12} u^4 - 2,41 \times 10^{-12} y^4 + 3,46 \times 10^{-10} y^3 - 2,63 \times 10^{-10} y^2 u . \quad (5.1)$$

Os dois primeiros termos de (5.1) ( $u^4$  e  $y^4$ ) foram considerados os mais importantes segundo o critério ERR. Tais termos sempre estiveram no topo da lista fornecida pelo ERR para os modelos identificados, tanto para modelos de diferentes número de termos, como para modelos identificados com diferentes números de centros. Porém, do terceiro termo mais importante em diante, a ordem dos termos variava em função do número de centros. Isso poderia indicar que a relevância desses termos em comparação com os outros é muito grande e que não há, segundo o ERR, uma ordem muito clara para os termos a partir do terceiro. Isso está de pleno acordo com a teoria de

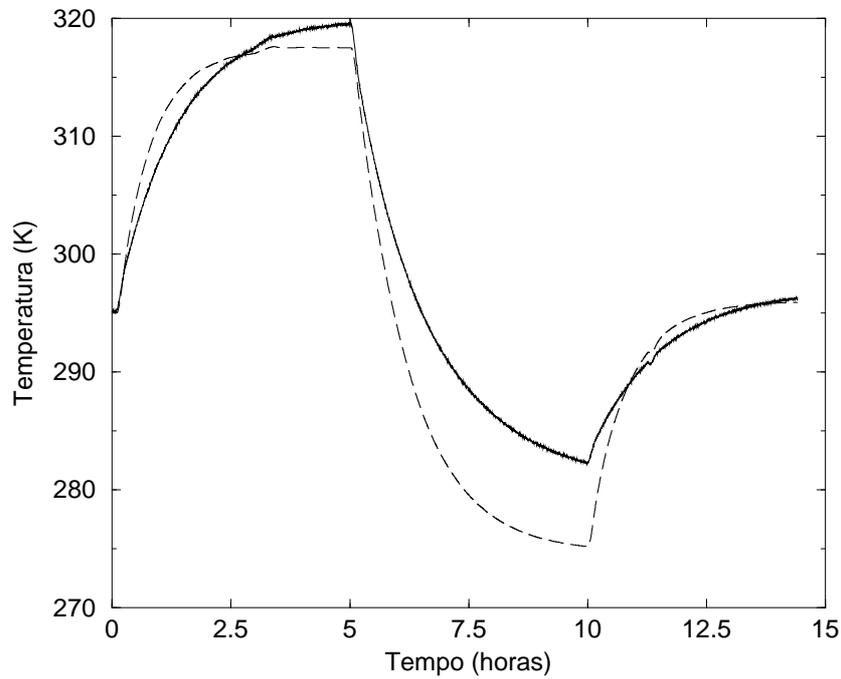


Figura 5.5: Validação do modelo de 12 termos. Modelo: linha tracejada, sistema: linha cheia.

transmissão de calor, segundo a qual o calor trocado por radiação entre dois corpos negros é proporcional à diferença das suas temperaturas absolutas elevadas à quarta potência.

A relevância dos termos  $y^4$  e  $u^4$  foi identificada pelo ERR sem auxílio de outras informações. Se ao invés de utilizar a detecção de estrutura, fossem procurados modelos polinomiais com número crescente de termos, incluindo os termos do menor grau para o maior, os termos mais importantes somente entrariam no modelo depois que todos os termos lineares, quadráticos e cúbicos já tivessem sido incluídos. Tais modelos muito provavelmente não seriam muito bons. É importante notar que essa forma de incluir termos em ordem crescente de graus não é uma boa idéia no presente caso, apesar de ser uma forma bastante intuitiva.

Como ilustração, na Figura 5.5 é mostrada a validação do modelo de 12 termos identificado com 100 centros. Na equação (5.2) é mostrado o modelo identificado.

$$\begin{aligned} \dot{y} = & -7,37 \times 10^{-11}u^4 - 2,58 \times 10^{-10}y^4 + 2,90 \times 10^{-7}y^3 + 2,69 \times 10^{-4}u^2 \\ & - 1,27 \times 10^{-4}y^2 - 8,10 \times 10^{-2}u + 2,67 \times 10^{-4}yu + 8,21 \times 10^{-11}y^2u^2 \\ & - 9,57 \times 10^{-7}yu^2 + 1,00 \times 10^{-9}yu^3 + 5,98 - 2,43 \times 10^{-7}u^3 . \end{aligned} \quad (5.2)$$

### 5.3 Identificação utilizando os modelos discretos

Com objetivo de comparação, foi feita a identificação do sistema utilizando modelos discretos.

Os dados utilizados na identificação dos modelos discretos foram os mesmos empregados na identificação dos modelos contínuos, sendo que foi adotado o mesmo procedimento de dividir os dados numa massa de identificação e numa massa de validação, ambas idênticas àsquelas da Seção 5.2. Houve apenas uma única diferença. Foram usados todos os pontos da massa de identificação, enquanto que no caso contínuo foram escolhidos alguns centros igualmente espaçados. Isso foi feito por motivo de simplicidade de implementação de código. Porém, empregar tal quantidade de pontos levou a problemas de mal-condicionamento numérico.

O resto do procedimento de identificação foi o mesmo, com temperaturas em Kelvin, grau máximo dos termos candidatos igual a 4, etc.

Novamente foram identificados 15 modelos, variando o número de termos. A Tabela 5.2 apresenta os erros de validação dos modelos identificados.

Nessa etapa, o modelo de melhor desempenho é o de 4 termos. Ele é mostrado na equação (5.3). A simulação desse modelo pode ser vista na Figura 5.6.

$$\begin{aligned} y(k) = & 1,00y(k-1) - 3,61 \times 10^{-5}u(k-1) - 2,06 \times 10^{-11}y^4(k-1) \\ & + 1,94 \times 10^{-11}u^4(k-1) . \end{aligned} \quad (5.3)$$

O modelo da equação (5.3) é bastante consistente com os demais contínuos. Com uma pequena manipulação na equação (5.3) tem-se

$$\begin{aligned} \frac{y(k) - 1,00y(k-1)}{10} = & -3,61 \times 10^{-6}u(k-1) - 2,06 \times 10^{-12}y^4(k-1) \\ & + 1,94 \times 10^{-12}u^4(k-1) , \end{aligned} \quad (5.4)$$

Tabela 5.2: Relação entre o número de termos do modelo identificado e o erro de predição para os modelos discretos.

Número de termos	Somatório dos erros quadráticos
1	$3,5825 \times 10^6$
2	$9,1887 \times 10^4$
3	$2,5841 \times 10^5$
4	2100,6
5	3295,9
6	$1,3746 \times 10^4$
7	$1,5588 \times 10^4$
8	$1,9811 \times 10^4$
9	$2,5680 \times 10^4$
10	$2,0043 \times 10^4$
11	$2,0033 \times 10^4$
12	$2,0908 \times 10^4$
13	$1,9858 \times 10^4$
14	$4,2662 \times 10^4$
15	$5,5394 \times 10^4$

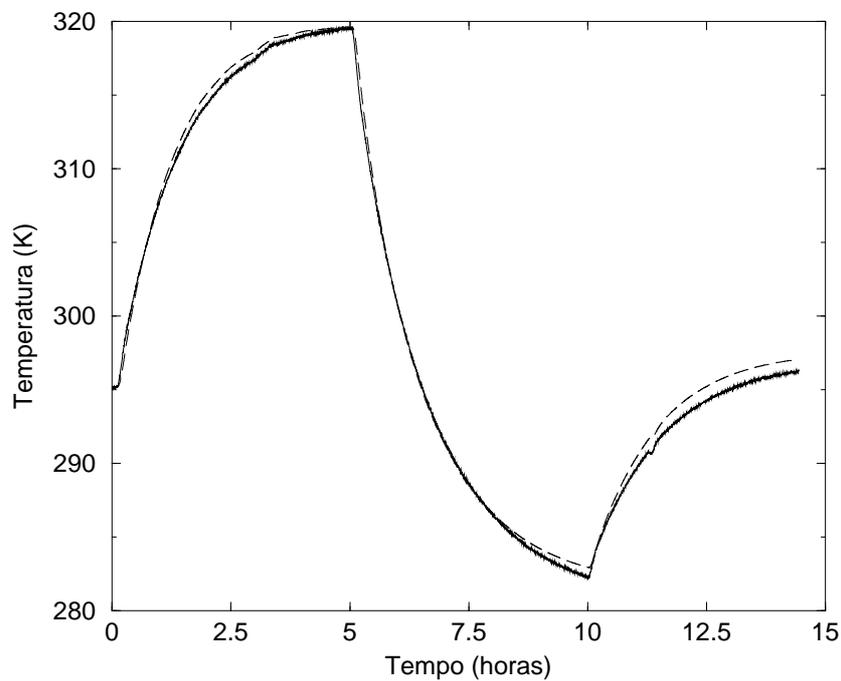


Figura 5.6: Validação do modelo discreto de 4 termos. Modelo: linha tracejada, sistema: linha cheia.

sendo que a constante 10 é o período de amostragem. O primeiro membro de (5.4) é uma aproximação para o sinal  $\dot{y}$ . Pode-se notar que os termos  $y^4(k-1)$  e  $u^4(k-1)$  têm coeficientes bastante próximos daqueles dos termos  $y^4$  e  $u^4$  da equação (5.1).

Por outro lado, na identificação discreta, esses termos não estão em tão grande evidência quanto no caso contínuo. Isso é possivelmente causado pela representação discreta necessitar do termo  $y(k-1)$  para poder representar corretamente essa dinâmica. Dessa forma, tal termo teria uma grande relevância e a sua inclusão no modelo de certa forma influenciaria a escolha dos demais termos.

O pior desempenho dos modelos discretos em relação aos contínuos pode ser explicado, em parte, pelos problemas de mal-condicionamento já citados. De qualquer forma, o desempenho do melhor modelo discreto é comparável aos dos melhores modelos contínuos. O valor eficaz dos erros (raiz quadrada da média dos valores quadráticos) desses modelos foram 0,37 K para o contínuo e 0,64 K para o discreto, o que é um bom desempenho.

## 5.4 Conclusão

O presente Capítulo mostrou a aplicação do método proposto no Capítulo 3 na identificação de uma câmara de termo-vácuo. O método se mostrou eficiente em capturar a dinâmica do sistema, principalmente no que se refere a identificar os termos fisicamente relevantes do processo.

Os modelos identificados foram validados com dados diferentes daqueles usados na identificação. Os melhores modelos obtidos foram capazes de prever adequadamente o sistema real.

Modelos de boa qualidade foram obtidos para com poucos termos, enquanto nenhum modelo muito complexo obteve um bom desempenho. Isso é uma forte indicação de que bons modelos não poderiam ser obtidos sem uso de detecção de estrutura.

Foi evidenciada a sensibilidade do algoritmo em relação ao número de centros usado na identificação.

A título de comparação, foram identificados modelos discretos para os mesmos dados. Tais modelos obtiveram um desempenho comparável, embora pior, àquele dos modelos contínuos, no que se refere à predição. Quanto a identificar os termos mais relevantes, os modelos contínuos foram capazes de melhor identificar os termos relevantes do que os discretos.

# Capítulo 6

## Discussão dos resultados e conclusões

### 6.1 Discussão

O presente trabalho apresentou um algoritmo de identificação de modelos dinâmicos contínuos não lineares que faz uso de técnicas de detecção de estrutura. Esse algoritmo foi descrito em detalhes e depois testado na identificação de três sistemas distintos: dois sistemas caóticos e um sistema térmico. Foram usados dados reais na identificação de dois desses sistemas: o circuito de Chua e o sistema térmico. No terceiro exemplo, o sistema de Lorenz, foram usados dados de simulação. Os sistemas caóticos foram usados no teste porque possuem comportamento bastante complexo e, assim, possibilitariam uma melhor análise do comportamento do procedimento de identificação proposto.

Na identificação de um dos sistemas caóticos, o sistema de Lorenz, o algoritmo foi capaz de, adequadamente, recuperar as características dinâmicas do sistema original. Já a identificação do circuito de Chua se mostrou bem mais difícil, não sendo possível a obtenção de bons modelos pela aplicação direta do procedimento. Porém, as técnicas de detecção de estrutura se mostraram capazes de melhorar a qualidade de modelos identificados sem detecção de estrutura. Particularmente, o uso de informação *a priori* sobre a simetria do sistema foi responsável pelo melhor resultado obtido.

Com relação à identificação de sistemas mais simples e de aplicação mais imediata, a identificação do sistema térmico — uma câmara de ensaio de satélites — mostrou o bom comportamento do método proposto. O seu desempenho foi tão bom quanto o do método discreto, senão melhor.

O algoritmo de identificação proposto se comportou bem, de maneira ge-

ral, sendo capaz de obter modelos válidos para sistemas com comportamento complexo.

Porém, algumas dificuldades foram observadas no processo de identificação. Primeiro, existem dificuldades associadas ao processo em si, tal como implementado nesse trabalho. Uma delas é a necessidade do cálculo numérico de derivadas de alta ordem de sinais amostrados. Esse cálculo é complicado e introduz incertezas no processo.

Uma outra dificuldade está relacionada à função de erro usada no método. Todo o processo de identificação é orientado de forma a minimizar o erro de estimação de uma derivada calculada numericamente. Porém, o que se deseja normalmente dos modelos identificados é um bom comportamento dinâmico. Como existem erros no cálculo dessa derivada, garantir que o modelo seja um bom estimador desse sinal não garante necessariamente a boa qualidade do seu comportamento dinâmico.

Isso pode explicar o melhor desempenho da identificação de modelos discretos para o circuito de Chua. Na identificação discreta, o que é minimizado é o erro de predição de um passo à frente. A predição de um passo à frente está mais diretamente relacionada à evolução dinâmica do sistema do que uma derivada de ordem elevada calculada numericamente. Talvez uma melhora significativa pudesse ser obtida se fosse utilizado o erro de predição livre. Uma indicação nesse sentido são os resultados apresentados em Timer et al. (2000). Nesse trabalho, os autores usam um método de estimação de parâmetros que minimiza o erro de predição de  $k$  passos à frente. Os resultados são bem promissores, principalmente tendo em vista que o sistema identificado é caótico. Porém, o problema de otimização associado ao método de estimação de parâmetros é bastante complexo.

Também foi observada uma deficiência na técnica de detecção de estrutura baseada no ERR. Ela se mostrou bem sensível ao conjunto de termos candidatos, pelo menos na identificação de um sistema complexo.

De qualquer forma, essas dificuldades não são fundamentais a ponto de inutilizar o método. O algoritmo de identificação funciona e é capaz de obter modelos para sistemas complexos, como sistemas caóticos.

O método de identificação apresentado é de utilização mais complexa do que sua alternativa discreta, os modelos NARMAX polinomiais. Porém, apresenta algumas vantagens. Um modelo contínuo é, em muitas aplicações, mais elucidativo do que um modelo discreto. Quando o objetivo da identificação não é apenas obter um modelo para predição, mas também obter informação sobre o funcionamento de um sistema, os modelos contínuos podem ser mais adequados do que modelos discretos. As variáveis do modelo podem ter significado físico (velocidades, acelerações), coisa que dificilmente ocorre com modelos discretos. Assim, um modelo obtido pelo método aqui

apresentado pode dar alguma indicação sobre os fenômenos que ocorrem no sistema em estudo.

Uma outra vantagem dos modelos contínuos é a facilidade com que informação sobre os pontos fixos do sistema pode ser incluída no processo de identificação. Apenas poucos parâmetros determinam os pontos fixos de um modelo contínuo monovariável, enquanto que, nos modelos NARMAX polinomiais, todos os parâmetros influenciam a localização dos mesmos (Aguirre e Souza, 1998). Isso é muito importante quando se deseja usar conhecimento *a priori* sobre os pontos fixos do sistema original no processo de identificação.

Também existem aplicações que exigem modelos contínuos. Por exemplo, muitas estratégias de controle de sistemas caóticos supõem o conhecimento do modelo do sistema, e esse modelo deve ser contínuo (Tôrres e Aguirre, 2000). Uma possibilidade seria identificar um sistema caótico e utilizar o modelo obtido para controle. Isso não seria tão direto se fossem identificados modelos discretos. Nesse caso, seria necessário adaptar a estratégia de controle para sistemas discretos ou obter um modelo contínuo a partir do modelo discreto.

## 6.2 Conclusões

A aplicação da detecção de estrutura na identificação de modelos contínuos não lineares se mostrou viável e importante no processo de obtenção de modelos. O algoritmo proposto foi capaz de identificar sistemas de comportamento dinâmico complexo como sistemas caóticos, bem como sistemas mais simples.

Mesmo quando o algoritmo não foi capaz de identificar modelos diretamente, ele melhorou a qualidade de modelos identificados sem detecção de estrutura.

A detecção de estrutura baseada no critério ERR se mostrou sensível ao conjunto de termos candidatos.

## 6.3 Propostas de continuidade

As possibilidades de investigação sobre o assunto de identificação de modelos contínuos não lineares estão longe de ser esgotadas. Muito trabalho ainda pode e deve ser feito a respeito.

Um dos caminhos a seguir é o uso de outros tipos de representações na identificação de modelos contínuos, como redes neurais MLP, redes RBF, e sistemas nebulosos. Essas representações têm, em geral, uma maior flexibilidade que pode beneficiar o processo de obtenção de modelos. Pode-se ainda estudar o uso de técnicas de detecção de estrutura na identificação

de modelos contínuos com essas representações, principalmente técnicas que envolvem o uso de informação *a priori*.

Outro ponto que merece investigação é o uso de outros tipos de erro na identificação, como o erro de predição livre. Isso pode ser crucial para a identificação bem sucedida de sistemas muito complexos.

Finalmente, seria importante aplicar o algoritmo apresentado nesse trabalho em mais sistemas, principalmente em aplicações de controle. Outra aplicação possível é a extração de parâmetros dos sistemas identificados através dos modelos identificados.

# Apêndice A

## Guia de Rotinas

### A.1 Introdução

Entre as formas de se obter modelos matemáticos para um sistema qualquer, uma que se destaca é o uso de técnicas de identificação de sistemas. Através dessas técnicas, pode-se obter modelos para sistemas complexos dispondo apenas de dados coletados desses sistemas, sem ter que passar pelo complicado (e nem sempre factível) procedimento de se derivar equações a partir da física do processo.

Existem vários tipos de representações que podem ser usadas na identificação de sistemas. Uma representação que tem atraído interesse recentemente (Gouesbet e Letellier, 1994) utiliza equações diferenciais ordinárias (EDO) não lineares para descrever o sistema em estudo. Tais EDO's são da forma

$$\frac{d\mathbf{x}}{dt} \equiv \dot{\mathbf{x}} = f(\mathbf{x}) , \quad (\text{A.1})$$

sendo  $\mathbf{x} \in \mathbf{R}^n$  é um vetor de variáveis de estado que possa caracterizar o sistema e  $f(\cdot): \mathbf{R}^n \rightarrow \mathbf{R}^n$  é uma função não linear polinomial.

O problema de identificação, nesta representação, consiste em determinar  $f(\cdot)$  em (A.1) de forma a representar satisfatoriamente o sistema em estudo, dispondo somente de séries temporais coletadas no sistema.

Para resolver esse problema, foram criadas uma série de rotinas que rodam em MATLAB e em GNU Octave. Esse pacote de rotinas baseou-se em um outro pacote utilizado para fazer a identificação de modelos discretos do tipo NARMAX polinomiais (Mendes e Aguirre, 1995).

As rotinas discretas foram utilizadas por dois motivos. Primeiramente, tanto o caso discreto quanto o contínuo se reduzem, no final, a um sistema de equações lineares sobredeterminado. O pacote discreto resolve esse problema

por um algoritmo de mínimos quadrados ortogonal. Como o problema é o mesmo, vale a pena utilizar as rotinas já prontas.

O mais importante motivo, no entanto, é que o pacote discreto utiliza uma técnica de detecção de estrutura baseada no critério ERR (Chen et al., 1989). Uma das motivações do trabalho era justamente aplicar essa técnica à identificação de modelos contínuos. Como essa técnica realiza a detecção de estrutura no sistema linear, é possível aproveitar as rotinas já codificadas.

O pacote de rotinas para identificação de modelos contínuos utiliza várias rotinas do pacote discreto. Nesse texto, apenas as informações relevantes sobre essas rotinas são fornecidas. Maiores detalhes podem ser obtidos em Mendes e Aguirre (1995).

Esse Apêndice tem por objetivo atuar como guia na utilização dessa rotinas.

## A.2 Coleta de dados e cálculo de derivadas

O primeiro passo na identificação é a coleta de dados.

Uma vez de posse dos dados, procede-se ao cálculo de suas derivadas. Se os modelos a serem identificados são do tipo multivariável, apenas a primeira derivada de cada variável de estado deve ser calculada.

A função que faz esse cálculo é a *deriva2.m*. Essa função calcula a derivada aproximando uma reta (por regressão linear) para uma pequena janela da série temporal e pegando a inclinação dessa reta como o valor da derivada no centro da janela. Assim, o tamanho da janela deve ser ímpar para que haja um valor central no qual se possa avaliar a inclinação da reta.

A função calcula a derivada para cada ponto da série temporal coletada, exceto para poucos pontos no início e no final da série. É interessante que esses pontos sejam descartados da série temporal antes de se proceder à identificação. Para isso, se  $k$  é o tamanho da janela, deve-se descartar os primeiros e os últimos  $\frac{k-1}{2}$  pontos da série temporal e da sua derivada.

O cálculo da derivada é feito dessa maneira para minimizar a influência do ruído. O tamanho da janela deve ser tanto maior quanto for o nível de ruído presente nos dados, porém não tão grande a ponto de obliterar a dinâmica do sistema. É necessária alguma experimentação na determinação desse tamanho de janela, mas um valor em torno de 5 é um bom começo.

Essa função assume que o período de amostragem é unitário. Se isso não for verdade, deve-se dividir o vetor calculado pelo período de amostragem para se obter o valor correto.

A função *deriva2.m* recebe como parâmetro uma matriz cujas colunas são as séries temporais das quais se deseja calcular as derivadas e retorna outra

matriz do mesmo tamanho cujas colunas são as derivadas das respectivas colunas da matriz de entrada. Assim, pode-se calcular de uma só vez as derivadas de várias séries temporais.

Se os modelos a serem identificados são do tipo monovariável, é necessário calcular derivadas de ordem elevada, tipicamente maior que terceira. Tal cálculo não é simples. Uma alternativa é utilizar sucessivamente a função *deriva2.m*, porém é possível que a acumulação de erros no processo torne tal abordagem impraticável.

### A.3 O processo de identificação

O cálculo das derivada é lento e independente do resto do processo de identificação. Porém, as demais etapas são mais rápidas e, portanto, foram agrupadas numa rotina mestre que chama as demais funções. Existem duas rotinas mestres: uma para modelos monovariáveis e outra para modelos multivariáveis. Essas rotinas se encontram reproduzidas aqui e as etapas são explicadas passo-a-passo.

É aconselhável utilizar essas rotinas mestres como base para o processo de identificação, apenas alterando os parâmetros que foram necessários.

#### A.3.1 Modelos monovariáveis — a rotina *contin.m*

De agora em diante, trechos de rotinas serão impressos em tipos diferentes como:

```
% Rotina de obtensao de modelos Continuos
% Ubiratan Santos Freitas (04/98)
```

Este é o começo do arquivo *contin.m*. A seguir, tem-se alguma inicialização.

```
% Limpa o espaco de trabalho
clear
clear global;
```

```
%Declara variavel global que sera
%utilizada na simulacao do modelo
global xn;
```

O nome da variável *xn* deve ser mantido pois essa variável é usada para transferência de dados para outra rotina (a *si.m.m*).

```
%Tamanho da janela de dados
jan=4500;
```

Esse é o tamanho da janela da qual serão escolhidos os centros para a estimação de parâmetros. Essa janela não tem nenhuma relação com a janela utilizada no cálculo de derivadas.

```
% Carrega e prepara os dados
disp('Carregando os dados...');
load lor.mat
clear y;
```

Neste ponto são lidas a série temporal e suas derivadas previamente calculadas de um arquivo. Os dados devem estar na forma de uma matriz cuja primeira coluna contém a série temporal; a segunda, sua derivada primeira; a terceira, a derivada segunda e assim por diante. Uma quantidade de linhas dessa matriz, maior ou igual ao tamanho da janela, deve ser copiada para a variável *vc5d*. Apenas as primeiras *jan* linhas serão utilizadas como domínio para a escolha de centros.

```
% Normalizacao:
[vc5d,a]=normono(vc5d);
```

Esse comando chama a função *normono.m*, responsável por fazer a normalização da matriz de dados para modelos monovariáveis. Tal normalização tem o objetivo de deixar todas as colunas da matriz de dados com aproximadamente a mesma ordem de grandeza para evitar problemas de mal condicionamento numérico na estimação de parâmetros. Isso é feito encontrando um período de amostragem fictício para o qual as derivadas sucessivas tenham a mesma ordem de grandeza do sinal original. As contas realizadas são

$$a = \sqrt[n]{\frac{\max |x_i|}{\max \left| \frac{d^n}{dt^n} x_i \right|}}, \quad i = 1, 2, \dots, N \quad (\text{A.2})$$

$$xn_i^{(p)} = a^p \frac{d^p}{dt^p} x_i \quad i = 1, 2, \dots, N \quad p = 1, 2, \dots, n$$

sendo que  $a$  é o fator de normalização,  $n$  é a ordem do modelo a ser identificado,  $x_i$  são as amostras do sinal,  $N$  é o número de amostras e  $xn_i^{(p)}$  é a  $p$ -ésima derivada normalizada.

A função *normono.m* recebe como argumento a matriz de dados e retorna a mesma matriz normalizada e o fator  $a$ . Esse fator pode então ser utilizado

na desnormalização do modelo identificado e na determinação do passo de integração da simulação dos modelos identificados, caso esta se realize com os modelos normalizados.

```
vc5d=vc5d(1:jan,:);

%Ordem
[ans,Ord]=size(vc5d);
Ord=Ord-1;

%Numero de iteracoes para o modelo de ruido
N=0;

%Numero de termos lineares de ruido a acrescentar
nno=0;

%      Numero de centros
nc=1125;
```

Nesse ponto a matriz de dados é truncada para ter *jan* linhas. A seguir, a ordem do modelo a ser identificado é determinada a partir do número de colunas da matriz de dados. Assim, para aumentar a ordem do modelo a ser identificado, basta calcular mais derivadas e aumentar a matriz de dados.

O valor de *N* só tem sentido se for utilizada a modelagem de ruído. É necessário analisar os resíduos de identificação para verificar a convergência do processo iterativo, porém um valor para *N* em torno de 5 ou 6 normalmente é suficiente.

Por uma questão de simplicidade, apenas termos lineares de ruído são acrescentados ao conjunto de termos candidatos. O valor de *nno* determina quantos termos são acrescentados e também quantos serão utilizados no modelo. Caso de deseje um comportamento diferente deve-se alterar a rotina.

O número de centros *nc* é bastante importante no processo de identificação. Apenas *nc* linhas da matriz de dados são utilizadas na identificação, mesmo depois que ela foi reduzida para ter *jan* linhas. Essas linhas são escolhidas igualmente espaçadas ao longo da matriz de dados pela função *esc\_cen.m*. Devido à forma como a escolha de centros é feita, é necessário que *nc* seja pelos menos 6 unidades menor que *jan*.

```
% Nome do arquivo de saida
sai='contin.txt';
```

```

% Abre arquivo de resposta
arq=fopen(sai, 'wt');
fprintf(arq, '          Resultado continuo lorenz y\n\n');
fprintf(arq, 'Taman. jan.: %d, N: %d, nno: %d, ...
          Centros: %d, Ord.: %d\n\n', ...
          jan, N, nno, nc, Ord);
fprintf(arq, 'Tipo Sim.  np  TC  Atrator  Pontos Fixos \n');
fclose(arq);

```

Esse trecho da rotina cria o arquivo de saída e escreve o seu cabeçalho. O arquivo de saída serve para se manter um controle sobre o processo de identificação, fornecendo um registro de informações relevantes. O arquivo de saída é especialmente importante quando se identificam vários modelos de uma só vez.

Esse arquivo é aberto e fechado após a identificação e simulação de cada modelo, de modo a garantir a sua integridade caso o processo seja interrompido. Os dois primeiros comandos *fprintf* escrevem o título e um cabeçalho com informações sobre a batelada de identificações. O terceiro comando *fprintf* descreve o tipo de informação que será colocada para cada modelo identificado, marcando as colunas de uma tabela. As linhas da tabela serão preenchidas, uma por modelo, após a identificação e simulação dos mesmos.

```

%Contador de simulacao
si_c=1;

% Grau de nao linearidade
Grau=5;

% Gera os termos candidatos
[C,total]=genterms(Grau,Ord);

```

O contador de simulação é incrementado a cada modelo identificado. A variável *Grau* contém o grau de não linearidade máximo dos monômios candidatos gerados. A função *genterms.m* gera os termos candidatos.

Essa função pertence ao pacote de identificação NARMAX. Ela recebe o grau máximo dos monômios e a ordem do modelo a ser identificado e gera todos os termos possíveis com essas características. Os termos são retornados numa matriz. O número total de termos gerados também é retornado.

Esses termos utilizam uma forma própria de representação que é detalhada a seguir.

Tabela A.1: Exemplos de representações de termos

Termo	Representação
[ <i>constante</i> ]	0 0 0
$x_1$	1001 0 0
$x_2^2$	1002 1002 0
$x_1^2 x_3$	1001 1001 1003
$x_1^3$	1001 1001 1001

A matriz de termos tem tantas linhas quanto o número de termos e tantas colunas quanto o grau do termo de maior grau. Cada linha representa um termo e é formada por uma série de números inteiros. O número de elementos não nulos dessa linha é igual ao grau do termo por ela representado. Cada elemento não nulo é um inteiro de quatro dígitos, sendo que os dois primeiros são sempre 1 e 0 e os dois últimos formam um número que varia de 01 a 99.

Cada um desses números inteiros representa uma variável. 1001 é o sinal amostrado ( $x_1$ ), 1002 é a primeira derivada do sinal ( $x_2$ ), 1003 é a segunda derivada do sinal ( $x_3$ ) e assim por diante.

Dessa forma, um termo do tipo  $x_1^2 x_3$  seria formado pela linha

$$1001 \ 1001 \ 1003 .$$

A tabela A.1 mostra mais alguns exemplos de representações de termos.

`% Retira os termos tipo x^n`

```
[C,total]=ranca(C,total);
C=[1001 zeros(1,Grau-1);1001 1001 1001 zeros(1,Grau-3);C];
total=total+2;
Cand=[Cand;(3001:3000+nno)' zeros(nno,Grau-1)];
```

A função *ranca.m* retira do conjunto de termos candidatos todos os termos do tipo  $x_1^n$ , para todo  $n$ . Isso é feito quando se deseja aplicar algum eventual conhecimento *a priori* sobre os pontos fixos do sistema na seleção de estrutura.

No caso dos modelos monovariáveis tratados aqui, o número e os valores dos pontos fixos dos modelos identificados são determinados pelos coeficientes dos termos do tipo<sup>1</sup>  $x_1^n$ . Assim, informações sobre o número e/ou a simetria dos pontos podem ser utilizadas para determinar quais termos do tipo  $x_1^n$  incluir no conjunto de termos candidatos.

<sup>1</sup>Mais precisamente, os pontos fixos são as raízes da equação  $\sum_{i=0}^{\infty} \theta_i r^i = 0$ , sendo  $\theta_i$  os coeficientes dos termos  $x_1^i$ .

Essas informações podem ser obtidas através de rotinas de cálculo de pontos fixos (Aguirre e Souza, 1998).

Para incluir apenas determinados termos deve-se primeiro excluir todos os termos do tipo  $x_1^n$  dos termos candidatos e depois incluir os termos desejados.

A função *ranca* recebe uma matriz com os termos candidatos e o número de termos e retorna outra matriz sem os termos  $x_1^n$  e o número de termos dessa nova matriz.

A seguir, dois termos, um do tipo  $x_1$  e o outro do tipo  $x_1^3$ , são incluídos no começo da matriz de termos candidatos. Dessa forma o modelo identificado terá 3 pontos fixos, sendo um nulo e os outros dois simétricos.

Para que esses termos sejam realmente incluídos no modelo identificado, é necessário que eles ocupem as primeiras posições da matriz de termos candidatos e a função *corthre2* seja alterada para tal. Para alterar *corthre2* deve-se editar o arquivo, descomentar as linhas que fazem referência à função *housels2* e comentar as linhas referentes à função *housels*.

```
% Laco do numero de termos candidatos
for TC=30:min([56, total])
    Cand=C(1:TC, :);
    tot=TC;
```

```
% Laco do numero de termos
for np=20:min([30,tot])
```

Esse trecho mostra os dois laços de identificação utilizados, um para o número de termos candidatos e o outro para o número de termos incluídos no modelo.

*A priori*, não é conhecido o número de termos que o modelo identificado deve conter. O algoritmo do ERR apenas ordena os termos em ordem decrescente de importância. Um dos parâmetros que o usuário deveria informar para a rotina de identificação seria o número de termos que o modelo deve ter. Foi observado que a qualidade do modelo identificado é muito dependente desse parâmetro. Se o número de termos incluídos for muito pequeno, o modelo pode não ter flexibilidade suficiente para representar a função de interesse. Por outro lado, um excesso de termos acarreta problemas de sobreparametrização (Aguirre e Billings, 1995).

Outro parâmetro sensível é o número de termos candidatos. Em alguns casos, foi observado que conjuntos de termos candidatos muito grandes dificultaram o processo de identificação.

Como esses dois parâmetros se mostraram críticos no processo e ainda não existe uma forma segura de determiná-los, procede-se a uma busca exaustiva em duas dimensões: uma para os termos candidatos  $TC$  e outra para o número de termos do modelo identificado  $np$ .

Dos dois laços, o mais importante é o interno. Se a pesquisa em duas variáveis for muito lenta, deve-se travar o primeiro laço.

```
disp(sprintf('Simulacao: %d',si_c))

% Identificacao
disp('Identificando...')
[model,xls,e,va,y0]=corthre2(Cand,vc5d,[np nno],N,nc);
```

Nesse ponto, duas mensagens são impressas na tela e a rotina de identificação *corthre2* é chamada.

Essa rotina recebe cinco parâmetros e retorna também cinco parâmetros.

O primeiro parâmetro de entrada é a matriz contendo os termos candidatos na forma explicada anteriormente. O segundo parâmetro é a matriz contendo a série temporal e suas derivadas. A ordem do modelo identificado é o número de colunas dessa matriz menos um. O terceiro parâmetro é um vetor de duas posições, sendo que a primeira é o número de termos do modelo identificado e a segunda é o número de termos de ruído acrescentados ao modelo. O quarto parâmetro é o número de iterações para o modelo de ruído. O quinto parâmetro é o número de centros que serão usados na identificação.

O primeiro parâmetro de saída é uma matriz com os termos do modelo identificado. O segundo parâmetro retornado consiste de uma matriz cuja primeira coluna contém os coeficientes dos termos do modelo identificado; a segunda, os valores ERR para os respectivos termos; e a terceira, os desvios padrão para os respectivos coeficientes. O terceiro parâmetro retornado é o vetor dos resíduos de identificação. O quarto parâmetro é a variância. O quinto parâmetro retornado é um vetor com as coordenadas no espaço de estados reconstruído do primeiro centro usado na identificação. Esse parâmetro é retornado para ser usado como condição inicial na simulação do modelo identificado.

A função *corthre2* é a principal rotina de identificação. Essa rotina recebe seus parâmetros e faz neles testes de consistência. A seguir, chama a função *cbl\_d\_pr2* passando como parâmetros a matriz de termos candidatos, a matriz com a série temporal e suas derivadas e o número de centros.

A função *cbl\_d\_pr2* é responsável por montar o problema de mínimos quadrados. Esse problema tem a forma

$$\Psi\theta = y, \quad (\text{A.3})$$

sendo  $\theta$  o vetor de coeficientes dos termos do modelo a ser identificado,  $y$  o vetor independente (no caso, a maior derivada calculada a partir da série temporal), e  $\Psi$  a matriz de regressores. Nessa matriz, cada coluna representa um termo e cada linha, um centro.

A equação (A.3) representa

$$\frac{d^{n+1}x_i(t)}{dt^{n+1}} = \sum_{p=0}^{p=np} \theta_p P_p(t), \quad (\text{A.4})$$

escrita para todos os centros. Em (A.4),  $x_i$  é a série temporal observada,  $n$  é a ordem do modelo identificado e os  $P_p$  são os termos.

O que a função *cbl\_d\_pr2* faz é criar a matriz  $\Psi$  e o vetor  $y$  de acordo com o conjunto de termos candidatos e com a posição dos centros. Essa função retorna três parâmetros. O primeiro é a matriz de regressores  $\Psi$ . O segundo é o vetor independente  $y$ . O terceiro é o vetor de condições iniciais retornado por *corthre2*.

A função *cbl\_d\_pr2* chama a função *esc\_cen* para calcular a posição dos centros. Essa função retorna um vetor com os índices dos centros. Ela recebe como parâmetros de entrada a série temporal, o número de centros e um número inteiro que informa quantas amostras do começo e do final da série devem ser ignoradas na escolha dos centros. A função *esc\_cen* escolhe os centros igualmente espaçados no *tempo*.

Uma vez que o problema de mínimos quadrados (A.3) está montado, a identificação dos modelos contínuos é idêntica a dos modelos discretos. Portanto, a rotina *corthre2* chama a função *houses* usada na identificação de modelos discretos. A única diferença em relação ao processo discreto é que existe a possibilidade de se chamar a rotina *houses2* para que os dois primeiros termos candidatos estejam presentes no modelo identificado.

```
xn=xls(:,1);
```

```
% Calcula os pontos fixos
pf=pon_fi(model,xls(:,1));
```

A seguir, os coeficientes dos termos do modelo identificado são copiados para a variável global  $xn$ . Essa variável será utilizada na simulação do modelo identificado.

A chamada de função seguinte calcula os pontos fixos do modelo identificado. A função *pon\_fi* recebe a matriz de termos do modelo identificado e os seus coeficientes e retorna um vetor com os pontos fixos do modelo.

Os pontos fixos de um modelo contínuo monovariável identificado são as raízes da equação

$$\theta_0 + \sum_{k=1}^l \theta_k \rho^k = 0, \quad (\text{A.5})$$

sendo que  $\theta_0$  é o termo independente e os  $\theta_k$  são os parâmetros dos termos do modelo identificado que dependem apenas da série temporal.

```
% Prepara a simulacao e simula
disp('Simulando...')
psim(model(1:np,:),Ord);
!smartdrv /c
[t,ym]=ode45p('si_m',0,28000/(a),y0',1/(a)*7,0);
```

Nesse ponto, depois de imprimir outra mensagem, é chamada a função *psim*. Essa função recebe a matriz com os termos do modelo identificado e a ordem do modelo e não retorna nada. Ao invés de retornar parâmetros, essa função cria um arquivo chamado *si\_m.m* que é uma função do MATLAB. Essa função está organizada de tal forma que pode ser chamada pela função de simulação de sistemas dinâmicos do MATLAB quando da simulação do modelo identificado. A simulação é feita dessa forma para diminuir o tempo de processamento, já que esta é a etapa mais lenta do processo de identificação.

A função *si\_m* utiliza a variável global *xn* para receber os coeficientes do modelo identificado.

A seguir, é executada uma chamada ao sistema operacional para que seja dado um *flush* no *cache* de disco. Isso é necessário quando a função *psim* sobrescreve um arquivo *si\_m.m* previamente existente. Sem essa operação de *flush*, o MATLAB carregará o arquivo anterior.

Depois é chamada a função de simulação. Deve-se notar que o modelo foi identificado utilizando dados normalizados. Como a normalização é um escalonamento do tempo, esse escalonamento deve ser levado em conta no momento da simulação. Uma alternativa é retornar o modelo identificado para o espaço não normalizado. Isso pode ser feito através da função *dnormon*.

A função *dnormon* recebe quatro parâmetros e retorna um. O primeiro parâmetro de entrada é o vetor com os coeficientes do modelo identificado. O segundo parâmetro é a matriz com os termos do modelo. O terceiro parâmetro é a ordem do modelo identificado e o quarto é o fator de escala de tempo. A função retorna um vetor com os coeficientes do modelo identificado no espaço não normalizado.

```
% "Mede" o tamanho do atrator
n=conta(ym(1:4000,1),12,30,-30,2000,0);
```

De posse do resultado da simulação procede-se à validação do modelo. Nessa etapa, é feita apenas uma validação muito simples, o suficiente para se ter uma idéia aproximada de quais são os maiores valores para  $TC$  e  $np$ . Se o sistema a ser identificado exibe comportamento caótico, a simples comparação da saída do modelo identificado com a do sistema original não diz nada a respeito da qualidade do modelo identificado. Uma alternativa é a comparação dos atratores do modelo e do sistema original.

Para isso pode-se utilizar a função *conta*. Essa função gera uma reconstrução por coordenadas de atraso do atrator do modelo e projeta-o sobre um plano. Depois divide esse plano em 22500 ( $150 \times 150$ ) regiões e conta quantas dessas regiões são ocupadas pelo atrator.

A função *conta* recebe seis parâmetros de entrada e retorna o número de regiões diferentes ocupadas. O primeiro parâmetro de entrada é a série temporal gerada pelo modelo identificado. O segundo parâmetro é o atraso de reconstrução. O terceiro e quarto parâmetros são os limites superior e inferior da região de procura. O quinto parâmetro é o número de amostras que se deve descartar do início da série temporal para se evitar transitórios. O sexto parâmetro é um *flag*. Se esse *flag* for diferente de zero uma figura é plotada mostrando as caixas que estão ocupadas.

Para se ter uma base de comparação, deve-se rodar a função *conta* com a série temporal do sistema original e anotar o resultado antes de se executar a rotina com séries produzidas por modelos identificados.

```
    arq=fopen('contin.txt','at');
    fprintf(arq,'lory %2d %2d %2d %5d      ',...
            si_c,np,TC,n);
    for i=1:length(pf)
        fprintf(arq,'%f ',pf(i));
    end,

    fprintf(arq,'\n');
    fclose(arq);

    si_c=si_c+1;
end
end
```

Essa parte é o final do arquivo *contin.m*. Nesse ponto, as informações sobre o processo de identificação são escritas no arquivo de saída.

A seguir, o contador de simulação é incrementado. Os dois comandos *end* finalizam os laços de número de termos e número de termos candidatos.

Esse é o fim do arquivo *contin.m*.

# Referências Bibliográficas

- Aguirre, L. A. (1995). A nonlinear correlation function for selecting the delay time in dynamical reconstructions. *Phys. Lett.*, 203A(2,3):88–94.
- Aguirre, L. A. (2000a). *Introdução à Identificação de Sistemas: Técnicas Lineares e Não Lineares Aplicadas a Sistemas Reais*. Editora UFMG.
- Aguirre, L. A. (2000b). A nonlinear dynamical approach to system identification. *IEEE Circuits Syst. Newsletter*, 11(2):10–23.
- Aguirre, L. A. e Billings, S. A. (1994). Validating identified nonlinear models with chaotic dynamics. *Int. J. Bifurcation and Chaos*, 4(1):109–125.
- Aguirre, L. A. e Billings, S. A. (1995). Dynamical effects of overparametrization in nonlinear models. *Physica D*, 80(1,2):26–40.
- Aguirre, L. A., Freitas, U. S., Letellier, C., e Maquet, J. (2001). Structure selection techniques applied to continuous-time nonlinear models. *Physica D*, 158(2):1–18.
- Aguirre, L. A., Freitas, U. S., Letellier, C., Sceller, L. L., e Maquet, J. (2000). State space reconstruction and topological analysis of attractors produced by an electronic oscillator. In Broomhead, D. S., Luchinskaya, E. A., McClintock, P. V. E., e Mullin, T., editors, *Stochastic and Chaotic Dynamics in the Lakes: STOCHAOS*, volume 502, pages 649–654. American Institute of Physics. Conference Location and Date: Ambleside, Cumbria, UK, August 16-20, 1999.
- Aguirre, L. A. e Mendes, E. M. (1996). Nonlinear polynomial models: Structure, term clusters and fixed points. *Int. J. Bifurcation and Chaos*, 6(2):279–294.
- Aguirre, L. A., Rodrigues, G. G., e Mendes, E. M. (1997). Nonlinear identification and cluster analysis of chaotic attractors from a real implementation of Chua’s circuit. *Int. J. Bifurcation and Chaos*, 7(6):1411–1423.

- Aguirre, L. A. e Souza, A. V. P. (1998). An algorithm for estimating fixed points of dynamical systems from time series. *Int. J. Bifurcation and Chaos*, 8(11):2203–2213.
- Aguirre, L. A. e Tôrres, L. A. B. (1997). Fixed point stability analysis of chua's circuit: A case study with a real circuit. *Journal of Circuits, Systems and Computers*, 7(2):111–115.
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Trans. Automat. Contr.*, 19(6):716–723.
- Araújo Filho, J. E. (2000). Projeto e desenvolvimento do sistema de controle da câmara de teste termo-vácuo. Relatório Técnico.
- Araújo Filho, J. E. e Macau, E. E. N. (1999). Thermal vacuum system controlled by fuzzy reasoning. In *3rd World Multiconference on Systemics Cybernetics and Informatics and 5th International Conference on Information Systems Analysis and Synthesis*, volume 7, pages 1–7, Orlando, USA.
- Baake, E., Baake, M., Bock, H. G., e Briggs, K. M. (1992). Fitting ordinary differential equations to chaotic data. *Phys. Rev. A*, 45(8):5524–5529.
- Bezruchko, B. P. e Smirnov, D. A. (2000). Constructing nonautonomous differential equations from experimental time series. *Phys. Rev. E*, 63(1):1–7.
- Billings, S. A. e Chen, S. (1989). Identification of nonlinear rational systems using a predictor-error estimation algorithm. *Int. J. Systems Sci.*, 20(3):467–494.
- Billings, S. A., Chen, S., e Korenberg, M. J. (1989). Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator. *Int. J. Control*, 49(6):2157–2189.
- Billings, S. A. e Voon, W. S. F. (1986). Correlation based model validity tests for non-linear models. *Int. J. Control*, 44(1):235–244.
- Box, G. E. P. e Jenkins, G. M. (1970). *Time series analysis, Forecasting and Control*. Holden-Day, San Francisco.
- Broomhead, D. S. e Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems*, 2:321–355.

- Brown, R., In, V., e Tracey, E. R. (1997). Parameters uncertainties in models of equivariant dynamical systems. *Physica D*, 102.
- Brown, R., Rulkov, N., e Tufillaro, N. (1994). Nonlinear prediction of chaotic time series. *Phys. Rev. E*, 50(6):4488–4508.
- Caminhas, W. M. e Gomide, F. A. C. (2000). A fast learning algorithm for neofuzzy networks. In *8th IPMU*, pages 1784–1790, Madrid, Spain.
- Casdagli, M., Jardins, D. D., Eubank, S., Farmer, J. D., Gibson, J., Theiler, J., e Hunter, N. (1992). Nonlinear modeling of chaotic time series: theory and applications. In Kim, J. H. e Stringer, J., editors, *Applied Chaos*, chapter 15, pages 335–380. John Wiley & Sons., New York.
- Cassini, C. (1999). Identificação recursiva de características estáticas não lineares utilizando modelos polinomiais NARMAX. Dissertação de mestrado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Celka, P., Vesin, J., Vetter, R., Grueter, R., Thonet, G., Pruvot, E., Duplain, H., e Scherrer, U. (1999). Parsimonious modeling of biomedical signals and systems: Applications to the cardiovascular system. In Akay, M., editor, *Nonlinear Biomedical Signal Processing Part II*. IEEE Press.
- Chen, C. T. (1993). *Analog and Digital Control System Design*. Saunders, London.
- Chen, S., Billings, S. A., e Luo, W. (1989). Orthogonal least squares methods and their application to nonlinear system identification. *Int. J. Control*, 50(5):1873–1896.
- Chua, L. O. (1992). The genesis of Chua's circuit. *Archiv für Elektronik und Übertragungstechnik*, 46(4):250–257.
- Corrêa, M. (1997). Identificação de sistemas dinâmicos não lineares utilizando modelos NARMAX racionais — aplicação a sistemas reais. Dissertação de mestrado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Corrêa, M. V. (2001). *Identificação Caixa Cinza de Sistemas Não Lineares Utilizando Representações NARMAX Polinomiais e Racionais*. Tese de doutorado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.

- Cremers, J. e Hübler, A. (1987). Construction of differential equations from experimental data. *Z. Naturforsch*, 42a:797–802.
- Elgar, S. and Vanhoff, B., Aguirre, L., Freitas, U., e Chandran, V. (1998). Higher-order spectra of nonlinear polynomial models for Chua's circuit. *Int. J. Bifurcation and Chaos*, 8(12):2425–2431.
- Elsner, J. B. (1992). Predicting time series using a neural network as a method of distinguishing chaos from noise. *J. Phys. A: Math. Gen.*, 25:843–850.
- Ferreira, A. B. d. H. (1986). *Novo Dicionário da Língua Portuguesa*. Editora Nova Fronteira, segunda edição.
- Fiedler-Ferrara, N. e Prado, C. P. C. (1994). *Caos: uma introdução*. Editora Edgard Blücher LTDA.
- Fonseca, C. M., Mendes, E. M. A. M., Fleming, P. J., e Billings, S. A. (1993). Nonlinear model term selection with genetic algorithms. In *IEE/IEEE Workshop on Natural Algorithms in Signal Processing*, pages 27/1–27/8, Chelmsford, Essex.
- Freitas, U. S. (1998). Identificação e validação de modelos NARMAX para um oscilador caótico. In *VII Semana de Iniciação Científica da Universidade Federal de Minas Gerais*, page 315.
- Garcia, C. (1997). *Modelagem e simulação de processos industriais e de sistemas eletromecânicos*. EDUSP.
- Gibson, J. F., Farmer, J. D., Casdagli, M., e Eubank, S. (1992). An analytic approach to practical state space reconstruction. *Physica D*, 57:1.
- Giona, M., Lentini, F., e Cimagalli, V. (1991). Functional reconstruction and local prediction of chaotic time series. *Phys. Rev. A*, 44(6):3496–3502.
- Gleick, J. (1991). *Caos: a criação de uma nova ciência*. Editora Campus, quarta edição.
- Golub, G. H. e Van Loan, C. F. (1989). *Matrix Computations*. 2nd Edition, Johns Hopkins, London.
- Gouesbet, G. e Letellier, C. (1994). Global vector field reconstruction by using a multivariate polynomial  $L_2$  approximation on nets. *Phys. Rev. E*, 49(6):4955–4972.

- Haber, R. (1985). Nonlinearity tests for dynamic processes. In *IFAC Identification and System Parameter Estimation*, pages 409–413.
- Henrique, H. M., Lima, E. L., e Seborg, D. E. (2000). Model structure determination in neural networks models. *Chemical Engineering Science*, 55:5457–5469.
- Kadtke, J. B., Brush, J., e Holzfuss, J. (1993). Global dynamical equations and Lyapunov exponents from noisy chaotic time series. *Int. J. Bifurcation and Chaos*, 3(3):607–616.
- Lainscsek, C. S. M., Schürerer, F., e Kadtke, J. B. (1998). A general form for global dynamical data models for three-dimensional systems. *Int. J. Bifurcation and Chaos*, 8(5):899–914.
- Leontaritis, I. J. e Billings, S. A. (1985a). Input-output parametric models for nonlinear systems part I: deterministic nonlinear systems. *Int. J. Control*, 41(2):303–328.
- Leontaritis, I. J. e Billings, S. A. (1985b). Input-output parametric models for nonlinear systems part II: stochastic nonlinear systems. *Int. J. Control*, 41(2):329–344.
- Leontaritis, I. J. e Billings, S. A. (1987). Experimental design and identifiability for nonlinear systems. *Int. J. Systems Sci.*, 18(1):189–202.
- Letellier, C., Le Sceller, L., Dutertre, P., Gouesbet, G., Fei, Z., e Hudson, J. L. (1995). Topological characterization and global vector field reconstruction of an experimental electrochemical system. *Journal of Physical Chemistry*, 99(18):7016–7027.
- Letellier, C., Maquet, J., Le Sceller, L., Gouesbet, G., e Aguirre, L. A. (1998). On the non-equivalence of observables in phase-space reconstructions from recorded time series. *J. Phys. A: Math. Gen.*, 31:7913–7927.
- Ljung, L. (1987). *System Identification, Theory for the User*. Prentice Hall, New Jersey.
- Ljung, L. e Söderström, T. (1983). *Theory and practice of recursive identification*. MIT press, Cambridge, MA.
- Lorenz, E. (1963). Deterministic nonperiodic flow. *J. Atmos. Sci.*, 20:282–293.

- Mao, K. Z. e Billings, S. A. (1997). Algorithms for minimal model structure detection in nonlinear dynamic system identification. *Int. J. Control*, 68(2):311–330.
- Mendes, E. M. A. e Aguirre, L. A. (1995). Orthogonal estimation of nonlinear models. Relatório técnico, UFMG. CPDEE/LCPI/01-95.
- Mendes, E. M. A. M. e Billings, S. A. (1998). On overparametrization of nonlinear discrete systems. *Int. J. Bifurcation and Chaos*, 8(3):535–556.
- Ogata, K. (1993). *Engenharia de Controle Moderno*. Segunda Edição, Prentice Hall do Brasil, São Paulo.
- Oliveira, P. J. R. (1998). Estimação de parâmetros do motor de indução utilizando um modelo contínuo no tempo. Dissertação de mestrado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Peixoto, Z. M. A. (2000). *A Estimação de Parâmetros e os Conversores Multiníveis com Capacitor Flutuante no Acionamento de Máquinas de Indução*. Tese de doutorado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Perona, P., Porporato, A., e Ridolfi, L. (2000). On the trajectory method for the reconstruction of differential equations from time series. *Nonlinear Dynamics*, 23(1):13–33.
- Reed, R. (1993). Pruning algorithms — a survey. *IEEE Trans. Neural Networks*, 4(5):740–746.
- Rodrigues, G. G. (1996). Identificação de sistemas dinâmicos não lineares utilizando modelos NARMAX polinomiais – aplicação a sistemas reais. Dissertação de mestrado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Rodrigues, G. G., Aguirre, L. A., Freitas, U. S., e Pena, R. T. (1996). Identificação de um forno elétrico não linear utilizando modelos polinomiais NARMAX. In *XI Congresso Brasileiro de Automática*, pages 953–958, São Paulo, S.P.
- Rosenstein, M. T., Collins, J. J., e De Luca, C. J. (1994). Reconstruction expansion as a geometry-based framework for choosing proper delay times. *Physica D*, 73:82–98.
- Rowlands, G. e Sprott, J. C. (1992). Extraction of dynamical equations from chaotic data. *Physica D*, 58:251–259.

- Swain, A. K. e Billings, S. A. (1998). Weighted complex orthogonal estimator for identifying linear and non-linear continuous time models from generalised frequency response functions. *Mechanical Systems and Signal Processing*, 12(2):269–292.
- Takagi, T. e Sugeno, M. (1985). Fuzzy identification of systems and its applications to modeling and control. *IEEE Trans. Syst., Man, Cybern.*, 15(1):116–132.
- Takens, F. (1980). Detecting strange attractors in turbulence. In Rand, D. A. e Young, L. S., editors, *Dynamical systems and turbulence, Lecture Notes in Mathematics, vol. 898*, pages 366–381. Springer Verlag, Berlin.
- Timmer, J., Rust, H., Horbelt, W., e Voss, H. U. (2000). Parametric, nonparametric and parametric modelling of a chaotic circuit time series. *Physics Letters A*, 274:123–134.
- Tôrres, L. e Aguirre, L. (1995). PEDINAL - protótipo para o estudo de dinâmica não linear. Relatório Técnico CPDEE/LCPI/03-95.
- Tôrres, L. A. B. (2001). *Controle e Sincronismo de Osciladores Caóticos*. Tese de doutorado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.
- Tôrres, L. A. B. e Aguirre, L. A. (2000). Inductorless Chua's circuit. *Electronics Letters*, 36(23):1915–1914.
- Yamakawa, T., Uchino, E., Miki, T., e Kusanagi, T. (1992). A neo fuzzy neuron and its applications to system identification and predictions to system behavior. In *Proceedings of the II IIZUKA*, pages 477–483, Japan.
- Yared, G. F. G. (2001). Síntese do movimento facial durante a fala a partir da atividade neuromuscular. Dissertação de mestrado, PPGEE, Universidade Federal de Minas Gerais, Belo Horizonte, Brasil.