Mateus Giesbrecht

PROPOSTAS IMUNO-INSPIRADAS PARA IDENTIFICAÇÃO DE SISTEMAS E REALIZAÇÃO DE SÉRIES TEMPORAIS MULTIVARIÁVEIS NO ESPAÇO DE ESTADO

Campinas
2013

UNIVERSIDADE ESTADUAL DE CAMPINAS
FACULDADE DE ENGENHARIA ELÉTRICA E DE COMPUTAÇÃO

Mateus Giesbrecht

PROPOSTAS IMUNO-INSPIRADAS PARA IDENTIFICAÇÃO DE SISTEMAS E REALIZAÇÃO DE SÉRIES TEMPORAIS MULTIVARIÁVEIS NO ESPAÇO DE ESTADO

Orientador: Prof. Dr. Celso Pascoli Bottura

Tese de doutorado apresentada ao Programa de Pós Graduação em Engenharia Elétrica da Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas como parte dos requisitos exigidos para a obtenção do título de Doutor em Engenharia Elétrica.
Área de concentração: Automação.

Este exemplar corresponde à versão final da tese defendida pelo aluno Mateus Giesbrecht, e orientada pelo Prof. Dr. Celso Pascoli Bottura

_____

Campinas
2013

FICHA CATALOGRÁFICA ELABORADA PELA
BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

G363p	Giesbrecht, Mateus Propostas imuno-inspiradas para identificação de sistemas e realização de séries temporais multivariáveis no espaço de estado / Mateus Giesbrecht. --Campinas, SP: [s.n.], 2013. Orientador: Celso Pascoli Bottura. Tese de Doutorado - Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação. 1. Análise de séries temporais. 2. Identificação de sistemas. 3. Métodos de espaço de estado. 4. Processo estocástico. 5. Algoritmos evolutivos. I. Bottura, Celso Pascoli, 1938-. II. Universidade Estadual de Campinas. Faculdade de Engenharia Elétrica e de Computação. III. Título.

Título em Inglês: Immuno-inspired approaches for state space multivariable system identification and time series realization
Palavras-chave em Inglês: Time series analysis, System Identification, State Space methods, Stochastic processes, Evolutionary algorithms
Área de concentração: Automação
Titulação: Doutor em Engenharia Elétrica
Banca examinadora: Annabel del Real Tamariz, João Viana da Fonseca Neto, Gilmar Barreto, Fernando José Von Zuben
Data da defesa: 20-02-2013
Programa de Pós Graduação: Engenharia Elétrica

COMISSÃO JULGADORA - TESE DE DOUTORADO

Candidato: Mateus Giesbrecht

Data da Defesa: 20 de fevereiro de 2013

Título da Tese: "Propostas Imuno-inspiradas para Identificação de Sistemas e Realização de Séries Temporais Multivariáveis no Espaço de Estado"

Prof. Dr. Celso Pascoli Bottura (Presidente): _____

Profa. Dra. Annabell Del Real Tamariz: _____

Prof. Dr. João Viana da Fonseca Neto: _____

Prof. Dr. Gilmar Barreto: _____

Prof. Dr. Fernando José Von Zuben: _____

.

À minha querida esposa Sara

.

Agradecimentos

Ao meu SENHOR, principalmente pela redenção, mas também por todo sustento, todo cuidado, toda direção e todas as outras bençãos sem as quais eu não poderia ter feito esta tese ou qualquer outra coisa.

Ao Prof. Dr. Celso Pascoli Bottura, meu orientador, por todo apoio e orientação durante este trabalho, e também à sua esposa, Carminha, por todos os cafezinhos.

Ao Prof. Dr. Gilmar Barreto, pelo encorajamento para que eu iniciasse este trabalho.

À minha querida esposa Sara, pelo carinho, cuidado, companheirismo e compreensão durante estes anos.

Aos meus pais, Mauro e Hulda, e à minha tia Ruth, pelo apoio e constantes orações.

À minha irmã, Dra. Érica Giesbrecht, pelo exemplo e encorajamento.

Ao meu sogro e minha sogra, João e Maria Aparecida, por toda a ajuda.

Ao Prof. Dr. Fernando José Von Zuben, pela introdução dos métodos de computação natural, pelas sugestões e pelas indicações de referências.

À empresa Andritz Hydro Inepar, em especial aos engenheiros Sérgio Cuyabano e Lamartine Silva, pelo tempo cedido para a execução deste trabalho.

.

Inútil vos será levantar de madrugada, repousar tarde, comer o pão que penosamente granjeastes; aos seus amados Ele o dá enquanto dormem

Salmo 127, v. 2

.

Resumo

Nesta tese é descrito como alguns problemas relacionados à identificação de sistemas discretos multivariáveis, à realização de séries temporais discretas multivariáveis e à modelagem de séries temporais discretas multivariáveis, podem ser formulados como problemas de otimização. Além da formulação dos problemas de otimização, nesta tese também são apresentadas algumas propostas imuno-inspiradas para a solução de cada um dos problemas, assim como os resultados e conclusões da aplicação dos métodos propostos.

Os métodos aqui propostos apresentam resultados e performance melhores que aqueles obtidos por métodos conhecidos para solução dos problemas estudados, e podem ser aplicados em problemas cujas condições não sejam favoráveis para aplicação dos métodos conhecidos na literatura.

Palavras-chave: Análise de séries temporais, Identificação de sistemas, Métodos de espaço de estado, Processo estocástico, Algoritmos evolutivos.

Abstract

In this thesis it is described how some problems related to multivariable system identification, multivariable time series realization and multivariable time series modeling, can be formulated as optimization problems. Additionally, in this thesis some immuno-inspired methods to solve each problem are also shown, and also the results and conclusions resultant from the application of the proposed methods.

The performance and the results obtained with the methods here proposed are better than the results produced by known methods to solve the studied problems and can be applied even if the problem conditions are not suitable to the methods presented in the literature.

Keywords: Time series analysis, System identification, State space methods, Stochastic processes, Evolutionary algorithms .

Lista de Figuras

Lista de Tabelas

Trabalhos Publicados Pelo Autor

1. GIESBRECHT, M.; BOTTURA, C. P. An immuno-inspired approach to find the steady state solution of Riccati equations not solvable by Schur method. *Proceedings of the 2010 IEEE World Congress on Computational Inteligence* (Barcelona, July 2010).

2. GIESBRECHT, M.; BOTTURA, C. P. Uma proposta imuno-inspirada para a solução algébrica da equação de Riccati no problema de identificação de séries temporais no espaço de estado. *Anais do XVIII Congresso Brasileiro de Automática* (Bonito, Setembro 2010).

3. GIESBRECHT, M.; BOTTURA, C. P. Uma proposta imuno-inspirada para a modelagem de séries temporais discretas no espaço de estado. *Anais do X Simpósio Brasileiro de Automação Inteligente* (São João del-Rei, Setembro 2011).

4. GIESBRECHT, M.; BOTTURA, C. P. An immuno inspired approach to generate white noise. *Proceedings of the fourth International Workshop on Advanced Computational Intelligence* (Wuhan, October 2011).

5. GIESBRECHT, M.; BOTTURA, C. P. Immuno inspired approaches to model discrete time series at state space. *Proceedings of the fourth International Workshop on Advanced Computational Intelligence* (Wuhan, October 2011).

6. TOBAR J.; BOTTURA, C. P.; GIESBRECHT, M. Computational modeling of multivariable non-stationary time series in the state space by the Aoki VAR algorithm. *IAENG International Journal of Computer Science 37*, November (2010).

Sumário

Capítulo 1

Introdução

A identificação de sistemas é uma forma de encontrar alguma representação matemática que realize um sistema que se quer estudar, ou seja, o objetivo que se tem ao identificar um sistema é encontrar alguma equação, seja ela escalar, matricial, diferencial, a diferenças, não linear, linear, dentre outras, que seja capaz de traduzir para uma linguagem matemática a relação entre grandezas quantificáveis através do sistema. Deve ficar claro então que modelos são apenas modelos, e não se deve supor que algum fenômeno acontece por causa de relações matemáticas. Por exemplo, uma maçã que se solte de uma árvore, não cai porque:

$$F = \frac{Gm_1m_2}{d^2},$$

mas sim porque ela cai. Ao cair, esta maçã desenvolve uma grandeza que pode ser quantificada, e que resolveram definir como velocidade. Esta grandeza definida como velocidade varia de acordo com algo também quantificado, que resolveram definir como aceleração da gravidade. Percebeu-se que esta aceleração tem relação linear com outra grandeza, definida como força, e que a constante de proporcionalidade entre força e aceleração é uma propriedade da maçã, que resolveram definir como massa. A força, que é proporcional à aceleração, que faz com que a velocidade da maçã varie, tem relação com a massa da Terra, e com outra grandeza, que é a distância da maçã à Terra. Notou-se que a força varia proporcionalmente à massa da Terra, à massa da maçã e com o inverso do quadrado da distância entre a maçã e a Terra. No entanto, como a grandeza definida como força, da maneira que foi definida, não se relaciona exatamente com o produto entre a massa da Terra, a massa da maçã e o inverso do quadrado da distância entre a maçã e a Terra, foi criada uma constante que ajusta o resultado, e a ela foi dado o nome de constante gravitacional. Esta constante muda, dependendo da maneira como for medida a massa da maçã, a massa da Terra e a distância entre a maçã e a Terra. A maçã, no entanto, não tem conhecimento disto, e cairia mesmo que um modelo físico-matemático para sua queda não existisse.

A equação criada para modelar a queda da maçã pode não descrever seu movimento de maneira exata. A maçã, ao cair, eventualmente pode estar sujeita a algum vento, que tenha algum componente que altere sua velocidade vertical. A própria maçã produz um deslocamento de ar, que também produz uma componente de força vertical que altera sua velocidade. Aliás, não se garante que a maçã chegará ao chão, uma vez que algum animal pode interceptar o seu vôo e comê-la.

Por todos esses motivos, deve-se ter em mente que os modelos matemáticos nem sempre refletem

exatamente o que de fato ocorrerá com um fenômeno. No entanto, os eventos não previstos podem não ter grande influência na relação entre as grandezas observadas de um certo fenômeno e podem ser tratados como pequenos desvios, ou seja, pequenos ruídos. O conjunto entre grandeza, ou sinal, a ser medido e ruídos pode apresentar características estatísticas marcantes, que permitam que se modele de uma maneira um pouco mais precisa o que ocorre, mesmo sem o conhecimento pleno sobre todos os eventos que possam acontecer naquele fenômeno. Por este motivo, a identificação de sistemas com ruído começou a ser estudada.

Muitas vezes também se está interessado no estudo de sinais puramente aleatórios. Estes sinais, definidos como processos estocásticos, ou como séries temporais, podem ter significado em si só. Exemplos disto são as séries temporais econômicas, em que se tem o interesse de conhecer o comportamento de determinado evento ao longo do tempo, as séries temporais relacionadas à natureza, como índice de pluviosidade, taxa de reprodução de animais, dentre outros. O tratamento deste tipo de dados pode ser feito da mesma forma como se tratam os ruídos observados em grandezas físicas e é possível propor modelos matemáticos que gerem ruídos com características estatísticas semelhantes às de ruídos observados. Ao estudo da definição destes modelos matemáticos se dá o nome de realização de séries temporais.

Nesta tese, é apresentada a pesquisa desenvolvida a respeito da aplicação de algoritmos imuno-inspirados a problemas relacionados à identificação de sistemas e à realização de séries temporais no espaço de estado. A aplicação de algoritmos de inteligência computacional a problemas de identificação e controle é um tema recente e já tem sido tratado no Laboratório de Controle e Sistemas Inteligentes (LCSI), com as teses de doutorado [19], [53], [56] dentre outras.

Os problemas estudados e as contribuições desta pesquisa são descritos de forma resumida a seguir.

1.1 Objetivos e justificativas

Tanto a identificação de sistemas multivariáveis no espaço de estado, quanto a realização de séries temporais multivariáveis no espaço de estado, são problemas que requerem um amplo tratamento estatístico dos dados coletados a partir de amostras de entradas e saídas dos sistemas a serem identificados, ou de amostras da série temporal a ser realizada, respectivamente. Para que este tratamento estatístico seja efetivo, é necessário que um volume relativamente grande de dados relativamente bem comportados seja coletado.

Caso o volume de dados coletados não seja suficientemente grande, as características estatísticas dos sinais podem não ser fielmente retratadas. Seja, por exemplo, o lançamento de uma moeda. Se uma moeda for lançada infinitas vezes, cada uma das faces será sorteada em exatamente metade das vezes. No entanto, se a moeda for jogada apenas duas vezes, não é impossível que a mesma face apareça as duas vezes. Isto não significa necessariamente que a moeda é viciada, ou que a probabilidade do aparecimento da face que foi sorteada seja unitária, mas apenas significa que o número de amostras do experimento não é grande o suficiente para que se conheça suas características estatísticas com exatidão.

O objetivo desta pesquisa é a aplicação de métodos de computação imuno-inspirada para a solução dos problemas de identificação de sistemas e de realização de séries temporais quando o volume de dados coletados não é grande o suficiente, ou não é bem comportado o suficiente, para que os métodos

estatísticos existentes impliquem em bons resultados. Em geral, os métodos de computação imuno-inspirada podem ser aplicados ao se transformar os problemas resultantes da falta de qualidade, ou quantidade, das amostras coletadas em problemas de otimização.

Desta forma, parte das contribuições desta pesquisa é a interpretação de diversos problemas ligados à identificação de sistemas e à realização de séries temporais como problemas de otimização. Para cada um destes problemas de otimização, as contribuições incluem sua definição, a definição das soluções candidatas, da função objetivo, do espaço de busca e, quando aplicável, das restrições. Outra parte das contribuições é a implementação de diversas variantes de algoritmos imuno-inspirados para tratar de cada um dos problemas definidos ao longo desta pesquisa. Como, em alguns casos, os problemas de otimização encontrados têm restrições, outra contribuição desta pesquisa é uma proposta de modificação dos algoritmos imuno-inspirados para lidar com este tipo de problemas.

De maneira resumida, os objetivos e os problemas descritos nesta tese são apresentados nas próximas subseções.

1.1.1 Geração de ruídos brancos

Como apresentado de maneira mais detalhada no capítulo 5, os resultados dos métodos de realização de séries temporais no espaço de estado são as matrizes de um modelo em espaço de estado e a covariância para atraso zero do ruído branco que, quando aplicado como entrada ao modelo encontrado, faz com que a saída seja uma realização da série temporal que se quer realizar. Sendo assim, para se gerar uma realização de uma determinada série temporal com um número limitado de amostras, é necessário aplicar um ruído branco com um limitado número de amostras como entrada ao sistema encontrado.

Durante esta pesquisa, notou-se que os sinais com número limitado de amostras criados com um gerador pseudo-aleatório comum não satisfazem a definição de ruído branco no domínio do tempo, detalhada no capítulo 7. Sendo assim, um dos objetivos desta pesquisa é a obtenção de um ruído branco com número de amostras limitado para gerar realizações de séries temporais com número limitado de amostras. Como se está tratando de modelos multivariáveis, o método proposto deve ser capaz de lidar com casos multidimensionais. Este objetivo foi alcançado e seus resultados estão publicados no artigo [37] e são detalhados no capítulo 7 desta tese.

1.1.2 Solução da equação algébrica de Riccati

Para a determinação das matrizes do modelo em espaço de estado que realiza uma determinada série temporal pelo método proposto por Aoki, é necessária a solução de uma equação algébrica de Riccati, conforme descrito no capítulo 5 desta tese. Os termos desta equação são matriciais e sua solução depende da possibilidade de manipulação destas matrizes. Estas matrizes são criadas a partir das amostras da série temporal que se quer realizar. Desta forma, dependendo dos dados coletados, as matrizes presentes na equação de Riccati são de tal forma que a solução não pode ser encontrada pelos métodos convencionais descritos na seção 5.1.3.

Um dos objetivos desta pesquisa é a determinação de soluções aproximadas para a equação de Riccati no caso em que os métodos de solução convencionais não levam a soluções por problemas numéricos durante a manipulação das matrizes. Este objetivo foi alcançado e seus resultados estão publicados nos artigos [39] e [40] e são detalhados no capítulo 7 desta tese.

1.1.3 Modelagem de séries temporais

A partir do problema de realização de séries temporais no espaço de estado, pode-se definir um problema mais simples, que é a modelagem de séries temporais no espaço de estado. Neste problema, considera-se que, tanto a entrada, quanto a saída do modelo que gera a série temporal, estão disponíveis para a aplicação do método. Sendo assim, este problema se aproxima mais do problema de identificação de sistemas no espaço de estado do que propriamente do problema de realização de séries temporais no espaço de estado.

Um dos objetivos desta pesquisa é a proposta de um algoritmo imuno-inspirado para a solução deste problema. A justificativa para isto é a criação de uma base que pode ser usada para a solução de problemas mais complexos, como a identificação de sistemas multivariáveis variantes no tempo, resumida na próxima subseção. Este objetivo também foi alcançado e os resultados foram publicados nos artigos [38] e [41] e são também detalhados no capítulo 7 desta tese.

1.1.4 Identificação de sistemas multivariáveis variantes no tempo no espaço de estado

Existem diversos métodos de identificação de sistemas multivariáveis invariantes no tempo, conforme apresentado no capítulo 3. Uma das maneiras de se identificar sistemas multivariáveis variantes no tempo é a aplicação dos métodos de identificação de sistemas multivariáveis invariantes no tempo a intervalos de dados para os quais não há variação significativa. No entanto, para se encontrar soluções dos problemas de identificação de sistemas multivariáveis invariantes no tempo, é necessário que um grande número de amostras esteja disponível. Desta forma, métodos de identificação de sistemas multivariáveis variantes no tempo baseados na partição temporal dos conjuntos de amostras falham se a variação do sistema for muito rápida, de forma a não haver um número de amostras suficientes para a identificação quando o sistema está estacionado em um estado.

Um dos objetivos desta pesquisa é a criação de um algoritmo para a identificação de sistemas multivariáveis variantes no tempo a partir de um número mínimo de amostras de entrada e saída dos sistemas. Desta forma, mesmo variações mais rápidas do sistema podem ser modeladas. Este objetivo foi alcançado, os resultados obtidos foram submetidos para publicação e são detalhados no capítulo 7 desta tese.

1.2 Organização desta tese

Esta tese está organizada da seguinte maneira: No capítulo 2 os processos estocásticos, ou séries temporais, são definidos e suas propriedades são descritas. Esse capítulo serve como base para os capítulos 4, em que é introduzido um estimador ótimo de estados, conhecido como filtro de Kalman, e 5, em que são introduzidos os problemas de realização e modelagem de séries temporais discretas no espaço de estado e alguns dos métodos de resolução destes dois problemas são descritos. Os capítulos 4 e 5 foram escritos de maneira a tornar clara a relação entre o filtro de Kalman e a realização de séries temporais no espaço de estado. No capítulo 3, os métodos de identificação de sistemas multivariáveis discretos no espaço de estado são discutidos. Este capítulo também está escrito de maneira que a relação entre o problema tratado nele e o problema tratado no capítulo 5 se torna clara. No capítulo

6 são descritos os algoritmos imuno-inspirados para otimização. Esse tipo de algoritmo foi escolhido devido a suas propriedades favoráveis à solução dos problemas estudados nesta tese. No capítulo 7, as contribuições desta pesquisa, que são os métodos baseados nos conceitos descritos no capítulo 6, para solução de problemas relacionados aos capítulos 3 e 5, são detalhadas. Por fim, no capítulo 8, as conclusões obtidas ao longo desta pesquisa e os trabalhos futuros que podem ser realizados são apresentados.

Capítulo 2

Processos Estocásticos

Neste capítulo, são apresentados os conceitos fundamentais envolvidos no estudo de processos estocásticos. O objetivo deste estudo é fornecer as bases necessárias para o desenvolvimento dos conceitos envolvidos no problema de realização de séries temporais multivariáveis, tratado no capítulo 5, que é um dos assuntos principais desta tese. As referências básicas para este capítulo são as notas de aula [9], os capítulos iniciais das referências [47] e [51], o capítulo 4 de [46] e também e os livros [30] e [52].

Ao longo do capítulo há notas de rodapé contando um pouco da história por trás das pessoas que foram homenageadas com nomes de teoremas, desigualdades, etc. com o objetivo de tornar a leitura mais instrutiva e permitir que o leitor conheça a ordem cronológica em que os conceitos foram percebidos.

2.1 Noções iniciais

O primeiro passo para o estudo estatístico é a definição de eventos e de probabilidades. Usualmente estas entidades são definidas de forma ingênua e sem a abordagem de aspectos como a sigma álgebra e a teoria da medida. O objetivo desta seção é o de introduzir estes conceitos de uma forma mais axiomática conforme feito por Kolmogorov[1].

A princípio vamos definir um espaço amostral, geralmente denotado por Ω. O **espaço amostral** é um espaço em que há pelo menos dois subconjuntos: o vazio ($\emptyset$) e o todo (Ω). Este espaço pode ser, por exemplo, um conjunto com todos os resultados possíveis de um experimento, um espaço de números como o $\Re^n$, dentre outros. Um dos exemplos mais básicos de espaço amostral é o conjunto formado por *cara* e *coroa*, que são os resultados possíveis do lançamento de uma moeda.

Dentro de um espaço amostral Ω pode-se encontrar uma determinada família de subconjuntos $\mathcal{F}$ que apresente as seguintes propriedades:

- $\emptyset \in \mathcal{F}$.
- Se F é um subconjunto que está contido na família $\mathcal{F}$ então seu complemento F^c também está contido na família $\mathcal{F}$.

[1] Andrey Nikolaevich Kolmogorov (1903-1987). Matemático soviético que contribuiu com avanços nas teorias de probabilidade, topologia, turbulência, dentre outros

- Se F_1, F_2,... são conjuntos contidos na família $\mathcal{F}$, então sua união infinita $\bigcup_{i=1}^{\infty} F_i$ também está contida na família $\mathcal{F}$.

Às famílias de conjuntos com estas três propriedades é dado o nome de **σ-álgebra**.

Para ilustrar este conceito, seja o experimento do lançamento de um dado. Os possíveis resultados pertencem ao conjunto Ω ={1, 2, 3, 4, 5, 6}. Se escolhermos uma família de conjuntos $\mathcal{F}$ formada pelos seguintes subconjuntos {1}, {2, 3, 4, 5, 6}, $\emptyset$ e Ω, nota-se que esta família tem todas as propriedades de uma σ-álgebra listadas acima, portanto esta família é uma σ-álgebra. Note também que se tomarmos a seguinte família de subconjuntos: {*Números pares*}, {*Números ímpares*}, $\emptyset$ e Ω teremos que esta família também satisfaz às propriedades acima e portanto também é uma σ-álgebra.

A cada subconjunto de elementos de Ω dá-se o nome de **evento**. Note que os eventos possíveis são determinados pela σ-álgebra definida para o espaço amostral. Um espaço amostral em que está definida uma σ-álgebra é chamado de **espaço mensurável**. Se há infinitos F_i eventos, então a intersecção e a soma infinita, ou seja:

$$\bigcup_{i=1}^{\infty} F_i$$

e

$$\bigcap_{i=1}^{\infty} F_i$$

também são eventos, sendo que a união infinita é evento devido à própria definição de sigma-álgebra

Pode-se definir uma medida P sobre o espaço mensurável. Neste contexto, entende-se por medida uma determinada função que leva do espaço amostral a um vetor numérico (que pode ser real, inteiro, degenerado para uma dimensão, etc..). Este vetor numérico indica o tamanho relativo de um determinado subconjunto do espaço amostral, sendo que este subconjunto é definido a partir de uma σ-álgebra. Se a medida P definida tiver as seguintes propriedades:

- $P(\emptyset) = 0$
- $P(\Omega) = 1$
- $P(\bigcup_{i=1}^{\infty} F_i) \leq \sum_{i=1}^{\infty} P(F_i)$

É dado à medida P o nome de **probabilidade**. A probabilidade apresenta ainda as seguintes propriedades:

$$0 \leq P(A) \leq P(B) \leq 1 \quad \forall A,\, B \in \mathcal{F}\, com\, A \subseteq B$$

$$P(A^c) = 1 - P(A)$$

em que A^c denota o evento complementar ao evento A.

Deve-se notar que a escolha da σ-álgebra influencia na medida de probabilidade. Voltando ao exemplo do lançamento de um dado, se for escolhida a primeira σ-álgebra exemplificada, ou seja, {1}, {2, 3, 4, 5, 6}, $\emptyset$ e Ω, a medida de probabilidade do subconjunto {1} é de 1/6 (supondo que o dado seja honesto). Caso seja escolhida a segunda σ-álgebra citada, ou seja, {*Números pares*}, {*Números*

ímpares}, $\emptyset$ e Ω, a probabilidade de ocorrência de cada elemento é 1/2. Independentemente da σ-álgebra escolhida, a probabilidade pode ser vista como uma medida normalizada de subconjuntos de um determinado espaço amostral.

Definidos um espaço amostral, uma σ-álgebra e uma medida de probabilidade, tem-se uma tripla $\{\Omega, \mathcal{F}, P\}$ chamada de **espaço de probabilidade**. É importante que fique clara a necessidade de cada um dos termos desta tripla. Voltando ao exemplo do lançamento de dados, é óbvio que se não houvesse o dado, ou seja, o espaço amostral, não faria sentido a definição de probabilidades. Se não fossem definidos diferentes eventos, ou seja, subconjuntos do espaço amostral como os números das faces ou o fato de eles serem pares ou ímpares, também não faria sentido se falar de probabilidades, ou seja, a σ-álgebra é fundamental para que se faça esta definição. E finalmente, se não for definida uma medida para cada um dos subconjuntos em que o espaço amostral foi dividido, não se pode falar em probabilidade. Desta forma fica clara a importância e a necessidade de cada um dos elementos desta tripla. Embora este parágrafo seja redundante com relação ao resto do texto, é importante que fique clara a definição de cada um dos elementos da tripla fundamental para uma definição axiomática da probabilidade.

Para uma determinada família de subconjuntos do espaço amostral pode-se definir a menor σ-álgebra gerada por esta família. Por exemplo, a família formada pelos subconjuntos Ω, $\emptyset$ determina a mais simples σ-álgebra que pode ser definida sobre um espaço amostral, que é chamada de **σ-álgebra trivial**.

Voltando ao experimento do lançamento de um dado, a primeira σ-álgebra definida continha o subconjunto {1}. No entanto, note que esta σ-álgebra não é a única que pode conter aquele subconjunto. Por exemplo, a família de subconjuntos {{1}, {2}, {2, 3, 4, 5, 6}, {1, 3, 4, 5, 6}, $\emptyset$, Ω} contém {1} e também satisfaz todas as condições para ser uma σ-álgebra, podendo portanto ser definida desta forma. A primeira σ-álgebra tem, por outro lado, uma propriedade interessante: ela é a menor σ-álgebra que pode ser definida contendo o subconjunto {1}, e ela está contida na σ-álgebra apresentada neste parágrafo. Desta discussão pode-se notar que as σ-álgebras podem estar contidas, serem maiores ou menores entre si. Além disto, é importante notar que a menor σ-álgebra contendo determinado subconjunto é a intersecção de todas as σ-álgebras possíveis que podem conter aquele subconjunto.

No caso em que o espaço amostral é o $\Re^n$, a σ-álgebra gerada por todos os subconjuntos semi-infinitos limitados superiormente deste espaço, ou seja, por todos os vetores $x \in \Re^n$ tais que

$$x \in \Re^n;\ -\infty < x \leq a$$

em que a é um número real qualquer, é chamada de **σ-álgebra de Borel**[2], denotada por $\mathcal{B}$. Aos conjuntos que geram esta σ-álgebra é dado o nome de **conjuntos de Borel**, e eles são normalmente simbolizados por B. Esta σ-álgebra é particularmente importante quando definida sobre a reta $\Re$ uma vez que é possível criar uma função $\Re \rightarrow \Re$ que contenha todas as informações de uma função $\Omega \rightarrow \Re$ conforme será apresentado mais adiante.

[2]Nome dado em homenagem ao matemático francês Félix Édouard Justin Émile Borel (1871-1956), que contribuiu significativamente para o estudo da teoria de probabilidade. É conhecido também por ter enunciado o teorema do Macaco Infinito, no qual dizia que se fosse dada a um macaco uma máquina de escrever ele poderia, em tempo infinito, datilografar um texto exatamente igual às obras completas de Shakespeare. Qualquer relação entre isto e as nacionalidades de Borel e de Shakespeare não deve ser mera coincidência.

A partir da definição do espaço amostral, pode-se definir funções que levem deste espaço a quaisquer outros espaços. Se uma função X que leva de Ω a $\Re^n$ é tal que o seu domínio está contido em uma determinada σ-álgebra $\mathcal{F}$ de Ω, dizemos que esta função $X : \Omega \rightarrow \Re^n$ é **$\mathcal{F}$-mensurável**. Matematicamente, X é $\mathcal{F}$-mensurável se:

$$X^{-1}(U) = \{\omega \in \Omega\} \in \mathcal{F}\,, U \subset \Re^n$$

em que U é um subconjunto qualquer da imagem de X. Note que a medida de probabilidade sobre um espaço mensurável $\{\Omega, \mathcal{F}\}$ é uma função $\mathcal{F}$-mensurável.

Dado um espaço de probabilidade $\{\Omega, \mathcal{F}, P\}$, uma função $X : \Omega \rightarrow \Re^n$ $\mathcal{F}$-mensurável é chamada de **variável aleatória**. Embora seja conhecida como variável, deve ficar claro que a variável aleatória é uma função. Como o domínio das variáveis aleatórias é um conjunto, e como o manuseio de funções que envolvam conjuntos é algo não muito intuitivo, foi criado um artifício para o estudo das variáveis aleatórias. Este artifício consiste em criar uma função $F : \Re^n \rightarrow [0, 1]$ que vá da imagem da variável aleatória até o intervalo $[0, 1]$. Esta função é tal que seu valor é definido como sendo a probabilidade de ocorrência do subconjunto formado por todos os eventos do espaço amostral tais que, ao se tomar a variável aleatória sobre estes eventos, o resultado é menor ou igual ao ponto do domínio de F sobre o qual F está sendo calculada. À função F é dado o nome de **função de distribuição**, ou ainda **probabilidade acumulativa**. O termo *acumulativa* vem do fato de um ponto na imagem da função de probabilidade acumulativa retratar a probabilidade de ocorrência de todos os eventos que, quando submetidos à variável aleatória, apresentam resultados menores ou iguais à aplicação da variável aleatória a um determinado evento. Na figura 2.1 são ilustrados os conceitos de variável aleatória e da função de distribuição.

Por exemplo, seja um determinado x na imagem de uma variável aleatória X, que também é o domínio de F. O valor de $F(x)$ será igual à probabilidade de todos os subconjuntos ω tais que $X(\omega) < x$, ou seja:

$$F(x) = P(\omega \in \Omega | X(\omega) \leq x) \tag{2.1}$$

Ao se definir a função de distribuição $F(x)$ criou-se algo que vai de $\Re^n \rightarrow [0, 1]$, mas mais do que isto, criou-se uma função que atribui valores no intervalo [0,1] a subconjuntos semiinfinitos limitados superiormente, ou seja, a subconjuntos de uma σ-álgebra de Borel. Sendo assim, definindo:

$$P_X(B) = P(\{\omega \in \Omega | X(\omega) \in B\})\, B \in \mathcal{B} \tag{2.2}$$

temos que a tripla ($\Re$, $\mathcal{B}$, P_X) é um espaço de probabilidade que contém as mesmas informações que o espaço (Ω, $\mathcal{F}$, P), e portanto a função $F(x)$ carrega todas as informações da variável aleatória X, podendo ser usada em seu lugar.

Conforme discutido, a função $F(x)$ reflete a probabilidade de todos os eventos tais que a aplicação da variável aleatória $X(\omega)$ sobre eles implica em um valor menor ou igual à aplicação da variável aleatória a um determinado evento, e por isso é definida como função de probabilidade acumulativa. No entanto, em alguns casos se quer saber qual é a probabilidade de um determinado grupo de eventos, que não necessariamente são todos tais que a aplicação da variável aleatória sobre eles seja menor ou igual a certo valor. Mais interessante ainda é conhecer a probabilidade daquele grupo de eventos com relação ao tamanho do intervalo que eles ocupam no espaço imagem da variável aleatória ou, em outras palavras, qual é a **densidade de probabilidade** daquele conjunto de eventos. Mais formalmente,

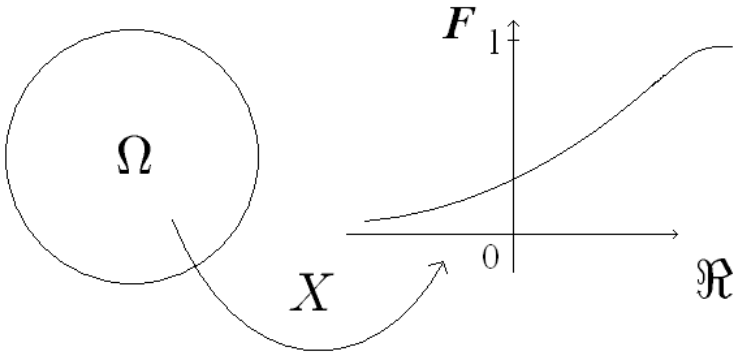

Fig. 2.1: Variável aleatória X levando de Ω a $\Re$ e função de distribuição F levando da imagem da variável aleatória ao intervalo [0,1].

suponha que se queira determinar a densidade de probabilidade de todos os eventos $\omega \in \Omega$ tais que, quando sujeitos a uma variável aleatória X, implicam em resultados entre $x_1 \in \Re$ e $x_2 \in \Re$ tal que $x_2 > x_1$. Obviamente, se for feita a diferença entre o valor de $F(x_2)$ que reflete a probabilidade de todos os eventos tais que, quando sujeitos à variável aleatória X, implicam em valores menores ou iguais a x_2, e o valor de $F(x_1)$, que é a probabilidade de todos os eventos tais que, quando sujeitos à variável aleatória X, implicam em valores menores ou iguais a x_1, se terá a probabilidade de todos eventos que, quando sujeitos à variável aleatória X, terão resultados entre x_1 e x_2. Ou seja:

$$P(\{\omega \in \Omega | x_1 \leq X(\omega) \leq x_2\}) = F(x_2) - F(x_1) \tag{2.3}$$

Desta forma, para se ter a densidade de probabilidade destes eventos, basta dividir a probabilidade deste conjunto de eventos, que é a diferença $F(x_2) - F(x_1)$, pelo tamanho do intervalo que este grupo de eventos ocupa na imagem da variável aleatória, ou seja, $x_2 - x_1$. Quando $x_2 \to x_1$, ou seja, quando o intervalo entre x_1 e x_2 é infinitesimal, se tem a definição da função de densidade de probabilidade $f(x)$ conforme abaixo:

$$f(x) = \lim_{x_2 \to x_1} = \frac{F(x_2) - F(x_1)}{x_2 - x_1} = \frac{d}{dx} F(x) \tag{2.4}$$

Com isto, para dois números reais x_1 e x_2, tem-se que a probabilidade de todos os eventos ω tais que os resultados da aplicação da variável aleatória estão no intervalo entre x_1 e x_2 é igual a:

$$P(\{\omega \in \Omega | x_1 \leq X(\omega) \leq x_2\}) = \int_{x_1}^{x_2} f(x) dx \tag{2.5}$$

2.2 Definição de processo estocástico

Recordados os conceitos importantes, parte-se agora para o estudo detalhado dos processos estocásticos. Por definição, um **processo estocástico** é um tipo de função temporal que atribui para cada instante de tempo uma variável aleatória. Um processo estocástico x é tal que $x(\omega, t) : (\Omega, t) \to \Re^n$,

ou seja, um processo estocástico é uma função que leva de um domínio, que inclui um espaço amostral Ω e um espaço de tempos, a uma imagem, que é o espaço real de dimensão n natural. Se $n = 1$ o processo estocástico é definido como **monovariável**. Caso $n > 1$, o processo estocástico é definido como **multivariável**, ou vetorial.

Deve-se notar que, se um determinado instante de tempo for fixado, ter-se-á apenas uma variável aleatória, pois a função x levará de um espaço amostral a um espaço real. Por outro lado, caso se observe a ocorrência do processo ao longo do tempo, se diz que se tem uma **função de amostragem**, também conhecida como **realização de um processo estocástico**, ou ainda como **série temporal**. Em outras palavras, uma realização de um processo estocástico é uma sequência numérica em que, a cada instante de tempo, é sorteado um valor, de acordo com as características estatísticas da variável aleatória que representa o processo naquele instante de tempo. O conceito de realização de um processo estocástico é tratado com mais detalhes no capítulo 5.

Seja um determinado processo estocástico definido nos seguintes k instantes de tempo: $t_1, \ldots, t_k$. Define-se como **distribuição conjunta** destas variáveis aleatórias a seguinte medida de probabilidade:

$$P\{x(t_1) \leq a_1, \ldots, x(t_k) \leq a_k\} =$$

$$= \int_{-\infty}^{a_1} \cdots \int_{-\infty}^{a_k} f_{t1,\ldots,tk}(x_1, \ldots, x_k)\, dx_1 \ldots dx_k \tag{2.6}$$

Em que $f_{t1,\ldots,tk}(x_1, \ldots, x_k)$ é a função de densidade de probabilidade conjunta das variáveis aleatórias obtidas ao se fixar os instantes de tempo de $t_1, \ldots, t_k$. Neste caso a distribuição conjunta tem dimensão finita. A distribuição de um processo estocástico pode ser determinada por todas as distribuições finitas das variáveis aleatórias formadas ao se fixar instantes de tempo.

Se o domínio de tempo de amostragem for toda a reta real, o processo estocástico é dito **contínuo**. Caso o domínio do tempo seja formado apenas por alguns pontos da reta, o processo estocástico é dito **discreto**. Nesta tese se tem interesse principalmente nos processos estocásticos discretos.

2.3 Propriedades de processos estocásticos monovariáveis

A definição e o estudo dos processos estocásticos não é algo trivial. Sendo assim, ao longo do tempo, foram definidas algumas propriedades que um processo estocástico pode apresentar de forma que seu estudo seja simplificado. Estas propriedades são interessantes pois se aplicam a uma grande classe de processos estocásticos. Com isto, ao se investigar um determinado processo estocástico em particular, estas propriedades podem ser supostas, facilitando o estudo. Por simplicidade, nesta seção são tratadas as propriedades dos processos estocásticos monovariáveis. Na seção 2.4 os processos estocásticos multivariáveis são introduzidos e as peculiaridades do caso multivariável das propriedades definidas nesta seção são tratadas.

A idéia de se definir as propriedades de processos estocásticos é semelhante à idéia por trás das hipóteses de linearidade, invariância com o tempo, dentre outras, aplicadas ao estudo de sistemas dinâmicos. Praticamente tudo aquilo que se estuda é não linear ou variante no tempo. No entanto, ao se trabalhar com um sistema, pode-se adotar estas hipóteses, mesmo que restritas a algumas condições

ou a algumas regiões do problema, e se encontrar soluções satisfatórias. Da mesma forma, ao se estudar um processo estocástico, pode-se adotar as hipóteses de que ele é markoviano, estacionário, ergódico, etc para simplificar seu estudo.

2.3.1 Processos Markovianos

Seja um determinado processo estocástico discreto $x(t)$. O conjunto de observações deste processo estocástico até um instante $s \leq t$ pode ser visto como um subconjunto do espaço gerado pelas observações até o instante de tempo t. Portanto, pode-se encontrar a mínima σ-álgebra $\mathcal{F}_s$ gerada por este subconjunto. Note que a cada novo instante de tempo, nova informação é fornecida ao sistema, de forma que se $t_1 < t_2$ então $\mathcal{F}_{t1} \subset \mathcal{F}_{t2}$. Em outras palavras, a cada instante de tempo se está adicionando mais informações, ou seja, **inovações** à σ-álgebra. Definida a σ-álgebra $\mathcal{F}_s$ como sendo a σ-álgebra gerada por amostras de um processo estocástico até o instante s, um processo estocástico é dito **markoviano**[3] se:

$$P\{x(s+1) \leq a|\mathcal{F}_s\} = P\{x(s+1) \leq a|\sigma(x(s))\} \tag{2.7}$$

em que $\sigma(x(s))$ denota a σ-álgebra gerada pela amostra $x(s)$.

Ou seja, o processo é markoviano se com apenas a medida no instante de tempo anterior pode-se determinar o futuro com a mesma precisão que se teria ao se conhecer todo o passado do processo. Ou ainda, o processo é markoviano se toda a informação passada do processo é resumida no último instante de tempo, ou ainda, $\sigma(x(s)) = \mathcal{F}_s$. Esta propriedade pode ser vista como um caso especial de causalidade, ou seja, o comportamento do processo depende apenas de seu passado (causalidade) e este passado está resumido no último instante de tempo. Obviamente, um processo markoviano é causal, mas nem todo processo causal é markoviano.

Em termos da densidade de probabilidade condicional, dizer que um processo é markoviano significa dizer que:

$$p(x(s)|x(s-1), \ldots, x(0)) = p(x(s)|x(s-1)) \tag{2.8}$$

A função de densidade conjunta de um processo markoviano pode ser encontrada com o uso da regra de Bayes[4] conforme demonstrado abaixo:

[3]Nome dado em homenagem ao matemático russo Andrey Andreyevich Markov (1856-1922), professor da Universidade de São Petesburgo tendo ocupado a cadeira deixada por Chebyshev. Markov ficou conhecido por solucionar a equação $(1+x^2)\frac{dy}{dx} = n(1+y^2)$ e por ter se retirado da Universidade de São Petersburgo após sofrer represálias ao ter negado a vigiar seus alunos durante revoltas estudantis no ano de 1908.

[4]Thomas Bayes (1702-1761). Matemático e pastor presbiteriano inglês, que teve o teorema que leva seu nome enunciado postumamente. Seus dois principais trabalhos mostram as duas atividades a que ele se dedicava: *Divine Benevolence, or an Attempt to Prove That the Principal End of the Divine Providence and Government is the Happiness of His Creatures* (1731) e *An Introduction to the Doctrine of Fluxions, and a Defence of the Mathematicians Against the Objections of the Author of the Analyst* (1736).

$$\begin{aligned} p(x(t_1), \ldots, x(t_k)) &= p(x(t_k)|x(t_1), \ldots, x(t_{k-1}))p(x(t_1), \ldots, x(t_{k-1})) \\ &= p(x(t_k)|x(t_{k-1}))\cdot \\ &\quad \cdot p(x(t_{k-1})|x(t_1), \ldots, x(t_{k-2}))p(x(t_1), \ldots, x(t_{k-2})) = \\ &= p(x(t_k)|x(t_{k-1}))\cdot \\ &\quad \cdot p(x(t_{k-1})|x(t_{k-2}))\cdot \\ &\quad \cdot p(x(t_{k-2})|x(t_1), \ldots, x(t_{k-3}))p(x(t_1), \ldots, x(t_{k-3})) = \\ &\quad \vdots \\ &= p(x(t_1)) \prod_{i=2}^{k} p(x(t_i)|x(t_{i-1})) \end{aligned} \tag{2.9}$$

mas

$$p(x(t_i)|x(t_{i-1})) = \frac{p(x(t_i),\, x(t_{i-1}))}{p(x(t_{i-1}))} \tag{2.10}$$

então:

$$p(x(t_1), \ldots, x(t_k)) = p(x(t_1)) \prod_{i=2}^{k} \left[\frac{p(x(t_i),\, x(t_{i-1}))}{p(x(t_{i-1}))} \right] \tag{2.11}$$

Portanto, tem-se que a densidade de probabilidade conjunta de um processo de Markov é determinada apenas pelas funções de densidade de primeira $p(x(t_i))$ e de segunda ordem $p(x(t_i),\, x(t_{i-1}))$. Este fato é fundamental, pois permite que se defina o processo apenas com estas duas medidas, que são equivalentes ao primeiro e segundo momentos, ou seja, média e autocovariância, conforme será visto a seguir. Este fato é uma das bases do estudo feito no capítulo 5.

2.3.2 Momentos de um processo estocástico

Para um determinado processo estocástico, define-se o k-ésimo **momento** $M(t_1, \ldots, t_k)$ como sendo a esperança do produto das variáveis aleatórias obtidas ao se fixar o processo estocástico x nos instantes $t_1, \ldots, t_k$, ou seja:

$$M(t_1, \ldots, t_k) = E[x(t_1) \ldots x(t_k)] \tag{2.12}$$

No estudo de processos estocásticos markovianos, se tem interesse pela função de **média** $\mu_x(t)$, que é o momento de primeira ordem, e pela função de **autocovariância** $\Lambda_{xx}(t, s)$, que é o momento de segunda ordem centrado, ou seja, as variáveis aleatórias são subtraídas de sua média antes de se tomar a esperança de seu produto. Caso a covariância seja tomada em dois processos estocásticos distintos, ela recebe o nome de **covariância cruzada** ($\Lambda_{xy}(t, s)$). Caso a covariância seja tomada no mesmo instante de tempo do mesmo processo ($\Lambda_{xx}(t, t)$), tem-se a **variância** do processo estocástico, normalmente denotada por σ^2.

Em resumo, os momentos objetos de estudo geralmente são:

- **Media:**
  $\mu_x(t) = E[x(t)]$
- **Autocovariância:**
  $\Lambda_{x,x}(t,s) = E[(x(t) - \mu_x(t))(x(s) - \mu_x(s))]$
- **Covariância cruzada:**
  $\Lambda_{x,y}(t,s) = E[(x(t) - \mu_x(t))(y(s) - \mu_y(s))]$
- **Variância:**
  $\sigma^2 = \Lambda_{x,x}(t,t) = E[(x(t) - \mu_x(t))(x(t) - \mu_x(t))]$

Um processo estocástico com variância finita é chamado de **processo de segunda ordem**.

2.3.3 Estacionariedade

Definidos os momentos, pode-se agora definir a estacionariedade de um processo estocástico. Um processo estocástico é dito **fortemente estacionário** quando todos seus momentos não dependem do intervalo de tempo em que são tomados. Algebricamente, $x(t)$ é fortemente estacionário se:

$$p_{t1,\ldots,tk}(x_1, \ldots, x_k) = p_{t1+l,\ldots,tk+l}(x_1, \ldots, x_k) \quad \forall\, k \tag{2.13}$$

Um processo é dito **fracamente estacionário** se apenas seus primeiro e segundo momentos forem independentes do instante de tempo em que se observa o processo. Para o primeiro momento (média) isto significa um valor constante em qualquer instante de tempo. Para o segundo momento (covariância), tem-se que seu valor depende apenas do intervalo de tempo entre os dois momentos em que ela é tomada e não dos instantes de tempo absolutos. Ou seja, um processo estocástico discreto é fracamente estacionário se:

$$\mu_x(l) = \mu_x \quad \Lambda_{xx}(t+l,t) = \Lambda_{xx}(l) \quad \forall\, l \in \mathcal{Z}$$

Um processo fortemente estacionário é fracamente estacionário mas um processo fracamente estacionário não é necessariamente um processo fortemente estacionário.

A definição de processo fortemente estacionário também pode ser feita sem o uso do conceito de momentos, conforme demonstrado em [47]. Seja $F_{t1,\,t2,\ldots,\,tn}$ a distribuição conjunta de uma variável aleatória X nos instantes de tempo t_1, t_2, t_n. O processo será dito fortemente estacionário se, para um dado h real, a seguinte expressão for verificada:

$$F_{t1,\,t2,\ldots,\,tn} = F_{t1+h,\,t2+h,\ldots,\,tn+h} \tag{2.14}$$

para todo t definido sobre o espaço T que faz parte do domínio da variável aleatória.

Deve-se notar que esta definição é equivalente à definição feita a partir dos momentos, uma vez que o enésimo momento de uma variável aleatória depende da densidade de probabilidade de ordem n, que por sua vez é a derivada da função de distribuição conjunta de uma variável aleatória em diversos instantes de tempo diferentes.

Para um processo estacionário pode-se enunciar o seguinte lema:

Lema 2.1 *Em um processo estacionário a função de covariância $\Lambda_{xx}(l)$ tem as seguintes propriedades:*

1. *É limitada, ou seja, $|\Lambda_{xx}(l)| \leq \Lambda_{xx}(0) \, \forall \, l \in \mathcal{Z}$*
2. *É simétrica, ou seja, $\Lambda_{xx}(l) = \Lambda_{xx}(-l)$*

Prova:

1. Da desigualdade de Cauchy[5]-Schwarz[6] tem-se que

$$| < x, y > |^2 \leq < x, x > \cdot < y, y >$$

em que $< x, y >$ denota o produto interno entre os vetores x e y de um espaço vetorial qualquer.

No caso do espaço vetorial formado pelos processos estocásticos, a norma é definida como sendo a covariância, portanto, se tomarmos o processo estocástico estacionário x nos instantes 0 e l, temos que:

$$E[x(l)x(0)]^2 \leq E[x(l)^2]E[x(0)^2] \Rightarrow$$

$$\Rightarrow E[x(l)x(0)]^2 \leq E[x(l)x(l)]E[x(0)x(0)] \Rightarrow$$

$$\Rightarrow E[x(l)x(0)]^2 \leq \Lambda_{xx}(0)\Lambda_{xx}(0) \Rightarrow$$

$$\Rightarrow E[x(l)x(0)] \leq \Lambda_{xx}(0)$$

uma vez que $\Lambda_{xx}(0) > 0$ por se tratar de um termo quadrático.

2. Como para o processo estacionário a covariância depende apenas do tamanho do intervalo entre os dois instantes de tempo escolhidos, não faz diferença se este intervalo for l ou $-l$.

2.3.4 Ergodicidade

As características dos processos estocásticos apresentadas até este ponto são definidas para infinitas realizações. No estudo de processos estocásticos, muitas vezes se tem apenas uma realização, com um número grande o suficiente de instantes de tempo. Desta forma, é importante saber se um processo estocástico tem alguma propriedade que permita que, com apenas uma realização, seja possível determinar as características definidas para infinitas realizações. Os processos estocásticos que têm esta propriedade são definidos como processos **ergódicos**. Este termo vem da junção das palavras

[5]Augustin Louis Cauchy (1789-1857). Matemático francês pioneiro em análise matemática. Em vida foi amigo de Lagrange e de Laplace

[6]Karl Hermann Amandus Schwarz (1843-1921). Matemático alemão que trabalhou principalmente com teoria de funções, geometria diferencial e cálculo variacional. Foi membro da academia de ciências de Berlim e professor da Universidade de Berlim.

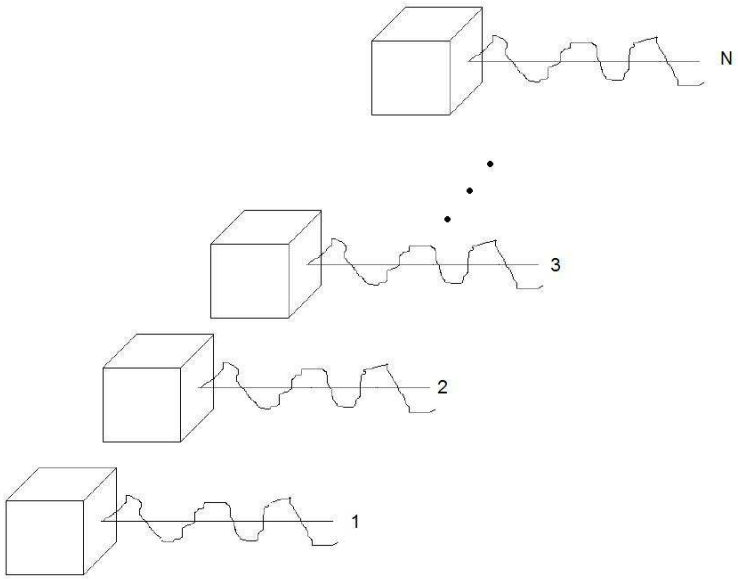

Fig. 2.2: Processo estocástico ergódico. Neste caso, obter as propriedades estatísticas do processo considerando todas as realizações 1, 2, 3... N é equivalente a se obter as propriedades ao longo de apenas uma das realizações.

gregas para trabalho (*έργον*) e caminho (*οδός*) e foi introduzido por Boltzmann[7] no estudo de dinâmica de partículas para definir um sistema em que a órbita passa através de todo ponto da superfície de energia [58].

Em outras palavras, um processo estocástico é dito ergódico quando suas características estatísticas são as mesmas ao se tomar medidas em um conjunto de realizações e ao se tomar medidas de apenas uma realização ao longo do tempo. Por exemplo, se houver um conjunto de partículas em um volume fechado e se a velocidade das partículas for um processo ergódico, a média das velocidades de todas as partículas ao longo de um número grande o suficiente de instantes de tempo tende à média da velocidade de apenas uma das partículas ao longo de um número grande o suficiente de instantes de tempo.Na figura 2.2 o conceito de ergodicidade é ilustrado.

De maneira mais formal, seja $x(t)$ um processo estocástico estacionário de Markov, de segunda ordem e com média nula. Se houver apenas uma realização do processo estocástico $x(t)$, pode-se definir uma média $m(N)$ e uma covariância $r_{xx}(l)$ desta realização da seguinte forma:

$$m(N) = \frac{1}{2N+1} \sum_{t=-N}^{N} x(t) \tag{2.15}$$

[7]Ludwig Eduard Boltzmann (1844-1906). Físico e matemático austríaco conhecido como um dos fundadores da mecânica estatística. Contemporâneo de Kirchhoff e Hemholtz, gradou-se e defendeu seu doutorado sobre teoria cinética dos gases na Universidade de Viena. Tornou-se professor das Universidades de Viena e de Graz, e se casou com Henriette von Aigentler, sua aluna em Graz. Passou também pelas Universidades de Munique e Leipzig, na Alemanha. Seu busto pode ser visto na galeria dos professores da Universidade de Viena, assim como em outros lugares.

$$r_{xx}(l) = \lim_{N\to\infty} \frac{1}{2N+1} \sum_{t=-N}^{N} x(t+l)x(t) \quad l \in \mathcal{Z} \tag{2.16}$$

Se a média de $m(N)$ coincidir com a média do processo estocástico x ou seja, $E[m(N)] = \mu_x(t)$ e se a covariância de $m(n)$ for nula, ou seja, $E[(m(N) - \mu_x)^2] = 0$ então o primeiro momento do processo $x(t)$ será característico de um processo ergódico. Da mesma forma, se a média da covariância $r_{xx}(l)$ for $\Lambda_{xx}(l)$ e a variância da variância for nula, o processo terá características de ergodicidade com relação aos momentos de segunda ordem. Como o processo é de Markov e de segunda ordem, garantir sua ergodicidade com relação aos dois primeiros momentos é suficiente para garantir sua ergodicidade completa.

O objetivo agora é estudar as condições para que isto ocorra. Para a primeira condição do primeiro momento tem-se o seguinte:

$$E[m(N)] = E\left[\frac{1}{2N+1}\sum_{t=-N}^{N} x(t)\right] = \frac{1}{2N+1}\sum_{t=-N}^{N} E[x(t)] = \mu_x(t) \tag{2.17}$$

ou seja, a primeira condição é sempre satisfeita.

Para a segunda condição tem-se o seguinte:

$$\begin{aligned}
E[(m(N)-\mu_x)^2] = E\left[\left(\frac{1}{2N+1}\sum_{t=-N}^{N}(x(t)-\mu_x)\right)^2\right] = \\
= \frac{1}{(2N+1)^2}\{E[(x_{-N}-\mu_x)(x_{-N}-\mu_x)] + \\
+E[(x_{-N}-\mu_x)(x_{-N+1}-\mu_x)] + \ldots + \\
+E[(x_{-N}-\mu_x)(x_N-\mu_x)] + \\
+E[(x_{-N+1}-\mu_x)(x_{-N}-\mu_x)] + \\
+E[(x_{-N+1}-\mu_x)(x_{-N+1}-\mu_x)] + \ldots + \\
+E[(x_{-N+1}-\mu_x)(x_N-\mu_x)] + \ldots + \\
+E[(x_N-\mu_x)(x_{-N}-\mu_x)] + \\
+E[(x_N-\mu_x)(x_{-N+1}-\mu_x)] + \ldots + \\
+E[(x_N-\mu_x)(x_N-\mu_x)]\} =
\end{aligned}$$

$$= \frac{1}{(2N+1)^2} \sum_{t=-N}^{N} \sum_{s=-N}^{N} \Lambda_{xx}(t-s) \tag{2.18}$$

Note que na somatória do final da equação 2.18 tem-se as seguintes quantidades de funções de covariância:

Covariância	**Quantidade**	**Ocorrências**
$\Lambda_{xx}(-2N)$	1	$\Lambda_{xx}(N,-N)$
$\Lambda_{xx}(-2N+1)$	2	$\Lambda_{xx}(N,-N+1)$ e $\Lambda_{xx}(N-1,-N)$
$\Lambda_{xx}(-2N+2)$	3	$\Lambda_{xx}(N,-N+2)$, $\Lambda_{xx}(N-1,-N+1)$ e $\Lambda_{xx}(N-2,-N)$
$\vdots$	$\vdots$	$\vdots$
$\Lambda_{xx}(-1)$	2N	$\Lambda_{xx}(-N+1,-N), \ldots, \Lambda_{xx}(N-1,N)$
$\Lambda_{xx}(0)$	2N+1	$\Lambda_{xx}(-N,-N), \ldots, \Lambda_{xx}(N,N)$
$\Lambda_{xx}(1)$	2N	$\Lambda_{xx}(-N,-N+1), \ldots, \Lambda_{xx}(N,N-1)$
$\vdots$	$\vdots$	$\vdots$
$\Lambda_{xx}(2N-2)$	3	$\Lambda_{xx}(-N+2,N)$, $\Lambda_{xx}(-N+1,N-1)$ e $\Lambda_{xx}(-N,N-2)$
$\Lambda_{xx}(2N-1)$	2	$\Lambda_{xx}(-N+1,N)$ e $\Lambda_{xx}(-N,N-1)$
$\Lambda_{xx}(-2N)$	1	$\Lambda_{xx}(-N,N)$

Ou seja,

$$E[(m(n)-\mu_x)^2] = \frac{1}{(2N+1)^2}(\Lambda_{xx}(2N) + \ldots + (2N+1)\Lambda_{xx}(0)\ldots + \Lambda_{xx}(2N))$$

Portanto, se tomarmos os termos entre parênteses e colocarmos em uma somatória em k indo de $-2N$ a $2N$, teremos $2N+1$ vezes cada termo. Isto se aplica quando $k=0$, no entanto, para $k \neq 0$ isto não se aplica. Por este motivo, um dos dois $2N+1$ do denominador da fração que multiplica a somatória entra para dentro da somatória e os $\Lambda_{xx}(k)$ são multiplicados pelo fator $1 - \frac{|k|}{2N+1}$. De fato, note que:

$$\begin{array}{ll} k = 2N & 1 - \frac{|k|}{2N+1} = \frac{1}{2N+1} \\ k = 2N-1 & 1 - \frac{|k|}{2N+1} = \frac{2}{2N+1} \\ \vdots & \vdots \\ k = 1 & 1 - \frac{|k|}{2N+1} = \frac{2N}{2N+1} \\ k = 0 & 1 - \frac{|k|}{2N+1} = \frac{2N+1}{2N+1} \end{array}$$

Portanto, temos finalmente que:

$$E[(m(n)-\mu_x)^2] = \frac{1}{2N+1} \sum_{k=-2N}^{2N} \left(1 - \frac{|k|}{2N+1}\right) \Lambda_{xx}(k) \tag{2.19}$$

Ao observar a equação 2.19 pode-se notar que se a seguinte expressão for válida:

$$\lim_{N\to\infty} \frac{1}{N+1} \sum_{k=0}^{N} \Lambda_{xx}(k) = 0 \tag{2.20}$$

tem-se que $E[(m(n) - \mu_x)^2] = 0$ e, portanto, o primeiro momento do processo é o de um processo ergódico. A somatória em k presente na equação 2.20 é conhecida como soma de Cesàro[8].

Com relação ao segundo momento, o valor esperado da variância r_{xx} é:

$$\begin{aligned} E[r_{xx}(l,N)] &= E\left[\tfrac{1}{2N+1}\textstyle\sum_{t=-N}^{N} x(t+l)x(t)\right] = \\ &= \tfrac{1}{2N+1}\textstyle\sum_{t=-N}^{N} E[x(t+l)x(t)] = \\ &= \Lambda_{xx}(l) \end{aligned} \tag{2.21}$$

Ou seja, sob este critério sempre se satisfaz a condição para a qual o processo é ergódico. Portanto, se a variância da variância $r_{xx}(l,N)$ for nula, o processo terá o segundo momento característico de um processo ergódico. Para calcular a variância da variância é introduzida a variável auxiliar $\xi(l) = x(t+l)x(t)$ de forma que $\mu_\xi(l) = \Lambda_{xx}(l)$ e a variância do processo ξ para um intervalo k é calculada da seguinte forma:

$$\begin{aligned} \Lambda_{\xi\xi}(k) &= E[(x(t+l+k)x(t+k) - \mu_\xi(l))(x(t+l)x(t) - \mu_\xi(l))] \\ &= E[(x(t+l+k)x(t+k)x(t+l)x(t)] - E[x(t+l+k)x(t+k)]\mu_\xi(l) - \\ &\quad -\mu_\xi(l)E[x(t+l)x(t)] + \mu_\xi(l)^2 \\ &= E[x(t+l+k)x(t+k)x(t+l)x(t)] - \mu_\xi(l)^2 - \mu_\xi(l)^2 + \mu_\xi(l)^2 \\ &= E[x(t+l+k)x(t+k)x(t+l)x(t)] - \mu_\xi(l)^2 \end{aligned} \tag{2.22}$$

A partir disto pode-se calcular a variância da variâcia da seguinte forma:

$$\begin{aligned} E[(r_{xx}(l,N) - \Lambda_{xx}(l))^2] &= E\left[\left(\tfrac{1}{2N+1}\textstyle\sum_{t=-N}^{N}(x(t+l)x(t) - \Lambda_{xx}(l))\right)^2\right] \\ &= E\left[\left(\tfrac{1}{2N+1}\textstyle\sum_{t=-N}^{N}(\xi(l) - \mu_\xi(l))\right)^2\right] \end{aligned}$$

Com procedimentos semelhantes aos feitos para o primeiro momento encontra-se que:

$$E[(r_{xx}(l,N) - \Lambda_{xx}(l))^2] = \frac{1}{2N+1}\sum_{k=-2N}^{2N}\left(1 - \frac{|k|}{2N+1}\right)\Lambda_{\xi\xi}(k) \tag{2.23}$$

Com isto, tem-se que o processo tem momento de segunda ordem caracterísitico de um processo ergódico apenas se a seguinte condição for satisfeita:

[8]Ernesto Cesàro (1859-1906). Matemático italiano que trabalhou no campo da geometria diferencial e das séries infinitas. Ficou conhecido justamente pela média de Cesàro, que é a média de uma série divergente obtida como soma de uma outra série. Ele também foi homenageado com o nome de um teorema que apresenta um resultado sobre séries infinitas.

$$\lim_{N\to\infty} \frac{1}{N+1} \sum_{k=0}^{N} \Lambda_{\xi\xi}(k) = 0 \tag{2.24}$$

As condições para que um processo seja ergódico determinadas acima formam um teorema ergódico. Provas mais formais de teoremas ergódicos no contexto da mecânica estatística foram desenvolvidas por Von Neumann[9] [63] e por Birkhoff[10] [7].

2.4 Processos estocásticos multivariáveis

Ao se formar um vetor a partir da concatenação de dois ou mais processos estocásticos monovariáveis, tem-se um vetor de processos estocásticos, também definido como **processo estocástico multivariável**. Seja, por exemplo, um processo estocástico multivariável $z(t)$ definido como a concatenação de dois processos estocásticos monovariáveis $x(t)$ e $y(t)$, ou seja:

$$z(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix}$$

A média do processo $z(t)$, denotada por μ_z, é um vetor definido da seguinte maneira:

$$\mu_z = E[z(t)] = E\left[\begin{bmatrix} x(t) \\ y(t) \end{bmatrix}\right] = \begin{bmatrix} E[x(t)] \\ E[y(t)] \end{bmatrix} = \begin{bmatrix} \mu_x \\ \mu_y \end{bmatrix}$$

em que μ_x é a média do processo estocástico $x(t)$ e μ_y é a média do processo estocástico $y(t)$.

Supondo que $x(t)$ e $y(t)$ tenham média nula, a covariância de $z(t)$ será uma matriz igual a:

$$\begin{aligned} \Lambda_z(t+l,t) &= E[z(t+l)z(t)^T] = E\left[\begin{bmatrix} x(t+l) \\ y(t+l) \end{bmatrix} [x(t) \quad y(t)]\right] \\ &= \begin{bmatrix} \Lambda_{xx}(t+l,t) & \Lambda_{xy}(t+l,t) \\ \Lambda_{yx}(t+l,t) & \Lambda_{yy}(t+l,t) \end{bmatrix} \end{aligned}$$

em que $\Lambda_{xx}(t+l,t)$ é a autocovariância de $x(t)$ entre os instantes t e $t+l$, $\Lambda_{xy}(t+l,t)$ é a covariância cruzada entre os processos $x(t)$ e $y(t)$ entre os instantes t e $t+l$ e $\Lambda_{yy}(t+l,t)$ é a autocovariância do processo $y(t)$ entre os instantes t e $t+l$.

[9] Margittai Neumann János Lajos (Johann von Neumman ou John von Neumann) (1903-1957). Matemático húngaro de origem judia, nacionalizado estadunidense, considerado um dos maiores matemáticos do século XX. Nascido em Budapeste, à época no império Austro-Húngaro, se destacava desde pequeno por seu raciocínio rápido e facilidade de memória. Aos 12 anos havia entendido o livro *Théorie des Fonctions* de Borel. Em 1921 foi estudar engenharia química em Berlim, onde teve aulas com Albert Einstein, e paralelamente fez um doutorado em matemática na Universidade de Budapeste. Posteriomente foi estudar engenharia química em Zurique e em 1926, aos 23 anos, se tornou o mais novo professor da Universidade de Berlim. Em 1930 se mudou para os Estados Unidos devido à perseguição do regime nazista. Nos Estados Unidos foi professor de Princenton e trabalhou também no projeto Manhattan, de desenvolvimento da bomba atômica. Propoz uma arquitetura de computadores que é utilizada até os dias de escrita desta tese.

[10] George David Birkhoff (1884-1944). Matemático estadunidense conhecido principalmente pela prova do teorema ergódico. Graduado e mestre por Harvard, completou seu doutorado em 1907 na Universidade de Chicago. Foi professor das Universidades de Wisconsin, Princeton e Harvard. Nos últimos anos de vida se interessou por temas como estética, arte, música e poesia, tendo publicado um livro a respeito da teoria matemática da estética.

Se os processos $x(t)$ e $y(t)$ forem descorrelacionados, os termos da matriz $\Lambda_z(t+l,t)$ que estão fora dos blocos diagonais principais, ou seja, os termos que representam as covariâncias cruzadas, são nulos. No contexto de espaços vetoriais de Hilbert[11], se o produto interno entre dois processos estocásticos for definido como sendo sua função de variância, ter este produto igual a zero significa ortogonalidade entre os dois processos. Portanto dizer que os processos são descorrelacionados é equivalente a dizer que eles são ortogonais no espaço em que eles estão definidos. Estes conceitos são detalhados nas seções 2.7.1 e 2.7.2.

Se um processo estocástico multivariável for formado pela concatenação de dois ou mais processos estocásticos estacionários, diz-se que o novo processo estocástico é **conjuntamente estacionário**. De fato, ao se observar a covariância $\Lambda_z(t+l,t)$, nota-se que se $x(t)$ e $y(t)$ são estacionários, $\Lambda_{xx}(t+l,t) = \Lambda_{xx}(l)$, $\Lambda_{xy}(t+l,t) = \Lambda_{xy}(l)$ e $\Lambda_{yy}(t+l,t) = \Lambda_{yy}(l)$, portanto $\Lambda_z(t+l,t) = \Lambda_z(l)$.

As covariâncias cruzadas entre dois processos estacionários, que fazem parte da matriz de covariância de um processo conjuntamente estacionário, obedecem ao seguinte lema:

Lema 2.2 *A função de covariância $\Lambda_{xy}(l)$ entre dois processos estocásticos estacionários $x(t)$ e $y(t)$ tem as seguintes propriedades:*

1. *Antisimetria, ou seja, $\Lambda_{xy}(l) = \Lambda_{yx}(-l)$*
2. *É limitada, ou seja, $|\Lambda_{xy}(l)|^2 \leq \Lambda_{xx}(0)\Lambda_{yy}(0)$*

Prova:

1. $\Lambda_{xy}(l) = E[x(t+l)y(t)] = E[y(t-l)x(t)] = \Lambda_{yx}(-l)$
2. Da desigualdade de Cauchy-Schwarz tem-se que

$$|\Lambda_{xy}(l)|^2 = E[x(l)y(0)]^2 \leq E[x(l)^2]E[y(0)^2]$$

mas, conforme enunciado no lema 2.1:

$$0 \leq E[x(l)^2] \leq \sigma_x^2 \quad 0 \leq E[y(l)^2] \leq \sigma_y^2$$

Portanto,

$$E[x(l)y(0)]^2 \leq \sigma_x^2\sigma_y^2 = \Lambda_{xx}(0)\Lambda_{yy}(0)$$

2.5 Exemplos de processos estocásticos

2.5.1 Ruído branco

Um exemplo importante de processo estocástico é o **ruído branco**. A característica do ruído branco é ter média nula e descorrelação entre as variáveis aleatórias tomadas em instantes de tempo diferentes. Ou seja, $v(t)$ é um ruído branco se:

[11] David Hilbert (1862-1943). Matemático alemão que se tornou famoso por ter generalizado o conceito do espaço euclidiano para infinitas dimensões. Por este motivo, qualquer espaço em que esteja definido um produto interno é hoje conhecido como espaço de Hilbert. Professor da Universidade de Götingen viu seus colegas serem exterminados pelo regime nazista e chegou a protestar de forma irônica sobre o problema com o ministro da educação Bernhard Rust.

- $E[v(t)] = 0$
- $E[v(t), v(s)] = \begin{cases} 0 & t \neq s \\ \sigma_v^2 & t = s \end{cases}$

em que σ_v é uma constante e seu quadrado é definido como a **covariância para atraso zero** do ruído branco $v(t)$.

A partir de sua covariância, que significa que amostras do ruído branco tomadas em intantes de tempo diferentes são independentes, tem-se que a função de densidade de probabilidade conjunta para um ruído branco é:

$$p_{t,s}(v(t),\ v(s)) = p_t(v(t))p_s(v(s))$$

O ruído branco é um processo fracamente estacionário uma vez que os seus dois primeiros momentos não são função dos instantes de tempo em que foram tomados. É também um processo de segunda ordem, uma vez que sua variância é finita. Como suas amostras são descorrelacionadas, o processo também é Markoviano, ou seja, a previsão do processo em um determinado instante de tempo a partir de todas as amostras passadas é a mesma obtida apenas com o instante de tempo imediatamente anterior.

A partir do processo estocástico ruído branco é possível gerar uma realização de qualquer outro processo estocástico através de uma filtragem. O problema de determinação do filtro que transforma um ruído branco em uma realização de processo estocástico desejado é conhecido como **problema de realização de séries temporais** e é tratado com detalhes no capítulo 5.

Ao se fazer a concatenação de dois ou mais ruídos brancos descorrelacionados se tem um **ruído branco multivariável**. Sejam $v(t)$ e $w(t)$ dois ruídos brancos descorrelacionados, com covariâncias para atraso zero respectivamente iguais a σ_v^2 e σ_w^2, concatenados formando um ruído branco multivariável $e(t)$. A média de $e(t)$ será um vetor bidimensional nulo, e sua matriz de covariância será a seguinte:

$$E[e(t)e(s)^T] = \begin{cases} \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} & t \neq s \\ \begin{bmatrix} \sigma_v^2 & 0 \\ 0 & \sigma_w^2 \end{bmatrix} & t = s \end{cases}$$

Semelhantemente ao caso monovariável, também é possível gerar realizações de processos estocásticos multivariáveis a partir da filtragem de ruídos brancos multivariáveis. A determinação do filtro que transforma um ruído branco multivariável em uma realização de um processo estocástico multivariável é definida como **problema de realização de séries temporais multivariáveis**. Neste caso, a análise dos filtros no espaço de estado reduz consideravelmente a complexidade do problema de realização. Nesta tese, são propostas algumas contribuições para a solução do problema de realização de séries temporais multivariáveis no espaço de estado. Estas contribuições são detalhadas no capítulo 7.

2.5.2 Passeio aleatório

Um exemplo simples de processo estocástico formado pela filtragem de um ruído branco é o **passeio aleatório** $x(t)$ definido a seguir:

$$x(t) = \sum_{i=0}^{t} v(i) \tag{2.25}$$

sendo $x(0) = 0$. Isto pode ser escrito de forma equivalente como:

$$x(t) = x(t-1) + v(t) \tag{2.26}$$

O passeio aleatório é um processo markoviano, uma vez que o valor do processo em um instante de tempo seguinte ao atual é determinado apenas pelo valor do processo no instante atual adicionado a um componente aleatório, conforme fica claro na equação 2.26.

O valor esperado do passeio aleatório é o seguinte:

$$\mu_x(t) = E[x(t)] = E\left[\sum_{i=0}^{t} v(i)\right] = \sum_{i=0}^{t} E[v(i)] = 0$$

uma vez que o processo $v(t)$ é um ruído branco e, portanto, tem média nula. Para os instantes de tempo s e t, tais que $s < t$, a covariância do *passeio aleatório* será:

$$\Lambda_{xx}(t,s) = E[x(s)x(t)] = E\left[\sum_{i=0}^{s} v(i) \sum_{j=0}^{t} v(j)\right]$$

como $E[v(s)v(t)] = 0 \,\forall\, s \neq t$, tem-se que, ao se passar o operador de valor esperado para dentro das chaves, os únicos termos que não se anulam são aqueles em que há um produto dos dois processos estocásticos no mesmo instante de tempo, portanto:

$$E\left[\sum_{i=0}^{s} v(i) \sum_{j=0}^{t} v(j)\right] = \sum_{i=0}^{s} E[v(i)^2] = s\sigma^2$$

Em que σ^2 é a variância do processo $v(t)$. Note que na última igualdade se considerou a estacionariedade do ruído branco. Pela definição deste processo nota-se que esta propriedade de fato se aplica. O *passeio aleatório* no entanto não é um processo estacionário pois a covariância sempre será função do menor instante de tempo escolhido para o cálculo e não do intervalo entre os instantes de tempo escolhidos.

2.5.3 Outros processos gerados pelo ruído branco

Suponha agora que $v(t)$ seja um ruído branco discreto de média nula e covariância σ^2, e que este ruído seja a entrada de um sistema discreto que tenha a seguinte função de transferência:

$$H(z) = \frac{\gamma_1 z^{-1} + \gamma_2 z^{-2} + \ldots + \gamma_q z^{-q}}{1 + \theta_1 z^{-1} + \theta_2 z^{-2} + \ldots + \theta_p z^{-p}} = \frac{\Gamma(z)}{\Theta(z)}$$

Este sistema é conhecido como *ARMA*, ou seja, **A**uto **R**egressivo pois a saída depende dela mesma em instantes de tempo passado, e com ***M**oving **A**verage*, pois toma-se uma média móvel das entradas ao longo do tempo [1]. Esta função de transferência será estável se as raízes de $\Theta(z)$ estiverem dentro do círculo de raio unitário. Se o polinômio $\Gamma(z)$ também tiver raízes dentro do círculo de raio

unitário, a inversa da função de transferência também será estável. Desta forma, é possível gerar um ruído branco ao se entrar com o sinal de saída $y(t)$ em um filtro com função de transferência igual ao inverso de $H(z)$. Este filtro é chamado de **branqueador**. Ao filtro com função de transferência $H(z)$ é dado o nome de filtro **formatador**[12].

O ruído branco pode ser visto como um análogo da função impulso definida para sistemas determinísticos. Esta analogia pode ser feita pois, da mesma forma que um sistema determinístico pode ser descrito sinteticamente por sua resposta ao impulso e, a partir desta resposta ao impulso, um modelo matemático pode ser construído para ter o mesmo comportamento que o sistema, um processo estocástico pode ser descrito sinteticamente por suas covariâncias, que são a resposta ao ruído branco de um modelo matemático que tem o mesmo comportamento que o processo estocástico. O problema de se encontrar um modelo com a mesma resposta ao impulso de um sistema dinâmico determinístico é tratado no capítulo 3 e o problema de se encontrar um modelo com a mesma resposta ao ruído branco, ou seja, com as mesmas covariâncias que uma determinada série temporal, é tratado no capítulo 5 desta tese.

2.6 Análise espectral

2.6.1 Caso monovariável

Se um processo estocástico $x(t)$ discreto, de segunda ordem, de média zero e com taxa de amostragem unitária é tal que a soma infinita do módulo da função de covariância não diverge, ou seja:

$$\sum_{l=-\infty}^{\infty} |\Lambda_{xx}(l)| \leq \infty$$

então pode-se definir sua função de **densidade espectral** em freqüência, como sendo a transformada discreta de Fourier[13] da função de covariância, ou seja:

$$\Phi_{xx}(z) = \sum_{l=-\infty}^{\infty} \Lambda_{xx}(l) z^{-l} \tag{2.27}$$

O espectro traz informações sobre a intensidade do processo estocástico em cada uma das freqüências uma vez que $z = e^{j\omega}$, $-\pi < \omega \leq \pi$.

Como a transformada de Fourier pode ser invertida, tem-se que a covariância é igual a:

$$\Lambda_{xx}(l) = \frac{1}{2\pi j} \int_{|z|=1} \Phi_{zz}(z) z^{-1} dz = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{æ\omega t} \Phi_{xx}(\omega)\, d\omega, \quad l \in \mathcal{Z} \tag{2.28}$$

[12]Tradução para o termo *shaping*

[13]Jean Baptiste Joseph Fourier (1768-1830). Matemático francês que lutou a favor da revolução francesa. Fourier recebeu uma cadeira da École Polytechnique, mas logo depois viajou ao Egito junto com Napoleão por quem foi nomeado governador do Baixo Egito, onde organizava a logística de armamentos para o exército francês e dava palpites matemáticos no Instituto Egípcio, fundado por Napoleão na cidade de Cairo. Seus estudos foram principalmente sobre sistemas térmicos mas sua fama veio através do enunciado que todas as funções, contínuas ou descontínuas, são formadas por combinações lineares de senos e cossenos. Foi contemporâneo de Laplace, Legendre, dentre outros.

Como o espectro e a covariância têm uma relação de ida e volta, tem-se que ambos carregam a mesma informação sobre o processo estocástico que eles representam[14].

Se a taxa de amostragem do processo estocástico não for unitária mas sim igual a Δt, é necessário que se faça uma mudança de escala na transformada de Fourier de forma que:

$$\Phi_{xx}(\nu) = \Delta t \sum_{l=-\infty}^{\infty} e^{-j\nu\Delta tl}\Lambda_{xx}(l) \quad \frac{-\pi}{\Delta t} < \nu \leq \frac{\pi}{\Delta t} \tag{2.29}$$

Em que $\nu = \frac{\omega}{\Delta t}$

Sobre a densidade espectral pode-se enunciar os seguintes lemas:

Lema 2.3 *A função de densidade espectral satisfaz às seguintes propriedades:*

1. *Simetria:* $\Phi_{xx}(\omega) = \Phi_{xx}(-\omega) \quad -\pi < \omega \leq \pi$
2. *Não negatividade:* $\Phi_{xx}(\omega) \geq 0 \quad -\pi < \omega \leq \pi$

Prova:

1. $\Phi_{xx}(\omega) = \sum_{l=-\infty}^{\infty} \Lambda_{xx}(l)e^{-j\omega l} = \sum_{l=\infty}^{-\infty} \Lambda_{xx}(-l)e^{j\omega l} =$
$= \sum_{l=-\infty}^{\infty} \Lambda_{xx}(l)e^{-j(-\omega)l} = \Phi_{xx}(-\omega)$
2. Como a densidade espectral é a somatória dos produtos de dois valores positivos que são a covariância e uma potência do número positivo e, ela também será positiva.

A partir do instante em que se define a densidade espectral do segundo momento de um processo estocástico, é possível que se descubra a densidade espectral da saída de um sistema, ou filtro, que tem como entrada o primeiro processo, desde que o sistema seja estável e que sua resposta ao impulso seja conhecida. Se os processos de entrada e de saída tiverem média nula[15] e forem markovianos, toda a informação necessária para conhecer a saída será conhecida através de sua densidade espectral. O lema a seguir formaliza esta discussão:

Lema 2.4 *Seja um sistema, ou filtro, linear, causal, invariante no tempo e com função de transferência $G(z)$ estável. Seja também um processo estocástico $u(t)$ estacionário, de segunda ordem, de média nula e markoviano com convariância satisfazendo a seguinte condição:*

$$\sum_{l=-\infty}^{\infty} |\Lambda_{uu}(l)| < \infty$$

Então a saída $y(t)$ deste sistema também será um processo estocástico, de segunda ordem, com média nula com a seguinte função de densidade espectral:

[14] Apesar de a transformada inversa ser apresentada como uma integral, normalmente são usados outros métodos para se calcular a transformada Z inversa, como por exemplo, a decomposição dos quocientes de polinômios em z em frações parciais. Mais informações podem ser encontradas nas referências [10] e [34].

[15] Se a média dos processos não for nula, a média pode ser considerada um termo determinístico da entrada do sistema. Como o sistema é linear, a saída será igual à soma dos efeitos da entrada determinísitica e da estocástica. Por isso, pode-se considerar para estudos que a média de um processo estocástico na entrada de um sistema linear é sempre nula.

$$\Phi_{yy}(z) = G(z)G(z^{-1})\Phi_{uu}(z) \tag{2.30}$$

Ou ainda, em função de ω:

$$\Phi_{yy}(\omega) = |G(e^{j\omega})|^2\Phi_{uu}(\omega) \tag{2.31}$$

A variância do processo y será dada por:

$$\sigma_y^2 = \frac{1}{2\pi}\int_{-\pi}^{\pi} |G(e^{j\omega})|^2\Phi_{uu}(\omega)\, d\omega \tag{2.32}$$

Prova: Primeiro vamos provar que a média do processo $y(t)$ é zero. Como $G(z)$ é estável tem-se que:

$$y(t) = \sum_{i=0}^{\infty} g(i)u(t-i)$$

Tomando o valor esperado de ambos os lados se tem:

$$E[y(t)] = E\left[\sum_{i=0}^{\infty} g(i)u(t-i)\right] = \sum_{i=0}^{\infty} g(i)E[u(t-i)] = 0$$

Vamos agora provar que o processo y é de segunda ordem, ou seja, que sua variância $\Lambda_{yy}(0)$ é finita. A covariância de y é dada por:

$$\begin{aligned}
\Lambda_{yy}(l) &= E[y(t+l)y(t)] = \\
&= E\left[\textstyle\sum_{i=0}^{\infty} g(i)u(t+l-i)\sum_{k=0}^{\infty} g(k)u(t-k)\right] = \\
&= \textstyle\sum_{i=0}^{\infty}\sum_{k=0}^{\infty} g(i)g(k)E[u(t+l-i)u(t-k)])] \\
&= \textstyle\sum_{i=0}^{\infty}\sum_{k=0}^{\infty} g(i)g(k)\Lambda_{uu}(l+k-i)
\end{aligned}$$

então:

$$\begin{aligned}
\textstyle\sum_{t=-\infty}^{\infty}|\Lambda_{yy}(l)| &= \textstyle\sum_{t=-\infty}^{\infty}\sum_{i=-\infty}^{\infty}\sum_{k=-\infty}^{\infty}|g(i)||g(k)||\Lambda_{uu}(l+k-i)| \\
&= = \left(\textstyle\sum_{i=0}^{\infty}|g(i)|\right)^2\sum_{l=-\infty}^{\infty}|\Lambda_{uu}(l+k-i)|
\end{aligned}$$

Como o sistema é estável, a somatória infinita de $|g(i)|$ converge. Como o processo de entrada u é de segunda ordem, a segunda somatória também converge. Portanto, a covariância da saída converge e a variância converge, o que é visto simplesmente ao se substituir l por zero na equação acima, e portanto, o processo de saída do sistema é de segunda ordem.

Para encontrar a densidade espectral da saída o procedimento é o seguinte:

$$\begin{aligned}
\Phi_{yy}(z) &= \textstyle\sum_{l=-\infty}^{\infty} \Lambda_{yy}(l) z^{-l} = \\
&= \textstyle\sum_{l=-\infty}^{\infty} z^{-l} \left(\sum_{i=0}^{\infty} \sum_{k=0}^{\infty} g(i) g(k) \Lambda_{uu}(l+k-i)\right) = \\
&= \textstyle\sum_{i=0}^{\infty} g(i) \sum_{k=0}^{\infty} g(k) \sum_{l=-\infty}^{\infty} z^{-l} \Lambda_{uu}(l+k-i) \\
&= \textstyle\sum_{i=0}^{\infty} g(i) z^{-i} z^{i} \sum_{k=0}^{\infty} g(k) z^{k} z^{-k} \sum_{l=-\infty}^{\infty} z^{-l} \Lambda_{uu}(l+k-i) \\
&= \textstyle\sum_{i=0}^{\infty} g(i) z^{-i} \sum_{k=0}^{\infty} g(k) z^{k} \sum_{l=-\infty}^{\infty} z^{-(l+k-i)} \Lambda_{uu}(l+k-i) = \\
&= G(z) G(z^{-1}) \Phi_{uu}(z)
\end{aligned} \tag{2.33}$$

Aplicando a transformada inversa desta equação e fazendo $l = 0$ se chega à variância do processo estocástico y.

2.6.2 Caso multivariável

Assim como foi feito para os processos monovariáveis, também é possível definir a densidade espectral nos casos multivariáveis. Seja um processo estocástico discreto, de média nula e n dimensional. A matriz de covariância é a seguinte:

$$\Lambda_{xx}(l) = E[x(t+l)x^T(t)]$$

Se a soma infinita dos elementos da diagonal principal desta matriz de covariância for convergente, ou seja, se todas os n processos que formam o processo $x(t)$ são de segunda ordem, ou ainda se:

$$\sum_{t=-\infty}^{\infty} |\Lambda_{ii}(l)| \leq \infty \quad 1 \leq i \leq n$$

Então define-se a densidade espectral matricial como sendo a transformada discreta de Fourier da matriz de covariâncias do processo x, ou seja:

$$\Phi_{xx}(z) = \sum_{l=-\infty}^{\infty} \Lambda_{xx}(l) z^{-l}$$

Ou ainda, em função de ω temos:

$$\Phi_{xx}(\omega) = \sum_{l=-\infty}^{\infty} \Lambda_{xx}(l) e^{-j\omega l} \quad -\pi \leq \omega < \pi$$

Também é possível determinar a transformada inversa com a seguinte integral:

$$\Lambda_{xx}(l) = \frac{1}{2\pi j} \int_{|z|=1} \Phi_{xx}(z) z^{t-l}\, dz = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega t} \phi_{xx}(\omega)\, d\omega \quad l \in \mathcal{Z}$$

Os elementos fora da diagonal principal da matriz de densidade espectral são conhecidos como densidades espectrais cruzadas e são funções de números complexos. Sobre a matriz de densidades espectrais pode-se enunciar o seguinte lema:

Lema 2.5 *A matriz de densidade espectral $\Phi_{xx}(\omega)$ tem as seguintes propriedades:*

1. *É hermitiana*[16,17]*, ou seja, $\Phi_{xx}(\omega) = \Phi_{xx}^H(-\omega)$*
2. *É não negativa: $\Phi_{xx}(\omega) \geq 0$*

Prova:

1. Como $\Lambda_{xx}(l) = \Lambda_{xx}^T(-l)$

$$\Phi_{xx}(\omega) = \sum_{l=-\infty}^{\infty} e^{-j\omega l}\Lambda_{xx}(l) = \sum_{l=-\infty}^{\infty} e^{-j\omega l}\Lambda_{xx}^T(-l) = \sum_{l=-\infty}^{\infty} e^{j\omega l}\Lambda_{xx}^T(l) = \Phi_{xx}^H(-\omega)$$

2. Seja $\xi(t) = \sum_{i=1}^{n} a_i x_i(t)$ em que a_i são coeficientes complexos. A covariância de ξ tem o seguinte valor:

$$\Lambda_{\xi\xi}(l) = \sum_{i,k=1}^{n} a_i \overline{a}_k \Lambda_{ik}(l)$$

   Em que $\overline{*}$ é o complexo conjugado de $*$. A transformada de Fourier de $\Lambda_{\xi\xi}$ é:

$$\Phi_{\xi\xi}(\omega) = \sum_{i,k=1}^{n} a_i \overline{a}_k \Phi_{ik}(\omega)$$

   Como $\Phi_{\xi\xi}(\omega) \geq 0$, Φ_{xx} é não negativa definida.

2.7 Processos estocásticos no espaço de Hilbert

2.7.1 Caso monovariável

Nesta seção é descrito o estudo dos processos estocásticos monovariáveis em um espaço definido para eles. Ao se definir os processos estocásticos em um espaço vetorial, em que estão definidos produto interno e norma, é possível que se use diversas ferramentas e definições como ortogonalidade, projeções, dentre outras. As técnicas de previsão do comportamento dos processos estocásticos são baseadas nestes princípios. Mais informações a respeito de espaços vetoriais podem ser encontradas em textos básicos de álgebra linear, como [8], [14], [50], [54], dentre outros.

O espaço gerado por um processo estocástico $y(k)$ estacionário, monovariável, de segunda ordem e média nula é denotado como:

$$\mathcal{H} = \left\{ \xi = \sum_{k=k_1}^{k_2} a_k y(k) | a_k \in \Re \right\}, \quad -\infty < k_1 \leq k_2 < \infty$$

[16]Uma matriz hermitiana é igual a sua complexa conjugada transposta

[17]Nome dado em homenagem a Charles Hermite (1822-1901), matemático francês que se dedicou ao estudo da teoria dos números, dos polinômios ortogonais, funções elípticas, dentre outros. Quando jovem foi afastado da École Polytechnique por ter um defeito de nascença em seu pé direito. Apesar disso conseguiu o título de Bacharel em 1847 e se tornou professor da École Politechnique em 1869. Durante a vida foi amigo de Cauchy e de Jacobi.

Ou seja, é o espaço gerado por todos os vetores ξ formados por combinações lineares dos vetores da base, que neste caso são as variáveis aleatórias obtidas ao se tomar o valor do processo estocástico y nos instantes de tempo entre k_1 e k_2.

Neste espaço se define o **produto interno** de dois vetores como sendo a covariância entre eles. Sejam dois vetores ξ e η contidos no espaço $\mathcal{H}$:

$$\xi = \sum_{i=i_1}^{i_2} a_i y(i) \quad \eta = \sum_{j=j_1}^{j_2} b_j y(j)$$

Como a média do processo y é nula o produto interno entre ξ e η é:

$$< \xi, \eta >_{\mathcal{H}} = E\left[\left(\sum_{i=i_1}^{i_2} a_i y(i)\right)\left(\sum_{j=j_1}^{j_2} b_j y(j)\right)\right] = \sum_{i,j\in D} E[y(i)y(j)]a_i b_j = \sum_{i,j\in D} \Lambda_{yy}(i-j)a_i b_j$$

em que D é o seguinte domínio: $D = \{(i,j)|i_1 \leq i \leq i_2;\ j_1 \leq j \leq j_2\}$.

Se a covariância $\Lambda_{yy}(i-j)$ for positiva [18], define-se a norma do vetor ξ como sendo:

$$||\xi||^2_{\mathcal{H}} = < \xi, \xi >_{\mathcal{H}} = \sum_{i,j\in D} \Lambda_{yy}(i-j)a_i a_j$$

Portanto, como no espaço $\mathcal{H}$ está definida uma norma, ele é um espaço de Hilbert conforme já havia sido adiantado.

Como exemplo seja o espaço gerado pelo ruído branco monovariável $e(t)$, que tem média nula e variância unitária:

$$\mathcal{H} = \left\{\xi_n = \sum_{k=0}^{\infty} a_k e(k) | \sum_{k=0}^{\infty} |a_k|^2 < \infty,\ a_k \in \Re\right\}$$

No limite, para $m > n \to \infty$, a norma da diferença entre dois vetores do espaço $\mathcal{H}$ será:

$$\begin{aligned} ||\xi_m - \xi_n||^2_{\mathcal{H}} &= ||\sum_{k=0}^{m} a_k e(k) - \sum_{k=0}^{n} a_k e(k)||^2_{\mathcal{H}} = ||\sum_{k=n+1}^{m} a_k e(k)||^2_{\mathcal{H}} \\ &= E\left[\sum_{k=n+1}^{m} a_k e(k) \sum_{i=n+1}^{m} a_i e(i)\right] = \sum_{k=n+1}^{m}\sum_{i=n+1}^{m} a_k a_i E[e_k e_i] = \\ &= \sum_{k=n+1}^{m} a_i^2 \end{aligned}$$

uma vez que e é branco com variância unitária, ou seja,$E[e_k e_i] = \delta_{k,i}$.

A somatória do quadrado de a tende a zero. Por este motivo ξ_n é um processo de Cauchy, ou seja, um processo que converge para um determinado valor depois de um intervalo de tempo suficientemente grande. Desta forma existe um limite quadrático:

$$\xi = q - \lim_{n\to\infty} \xi_n = q - \lim_{n\to\infty} \sum_{k=0}^{n} a_k e(k)$$

[18] Se a covariância não for positiva, a definição foge do conceito de norma.

Portanto, ao se agregar todos os possíveis limites de média quadrática, $\mathcal{H}$ se torna um espaço de Hilbert, então $\mathcal{H}$ pode ser escrito como:

$$\mathcal{H} = \left\{ \xi = q - \lim_{n\to\infty} \sum_{k=0}^{n} a_k e(k) | \sum_{k=0}^{\infty} |a_k|^2 < \infty,\ a_k \in \Re \right\}$$

Define-se como produto interno entre duas variáveis aleatórias monovariáveis a covariância entre elas. Se duas variáveis aleatórias monovariáveis ξ, η, que são vetores do espaço $\mathcal{H}$, são tais que $< \xi, \eta >_{\mathcal{H}} = 0$, elas são ditas **ortogonais**, o que é simbolizado por $\xi \perp \eta$. Da mesma forma se $\mathcal{W}$ é subespaço de $\mathcal{H}$ e todos os vetores w de $\mathcal{W}$ forem ortogonais a um vetor $\xi \in \mathcal{H}$ então o subespaço $\mathcal{W}$ é ortogonal a ξ, ou seja, $\mathcal{W} \perp \xi$. Neste caso o vetor ξ está contido no complemento ortogonal de $\mathcal{W}$, ou seja, no subespaço de $\mathcal{H}$ que tem intersecção vazia com $\mathcal{W}$ e que unido com $\mathcal{W}$ forma o espaço $\mathcal{H}$. O complemento ortogonal de $\mathcal{W}$ é denotado por $\mathcal{W}^{\perp}$.

Com o conceito de ortogonalidade, pode-se definir a projeção ortogonal. Seja $\mathcal{W}$ um subespaço do espaço de Hilbert $\mathcal{H}$. Para qualquer vetor $\xi \in \mathcal{H}$ há um único w_0 que satisfaz:

$$||\xi - w_0||_{\mathcal{H}} \leq ||\xi - w||_{\mathcal{H}} \quad \forall\, w \in \mathcal{W} \tag{2.34}$$

O vetor ω_0 é tal que $\xi - \omega_0 \perp \mathcal{W}$ e ω_0 é chamado de **projeção ortogonal** de ξ em $\mathcal{W}$. Se $\xi \in \mathcal{W}$, $\omega_0 = \xi$.

A partir deste conceito, a predição do valor futuro de um processo estocástico passa a ser vista como o procedimento de encontrar a projeção ortogonal do próximo valor do processo estocástico no subespaço definido pelos valores do processo nos instantes anteriores de tempo e, se o processo for markoviano, pelo subespaço definido pela saída do processo no instante de tempo atual. Esta idéia será explorada mais detalhadamente deste ponto em diante.

Seja $y(t)$ um processo estocástico monovariável estacionário de segunda ordem e média zero. Para prever o futuro do processo no instante $t + m$ como uma combinação linear das medidas obtidas nos instantes de tempo t, $t-1$, ... é definido um espaço $\mathcal{Y}_t$ de Hilbert gerado pelas medidas $y(t)$, $y(t-1)$, ...:

$$\mathcal{Y}_t = \left\{ \xi(t) = \sum_{k=0}^{\infty} a_k y(t-k) | \sum_{k=0}^{\infty} |a_k| < \infty,\ a_k \in \Re \right\}$$

Este espaço pode ser visto como a saída de um filtro com função de transferência $A(z) = \sum_{k=0}^{\infty} a_k z^{-k}$, excitado pela entrada $y(t)$. Por hipótese, é suposto que este filtro é estável. O espaço $\mathcal{Y}_t$ é linear, uma vez que ele é formado por combinações lineares de vetores. Além disto, este espaço é fechado com relação à soma e à multiplicação por escalar, conforme provado a seguir:

Sejam ξ e η dois vetores de $\mathcal{Y}_t$:

$$\xi = \sum_{k=0}^{\infty} a_k y(t-k) \quad \eta = \sum_{k=0}^{\infty} b_k y(t-k)$$

em que a soma infinita dos módulos de a_k e b_k são finitas. Então:

$$\xi + \eta = \sum_{k=0}^{\infty} (a_k + b_k) y(t-k)$$

Como $|a_k + b_k| \leq |a_k| + |b_k|$, $\sum_{k=0}^{\infty} |a_k + b_k| < \infty$, portanto $\xi + \eta \in \mathcal{Y}_t$, que, portanto, é fechado com relação à soma.

Seja c um escalar. Então:

$$c\xi = c\sum_{k=0}^{\infty} a_k y(t-k) = \sum_{k=0}^{\infty} ca_k y(t-k) < \infty$$

Portanto $c\xi \in \mathcal{Y}_t$, que, portanto, é fechado com relação à multiplicação por um escalar.

Como geralmente a medida $y(t+m)$ não pertence a $\mathcal{Y}_t$, a previsão de seu valor baseada nas medidas até o instante t consiste em encontrar uma aproximação $\hat{y}(t+m) \in \mathcal{Y}_t$ que seja a mais próxima possível da medida $y(t+m)$. Como foi visto acima, o melhor preditor é dado pela projeção ortogonal, ou seja:

$$\hat{y}(t+m) = \hat{E}[y(t+m)|\mathcal{Y}_t] \in \mathcal{Y}_t | [y(t+m) - \hat{y}(t+m)] \perp \mathcal{Y}_t$$

e o vetor $y(t+m) - \hat{y}(t+m)$ é definido como **erro de predição**.

A variância do erro de predição é definida como:

$$\sigma_m^2 = E[(y(t+m) - \hat{y}(t+m))^2], \quad m = 1, 2, \ldots$$

Como o processo y é estacionário por hipótese, o erro de predição é função apenas do intervalo m e não de t. Note que a variância do erro de predição é uma medida da incerteza sobre o futuro.

A cada instante de tempo se adquire uma informação sobre o processo. Se esta nova informação for linearmente dependente das informações já existentes, não há ganho de informação sobre o processo. No entanto, geralmente a nova informação tem algum componente linearmente independente das anteriores, e ela representa uma inovação. De qualquer forma, se $s \leq t$ forem dois instantes de tempo, $\mathcal{Y}_s \subset \mathcal{Y}_t$. Com isto, se o intervalo de tempo m entre o instante atual e o futuro for pequeno, a incerteza sobre o futuro será menor que se o intervalo m for maior. Portanto, a variância σ_m^2 é uma função não decrescente com relação a m:

$$0 \leq \sigma_1^2 \leq \sigma_2^2 \leq \ldots \tag{2.35}$$

Se a variância de um processo estacionário de segunda ordem for $\sigma_1^2 = 0$ não há incerteza sobre o futuro e o processo é dito **determinístico** ou **singular**. Se $\sigma_1^2 > 0$ o processo é dito **não determinístico** ou **regular**.

Se $\sigma_1^2 > 0$, pela desigualdade 2.35 tem-se que $\sigma_m^2 > 0 \, \forall \, m = 2, 3, \ldots$. De forma análoga, se $\sigma_1^2 = 0$, $\hat{y}(t+1) = y(t+1) \in \mathcal{Y}_t$. Mas $\hat{y}(t+2) = y(t+2) \in \mathcal{Y}_{t+1}$. Portanto, $\mathcal{Y}_{t+2} = \mathcal{Y}_t$ e $\sigma_2^2 = 0$. Seguindo este raciocínio conclui-se que se $\sigma_1^2 = 0$, $\sigma_m^2 = 0 \, \forall \, m = 2, 3, \ldots$.

Um processo estocástico $y(t)$ estacionário de segunda ordem e de média nula é decomposto de forma única como:

$$y(t) = u(t) + v(t) = \sum_{i=0}^{\infty} h(i)\epsilon(t-i) + v(t) \tag{2.36}$$

em que $v(t)$ é determinístico e descorrelacionado com a média móvel $u(t)$ que tem seus pesos $h(t)$ absolutamente somáveis, ou seja, a soma infinita dos valores absolutos de $h(t)$ é finita. Este é o

teorema de Wold[19].

Seja um processo $y(t)$ estacionário de média zero. A partir de sua função de densidade espectral $\Phi_{yy}(\omega)$ é possível determinar se ele é regular, ou seja, se ele é não determinístico. Supondo que a função $\log \Phi_{yy}(z)$ é definida na coroa $\rho < |z| < 1/\rho$, $0 < \rho < 1$ pode-se decompor esta função como uma soma de potências de z da seguinte forma:

$$\log \Phi_{yy}(z) = \sum_{l=-\infty}^{\infty} c_l z^{-l}, \quad \rho < |z| < 1/\rho \tag{2.37}$$

Como a função $\log \Phi_{yy}(\omega)$ é par, $c_{-l} = c_l$ no domínio $\rho < |z| < 1/\rho$ e a seguinte igualdade pode ser escrita:

$$\Phi_{yy}(z) = \exp\left(\sum_{l=-\infty}^{\infty} c_l z^{-l}\right) = e^{c_0} \exp\left(\sum_{l=-\infty}^{-1} c_l z^{-l}\right) \exp\left(\sum_{l=1}^{\infty} c_l z^{-l}\right)$$

O lado direito da equação 2.37 tem transformada inversa igual a uma soma de funções temporais em instantes l ponderadas por c_l. Ou seja, os termos c_l são os coeficientes da transformada inversa da função $\log \Phi_{yy}(z)$ portanto:

$$c_l = \frac{1}{2\pi} \int_{-\pi}^{\pi} e^{j\omega l} \log \Phi_{yy}(\omega)\, d\omega, \quad l \in \mathcal{Z} \tag{2.38}$$

O que para $l = 0$ dá a condição para a série ser não determinística que é:

$$c_0 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log \Phi_{yy}(\omega)\, d\omega > -\infty \tag{2.39}$$

uma vez que se $c_0 = -\infty$, $e^{c_0} = 0$ e $\Phi_{yy}(z) = 0$. Esta condição é conhecida como condição de regularidade de Szegö[20].

Definindo $H(z) = \exp\left(\sum_{i=0}^{\infty} h(i) z^{-i}\right), \quad |z| > \rho$, temos que a série deste expoente é analítica em $|z| > \rho$ e $H(\infty) = 1$. Portanto $H(z)$ pode ser expandido em série de Taylor, ou seja:

$$H(z) = \sum_{i=0}^{\infty} h(i) z^{-i}, \quad |z| > \rho, \quad h(0) = 1 \tag{2.40}$$

Além disso, a definição de $H(z)$ permite que se escreva a função de densidade espectral da seguinte forma:

$$\Phi_{yy}(z) = \sigma^2 H(z) H(z^{-1})$$

onde $\sigma^2 = e^{c_0}$.

[19]Herman Wold (1908-1992). Matemático norueguês naturalizado sueco. Graduou-se em matemática na Universidade de Estocolmo, tendo estudado com Harald Cramér. Fez seu doutorado em processos estocásticos sob a orientação de Cramér, tendo publicado em 1936 a tese *Um estudo na análise de séries temporais estacionárias*, em que enunciou a decomposição de Wold que é citada neste texto. Em 1942 se tornou professor de estatística da Universidade de Uppsala, onde ficou até 1970 quando se tornou professor da Universidade de Gotemburgo por onde se aposentou em 1975.

[20]Gábor Szegö (1895-1985). Matemático húngaro que se dedicou ao estudo das matrizes de Toeplitz e dos polinômios ortogonais, tendo publicado o livro *Polinômios Ortogonais* em 1939. Assim como Hilbert, também foi professor de János Von Neumman.

A série de potência da equação 2.40 converge para $|z| > \rho$ então $H(z)$ não tem pólos na região $|z| \geq 1$. Disto também segue que $|h(l)| \leq M\rho_1$ para qualquer $l \geq 0$ onde $M > 0$ e $0 < \rho_1 \leq \rho < 1$. Portanto:

$$\sum_{i=0}^{\infty} |h(l)| < \infty \Rightarrow \sum_{i=0}^{\infty} |h(l)|^2 < \infty$$

Além disso, $H(z)$ é tal que:

$$H(z)^{-1} = \exp\left(-\sum_{l=1}^{\infty} c_l z^{-l}\right)$$

que é analítica em $|z| > \rho$ e portanto $H(z)$ não tem zeros na região $|z| \geq 1$, portanto $H(z)$ é de fase mínima.

Como a densidade espectral do processo estocástico y é maior ou igual a zero, $\log \Phi_{yy}(\omega) \leq \Phi_{yy}(\omega)$. Portanto c_0 é menor que o coeficiente zero da série de Fourier de Φ_{yy}, ou seja:

$$c_o \leq \frac{1}{2\pi} \int_{-\pi}^{\pi} \Phi_{yy}(\omega)\, d\omega < \infty \tag{2.41}$$

Portanto há um limite superior para c_0. Se também há um limite inferior para c_o, ou seja, $c_0 < \infty$, o processo é não singular.

No entanto quando $c_0 > -\infty$ é possível que o processo tenha zeros sobre o círculo de raio unitário, de forma que a condição de regularidade de Szegö é mais fraca que o fato de a função $\log \Phi_{yy}(z)$ ser analítica conforme será demonstrado no exemplo a seguir. Seja $y(t)$ o seguinte processo de média móvel:

$$y(t) = \epsilon(t) - \epsilon(t-1) \tag{2.42}$$

em que ϵ é um ruído branco de média nula e variância $\sigma^2 = 1$.

O processo $y(t)$ pode ser visto como a saída de um filtro com função de transferência $H(z) = 1 - z^{-1}$ a uma entrada $\epsilon(t)$. Portanto a função de densidade espectral de $y(t)$ será:

$$\Phi_{yy}(z) = \sigma^2 H(z)H(z^{-1}) = 2 - (z + z^{-1}) \Rightarrow \Phi_{yy}(\omega) = 2 - 2\cos(\omega) \tag{2.43}$$

Para este processo, $\log \Phi_{yy}(z)|_{z=1} = -\infty$, portanto o logaritmo não é analítico sobre o círculo de raio unitário. No entanto, a integral deste logaritmo é maior que $-\infty$, o que satisfaz a condição de Szegö, conforme será demonstrado abaixo:

$$\int_{-\pi}^{\pi} \log \Phi_{yy}(\omega)\, d\omega = \int_{-\pi}^{\pi} \log(2 - 2\cos(\omega))\, d\omega = 0 > -\infty$$

2.7.2 Caso multivariável

Seja agora $y(k)$ um processo estocástico estacionário, multivariável com dimensão $n > 1$, de segunda ordem e média nula. O espaço $\mathcal{H}$ gerado por esse processo pode ser definido da seguinte maneira:

$$\mathcal{H} = \left\{ \xi = \sum_{k=k_1}^{k_2} a_k y(k) | a_k \in \Re \right\}, \quad -\infty < k_1 \leq k_2 < \infty$$

Ou seja, o espaço $\mathcal{H}$ é definido por todos os vetores ξ formados por combinações lineares de todos os elementos do processo estocástico vetorial $y(k)$ em todos instantes compreendidos entre k_1 e k_2.

Sejam dois vetores ξ e η contidos no espaço $\mathcal{H}$ definidos da seguinte maneira:

$$\xi = \sum_{k=k_1}^{k_2} a_k y(k) \quad \eta = \sum_{j=j_1}^{j_2} b_j y(j)$$

Neste espaço, define-se o produto interno entre dois vetores como sendo a somatória das covariâncias tomadas para cada termo do processo multivariável, ou seja, o produto interno entre dois vetores ξ e η definidos em $\mathcal{H}$ é:

$$< \xi, \eta >_{\mathcal{H}} = \sum_{i=1}^{n} E\left[\left(\sum_{k=k_1}^{k_2} a_k y_i(k) \right) \left(\sum_{j=j_1}^{j_2} b_j y_i(j) \right) \right] = \sum_{k,j \in D} a_k b_j \sum_{i=1}^{n} E[y_i(k) y_i(j)]$$

em que $y_i(k)$ é o i-ésimo termo do vetor $y(k)$, D é o domínio em que estão definidas as variáveis j e k, ou seja, $k_1 \leq k \leq k_2$ e $j_1 \leq j \leq j_2$, e $1 \leq i \leq n$. A expressão acima pode ser simplificada ao se tomar a matriz de covariância do processo estocástico multivariável $y(k)$ de média nula, conforme apresentado abaixo:

$$E[y(k)y(j)^T] = \begin{bmatrix} E[y_1(k)y_1(j)] & E[y_1(k)y_2(j)] & \dots & E[y_1(k)y_n(j)] \\ E[y_2(k)y_1(j)] & E[y_2(k)y_2(j)] & \dots & E[y_2(k)y_n(j)] \\ \vdots & \vdots & \ddots & \vdots \\ E[y_n(k)y_1(j)] & E[y_n(k)y_2(j)] & \dots & E[y_n(k)y_n(j)] \end{bmatrix}$$

portanto,

$$\sum_{i=1}^{n} E[y_i(k)y_i(j)] = tr(E[y(k)y(j)^T])$$

em que $tr(\bullet)$ é o traço da matriz $\bullet$, ou seja, a soma dos elementos de sua diagonal principal.

Desta forma, o produto interno em um espaço de Hilbert formado por processos estocásticos multivariáveis pode ser escrito da seguinte maneira:

$$< \xi, \eta >_{\mathcal{H}} = \sum_{k,j \in D} tr(E[y(k)y(j)^T]) a_k b_j$$

Consequentemente, a norma do vetor ξ é definida como:

$$||\xi||^2_{\mathcal{H}} = < \xi, \xi >_{\mathcal{H}} = \sum_{k=k_1}^{k_2} tr(E[y(k)y(k)^T]) a_k^2$$

Como a norma é o produto entre uma somatória de covariâncias e um termo quadrático, ela é sempre positiva e o espaço $\mathcal{H}$ é um espaço de Hilbert.

2.8 Predição de um processo estocástico

No exemplo a seguir é demonstrado como a predição de um processo estocástico pode ser feita levando em conta a análise espectral desenvolvida acima. Seja $y(t)$ um processo regular estacionário. Conforme comentado anteriormente, o processo $y(t)$ pode ser visto como a saída de um filtro que tem como entrada um ruído branco descorrelacionado $\epsilon(t)$, ou seja:

$$y(t) = \sum_{i=0}^{\infty} h(i)\epsilon(t-i) \tag{2.44}$$

Sendo que a função de transferência do filtro é:

$$H(z) = \sum_{i=0}^{\infty} h(i)z^{-i} \tag{2.45}$$

por hipótese este filtro é de fase mínima, ou seja, sua inversa é estável.

Como o processo $y(t)$ é formado por combinações lineares de ocorrências passadas do processo $\epsilon(t)$, o espaço $\mathcal{Y}_t$, formado pelos valores do processo $y(t)$ até o instante t, é igual ao espaço $\mathcal{E}_t$, formado pelo ruído branco até o instante t. Por este motivo, a previsão do valor do processo $y(t)$ no instante $t+m$, $m>0$, pode ser escrita tanto como uma combinação linear de valores passados de y, quanto de valores passados do processo ϵ, ou seja:

$$\hat{y}(t+m|t) = \hat{E}[y(t+m)|\mathcal{Y}_t] = \sum_{i=0}^{\infty} g_i y(t-i) \tag{2.46}$$

ou

$$\hat{y}(t+m|t) = \hat{E}[y(t+m)|\mathcal{E}_t] = \sum_{i=0}^{\infty} f_i \epsilon(t-i) \tag{2.47}$$

A partir das famílias de coeficientes g_i e f_i pode-se definir as seguintes funções de transferência:

$$G(z) = \sum_{i=0}^{\infty} g(i)z^{-i} \quad F(z) = \sum_{i=0}^{\infty} f(i)z^{-i} \tag{2.48}$$

Estes filtros têm uma relação entre si. Se o sinal $y(t)$ for dado como entrada para um filtro com função de transferência $1/H(z)$ (motivo pelo qual foi feita a hipótese de $H(z)$ ser de fase mínima.) na saída haverá o sinal $\epsilon(t)$. Se este sinal for dado como entrada a um filtro com função de transferência $F(z)$, o sinal de saída será $\hat{y}(t+m)$. Por outro lado, se o sinal $y(t)$ for dado como entrada a um filtro com função de transferência $G(z)$, a saída será $\hat{y}(t+m)$. Disto se conclui que:

$$G(z) = \frac{F(z)}{H(z)} \tag{2.49}$$

O parágrafo e a expressão acima estão ilustrados na figura 2.3.

Definindo $\overline{y}(t+m)$ como sendo o erro de predição, ou seja:

$$\overline{y}(t+m) = y(t+m) - \hat{y}(t+m) \tag{2.50}$$

tem-se que:

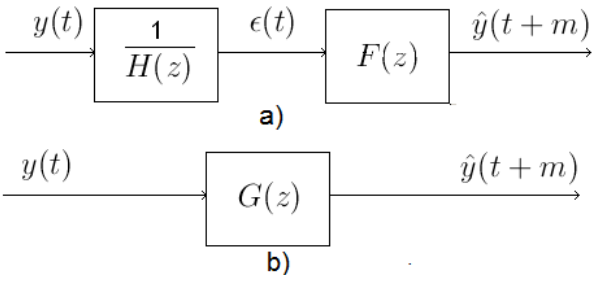

Fig. 2.3: Relação entre as funções de transferência $G(z)$, $F(z)$ e $H(z)$. a) o sinal $y(t)$ ao passar pelo filtro com função de transferência $\frac{1}{H(z)}$ produz o sinal $\epsilon(t)$, que ao passar pelo filtro com função de transferência $F(z)$ resulta no sinal $\hat{y}(t+m)$. b) o sinal $y(t)$ ao passar pelo filtro com função de transferência $G(z)$ resulta no sinal $\hat{y}(t+m)$.

$$\begin{aligned} \overline{y}(t+m) &= \textstyle\sum_{i=0}^{\infty} h(i)\epsilon(t+m-i) - \sum_{i=0}^{\infty} f(i)\epsilon(t-i) \\ &= \textstyle\sum_{i=0}^{m-1} h(i)\epsilon(t+m-i) + \sum_{i=0}^{\infty}[h(i+m) - f(i)]\epsilon(t-i) \end{aligned} \tag{2.51}$$

Ao se aplicar o valor esperado dos dois lados da equação acima fica claro que o valor esperado do erro de predição é nulo. A variância do erro de predição por sua vez é dada por:

$$E[\overline{y}^2(t+m)] = \sigma_e^2 \sum_{i=0}^{m-1} h^2(i) + \sigma_e^2 \sum_{i=0}^{\infty}[h(i+m) + f(i)]^2 \tag{2.52}$$

uma vez que o processo ϵ é um ruído branco descorrelacionado de variância σ_e^2.

Da equação acima nota-se que para minimizar a variância do erro deve-se fazer $f(i) = h(i+m)$, portanto o melhor preditor tem a seguinte forma:

$$\hat{y}(t+m|t) = \sum_{i=0}^{\infty} h(i+m)\epsilon(t-i) = \sum_{i=-\infty}^{t} h(t+m-i)\epsilon(i) \tag{2.53}$$

Também se tem que a função de transferência $F(z)$ é dada por:

$$F(z) = \sum_{i=0}^{\infty} h(i+m)z^{-i} = h(m) + h(m+1)z^{-1} + \ldots \tag{2.54}$$

mas ao se multiplicar $H(z)$ por z^m se tem o seguinte:

$$z^m H(z) = h(0)z^m + \ldots + h(m-1)z + h(m) + h(m+1)z^{-1} + \ldots \tag{2.55}$$

portanto $F(z)$ é igual à parte causal de $z^m H(z)$. Se $[*]_+$ denotar o isolamento da parte causal de uma expressão, então a função $G(z)$ que fornece a previsão com menor covariância de erro é dada por:

$$G(z) = \frac{[z^m H(z)]_+}{H(z)} \tag{2.56}$$

A variância mínima a que se chega usando o $G(z)$ ótimo é obtida ao se anular o segundo termo da equação 2.52, ou seja:

$$\sigma_m^2 = E[\overline{y}^2(t+m)] = \sigma_e^2 \sum_{i=0}^{m-1} h^2(i) \tag{2.57}$$

Como $y(t+m) = \hat{y}(t+m|t) + \overline{y}(t+m)$ com $\hat{y}(t+m|t)$ ortogonal a $\overline{y}(t+m)$ então:

$$\begin{aligned} \sigma_y^2 &= E[y(t+m)y(t+m)] = \\ &= E[(\hat{y}(t+m|t) + \overline{y}(t+m))(\hat{y}(t+m|t) + \overline{y}(t+m))] = \\ &= \sigma_{\hat{y}}^2 + \sigma_m^2 \Rightarrow \sigma_m^2 = \sigma_y^2 - \sigma_{\hat{y}}^2 \end{aligned} \tag{2.58}$$

em que $\sigma_y^2 = E[\hat{y}^2(t+m|t)]$. Como $Y(z) = H(z)\epsilon(z)$ e $\hat{Y}(z) = F(z)\epsilon(z)$ então a variância mínima pode ser escrita a partir da transformada inversa de Fourier como:

$$\sigma_m^2 = \frac{\sigma_e^2}{2\pi} \int_{-\pi}^{\pi} \left(|H(e^{j\omega})|^2 - |F(e^{j\omega})|^2 \right) d\omega \tag{2.59}$$

2.9 Sistemas estocásticos variantes no tempo

Seja um sistema linear variante no tempo sujeito apenas a ruídos e descrito pela seguinte equação de estado:

$$\begin{cases} x(t+1) = A(t)x(t) + w(t) \\ y(t) = C(t)x(t) + v(t) \end{cases} \tag{2.60}$$

Em que $x \in \Re^n$ é o vetor de estados, $y \in \Re^p$ é o vetor de saída, $w \in \Re^n$ é o ruído sobre a atualização dos estados, $v \in \Re^p$ é o ruído de observação e as matrizes $A(t) \in \Re^{nxn}$ e $C(t) \in \Re^{pxn}$ são funções determinísticas do tempo t. Os ruídos $v(t)$ e $w(t)$ são processos estocásticos brancos, gaussianos de média nula com as seguintes matrizes de covariância:

$$E\left[\begin{bmatrix} w(t) \\ v(t) \end{bmatrix} \begin{bmatrix} w(t)^T & v(t)^T \end{bmatrix} \right] = \begin{bmatrix} Q(t) & S(t) \\ S(t)^T & R(t) \end{bmatrix} \delta_{ts} \tag{2.61}$$

em que $Q(t) \in \Re^{nxn}$ é uma matriz não negativa definida e $R(t)$ é uma matriz positiva definida para qualquer t. O estado inicial x_0 por sua vez é gaussiano com média $\mu_x(0)$, com matriz de covariância $\Pi(0)$ e descorrelacionado com os ruídos $v(t)$ e $w(t)$.

A partir destas definições o interesse é o estudo das propriedades estatísticas dos processos x e y a partir das características dos outros processos envolvidos no problema. Para o estudo do processo x é interessante definir a matriz de transição de estados $\Phi(t,s)$ como sendo:

$$\Phi(t,s) = \begin{cases} \prod_{i=s}^{t-1} A(i) & s < t \\ I & s = t \end{cases} \tag{2.62}$$

Com esta definição a solução da primeira equação do sistema é:

$$\begin{aligned}
x(t) &= A(t-1)x(t-1) + w(t-1) = \\
&= A(t-1)[A(t-2)x(t-2) + w(t-2)] + w(t-1) = \\
&= A(t-1)[A(t-2)[A(t-3)x(t-3) + w(t-3)] + w(t-2)] + w(t-1) = \\
&= \ldots = \\
&= \textstyle\prod_{i=s}^{t-1} A(i)x(s) + w(t-1) + A(t-1)w(t-2) + A(t-1)A(t-2)w(t-3) + \\
&\quad + \ldots + A(t-1)A(t-2)\ldots A(s+1)w(s) = \\
&= \textstyle\prod_{i=s}^{t-1} A(i)x(s) + w(t-1) + \prod_{i=t-1}^{t-1} A(i)w(t-2) + \prod_{i=t-2}^{t-1} A(i)w(t-3) + \\
&\quad + \ldots + \textstyle\prod_{i=s+1}^{t-1} A(i)w(s)
\end{aligned} \tag{2.63}$$

usando a definição de $\Phi(t,s)$ tem-se:

$$\begin{aligned}
x(t) &= \Phi(t,s)x(s) + \Phi(t,t)w(t-1) + \Phi(t,t-1)w(t-2) + \Phi(t,t-2)w(t-3) + \\
&\quad + \ldots + \Phi(t,s+1)w(s) = \\
&= \Phi(t,s)x(s) + \textstyle\sum_{k=s}^{t-1} \Phi(t,k+1)w(k)
\end{aligned} \tag{2.64}$$

Se o estado inicial é x_0, ou seja, $s = 0$, então:

$$x(t) = \Phi(t,0)x(0) + \sum_{k=0}^{t-1} \Phi(t,k+1)w(k) \tag{2.65}$$

ou seja, o vetor $x(t)$ é uma combinação linear de um processo gaussiano x_0 e do ruído w nos instantes de tempo de 0 a $t-1$, portanto $x(t)$ também é gaussiano. O processo $x(t)$ também é markoviano, uma vez que ele é combinação linear de processos estocásticos descorrelacionados (x_0 e w), implicando que:

$$p(x(t+k)|x(t), x(t-1), \ldots, x(0)) = p(x(t+k)|x(t))$$

comprovando que $x(t)$ é de fato markoviano. Como um processo markoviano e gaussiano, $x(t)$ é completamente caracterizado pelo seu vetor de média:

$$\mu_x(t) = E[x(t)] = E\left[\Phi(t,0)x(0) + \sum_{k=0}^{t-1} \Phi(t,k+1)w(k)\right] = \Phi(t,0)\mu_x(0) \tag{2.66}$$

e por sua matriz de covariância:

$$
\begin{aligned}
\Lambda_{xx}(t,s) &= E[(x(t)-\mu_x(t))(x(s)-\mu_x(s))^T] = \\
&= E\left[\left(\left(\Phi(t,0)x(0)+\textstyle\sum_{l=0}^{t-1}\Phi(t,l+1)w(l)\right)-\mu_x(t)\right)* \right. \\
&\quad \left. *\left(\left(\Phi(s,0)x(0)+\textstyle\sum_{k=0}^{s-1}\Phi(s,k+1)w(k)\right)-\mu_x(s)\right)^T\right] = \\
&= E\left[\left(\Phi(t,0)(x(0)-\mu_x(0))+\textstyle\sum_{l=0}^{t-1}\Phi(t,l+1)w(k)\right)* \right. \\
&\quad \left. *\left(\Phi(s,0)(x(0)-\mu_x(0))+\textstyle\sum_{k=0}^{s-1}\Phi(s,k+1)w(k)\right)^T\right] = \\
&= \Phi(t,0)E[(x(0)-\mu_x(0))(x(0)-\mu_x(0))^T]\Phi^T(s,0)+ \\
&\quad +\Phi(t,0)E\left[(x(0)-\mu_x(0))(\textstyle\sum_{k=0}^{s-1}\Phi(s,k+1)w(k))^T\right]+ \\
&\quad +E[(\textstyle\sum_{l=0}^{t-1}\Phi(t,l+1)w(l))(x(0)-\mu_x(0))^T]\Phi^T(s,0)+ \\
&\quad +\textstyle\sum_{l=0}^{t-1}\Phi(t,l+1)E[w(l)w(k)^T]\sum_{k=0}^{s-1}\Phi^T(s,k+1) = \\
&= \Phi(t,0)\Pi(0)\Phi^T(s,0)+\textstyle\sum_{k=0}^{s-1}\Phi(t,k+1)Q(k)\Phi^T(s,k+1)
\end{aligned}
\tag{2.67}
$$

Na última passagem do cálculo abaixo, o segundo e o terceiro termos são nulos devido à descorrelação entre $x(0)$ e $w(t)$ e também devido ao fato de $w(t)$ ter média nula. No último termo, o produto de somatórias se torna apenas uma somatória pois o ruído w é descorrelacionado em instantes de tempo diferentes. Portanto, se $s < t$, a somatória vai até o instante $s-1$.

A matriz de variância $\Pi(t)$ do processo x é obtida simplesmente ao se substituir s por t na equação acima, ou seja:

$$
\Pi(t) = \Lambda_{xx}(t,t) = \Phi(t,0)\Pi(0)\Phi^T(t,0) + \sum_{k=0}^{t-1}\Phi(t,k+1)Q(k)\Phi^T(t,k+1) \tag{2.68}
$$

Esta matriz de variância também pode ser calculada recursivamente conforme demonstrado a seguir:

$$\begin{aligned}
\Pi(1) &= \Phi(1,0)\Pi(0)\Phi^T(1,0) + \Phi(1,1)Q(0)\Phi^T(1,1) \\
&= A(0)\Pi(0)A^T(0) + Q(0) \\
\Pi(2) &= \Phi(2,0)\Pi(0)\Phi^T(2,0) + \Phi(2,1)Q(0)\Phi^T(2,1) + \Phi(2,2)Q(1)\Phi^T(2,2) = \\
&= A(1)A(0)\Pi(0)A^T(0)A^T(1) + A(1)Q(0)A^T(1) + Q(1) = \\
&= A(1)[A(0)\Pi(0)A^T(0) + Q(0)]A^T(1) + Q(1) = \\
&= A(1)\Pi(1)A^T(1) + Q(1) \\
\vdots &\quad \vdots \\
\Pi(t+1) &= A(t)\Pi(t)A^T(t) + Q(t)
\end{aligned} \tag{2.69}$$

A equação 2.69 é conhecida como equação de Lyapunov[21] e tem sua solução explorada na referência [56] por meio de redes neurais e nos artigos [39] e [40] e no capítulo 7 desta tese por meio de algoritmos imuno-inspirados.

O processo $y(t)$ também é gaussiano pelos mesmos motivos que o processo $x(t)$ o é. e Sua média é a seguinte:

$$\mu_y(t) = E[y(t)] = E[C(t)x(t) + v(t)] = C(t)\mu_x(t) = C(t)\Phi(t,0)\mu_x(0) \tag{2.70}$$

O cálculo de sua variância é feito em duas partes. Primeiro encontra-se o valor de $y(t) - \mu_y(t)$ em termos dos processos conhecidos:

$$\begin{aligned}
y(t) - \mu_y(t) &= C(t)x(t) + v(t) - C(t)\Phi(t,s)\mu_x(s) \\
&= C(t)\left[\Phi(t,s)x(s) + \textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)w(k)\right] + v(t) - C(t)\Phi(t,s)\mu_x(s) = \\
&= C(t)\Phi(t,s)[x(s) - \mu_x(s)] + C(t)\textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)w(k) + v(t)
\end{aligned} \tag{2.71}$$

Com isto tem-se que:

[21] Aleksandr Mikhailovich Lyapunov (1857-1918). Matemático e físico russo, aluno de Chebyshev. Graduado pela Universidade de São Petesburgo dois anos após Markov, especializou-se nas áreas de hidrostática, teoria da probabilidade e sistemas dinâmicos. Foi professor da Universidade da Carcóvia e posteriormente da Universidade de São Petesburgo, ocupando a cadeira de Chebyshev. Se suicidou em Odessa, após a morte de sua esposa por tuberculose.

$$
\begin{aligned}
\Lambda_{yy}(t,s) &= E[(y(t)-\mu_y(t))(y(s)-\mu_y(s))^T] = \\
&= E\left[\Big(C(t)\Phi(t,s)[x(s)-\mu_x(s)] + C(t)\textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)w(k)+v(t)\Big) * \right. \\
&\quad \left. * \left(C(s)x(s)+v(s)-C(s)\mu_x(s)\right)^T\right] \\
&= E\left[\Big(C(t)\Phi(t,s)[x(s)-\mu_x(s)] + C(t)\textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)w(k)+v(t)\Big) * \right. \\
&\quad \left. * \left(C(s)(x(s)-\mu_x(s))+v(s)\right)^T\right] = \\
&= C(t)\Phi(t,s)E[(x(s)-\mu_x(s))(x(s)-\mu_x(s))^T]C^T(s) + \\
&\quad + C(t)\Phi(t,s)E[(x(s)-\mu_x(s))v(s)^T] + \\
&\quad + C(t)\textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)E[w(k)(x(s)-\mu_x(s))^T]C^T(s) + \\
&\quad + C(t)\textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)E[w(k)v(s)^T] + \\
&\quad + E[v(t)(x(s)-\mu_x(s))^T]C^T(s) + E[v(t)v(s)^T] = \\
&= C(t)\Phi(t,s)\Pi(s)C^T(s) + \\
&\quad + C(t)\textstyle\sum_{k=s}^{t-1}\Phi(t,k+1)E[w(k)v(s)^T] + \\
&\quad + E[v(t)v(s)^T]
\end{aligned}
\tag{2.72}
$$

Se $s < t$, a somatória do segundo termo da última equação acima tem um termo não nulo quando $k = s$ e este termo é $\Phi(t, s+1)S(s)$, enquanto o terceiro termo se anula. Se $s = t$ o segundo termo se anula e o terceiro é a variância do processo $v(t)$. É possível mostrar também que se $s > t$, a matriz de covariância é a transposta de quando $s < t$. Portanto, tem-se finalmente que:

$$
\Lambda_{yy}(t,s) = \begin{cases} C(t)\Phi(t,s)\Pi(s)C^T(s) + C(t)\Phi(t,s+1)S(s) & t > s \\ C(t)\Pi(t)C^T(t) + R(t) & t = s \\ \Lambda_{yy}^T(s,t) & t < s \end{cases} \tag{2.73}
$$

2.10 Sistemas estocásticos invariantes no tempo

Um sistema linear estocástico invariante no tempo é semelhante aos variantes no tempo a menos que as matrizes envolvidas ($A(t)$, $C(t)$, $Q(t)$, $R(t)$ e $S(t)$) não dependem do tempo t, ou seja, $A(t) = A$, $C(t) = C$, $Q(t) = Q$, $R(t) = R$ e $S(t) = S$.

Por analogia ao estudado no caso variante no tempo se tem que, para um sistema linear estocástico invariante no tempo representado por:

$$\begin{cases} x(t+1) = Ax(t) + w(t) \\ y(t) = Cx(t) + v(t) \end{cases} \tag{2.74}$$

com instante inicial igual a t_0, com $x(t_0)$ gaussiano com média $\mu_x(t_0)$ e matriz de covariância $\Pi(t_0)$, a matriz de transição de estado será:

$$\Phi(t,s) = A^{t-s} \tag{2.75}$$

se $t \leq s$. O processo x terá média:

$$\mu_x(t) = A^{t-t_0}\mu_x(t_0) \tag{2.76}$$

e a seguinte matriz de covariância:

$$\begin{aligned} \Pi(t) &= A^{t-t_0}\Pi(t_0)(A^T)^{t-t_0} + \textstyle\sum_{k=t_0}^{t-1} A^{t-k-1}Q(A^T)^{t-k-1} \\ &= A^{t-t_0}\Pi(t_0)(A^T)^{t-t_0} + \textstyle\sum_{k=0}^{t-t_0-1} A^{k}Q(A^T)^{k} \end{aligned} \tag{2.77}$$

e esta matriz satisfaz a equação de Lyapunov, ou seja:

$$\Pi(t+1) = A\Pi(t)A^T + Q \tag{2.78}$$

Se a matriz A for estável, matriz de covariância $\Pi(t)$ tende a um valor constante Π conforme será demonstrado a seguir:

Se A é estável então:

$$\lim_{t_0 \to -\infty} A^{t-t_0} = 0 \tag{2.79}$$

portanto:

$$\lim_{t_0 \to -\infty} \mu_x(t) = 0 \tag{2.80}$$

Com isto o limite de $\Pi(t)$ quando $t_0 \to -\infty$[22] é:

$$\lim_{t_0 \to -\infty} \left[A^{t-t_0}\Pi(t_0)(A^T)^{t-t_0} + \sum_{k=0}^{t-t_0-1} A^k Q(A^T)^k \right] = \sum_{k=0}^{\infty} A^k Q(A^T)^k = \Pi \tag{2.81}$$

Ou seja, $\Pi(t)$ não depende mais de t, portanto, pode ser definido como Π. Desta forma, a matriz de covariância $\Lambda(t+l,t)$ quando $t_0 \to -\infty$ tem um valor que depende apenas do intervalo l, uma vez que, a partir da equação 2.77:

$$\Lambda_{xx}(t,t) = A^{t-t_0}\Pi(t_0)(A^T)^{t-t_0} + \sum_{k=t_0}^{t-1} A^{t-k-1}Q(A^T)^{t-k-1} \tag{2.82}$$

[22]Tomar o limite quando $t_0 \to -\infty$ significa que o sistema está sendo analisado em seu regime, ou seja, o comportamento transitório do início de seu funcionamento não tem mais importância pois ele ocorreu há um tempo grande o suficiente.

e

$$\begin{aligned}\Lambda_{xx}(t+l,t) &= A^{t+l-t_0}\Pi(t_0)(A^T)^{t-t_0} + \textstyle\sum_{k=t_0}^{t-1} A^{t+l-k-1}Q(A^T)^{t-k-1} = \\ &= A^l\left[A^{t-t_0}\Pi(t_0)(A^T)^{t-t_0} + \textstyle\sum_{k=t_0}^{t-1} A^{t-k-1}Q(A^T)^{t-k-1}\right] = \\ &= A^l\Lambda_{xx}(t,t)\end{aligned} \tag{2.83}$$

Tomando o limite quando $t_0 \to -\infty$, encontra-se que $\Lambda_{xx}(t+l,t) = \Lambda_{xx}(l) = A^l\Pi$. Além disto, se $l < 0$, ou seja, se o intervalo entre as duas medidas é negativo, o segundo termo do argumento de Λ_{xx} deve ser maior que o primeiro termo. Como l é negativo, este valor deve ser subtraído de t no segundo termo para que o intervalo entre os dois instantes de tempo do argumento de Λ_{xx} seja negativo. Portanto:

$$\begin{aligned}\Lambda_{xx}(t,t-l) &= A^{t-t_0}\Pi(t_0)(A^T)^{t-l-t_0} + \textstyle\sum_{k=t_0}^{t-1} A^{t-k-1}Q(A^T)^{t-l-k-1} \\ &= \left[A^{t-t_0}\Pi(t_0)(A^T)^{t-t_0} + \textstyle\sum_{k=t_0}^{t-1} A^{t-k-1}Q(A^T)^{t-k-1}\right](A^T)^{-l} = \\ &= \Lambda_{xx}(t,t)(A^T)^{-l}\end{aligned} \tag{2.84}$$

Ao se tomar o limite quando $t_0 \to -\infty$ encontra-se que $\Lambda_{xx}(t,t-l) = \Lambda_{xx}(-l) = \Pi(A^T)^{-l}$.

Em um sistema linear estocástico invariante no tempo em que a matriz A é estável, o processo y será gaussiano e, se $t_0 \to -\infty$, sua média será nula, conforme demonstrado abaixo:

$$E[y(t)] = E[Cx(t) + v(t)] = C\mu_x(t) + E[v(t)] = 0 \tag{2.85}$$

A matriz de covariância $\Lambda_{yy}(l)$ do processo y quando $t_0 \to -\infty$, obtida por analogia com a equação 2.73, é a seguinte:

$$\Lambda_{yy}(l) = \begin{cases} CA^l\Pi C^T + CA^{l-1}S & t > s \\ C\Pi C^T + R & t = s \\ C\Pi(A^T)^{-l}C^T + S^T(A^T)^{-l-1}C^T & t < s \end{cases} \tag{2.86}$$

Definindo $M = C\Pi A^T + S^T$ tem-se que:

$$\Lambda_{yy}(l) = \begin{cases} CA^{l-1}M^T & t > s \\ C\Pi C^T + R & t = s \\ M(A^T)^{-l-1}C^T & t < s \end{cases} \tag{2.87}$$

2.10.1 Relação entre matrizes e correlações

É possível fazer estimativas sobre as matrizes do sistema a partir do estudo da correlação entre os sinais do mesmo. Os métodos de realização de séries temporais detalhados no capítulo 5 partem de uma análise semelhante a que é feita a seguir.

A matriz A, do sistema descrito na equação 2.74 pode ser determinada da seguinte forma:

$$\begin{aligned} & x(t+1) = Ax(t) + w(t) \Rightarrow \\ \Rightarrow \quad & E[x(t+1)x^T(t)] = E[Ax(t)x^T(t) + w(t)x^T(t)] \Rightarrow \\ \Rightarrow \quad & E[x(t+1)x^T(t)] = A\Pi \Rightarrow A = E[x(t+1)x^T(t)]\Pi^{-1} \end{aligned} \tag{2.88}$$

Analogamente, a matriz C é dada por:

$$\begin{aligned} & y(t) = Cx(t) + v(t) \Rightarrow \\ \Rightarrow \quad & E[y(t)x^T(t)] = E[Cx(t)x^T(t) + v(t)x^T(t)] \Rightarrow \\ \Rightarrow \quad & E[y(t)x^T(t)] = C\Pi \Rightarrow C = E[y(t)x^T(t)]\Pi^{-1} \end{aligned} \tag{2.89}$$

e a matriz M também pode ser calculada de forma semelhante:

$$\begin{aligned} & y(t) = Cx(t) + v(t) \Rightarrow \\ \Rightarrow \quad & E[y(t)x^T(t+1)] = E[(Cx(t)x^T(t+1) + v(t))x^T(t+1)] \Rightarrow \\ \Rightarrow \quad & E[y(t)x^T(t+1)] = CE[x(t)\left(Ax(t)\right)^T] + E[v(t)w(t)^T] = \\ & = C\Pi A^T + S^T = M \end{aligned} \tag{2.90}$$

2.10.2 Densidade espectral

Seja um sistema em que a matriz $S = 0$, ou seja, os ruídos w e v são independentes. Para este sistema a matriz de covariância de y será a seguinte:

$$\begin{aligned} \Lambda_{yy}(l) &= \begin{cases} CA^l\Pi C^T & t > s \\ C\Pi C^T + R & t = s \\ C\Pi(A^T)^{-l}C^T & t < s \end{cases} = \begin{cases} C\Lambda_{xx}(l)C^T & l \neq 0 \\ C\Lambda_{xx}(0)C^T + R & l = 0 \end{cases} \\ &= C\Lambda_{xx}(l)C^T + R\delta(l) \end{aligned} \tag{2.91}$$

A partir da matriz de covariância pode-se calcular a densidade espectral do processo y que é dada por:

$$\Phi_{yy}(z) = \mathcal{F}\left[\Lambda_{yy}(l)\right] = C\Phi_{xx}(z)C^T + R \tag{2.92}$$

em que Φ_{xx} é a matriz de densidade espectral do processo x. Esta matriz por sua vez é calculada da seguinte forma:

$$
\begin{aligned}
\Phi_{xx}(z) &= \mathcal{F}\left[\Lambda_{xx}(l)\right] = \textstyle\sum_{l=-\infty}^{\infty} \Lambda_{xx}(l) z^{-l} = \\
&= \textstyle\sum_{l=-\infty}^{-1} \Pi (A^t)^{-l} z^{-l} + \Pi + \sum_{i=1}^{\infty} A^k \Pi z^{-l} = \\
&= \Pi + \Pi \left(\textstyle\sum_{l=1}^{\infty} z^l (A^T)^l \right) + \left(\sum_{i=1}^{\infty} z^{-l} A^l \right) \Pi = \\
&= \Pi + \Pi \left(\textstyle\sum_{l=0}^{\infty} z^l (A^T)^l - I \right) + \left(\sum_{i=0}^{\infty} z^{-l} A^l - I \right) \Pi = \\
&= \Pi + \Pi \left((I - zA^T)^{-1} - I \right) + \left((I - z^{-1}A)^{-1} - I \right) \Pi = \\
&= \Pi + \Pi \left((I - (I - zA^T))(I - zA^T)^{-1} \right) + \left((I - z^{-1}A)^{-1} (I - (I - z^{-1}A)) \right) \Pi = \\
&= \Pi + \Pi z A^T (I - zA^T)^{-1} + (I - z^{-1}A)^{-1} z^{-1} A \Pi = \\
&= \Pi + \Pi A^T (z^{-1} I - A^T)^{-1} + (zI - A)^{-1} A \Pi = \\
&= (zI - A)^{-1}(zI - A) \left[\Pi + \Pi A^T (z^{-1} I - A^T)^{-1} + (zI - A)^{-1} A \Pi \right] * \\
&\quad *(z^{-1} I - A^T)(z^{-1} I - A^T)^{-1} = \\
&= (zI - A)^{-1} \left[(zI - A)\Pi(z^{-1} I - A^T) + (zI - A)\Pi A^T + A\Pi(z^{-1} I - A^T) \right] * \\
&\quad *(z^{-1} I - A^T)^{-1} = \\
&= (zI - A)^{-1} \left[\Pi - z\Pi A^T - z^{-1} A\Pi + A\Pi A^T + z\Pi A^T - A\Pi A^T + \right. \\
&\quad \left. + z^{-1} A\Pi - A\Pi A^T \right] (z^{-1} I - A^T)^{-1} = \\
&= (zI - A)^{-1} \left[\Pi - A\Pi A^T \right] (z^{-1} I - A^T)^{-1} \\
&= (zI - A)^{-1} Q (z^{-1} I - A^T)^{-1}
\end{aligned}
\tag{2.93}
$$

Portanto a matriz de densidade espectral do processo y será:

$$\Phi_{yy}(z) = C(zI - A)^{-1} Q (z^{-1} I - A^T)^{-1} C^T + R \tag{2.94}$$

Definindo $W(z) = C(zI - A)^{-1}$ temos que:

$$\Phi_{yy}(z) = W(z) Q W^T(z^{-1}) + R \tag{2.95}$$

No caso em que há correlação entre os sinais de ruído v e w, ou seja, quando $S \neq 0$, a matriz de densidade espectral de y é calculada a partir da equação 2.87 da seguinte forma:

$$
\begin{aligned}
\Phi_{yy}(z) &= \mathcal{F}[\Lambda_{yy}(z)] = \textstyle\sum_{-\infty}^{\infty} \Lambda_{yy}(l) z^{-l} = \\
&= \textstyle\sum_{l=-\infty}^{-1} M(A^T)^{-l-1} C^T z^{-l} + C\Pi C^T + R + \sum_{l=1}^{\infty} C A^{l-1} M^T z^{-l} = \\
&= M\left[\textstyle\sum_{l=1}^{\infty} (A^T)^{l-1} z^l\right] C^T + C\Pi C^T + R + C\left[\textstyle\sum_{l=1}^{\infty} A^{l-1} z^{-l}\right] M^T = \\
&= M\left[\textstyle\sum_{l=0}^{\infty} (A^T)^{l-1} z^l - (A^T)^{-1}\right] C^T + C\Pi C^T + R + C\left[\textstyle\sum_{l=0}^{\infty} A^{l-1} z^{-l} - A^{-1}\right] M^T = \\
&= M(A^T)^{-1}\left[\textstyle\sum_{l=0}^{\infty} (A^T z)^l - I\right] C^T + C\Pi C^T + R + C\left[\textstyle\sum_{l=0}^{\infty} (A z^{-1})^l - I\right] A^{-1} M^T = \\
&= M(A^T)^{-1}\left[(I - A^T z)^{-1} - I\right] C^T + C\Pi C^T + R + C\left[(I - A z^{-1})^{-1} - I\right] A^{-1} M^T = \\
&= M(A^T)^{-1}\left\{[I - (I - A^T z)](I - A^T z)^{-1}\right\} C^T + C\Pi C^T + R + \\
&\quad + C\left\{(I - A z^{-1})^{-1}[I - (I - A z^{-1})]\right\} A^{-1} M^T \\
&= M(A^T)^{-1} A^T z (I - A^T z)^{-1} C^T + C\Pi C^T + R + C(I - A z^{-1})^{-1} A z^{-1} A^{-1} M^T \\
&= M(z^{-1} I - A^T)^{-1} C^T + C\Pi C^T + R + C(zI - A)^{-1} M^T = \\
&= (C\Pi A^T + S^T)(z^{-1} I - A^T)^{-1} C^T + C\Pi C^T + R + C(zI - A)^{-1}(A\Pi C^T + S) = \\
&= C\Pi A^T (z^{-1} I - A^T)^{-1} C^T + S^T (z^{-1} I - A^T)^{-1} C^T + C\Pi C^T + R + \\
&\quad + C(zI - A)^{-1} A\Pi C^T + C(zI - A)^{-1} S = \\
&= C\Pi A^T (z^{-1} I - A^T)^{-1} C^T + S^T W^T(z^{-1}) + C\Pi C^T + R + \\
&\quad + C(zI - A)^{-1} A\Pi C^T + W(z) S = \\
&= R + W(z) S + S^T W^T(z^{-1}) + \\
&\quad + C\Pi A^T (z^{-1} I - A^T)^{-1} C^T + C\Pi C^T + C(zI - A)^{-1} A\Pi C^T = \\
&= R + W(z) S + S^T W^T(z^{-1}) + \\
&\quad + C(zI - A)^{-1}(zI - A) C^{-1} * \\
&\quad * \left[C\Pi A^T (z^{-1} I - A^T)^{-1} C^T + C\Pi C^T + C(zI - A)^{-1} A\Pi C^T\right] * \\
&\quad * (C^T)^{-1}(z^{-1} I - A^T)(z^{-1} I - A^T)^{-1} C^T =
\end{aligned}
$$

continuação:

$$
\begin{aligned}
\Phi_{yy}(z) &= R + W(z)S + S^T W^T(z^{-1}) + \\
&\quad + C(zI - A)^{-1} * \\
&\quad * \left[(zI - A)\Pi A^T + (zI - A)\Pi(z^{-1}I - A^T) + A\Pi(z^{-1}I - A^T) \right] * \\
&\quad * (z^{-1}I - A^T)^{-1} C^T \\
&= R + W(z)S + S^T W^T(z^{-1}) + \\
&\quad + W(z) * \\
&\quad * \left[z\Pi A^T - A\Pi A^T + \Pi - z\Pi A^T - z^{-1}A\Pi + A\Pi A^T + z^{-1}A\Pi - A\Pi A^T \right] * \\
&\quad * W^T(z^{-1}) = \\
&= R + W(z)S + S^T W^T(z^{-1}) + W(z)(\Pi - A\Pi A^T) W^T(z^{-1}) = \\
&= R + W(z)S + S^T W^T(z^{-1}) + W(z) Q W^T(z^{-1})
\end{aligned}
\tag{2.96}
$$

Para a validade das equações 2.93 e 2.96 é suposto que as somatórias envolvidas convergem, o que de fato ocorre na coroa $\rho(A) < |z| < \rho^{-1}(A)$ se a matriz A for estável, ou seja, se todos os autovalores $\rho(A)$ estiverem dentro do círculo de raio unitário, que é a hipótese inicial. De fato, a somatória $\sum_{l=0}^{\infty} Z^{-l} A^l$ converge para $|z| > \rho(A)$ e a somatória $\sum_{l=0}^{\infty} Z^l (A^T)^l$ converge para $|z| < \rho(A)^{-1}$.

Capítulo 3

Identificação de sistemas no espaço de estado

Neste capítulo são estudadas as técnicas de identificação de sistemas discretos, determinísticos e multivariáveis no espaço de estado, que sem perda de generalidade também podem ser usadas para sistemas determinísticos monovariáveis. A inclusão deste capítulo se deve a dois motivos principais. O primeiro é para ilustrar a semelhança entre a identificação de sistemas determinísticos e a realização de séries temporais, que é o problema tratado no capítulo 5. O segundo motivo é o embasamento de uma das contribuições desta tese, que é a identificação de sistemas multivariáveis variantes no tempo, apresentada no capítulo 7.

As referências principais para este estudo são as teses de doutorado [5], [13], [56] e a dissertação de mestrado [15].

3.1 Aspectos básicos

Nesta seção será explorado o conceito de resposta ao impulso. Embora seja um conceito amplamente conhecido no caso monovariável, este conceito é notavelmente peculiar no caso multivariável. Sendo assim, nesta seção será explorado o conceito de impulso discreto monovariável e, a partir da observação da essência deste conceito, será definida a resposta ao impulso de sistemas multivariáveis.

3.1.1 Resposta ao impulso discreto monovariável

Um sistema discreto, linear, causal, monovariável e invariante no tempo pode ser caracterizado por sua equação a diferenças. Esta equação relaciona a saída do sistema no instante de tempo presente com os valores da entrada e até mesmo da saída em instantes de tempo anteriores, Este conceito é amplamente conhecido e para se ter mais detalhes recomenda-se o estudo das referências [1], [10], [34] e das seções finais do capítulo 4 da dissertação [36], em que são exploradas as várias formas de representação de sistemas monovariáveis.

Se a natureza física do sistema é conhecida, a equação a diferenças pode ser estimada a partir das constantes envolvidas no processo, que em geral representam inércias, constantes de amortecimento, contantes de mola ou combinações destas constantes. Por outro lado, caso essas constantes não sejam conhecidas, pode-se fazer uma estimativa de seus valores ao se observar a resposta do sistema a alguma entrada que torne a relação entre a saída observada e a característica do sistema o mais fácil

possível. Desta forma, ao se observar a resposta a esta entrada, se conhecerá uma equação que modela o comportamento do sistema para qualquer outra entrada.

Sem perda de generalidade, seja um sistema monovariável em que a saída no instante k, denotada por $y(k)$, seja resultado da combinação linear de infinitas entradas $u(k) \dots u(k-\infty)$, ponderadas por infinitas constantes $g(0) \dots g(\infty)$, da seguinte forma:

$$y(k) = \sum_{i=0}^{\infty} g(i)u(k-i) \tag{3.1}$$

Para se conhecer as constantes $g(0) \dots g(\infty)$ diretamente, basta aplicar uma entrada tal que $u(k) = 1$ no instante de tempo 0 e $u(k) = 0$ nos instantes de tempo seguintes, ou seja:

$$u(k) = \begin{cases} 1 & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{3.2}$$

Desta forma se terá:

$$\begin{array}{c} y(0) = g(0) \\ y(1) = g(1) \\ \vdots \\ y(\infty) = g(\infty) \end{array} \tag{3.3}$$

À seqüência de entrada $u(k)$, $k = 0 \dots \infty$, definida na equação 3.2 é dado o nome de impulso discreto monovariável e à seqüência $g(k)$, $k = 0 \dots \infty$, é dado o nome de resposta ao impulso. Deve-se notar que a seqüência de resposta ao impulso é, em sua essência, uma seqüência de constantes da combinação linear da entrada do sistema em instantes de tempo do presente e do passado que resultará na saída do sistema no instante de tempo presente. Sendo assim, se a resposta ao impulso for conhecida, pode-se conhecer a saída do sistema para qualquer seqüência de entrada, permitindo que se realize o sistema de forma algébrica.

3.1.2 Resposta ao impulso discreto multivariável

Seja agora um sistema discreto linear causal multivariável e invariante no tempo com m entradas e l saídas. Da mesma forma que no caso monovariável, espera-se que haja uma seqüência de entes matemáticos que relacionem um vetor de saída no instante k com as entradas naquele instante e no passado. No entanto, os entes não serão mais escalares, mas sim matrizes de dimensão lXm, denotadas por $G(k)$, $k = 0 \dots \infty$. Com isto, se tem a seguinte representação:

$$y(k) = \sum_{i=0}^{\infty} G(i)u(k-i) \tag{3.4}$$

Ou seja, no instante 0 a saída será:

$$\begin{bmatrix} y_1(0) \\ y_2(0) \\ \vdots \\ y_l(0) \end{bmatrix} = \begin{bmatrix} g_{11}(0) & g_{12}(0) & \dots & g_{1m}(0) \\ g_{21}(0) & g_{22}(0) & \dots & g_{2m}(0) \\ \vdots & \vdots & \ddots & \vdots \\ g_{l1}(0) & g_{l2}(0) & \dots & g_{lm}(0) \end{bmatrix} \begin{bmatrix} u_1(0) \\ u_2(0) \\ \vdots \\ u_m(0) \end{bmatrix} \tag{3.5}$$

No instante 1 a saída será:

$$\begin{bmatrix} y_1(1) \\ y_2(1) \\ \vdots \\ y_l(1) \end{bmatrix} = \begin{bmatrix} g_{11}(0) & g_{12}(0) & \dots & g_{1m}(0) \\ g_{21}(0) & g_{22}(0) & \dots & g_{2m}(0) \\ \vdots & \vdots & \ddots & \vdots \\ g_{l1}(0) & g_{l2}(0) & \dots & g_{lm}(0) \end{bmatrix} \begin{bmatrix} u_1(1) \\ u_2(1) \\ \vdots \\ u_m(1) \end{bmatrix} + \\ + \begin{bmatrix} g_{11}(1) & g_{12}(1) & \dots & g_{1m}(1) \\ g_{21}(1) & g_{22}(1) & \dots & g_{2m}(1) \\ \vdots & \vdots & \ddots & \vdots \\ g_{l1}(1) & g_{l2}(1) & \dots & g_{lm}(1) \end{bmatrix} \begin{bmatrix} u_1(0) \\ u_2(0) \\ \vdots \\ u_m(0) \end{bmatrix} \tag{3.6}$$

e assim por diante.

Desta forma, se todos os elementos $g_{ij}(k)$, $i = 1 \dots l$, $j = 1 \dots m$, $k = 0 \dots \infty$, forem determinados, será possível realizar o sistema multivariável. Para conhecer estes elementos deve-se fazer m experimentos sendo que, em cada um deles, se observará a resposta do sistema a uma das entradas. Para isto, no j-ésimo experimento deve-se aplicar uma entrada que, no instante $k = 0$, tem a j-ésima entrada igual a 1 e as demais iguais a zero e nos instantes de tempo seguintes, todas as entradas serão nulas. Seja, por exemplo, o primeiro experimento. Nele se aplicará a seguinte entrada:

$$u(k) = \begin{cases} [\ 1 \quad 0 \quad \dots \quad 0\ ]^T & k = 0 \\ \mathbf{0} & k \neq 0 \end{cases} \tag{3.7}$$

em que **0** é a matriz nula de dimensões apropriadas. Consequentemente, a partir da equação 3.4 as saídas serão as seguintes:

$$\begin{array}{cccc} y_1(0) = g_{11}(0) & y_1(1) = g_{11}(1) & \dots & y_1(k) = g_{11}(k) \\ y_2(0) = g_{21}(0) & y_2(1) = g_{21}(1) & \dots & y_2(k) = g_{21}(k) \\ \vdots & \vdots & \ddots & \vdots \\ y_l(0) = g_{l1}(0) & y_l(1) = g_{l1}(1) & \dots & y_l(k) = g_{l1}(k) \end{array} \tag{3.8}$$

ou seja, ao se aplicar a entrada definida na equação 3.7 pode-se determinar a primeira coluna de todas as matrizes $G(k)$, desde $k = 0$ até $k = \infty$.

Fazendo-se um procedimento similar, percebe-se que em cada um dos m experimentos se encontrará uma das colunas de todas as matrizes G, de forma que ao fim destes experimentos se terá um modelo algébrico capaz de realizar o sistema multivariável em questão. À seqüência de matrizes G é dado o nome de resposta ao impulso discreto multivariável.

Deve-se notar que, diferentemente do caso monovariável, não se pode definir uma única entrada como sendo um impulso discreto multivariável. Desta forma, o termo resposta ao impulso multivariável é usado como uma analogia aos sistemas monovariáveis, e não como uma resposta de um sistema multivariável a apenas uma entrada impulso multivariável. Na verdade, o termo se refere às respostas a vários impulsos, sendo que cada um deles excita uma das entradas do sistema multivariável em questão.

3.2 Identificação de sistemas invariantes no tempo

Uma vez definidas as matrizes de resposta ao impulso, pode-se partir para a representação dos sistemas determinísticos multivariáveis invariantes no tempo. Uma das formas mais bem sucedidas de se representar um sistema multivariável é a representação no espaço de estados. Esta abordagem foi proposta por Kalman na década de 60 e será detalhada a seguir.

Nesta forma de representação, a saída do sistema no instante k é vista como uma combinação entre um vetor x de dimensão n, que resume todo o passado do sistema, ou ainda, que indica o estado do sistema no instante k, e a entrada no instante k. Ao vetor x é dado o nome de vetor de estado. Este vetor também tem sua evolução no tempo dada como uma combinação entre o estado anterior do sistema e a entrada no instante anterior. Algebricamente, o sistema pode ser representado da seguinte forma:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) \\ y(k) = Cx(k) + Du(k) \end{cases} \tag{3.9}$$

em que $A \in \Re^{nxn}$, $B \in \Re^{nxm}$, $C \in \Re^{lxn}$ e $D \in \Re^{lxm}$, ou ainda:

$$\begin{cases} \begin{bmatrix} x_1(k+1) \\ \vdots \\ x_n(k+1) \end{bmatrix} = \begin{bmatrix} a_{11} & \dots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \dots & a_{nn} \end{bmatrix} \begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix} + \begin{bmatrix} b_{11} & \dots & b_{1m} \\ \vdots & \ddots & \vdots \\ b_{n1} & \dots & b_{nm} \end{bmatrix} \begin{bmatrix} u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix} \\ \begin{bmatrix} y_1(k) \\ \vdots \\ y_l(k) \end{bmatrix} = \begin{bmatrix} c_{11} & \dots & c_{1n} \\ \vdots & \ddots & \vdots \\ c_{l1} & \dots & c_{ln} \end{bmatrix} \begin{bmatrix} x_1(k) \\ \vdots \\ x_n(k) \end{bmatrix} + \begin{bmatrix} d_{11} & \dots & d_{1m} \\ \vdots & \ddots & \vdots \\ d_{l1} & \dots & d_{lm} \end{bmatrix} \begin{bmatrix} u_1(k) \\ \vdots \\ u_m(k) \end{bmatrix} \end{cases} \tag{3.10}$$

Sendo assim, o objetivo das técnicas de identificação é determinar as matrizes A, B, C e D, de forma que, para um determinado conjunto de entradas multivariáveis $u(k) \in \Re^m$, se tenha as saídas multivariáveis $y(k) \in \Re^l$ o mais próximas possível das observadas ao se submeter o sistema real às mesmas entradas. Para se encontrar essas matrizes há diversos procedimentos propostos, que dependem de alguns conceitos para serem aplicados. Nesta seção, serão apresentados dois métodos de estimação das matrizes do sistema: o primeiro é baseado nas matrizes de resposta ao impulso discreto e o outro é baseado na decomposição LQ de uma matriz formada por dados de entrada e saída. Ao segundo método é dado o nome de método MOESP (**M**ultivariable **O**utput-**E**rror **S**tate **Sp**ace).

3.2.1 Identificação a partir da resposta ao impulso do sistema

Relação entre resposta ao impulso e as matrizes do sistema em espaço de estados - parâmetros de Markov

Supondo que o sistema tenha estado inicial nulo e seja aplicada a entrada definida na equação 3.7 se terá, para $k = 0$, as seguintes relações:

$$\begin{bmatrix} x_1(1) \\ x_2(1) \\ \dots \\ x_n(1) \end{bmatrix} = \begin{bmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{n1} \end{bmatrix} \quad \begin{bmatrix} y_1(0) \\ y_2(0) \\ \dots \\ y_l(0) \end{bmatrix} = \begin{bmatrix} d_{11} \\ d_{21} \\ \dots \\ d_{l1} \end{bmatrix} \tag{3.11}$$

para $k = 1$ tem-se:

$$\begin{bmatrix} x_1(2) \\ x_2(2) \\ \dots \\ x_n(2) \end{bmatrix} = A \begin{bmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{n1} \end{bmatrix} \quad \begin{bmatrix} y_1(1) \\ y_2(1) \\ \dots \\ y_l(1) \end{bmatrix} = C \begin{bmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{n1} \end{bmatrix} \tag{3.12}$$

para $k = 2$:

$$\begin{bmatrix} x_1(3) \\ x_2(3) \\ \dots \\ x_n(3) \end{bmatrix} = A^2 \begin{bmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{n1} \end{bmatrix} \quad \begin{bmatrix} y_1(2) \\ y_2(2) \\ \dots \\ y_l(2) \end{bmatrix} = CA \begin{bmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{n1} \end{bmatrix} \tag{3.13}$$

Portanto, para k qualquer tem-se que a resposta ao impulso do sistema é dada pela seguinte expressão:

$$\begin{bmatrix} g_{11}(k) \\ g_{12}(k) \\ \dots \\ g_{l1}(k) \end{bmatrix} = \begin{bmatrix} y_1(k) \\ y_2(k) \\ \dots \\ y_l(k) \end{bmatrix} = \begin{cases} \begin{bmatrix} d_{11} \\ d_{21} \\ \dots \\ d_{l1} \end{bmatrix} & k = 0 \\ CA^{k-1} \begin{bmatrix} b_{11} \\ b_{21} \\ \dots \\ b_{n1} \end{bmatrix} & k \neq 0 \end{cases} \tag{3.14}$$

No caso em que o sistema é submetido à entrada:

$$u(k) = \begin{cases} [\ 0 \quad 1 \quad 0 \quad \dots \quad 0\ ]^T & k = 0 \\ 0 & k \neq 0 \end{cases} \tag{3.15}$$

se terá, para $k = 0$, as seguintes relações:

$$\begin{bmatrix} x_1(1) \\ x_2(1) \\ \dots \\ x_n(1) \end{bmatrix} = \begin{bmatrix} b_{12} \\ b_{22} \\ \dots \\ b_{n2} \end{bmatrix} \quad \begin{bmatrix} y_1(0) \\ y_2(0) \\ \dots \\ y_l(0) \end{bmatrix} = \begin{bmatrix} d_{12} \\ d_{22} \\ \dots \\ d_{l2} \end{bmatrix} \tag{3.16}$$

para $k = 1$ tem-se:

$$\begin{bmatrix} x_1(2) \\ x_2(2) \\ \dots \\ x_n(2) \end{bmatrix} = A \begin{bmatrix} b_{12} \\ b_{22} \\ \dots \\ b_{n2} \end{bmatrix} \quad \begin{bmatrix} y_1(1) \\ y_2(1) \\ \dots \\ y_l(1) \end{bmatrix} = C \begin{bmatrix} b_{12} \\ b_{22} \\ \dots \\ b_{n2} \end{bmatrix} \tag{3.17}$$

para $k = 2$:

$$\begin{bmatrix} x_1(3) \\ x_2(3) \\ \dots \\ x_n(3) \end{bmatrix} = A^2 \begin{bmatrix} b_{12} \\ b_{22} \\ \dots \\ b_{n2} \end{bmatrix} \quad \begin{bmatrix} y_1(2) \\ y_2(2) \\ \dots \\ y_l(2) \end{bmatrix} = CA \begin{bmatrix} b_{12} \\ b_{22} \\ \dots \\ b_{n2} \end{bmatrix} \tag{3.18}$$

Portanto, para k qualquer tem-se que a resposta ao impulso do sistema é dada pela seguinte expressão:

$$\begin{bmatrix} g_{12}(k) \\ g_{22}(k) \\ \dots \\ g_{l2}(k) \end{bmatrix} = \begin{bmatrix} y_1(k) \\ y_2(k) \\ \dots \\ y_l(k) \end{bmatrix} = \begin{cases} \begin{bmatrix} d_{12} \\ d_{22} \\ \dots \\ d_{l2} \end{bmatrix} & k = 0 \\ CA^{k-1} \begin{bmatrix} b_{12} \\ b_{22} \\ \dots \\ b_{n2} \end{bmatrix} & k \neq 0 \end{cases} \tag{3.19}$$

Deve-se notar que, da mesma forma que a cada um dos m experimentos se determina uma coluna da matriz de resposta ao impulso do sistema, se encontra um termo que tem relação com as colunas das matrizes B e D da representação de estados do sistema. Com isto pode-se concluir que a matriz de resposta ao impulso tem a seguinte relação com as matrizes de estados:

$$G(k) = \begin{cases} D & k = 0 \\ CA^{k-1}B & k \neq 0 \end{cases} \tag{3.20}$$

Os termos da matriz de reposta ao impulso recebem também o nome de parâmetros de Markov.

Matrizes de Hankel, de atingibilidade e de observabilidade

Seja um sistema com estado inicial nulo. Suponha que este sistema seja submetido a entradas quaisquer, até um determinado instante de tempo $k > 0$, e que depois deste instante todas as entradas são zeradas. A saída do sistema do instante de tempo $k + 1$ em diante dependerá então somente do estado do sistema no instante $k+1$. Esta relação é facilmente obtida desenvolvendo-se a equação 3.9, conforme é demonstrado abaixo.

$$\begin{array}{ll} x(k+2) = Ax(k+1) & y(k+1) = Cx(k+1) \\ x(k+3) = A^2x(k+1) & y(k+2) = CAx(k+1) \\ x(k+4) = A^3x(k+1) & y(k+3) = CA^2x(k+1) \\ \vdots & \vdots \end{array} \tag{3.21}$$

e a relação entre saídas posteriores ao instante k e o estado em $k+1$ pode ser escrita de matricialmente da seguinte forma:

$$\begin{bmatrix} y(k+1) \\ y(k+2) \\ y(k+3) \\ \vdots \end{bmatrix} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \end{bmatrix} x(k+1) \tag{3.22}$$

A matriz que relaciona as saídas futuras (ou seja, do instante $k+1$ em diante) com o estado no instante $k+1$ demonstra quais aspectos do estado serão observados na saída futura. Por este motivo esta matriz é chamada de matriz de observabilidade e é denotada por $\mathcal{O}$. Por simplicidade, denota-se o vetor de saídas a partir do instante $k+1$ como $y^+(k+1)$. Com isto se tem a seguinte relação:

$$y^+(k+1) = \mathcal{O}x(k+1) \tag{3.23}$$

O estado no instante $k+1$ é resultado da influência das entradas aplicadas ao sistema e do estado no instante 0. Supondo que, inicialmente, o estado do sistema é nulo, a relação entre entradas passadas e estado no instante $k+1$ é a seguinte:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) = \\ &= A(Ax(k-1) + Bu(k-1)) + Bu(k) = \\ &= A^2x(k-1) + ABu(k-1) + Bu(k) = \\ &= A^2(Ax(k-2) + Bu(k-2)) + ABu(k-1) + Bu(k) = \\ &= A^3x(k-2) + A^2Bu(k-2) + ABu(k-1) + Bu(k) = \\ &\quad \vdots \end{aligned} \tag{3.24}$$

que também pode ser reescrita matricialmente da seguinte forma:

$$x(k+1) = \begin{bmatrix} B & AB & A^2B & \dots \end{bmatrix} \begin{bmatrix} u(k) \\ u(k-1) \\ u(k-2) \\ \vdots \end{bmatrix} \tag{3.25}$$

lembrando que o estado inicial é suposto nulo. Note que a matriz que relaciona entradas passadas com o estado no instante $k-1$ na verdade informa quais aspectos do estado podem ser alterados pela entrada, ou ainda, quais estados podem ser atingidos pelas entradas. Por este motivo esta matriz recebe o nome de matriz de atingibilidade e é denotada por $\mathcal{C}$. A matriz de entradas passadas pode ser denotada por $u^-(k)$, de forma que se pode escrever, de forma resumida, a seguinte relação:

$$x(k+1) = \mathcal{C}u^-(k) \tag{3.26}$$

Substituindo a equação 3.26 em 3.23 encontra-se que a relação entre entradas passadas e saídas futuras do sistema é dada por:

$$y^+(k+1) = \mathcal{O}\mathcal{C}u^-(k) = Hu^-(k) \tag{3.27}$$

A matriz $H = \mathcal{O}\mathcal{C}$ é conhecida como matriz de Hankel. Ao se fazer o produto entre $\mathcal{O}$ e $\mathcal{C}$ se tem o seguinte:

$$\mathcal{O}\mathcal{C} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \end{bmatrix} \begin{bmatrix} B & AB & A^2B & \dots \end{bmatrix} = \begin{bmatrix} CB & CAB & CA^2B & \dots \\ CAB & CA^2B & CA^3B & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.28}$$

Cada bloco desta matriz é um termo da resposta ao impulso, ou ainda, um parâmetro de Markov, conforme pode ser visto ao se observar a equação acima e compará-la com a equação 3.20. Desta forma, pode-se escrever:

$$H = \mathcal{OC} = \begin{bmatrix} G(1) & G(2) & G(3) & \dots \\ G(2) & G(3) & G(4) & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.29}$$

A partir destas matrizes e destas identidades pode-se fazer a identificação dos sistemas determinísticos conforme foi bem explorado na referência [5] e na seção abaixo.

Método de identificação

Uma forma simples de se fazer a identificação dos sistemas determinísticos invariantes no tempo, ou seja, de encontrar as matrizes A, B, C e D do sistema, é a partir da matriz de Hankel definida acima. Esta técnica é baseada na proposta em [42]. A matriz de Hankel é obtida diretamente através das respostas aos impulsos do sistema.

Seja, por definição, $H_{\uparrow}$ a matriz H deslocada para cima, ou seja:

$$H_{\uparrow} = \begin{bmatrix} CAB & CA^2B & CA^3B & \dots \\ CA^2B & CA^3B & CA^4B & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{3.30}$$

nota-se que esta matriz é, na verdade, igual ao produto $\mathcal{O}A\mathcal{C}$. Portanto, se as matrizes de observabilidade e de atingibilidade forem conhecidas, é possível obter a matriz A. Uma forma de se encontrar as matrizes $\mathcal{O}$ e $\mathcal{C}$ é se fazendo a decomposição da matriz H em valores singulares, ou seja, encontrar as matrizes U, Σ e V que satisfaçam:

$$\mathcal{OC} = H = U\Sigma V^T = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_{11} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \tag{3.31}$$

Desta forma, pode-se definir:

$$\mathcal{O} = U_1\Sigma_{11}^{1/2} \quad \mathcal{C} = \Sigma_{11}^{1/2}V_1^T \tag{3.32}$$

portanto:

$$H_{\uparrow} = \mathcal{O}A\mathcal{C} \Rightarrow A = \mathcal{O}^{\dagger}H_{\uparrow}\mathcal{C}^{\dagger} \tag{3.33}$$

É facil de notar que o primeiro bloco coluna da matriz H, que será denotado por $H(:,1)$, é a matriz de observabilidade multiplicada pela matriz B. Com isto, pode-se escrever o seguinte:

$$H(:,1) = \mathcal{O}B \Rightarrow B = \mathcal{O}^{\dagger}H(:,1) \tag{3.34}$$

Da mesma forma, o primeiro bloco linha de H, denotado por $H(1,:)$, é a matriz C multiplicada pela matriz de atingibilidade. Portanto:

$$H = C\mathcal{C} \Rightarrow C = H\mathcal{C}^{\dagger} \tag{3.35}$$

e a matriz D é obtida diretamente da equação 3.20, como sendo a matriz de resposta no instante 0, ou seja:

$$D = G(0) \tag{3.36}$$

Outra forma de se estimar as matrizes B e C seria tomar os primeiros blocos de dimensões adequadas das matrizes $\mathcal{O}$ e $\mathcal{C}$ obtidos ao se fazer a decomposição em valores singulares da matriz de Hankel.

3.2.2 Método MOESP

Outro método de identificação de sistemas é o MOESP, proposto por Verhaegen e Dewilde em [62] e [61]. A idéia principal deste método é a decomposição de uma matriz formada por dados de entrada e saída do instante 0 até o instante $t-1$, concatenados em duas matrizes via decomposição LQ. Este método é baseado na representação estendida no espaço de estado, apresentada a seguir.

Modelo estendido no espaço de estados

Da mesma forma que os sistemas podem ser representados pela equação 3.10, há também outras formas de se representar a relação entre entradas, saídas e estados. Apesar de envolverem matrizes de dimensões maiores que as de entrada, saída e estados, estas novas formas são úteis pois permitem que diferentes abordagens de identificação sejam adotadas. E atualmente, com a facilidade que se tem para fazer a manipulações e operações algébricas matriciais com o uso de computadores, estes métodos se tornaram viáveis. Três destas formas são explicadas com clareza na tese [5]. Nesta seção será apresentada a primeira delas, que é útil para o desenvolvimento do algoritmo MOESP.

Seja um instante de tempo t, antes do qual todas as entradas são nulas. Com isto, a partir da equação 3.10, pode-se escrever escrever a relação entre entradas, saídas e estados do instante t até um instante $k-1$ arbitrário da seguinte forma:

$$\begin{bmatrix} y(t) \\ y(t+1) \\ \vdots \\ y(t+k-1) \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{bmatrix} x(t) + \begin{bmatrix} D & 0 & 0 & 0 \\ CB & D & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{k-2}B & \dots & CB & D \end{bmatrix} \begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+k-1) \end{bmatrix} \tag{3.37}$$

Definindo:

$$y_{t|k-1} = \begin{bmatrix} y(t) \\ y(t+1) \\ \vdots \\ y(t+k-1) \end{bmatrix} \quad u_{t|k-1} = \begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+k-1) \end{bmatrix} \tag{3.38}$$

em que $y_{t|k-1} \in \Re^{klX1}$ e $u_{t|k-1} \in \Re^{kmX1}$, e definindo também a matriz de Toeplitz:

$$\Psi_k = \begin{bmatrix} D & 0 & 0 & 0 \\ CB & D & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{k-2}B & \dots & CB & D \end{bmatrix} \tag{3.39}$$

pode-se reescrever 3.37 da seguinte forma;

$$y_{t|k-1} = \mathcal{O}_k x(t) + \Psi_k u_{t|k-1} \tag{3.40}$$

Se as matrizes $y_{t|k-1}$ para $t = 0 \dots N-1$ forem concatenadas lado a lado, assim como as entradas $y_{t|k-1}$ e os estados, pode-se definir as seguintes matrizes:

$$U_{0|k-1} = \begin{bmatrix} u_{0|k-1} & u_{1|k} & \dots & u_{N-1|k+N-2} \end{bmatrix} \tag{3.41}$$

$$Y_{0|k-1} = \begin{bmatrix} y_{0|k-1} & y_{1|k} & \dots & y_{N-1|k+N-2} \end{bmatrix} \tag{3.42}$$

$$X_{N-1} = \begin{bmatrix} x(0) & x(1) & \dots & x(N-1) \end{bmatrix} \tag{3.43}$$

e com isto, se escrever o seguinte modelo estendido:

$$Y_{0|k-1} = \mathcal{O}_k X_{N-1} + \Psi_k U_{0|k-1} \tag{3.44}$$

a partir deste modelo estendido será desenvolvido o método MOESP.

Método de identificação

Definidas as matrizes $U_{0|k-1}$ e $Y_{0|k-1}$, pode-se decompor sua concatenação em duas matrizes L e Q da seguinte forma:

$$\begin{bmatrix} U_{0|k-1} \\ Y_{0|k-1} \end{bmatrix} = LQ = \begin{bmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{bmatrix} \begin{bmatrix} Q_1^T \\ Q_2^T \end{bmatrix} \tag{3.45}$$

em que $L_{11} \in \Re^{kmXkm}$ e $L_{22} \in \Re^{klXkm}$ são matrizes triangulares inferiores, $Q_1^T \in \Re^{kmXN}$ e $Q_2^T \in \Re^{klXN}$ são matrizes ortonormais e $L_{21} \in \Re^{kmXkl}$.

Com esta decomposição, pode-se reescrever a equação 3.44 da seguinte forma:

$$L_{21}Q_1^T + L_{22}Q_2^T = \mathcal{O}_k X_{N-1} + \Psi_k L_{11} Q_1^T \tag{3.46}$$

pós multiplicando os dois lados desta equação por Q_2, e tendo em mente a ortonormalidade entre as matrizes Q_1 e Q_2, tem-se:

$$L_{22} = \mathcal{O}_k X_{N-1} Q_2 \tag{3.47}$$

tomando a decomposição em valores singulares da matriz L_{22}, ou seja, encontrando-se as matrizes U, Σ e V tais que $L_{22} = U\Sigma V$ tem-se:

$$L_{22} = U\Sigma V = \begin{bmatrix} U_1 & U_2 \end{bmatrix} \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \mathcal{O}_k X_{N-1} Q_2 \tag{3.48}$$

Portanto pode-se dizer que:

$$\mathcal{O}_k = U_1 \Sigma_1^{\frac{1}{2}} \quad X_{N-1} Q_2 = \Sigma_1^{\frac{1}{2}} V_1 \tag{3.49}$$

A partir da matriz de observabilidade encontra-se facilmente as matrizes C e A. A primeira delas é o primeiro bloco de dimensão lxn de $\mathcal{O}_k$ e a segunda é obtida ao se observar que:

$$\mathcal{O}_{k\uparrow} = \begin{bmatrix} CA \\ CA^2 \\ \vdots \end{bmatrix} = \begin{bmatrix} C \\ CA \\ \vdots \end{bmatrix} A = \mathcal{O}_k A \tag{3.50}$$

portanto:

$$A = \mathcal{O}_k^{\dagger} \mathcal{O}_{k\uparrow} \tag{3.51}$$

Para se obter as matrizes B e D basta observar o seguinte:

$$U_2^T L_{22} = 0 \quad U_2^T \mathcal{O}_k = 0 \tag{3.52}$$

A primeira igualdade vem do fato que, conforme visto na equação 3.48, $L_{22} = U_1 \Sigma_1 V_1$. Como pelas propriedades da decomposição em valores singulares U_1 é ortogonal a U_2, a relação acima fica clara. A segunda igualdade se baseia no seguinte fato: $\mathcal{O}_k = U_1 \Sigma_1^{\frac{1}{2}}$. Portanto, pela mesma propriedade de ortogonalidade entre U_1 e U_2, a igualdade acima fica clara. Observadas estas duas igualdades, multiplica-se os dois lados da equação 3.46 por U_2^T obtendo-se o seguinte:

$$\begin{aligned} & U_2^T (L_{21} Q_1^T + L_{22} Q_2^T) = U_2^T (\mathcal{O}_k X_{N-1} + \Psi_k L_{11} Q_1^T) \Rightarrow \\ & \Rightarrow U_2^T L_{21} Q_1^T = U_2^T \Psi_k L_{11} Q_1^T \Rightarrow \\ & \Rightarrow U_2^T L_{21} = U_2^T \Psi_k L_{11} \\ & \Rightarrow U_2^T L_{21} L_{11}^{-1} = U_2^T \Psi_k \end{aligned} \tag{3.53}$$

Dividindo a matriz U_2^T em blocos com l colunas denominados L_i, dividindo a matriz $U_2^T L_{21} L_{11}^{-1}$ em blocos com m colunas denominados M_i e representando a matriz Ψ_k em função das matrizes B e D temos:

$$\begin{bmatrix} M_1 & M_2 & \dots & M_k \end{bmatrix} = \begin{bmatrix} L_1 & L_2 & \dots & L_k \end{bmatrix} \begin{bmatrix} D & 0 & \dots & 0 \\ CB & D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{k-2}B & CA^{k-3}B & \ddots & D \end{bmatrix} \tag{3.54}$$

de forma que:

$$
\begin{aligned}
&L_1D + L_2CB + \ldots + L_{k-1}CA^{k-3}B + L_kCA^{k-2}B = M_1 \\
&L_2D + L_3CB + \ldots + L_{k-1}CA^{k-4}B + L_kCA^{k-3}B = M_2 \\
&\vdots \\
&L_{k-1}D + L_kCB = M_{k-1} \\
&L_kD = M_k
\end{aligned}
\tag{3.55}
$$

o que pode ser escrito matricialmente da seguinte forma:

$$
\begin{bmatrix}
L_1 & L_2C + \ldots + L_{k-1}CA^{k-3} + L_kCA^{k-2} \\
L_2 & L_3C + \ldots + L_{k-1}CA^{k-4} + L_kCA^{k-3} \\
\vdots & \vdots \\
L_{l-1} & L_kC \\
L_k & 0
\end{bmatrix}
\begin{bmatrix} D \\ B \end{bmatrix} =
\begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_{k-1} \\ M_k \end{bmatrix}
\tag{3.56}
$$

portanto:

$$
\begin{bmatrix} D \\ B \end{bmatrix} =
\begin{bmatrix}
L_1 & L_2C + \ldots + L_{k-1}CA^{k-3} + L_kCA^{k-2} \\
L_2 & L_3C + \ldots + L_{k-1}CA^{k-4} + L_kCA^{k-3} \\
\vdots & \vdots \\
L_{k-1} & L_kC \\
L_k & 0
\end{bmatrix}^{\dagger}
\begin{bmatrix} M_1 \\ M_2 \\ \vdots \\ M_{k-1} \\ M_k \end{bmatrix}
\tag{3.57}
$$

completando assim o cálculo da quadrupla de matrizes do sistema.

Como se pode notar acima, o método MOESP depende de uma grande quantidade de dados para a determinação do modelo estendido, que culmina na determinação das matrizes do modelo no espaço de estado. Há casos em que os dados de entrada e saída do sistema são limitados, implicando em impossibilidade de aplicação deste método. Para lidar com estes casos, foi desenvolvido um método ao longo desta pesquisa, que é apresentado no capítulo 7 desta tese.

3.3 Identificação de sistemas variantes no tempo

Para a solução do problema de identificação de sistemas determinísticos variantes no tempo, um algoritmo baseado no MOESP e denominado MOESP-VAR foi proposto [56], [57].

Basicamente, o método consiste em, dado um sistema variante no tempo, definir intervalos, também denominados janelas, em torno de instantes de tempo específicos escolhidos para amostragem do sistema. O número de dados em cada janela é definido de maneira que o sistema não sofra mudanças durante todo o intervalo de tempo definido para aquela janela. O método MOESP para sistemas invariantes é então aplicado aos dados de entrada e saída do intervalo escolhido, e com isto se encontra as matrizes que representam o sistema neste intervalo. Este algoritmo está baseado na hipótese que um

sistema lentamente variante no tempo pode ser modelado como uma série de sistemas invariantes no tempo em intervalos arbitrários.

O algoritmo MOESP-VAR pode ser aplicado caso haja um grande número de dados disponível para cada janela, ou seja, caso as variações do sistema não sejam muito rápidas, garantindo que haja um conjunto de entradas e saídas suficientemente grande para a identificação via método MOESP em cada um dos intervalos. No caso em que as variações do sistema são rápidas, o método MOESP-VAR não implica em respostas muito precisas. Ao longo desta pesquisa foi desenvolvido um algoritmo imuno-inspirado para lidar com estes casos. Este algoritmo é apresentado no capítulo 7 desta tese.

Na tese [56] são apresentados dois experimentos em que se identifica sistemas variantes no tempo com o algoritmo MOESP-VAR. Nestes experimentos a variação do sistema é conhecida e é possível estabelecer intervalos de forma que o sistema fique bem representado. No entanto, em situações práticas, é necesssário que se faça várias propostas de intervalos para que se encontre uma que modele satisfatoriamente o sistema, a menos que se conheça a dinâmica que implica na variância no tempo do sistema.

Por exemplo, seja uma máquina elétrica sendo partida em um determinado instante de tempo. Supondo que a princípio ela esteja fria, seu aquecimento ao longo da operação implicará em variação da resistividade de seus enrolamentos ao longo do tempo, assim como suas permeâncias também se alteram devido à variação do entreferro, que por sua vez é devida aos efeitos térmicos e das forças magnéticas sobre o estator e ao somatório destes efeitos com as forças causadas pela rotação sobre o rotor. Além disso, a relação entre a densidade de fluxo magnética e a força magnetomotriz é não linear devido à saturação, de forma que o comportamento da máquina depende também da condição de carga. Portanto, a máquina elétrica é um sistema altamente variante no tempo. Desta maneira, para se poder aplicar o algoritmo MOESP-VAR ao problema de identificação de uma máquina elétrica, seria necessário se conhecer a dinâmica de aquecimento, que depende, dentre outros fatores, da dinâmica da carga aplicada à máquina. Em situações em que a temperatura da máquina cresce bastante, seja por uma carga elevada ou seja por estar em uma região de temperatura distante do regime, os intevalos de amostragem teriam que ser menores que os aplicados quando a máquina está próxima do regime de temperatura.

Por este motivo, um método de identificação variante no tempo capaz de atualizar recursivamente as matrizes do modelo em espaço de estado, e que possa ser aplicado a janelas pequenas o suficiente para que se determine qualquer variação, é de muito valor para a solução deste problema e de outros com características similares.

Capítulo 4

Filtro de Kalman

Neste capítulo, é demonstrado o desenvolvimento do algoritmo do filtro de Kalman para sistemas lineares discretos com entradas puramente estocásticas. O estudo do filtro de Kalman envolve o equacionamento que é utilizado no problema de realização de séries temporais, tratado com detalhes no capítulo 5, que por sua vez é um dos temas envolvido nas contribuições desta tese, detalhadas no capítulo 7. A teoria básica para acompanhamento deste capítulo é descrita no capítulo 2 desta tese. Neste capítulo, o desenvolvimento do filtro de Kalman é feito de forma que a relação entre seu equacionamento e o equacionamento do problema de realização de séries temporais, tratado no capítulo 5, é direta. O conteúdo desenvolvido neste capítulo é baseado principalmente nas referências [2], [4], [45] e [46].

O filtro de Kalman é um método para se estimar o vetor de estado de um sistema dinâmico linear, com componentes estocásticos, a partir da informação trazida por sua saída. Para um sistema puramente determinístico, não faria sentido criar um método de estimação de estado, uma vez que, supondo que o estado inicial e as entradas são informações conhecidas, pode-se determinar qualquer estado de forma exata a partir da equação de atualização de estado. No entanto, ao se considerar que o sistema está sujeito a ruídos, o cálculo preciso do vetor de estado não é possível. Por este motivo, Kalman[1] desenvolveu e publicou em 1960 [45] uma técnica de se encontrar um processo estocástico com média igual à média do vetor de estado. Além disto, a covariância da diferença entre o estado real e o referido processo estocástico é mínima. Além de tudo isto, o processo estocástico encontrado tem relação linear com as saídas. Ou seja, Kalman propôs um estimador de estado linear, não polarizado e de mínima variância.

O sistema linear discreto sobre o qual é desenvolvido o algoritmo do filtro de Kalman é descrito da seguinte forma:

$$\begin{cases} x(k+1) = A(k)x(k) + G(k)u(k) \\ y(k) = C(k)x(k) \end{cases} \tag{4.1}$$

[1]Nascido em 1930 na Hungria, Rudolf Kalman obteve o título de bacharel em engenharia elétrica em 1953 e o título de mestre em engenharia elétrica em 1954, ambos pelo MIT. Em 1957 obteve seu título de doutor pela Colúmbia University e no ano seguinte passou a trabalhar no Research Institute for Advanced Study, onde ficou até 1964. Nesta época publicou seu trabalho histórico sobre o método de filtragem que ganharia seu nome. Apesar de ser engenheiro eletricista, Kalman publicou seu trabalho histórico em uma revista de Engenharia Mecânica. O algoritmo de filtragem desenvolvido por ele foi usado no programa espacial norte americano durante a corrida espacial da guerra fria. Seu trabalho lhe rendeu muitos prêmios, além de inaugurar um novo paradigma na representação de sistemas dinâmicos.

em que $x(k) \in \Re^n$ é o vetor de estado do sistema no instante k, $u(k) \in \Re^m$ é um vetor de processos estocásticos de ruído de transição de estado, $y(k) \in \Re^l$ é o sinal de saída do sistema, $A(k)$, $G(k)$ e $C(k)$ são matrizes de dimensões apropriadas. Para o desenvolvimento do algoritmo do filtro de Kalman, parte-se da hipótese que as matrizes do sistema são conhecidas.

Para um sistema representado desta forma, é demonstrado neste capítulo o desenvolvimento do algoritmo do filtro de Kalman a partir das seguintes etapas:

- Revisão da função de densidade de probabilidade condicional de uma distribuição gaussiana multidimensional
- Solução do problema do estimador linear de mínima variância.
- Demonstração que a estimativa linear de mínima variância é igual ao valor esperado da probabilidade condicional para variáveis aleatórias gaussianas.
- Solução do problema de estimação ótima via projeções ortogonais.
- Demonstração que a solução de mínima variância coincide com a obtida por projeções ortogonais.
- Definição do conceito de inovação.
- Demonstração do algoritmo de estimação ótima por projeção ortogonal para predição e filtragem

4.1 Distribuição Gaussiana multivariável

4.1.1 Função de densidade condicional

Sejam $x \in \Re^n$ e $y \in \Re^l$ dois vetores de variáveis aleatórias gaussianas. Ou seja, são dois vetores em que cada um dos elementos é uma variável aleatória que cuja função de densidade de probabilidade é gaussiana. Sejam μ_x e μ_y dois vetores contendo as médias das variáveis aleatórias contidas em x e y respectivamente e seja Σ a matriz de covariância apresentada abaixo:

$$\Sigma = E[xy^T] = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix} \tag{4.2}$$

em que:

$$\Sigma_{xx} = \begin{bmatrix} E[(x_1 - \mu_{x1})(x_1 - \mu_{x1})^T] & \dots & E[(x_1 - \mu_{x1})(x_n - \mu_{xn})^T] \\ E[(x_2 - \mu_{x2})(x_1 - \mu_{x1})^T] & \dots & E[(x_2 - \mu_{x2})(x_n - \mu_{xn})^T] \\ \vdots & \ddots & \vdots \\ E[(x_n - \mu_{xn})(x_1 - \mu_{x1})^T] & \dots & E[(x_n - \mu_{xn})(x_n - \mu_{xn})^T] \end{bmatrix}$$

$$\Sigma_{xy} = \begin{bmatrix} E[(x_1 - \mu_{x1})(y_1 - \mu_{y1})^T] & \dots & E[(x_1 - \mu_{x1})(y_p - \mu_{yp})^T] \\ E[(x_2 - \mu_{x2})(y_1 - \mu_{y1})^T] & \dots & E[(x_2 - \mu_{x2})(y_p - \mu_{yp})^T] \\ \vdots & \ddots & \vdots \\ E[(x_n - \mu_{xn})(y_1 - \mu_{y1})^T] & \dots & E[(x_n - \mu_{xn})(y_p - \mu_{yp})^T] \end{bmatrix}$$

$$\Sigma_{yx} = \begin{bmatrix} E[(y_1 - \mu_{y1})(x_1 - \mu_{x1})^T] & \dots & E[(y_1 - \mu_{y1})(x_n - \mu_{xn})^T] \\ E[(y_2 - \mu_{y2})(x_1 - \mu_{x1})^T] & \dots & E[(y_2 - \mu_{y2})(x_n - \mu_{xn})^T] \\ \vdots & \ddots & \vdots \\ E[(y_p - \mu_{yp})(x_1 - \mu_{x1})^T] & \dots & E[(y_p - \mu_{yp})(x_n - \mu_{xn})^T] \end{bmatrix}$$

$$\Sigma_{yy} = \begin{bmatrix} E[(y_1 - \mu_{y1})(y_1 - \mu_{y1})^T] & \dots & E[(y_1 - \mu_{y1})(y_p - \mu_{yp})^T] \\ E[(y_2 - \mu_{y2})(y_1 - \mu_{y1})^T] & \dots & E[(y_2 - \mu_{y2})(y_p - \mu_{yp})^T] \\ \vdots & \ddots & \vdots \\ E[(y_p - \mu_{yp})(y_1 - \mu_{y1})^T] & \dots & E[(y_p - \mu_{yp})(y_p - \mu_{yp})^T] \end{bmatrix}$$

em que x_i, $i = 1 \dots n$ e y_j, $j = 1 \dots l$ são os elementos dos vetores x e y e $E[.]$ representa o operador esperança. Por hipótese, é suposto que a matriz Σ tenha inversa.

A função de densidade de probabilidade de uma distribuição gaussiana monovariável é dada por:

$$p(a) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{1}{2}\frac{(a - \mu_a)^2}{\sigma^2}\right) \tag{4.3}$$

A partir disto pode-se definir para o caso multidimensional a distribuição gaussiana de um determinado vetor z qualquer será:

$$p(z) = \frac{1}{(2\pi)^{\dim(z)/2}|\Sigma_{zz}|^{1/2}} \exp\left(-\frac{1}{2}(z - \mu_z)^T \Sigma_{zz}^{-1}(z - \mu_z)\right) \tag{4.4}$$

Se z for a concatenação dos vetores x e y apresentados acima, que são gaussianos por hipótese, existirá a seguinte distribuição conjunta:

$$p(x, y) = \frac{1}{(2\pi)^{(n+l)/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}[(x - \mu_x)^T \quad (y - \mu_y)^T]\Sigma^{-1} \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix}\right) \tag{4.5}$$

Se, por definição:

$$C = (2\pi)^{(n+l)/2}|\Sigma|^{1/2} \tag{4.6}$$

e

$$Q(x, y) = [(x - \mu_x)^T \quad (y - \mu_y)^T]\Sigma^{-1} \begin{bmatrix} x - \mu_x \\ y - \mu_y \end{bmatrix} \tag{4.7}$$

chega-se à uma equação simplificada da distribuição conjunta dos vetores gaussianos x e y:

$$p(x, y) = \frac{1}{C} \exp\left(-\frac{1}{2}Q(x, y)\right) \tag{4.8}$$

4.1.2 Probabilidade condicional de vetores de variáveis aleatórias gaussianas

A partir desta função de densidade de probabilidade, quer se encontrar a distribuição condicional do vetor x, sendo que já é conhecido o valor de y. Para encontrar a distribuição conjunta, basta aplicar a fórmula de Bayes abaixo:

$$p(x|y) = \frac{p(x,y)}{p(y)} \tag{4.9}$$

Como y é um vetor de variáveis aleatórias gaussianas, tem-se que sua distribuição é, conforme a equação 4.4, dada por:

$$p(y) = \frac{1}{(2\pi)^{l/2}|\Sigma_{yy}|^{1/2}} \exp\left(-\frac{1}{2}(y-\mu_y)^T\Sigma_{yy}^{-1}(y-\mu_y)\right) \tag{4.10}$$

Para conhecer o valor de x dado y, basta agora manipular a expressão de $p(x,y)$ e resolver a equação 4.9, o que é feito a seguir.

Seja, por definição:

$$\Sigma^{-1} = \begin{bmatrix} \Sigma_{xx} & \Sigma_{xy} \\ \Sigma_{yx} & \Sigma_{yy} \end{bmatrix}^{-1} = \begin{bmatrix} V_{xx} & V_{xy} \\ V_{yx} & V_{yy} \end{bmatrix} \tag{4.11}$$

em que, conforme pode ser facilmente provado por substituição, tem-se as seguintes igualdades:

$$V_{xx} = \Upsilon^{-1} \tag{4.12}$$

$$V_{xy} = -\Upsilon^{-1}\Sigma_{xy}\Sigma_{yy}^{-1} \tag{4.13}$$

$$V_{yx} = -\Sigma_{yy}^{-1}\Sigma_{yx}\Upsilon^{-1} \tag{4.14}$$

$$V_{yy} = \Sigma_{yy}^{-1} + \Sigma_{yy}^{-1}\Sigma_{yx}\Upsilon^{-1}\Sigma_{xy}\Sigma_{yy}^{-1} \tag{4.15}$$

em que $\Upsilon = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}$. Tem-se também que as seguintes relações são válidas:

$$\begin{aligned} \Upsilon^T &= (\Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx})^T = \\ &= \Sigma_{xx}^T - \Sigma_{yx}^T\Sigma_{yy}^{-1T}\Sigma_{xy}^T = \\ &= \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma yx = \\ &= \Upsilon \end{aligned} \tag{4.16}$$

como $\Upsilon^T = \Upsilon$, $\Upsilon^{-1T} = \Upsilon^{-1}$ dado que:

$$\Upsilon\Upsilon^{-1} = I \Rightarrow \Upsilon^T\Upsilon^{-1} = I \Rightarrow \Upsilon^{-1T}\Upsilon^T\Upsilon^{-1} = \Upsilon^{-1T} \Rightarrow \Upsilon^{-1} = \Upsilon^{-1T}$$

Além disto,

$$\begin{aligned} V_{xy}^T &= (-\Upsilon^{-1}\Sigma_{xy}\Sigma_{yy}^{-1})^T = \\ &= -\Sigma_{yy}^{-1T}\Sigma_{xy}^T\Upsilon^{-1T} \\ &= -\Sigma_{yy}^{-1}\Sigma_{yx}\Upsilon^{-1} \\ &= V_{yx} \end{aligned} \tag{4.17}$$

e também:

$$V_{xx}^{-1} = \Upsilon \Rightarrow V_{xx}^{-1T} = \Upsilon^T = \Upsilon = V_{xx}^{-1} \tag{4.18}$$

Sendo que as igualdades $\Sigma_{yy}^{-1T} = \Sigma_{yy}^{-1}$ e $\Sigma_{yx}^T = \Sigma_{xy}$ usadas acima vêm do fato que estas matrizes são covariâncias, ou o inverso de covariâncias, e, portanto, são de matrizes simétricas.

A partir das relações que vão da equação 4.12 até a 4.18 tem-se que:

$$\begin{aligned} Q(x,y) &= [(x-\mu_x)^T \quad (y-\mu_y)^T]\Sigma^{-1}\begin{bmatrix} x-\mu_x \\ y-\mu_y \end{bmatrix} = \\ &= [(x-\mu_x)^T \quad (y-\mu_y)^T]\begin{bmatrix} V_{xx} & V_{xy} \\ V_{yx} & V_{yy} \end{bmatrix}\begin{bmatrix} x-\mu_x \\ y-\mu_y \end{bmatrix} = \\ &= (x-\mu_x)^T V_{xx}(x-\mu_x) + (y-\mu_y)^T V_{yx}(x-\mu_x) + \\ &\quad +(x-\mu_x)^T V_{xy}(y-\mu_y) + (y-\mu_y)^T V_{yy}(y-\mu_y) = \\ &= (x-\mu_x)^T V_{xx}(x-\mu_x) + (y-\mu_y)^T V_{xy}^T V_{xx}^{-1T} V_{xx}(x-\mu_x) + \\ &\quad +(x-\mu_x)^T V_{xx} V_{xx}^{-1} V_{xy}(y-\mu_y) + (y-\mu_y)^T V_{yy}(y-\mu_y) \end{aligned}$$

adicionando o termo $(y-\mu_y)^T V_{xy}^T V_{xx}^{-1T} V_{xx} V_{xx}^{-1} V_{xy}(y-\mu_y)$ e subtraindo o mesmo termo, mas escrito da forma $(y-\mu_y)^T V_{yx} V_{xx}^{-1} V_{xy}(y-\mu_y)$, encontra-se:

$$\begin{aligned} Q(x,y) &= (x-\mu_x)^T V_{xx}(x-\mu_x) + (y-\mu_y)^T V_{xy}^T V_{xx}^{-1T} V_{xx}(x-\mu_x) + \\ &\quad +(x-\mu_x)^T V_{xx} V_{xx}^{-1} V_{xy}(y-\mu_y) + (y-\mu_y)^T V_{yy}(y-\mu_y) + \\ &\quad +(y-\mu_y)^T V_{xy}^T V_{xx}^{-1T} V_{xx} V_{xx}^{-1} V_{xy}(y-\mu_y) - (y-\mu_y)^T V_{yx} V_{xx}^{-1} V_{xy}(y-\mu_y) \end{aligned}$$

rearranjando a ordem dos termos temos:

$$\begin{aligned} Q(x,y) &= (x-\mu_x)^T V_{xx}(x-\mu_x) + (y-\mu_y)^T V_{xy}^T V_{xx}^{-1T} V_{xx}(x-\mu_x) + \\ &\quad +(x-\mu_x)^T V_{xx} V_{xx}^{-1} V_{xy}(y-\mu_y) + (y-\mu_y)^T V_{xy}^T V_{xx}^{-1T} V_{xx} V_{xx}^{-1} V_{xy}(y-\mu_y) + \\ &\quad +(y-\mu_y)^T V_{yy}(y-\mu_y) - (y-\mu_y)^T V_{yx} V_{xx}^{-1} V_{xy}(y-\mu_y) \end{aligned}$$

colocando termos comuns em evidência tem-se:

$$\begin{aligned} Q(x,y) &= [(x-\mu_x)+V_{xx}^{-1}V_{xy}(y-\mu_y)]^T V_{xx}[(x-\mu_x)+V_{xx}^{-1}V_{xy}(y-\mu_y)]+ \\ &\quad +(y-\mu_y)^T[V_{yy}-V_{yx}V_{xx}^{-1}V_{xy}](y-\mu_y) \end{aligned}$$

definindo:

$$\begin{aligned} \alpha &= \mu_x - V_{xx}^{-1}V_{xy}(y-\mu_y) = \mu_x - \Upsilon(-\Upsilon^{-1}\Sigma_{xy}\Sigma_{yy}^{-1})(y-\mu_y) = \\ &= \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y-\mu_y) \end{aligned} \tag{4.19}$$

e observando que:

$$\begin{aligned} V_{yy}-V_{yx}V_{xx}^{-1}V_{xy} &= \Sigma_{yy}^{-1}+\Sigma_{yy}^{-1}\Sigma_{yx}\Upsilon^{-1}\Sigma_{xy}\Sigma_{yy}^{-1}- \\ &\quad -(-\Sigma_{yy}^{-1}\Sigma_{yx}\Upsilon^{-1})\Upsilon(-\Upsilon^{-1}\Sigma_{xy}\Sigma_{yy}^{-1}) = \\ &= \Sigma_{yy}^{-1} \end{aligned}$$

tem-se que

$$Q(x,y) = (x-\alpha)^T\Upsilon^{-1}(x-\alpha)+(y-\mu_y)^T\Sigma_{yy}^{-1}(y-\mu_y) \tag{4.20}$$

de forma que a função de densidade de probabilidade, apresentada na equação 4.5, pode ser reescrita como:

$$\begin{aligned} p(x,y) &= \tfrac{1}{C}\exp\left(-\tfrac{1}{2}\left((x-\alpha)^T\Upsilon^{-1}(x-\alpha)+(y-\mu_y)^T\Sigma_{yy}^{-1}(y-\mu_y)\right)\right) \\ &= \tfrac{1}{C'}\exp\left(-\tfrac{1}{2}(x-\alpha)^T\Upsilon^{-1}(x-\alpha)\right)\cdot \\ &\quad \cdot\tfrac{1}{C''}\exp\left(-\tfrac{1}{2}(y-\mu_y)^T\Sigma_{yy}^{-1}(y-\mu_y)\right) \end{aligned} \tag{4.21}$$

em que $C' = (2\pi)^{n/2}|\Upsilon|^{1/2}$ e $C'' = (2\pi)^{l/2}|\Sigma_{yy}|^{1/2}$, o que de fato faz com que seja verdade que $C = C'C''$.

Substituindo a equação 4.21 na fórmula de Bayes (equação 4.9) e, lembrando da densidade de probabilidade do vetor gaussiano y, apresentada na equação 4.10, encontra-se a função de densidade de probabilidade de x dado y como sendo:

$$p(x|y) = \frac{1}{C'}\exp\left(-\frac{1}{2}(x-\alpha)^T\Upsilon^{-1}(x-\alpha)\right) \tag{4.22}$$

Da equação 4.22 nota-se que a probabilidade condicional também é uma gaussiana, com média α e variância Υ. Neste desenvolvimento, foi considerado que a matriz de covariância Σ tem inversa, ou seja, é não singular. Caso ela seja singular, o desenvolvimento é válido ao se substituir a inversa da matriz pela pseudoinversa $\Sigma^\dagger$.

4.2 Estimador linear de mínima variância

Seja uma variável aleatória gaussiana x, de média μ_x e variância σ_{xx}, que se quer estimar a partir de medidas de uma outra variável aleatória y, também gaussiana, de média μ_y e variância σ_{yy}. Uma das formas mais simples de se fazer a estimativa $\hat{x}$ para a variável aleatória x é supor que $\hat{x}$ é uma função linear de y, ou seja:

$$\hat{x} = ay + b \tag{4.23}$$

e no caso em que x e y são vetores de variáveis aleatórias de dimensão n e l respectivamente, a estimativa linear é feita de forma similar, a menos que a constante a é substituída por uma matriz $A \in \Re^{nXl}$ e b passa a ser um vetor de dimensão n, ou seja:

$$\hat{x} = Ay + b \tag{4.24}$$

Além disto, os vetores de variáveis aleatórias terão vetores de média μ_x e μ_y, matrizes de covariância própria Σ_{xx} e Σ_{yy} e matriz de covariância cruzada Σ_{xy}. Como y é uma variável aleatória gaussiana, a estimativa $\hat{x}$ também é uma variável aleatória gaussiana.

O erro de medição $\tilde{x}$ é definido como sendo a diferença entre a variável aleatória real x e a estimativa $\hat{x}$, de forma que este erro também é uma variável aleatória gaussiana. Portanto, o erro é completamente caracterizado pelos seus dois primeiros momentos, ou seja, por sua média e por sua variância.

As constantes A e b da equação 4.24 podem ser quaisquer, no entanto podem existir constantes tais que a variância do erro da estimação seja mínima, conforme será investigado a seguir.

Supondo uma estimativa linear como feito acima, a covariância do erro é dada pela seguinte expressão:

$$\begin{aligned} E[\tilde{x}\tilde{x}^T] &= E[(x - Ay - b)(x - Ay - b)^T] = \\ &= E[xx^T] - E[xy^T]A^T - E[x]b^T - AE[yx^T] + \\ &\quad + AE[yy^T]A^T + AE[y]b^T - bE[x^T] + bE[y^T]A^T + bb^T \end{aligned} \tag{4.25}$$

como:

$$\begin{aligned} \Sigma_{xx} &= E[(x - \mu_x)(x - \mu_x)^T] = E[xx^T] - \mu_x\mu_x^T \\ \Sigma_{yy} &= E[(y - \mu_y)(y - \mu_y)^T] = E[yy^T] - \mu_y\mu_y^T \\ \Sigma_{xy} &= E[(x - \mu_x)(y - \mu_y)^T] = E[xy^T] - \mu_x\mu_y^T \end{aligned} \tag{4.26}$$

temos:

$$\begin{aligned}
E[\tilde{x}\tilde{x}^T] &= \Sigma_{xx} + \mu_x\mu_x^T - (\Sigma_{xy} + \mu_x\mu_y^T)A^T - \mu_x b^T - A(\Sigma_{xy}^T + \mu_y\mu_x^T) + \\
&\quad + A(\Sigma_{yy} + \mu_y\mu_y^T)A^T + A\mu_y b^T - b\mu_x^T + b\mu_y^T A^T + bb^T = \\
&= \Sigma_{xx} - A\Sigma_{xy}^T - \Sigma_{xy}A^T + A\Sigma_{yy}A^T + \\
&\quad + \mu_x\mu_x^T - \mu_x\mu_y^T A^T - \mu_x b^T - A\mu_y\mu_x^T + A\mu_y\mu_y^T A^T + A\mu_y b^T - \\
&\quad - b\mu_x^T + b\mu_y^T A^T + bb^T = \\
&= [(A - \Sigma_{xy}\Sigma_{yy}^{-1})\Sigma_{yy}(A^T - \Sigma_{yy}^{-1}\Sigma_{xy}^T)] + \\
&\quad + \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{xy}^T + \\
&\quad + (\mu_x - A\mu_y - b)(\mu_x - A\mu_y - b)^T
\end{aligned} \tag{4.27}$$

Pode-se notar claramente que a variância do erro de um estimador linear qualquer é formado por três termos. Dois deles são quadráticos, portanto, semi definidos positivos, e dependem das constantes A e b. Se estes termos forem nulos, se terá a dupla $\{\hat{A}, \hat{b}\}$ que implica no estimador linear de variância mínima, ou seja, o estimador linear ótimo quando o critério é a variância do erro[2]. É fácil notar que os termos quadráticos são nulos quando:

$$\begin{aligned}
\hat{A} &= \Sigma_{xy}\Sigma_{yy}^{-1} \\
\hat{b} &= \mu_x - \hat{A}\mu_y = \mu_x - \Sigma_{xy}\Sigma_{yy}^{-1}\mu_y
\end{aligned} \tag{4.28}$$

Com isto, a estimativa linear de menor variância $\hat{x}$ é a seguinte:

$$\hat{x} = \hat{A}y + \hat{b} = \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y) \tag{4.29}$$

o que é exatamente igual à media α da probabilidade condicional definida na equação 4.19. Desta forma pode-se dizer o seguinte:

> *Para variáveis aleatórias gaussianas, a estimativa linear de mínima variância é igual ao valor esperado da probabilidade condicional.*

Além de levar a uma estimativa igual à probabilidade condicional, o estimador linear de mínima variância é também não polarizado, ou seja, o valor esperado da estimativa é igual ao valor esperado da variável aleatória real, conforme demonstrado abaixo:

$$E[\hat{x}] = E[\hat{A}y + \hat{b}] = \hat{A}\mu_y + \hat{b} = \hat{A}\mu_y + \mu_x - \hat{A}\mu_y = \mu_x \tag{4.30}$$

Alternativamente, pode-se demonstrar o fato de o estimador ser não polarizado ao se calcular a esperança do erro $\tilde{x}$, que é igual a zero conforme desenvolvido abaixo:

[2]Este critério não necessariamente é o único que pode ser usado para se encontrar estimadores ótimos.

$$\begin{aligned} E[\tilde{x}] &= E[x - \hat{x}] = E[x - \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y)] \\ &= \mu_x - \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(\mu_y - \mu_y) = 0 \end{aligned} \tag{4.31}$$

Obviamente, a partir da equação 4.27, covariância do erro ao se usar o estimador ótimo é:

$$E[\tilde{x}\tilde{x}^T] = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{xy}^T \tag{4.32}$$

que é exatamente igual à covariância Υ da densidade de probabilidade condicional.

A seguir é apresentado um exemplo de como estimar um vetor de variáveis aleatórias x a partir de outro vetor de variáveis aleatórias y, dado que a relação entre estas variáveis aleatórias é conhecida. Com este exemplo, algumas passagens algébricas envolvidas no desenvolvimento do filtro de Kalman são exploradas dando familiaridade com o que é desenvolvido mais adiante.

Seja a seguinte relação entre variáveis aleatórias:

$$y = Hx + v \tag{4.33}$$

Em que $x \in \Re^n$ é um vetor aleatório gaussiano de média μ_x e matriz de covariância P , $y \in \Re^l$, H é uma matriz de dimensões apropriadas e v um vetor aleatório de média nula e covariância dada pela matriz R . O primeiro momento da variável y é:

$$\mu_y = E[y] = HE[x] + E[v] = H\mu_x \tag{4.34}$$

A covariância entre x e y é dada por:

$$\begin{aligned} \Sigma_{xy} &= E[(x - \mu_x)(y - \mu_y)^T] \\ &= E[(x - \mu_x)(Hx + v - H\mu_x)^T] \\ &= E[xx^TH^T + xv^T - x\mu_x^TH^T - \mu_x x^TH^T - \mu_x v^T + \mu_x\mu_x^TH^T] = \\ &= E[xx^T - x\mu_x^T - \mu_x x^T + \mu_x\mu_x^T]H^T = \\ &= E[(x - \mu_x)(x - \mu_x)^T]H^T = PH^T \end{aligned} \tag{4.35}$$

e a variância própria de y dada por:

$$\begin{aligned}
\Sigma_{yy} &= E[(y-\mu_y)(y-\mu_y)^T] \\
&= E[(Hx+v-H\mu_x)(Hx+v-H\mu_x)^T] \\
&= E[Hxx^TH^T + Hxv^T - Hx\mu_x^TH^T + vx^TH^T + vv^T + v\mu_x^TH^T - \\
&\quad -H\mu_xx^TH^T - H\mu_xv^T + H\mu_x\mu_x^TH^T] = \\
&= HE[xx^T - x\mu_x^T - \mu_xx^T + \mu_x\mu_x^T]H^T + E[vv^T] = \\
&= HE[(x-\mu_x)(x-\mu_x)^T]H^T + E[vv^T] = HPH^T + R
\end{aligned} \tag{4.36}$$

Sendo assim, o estimador $\hat{x}$ de menor variância é:

$$\hat{x} = \alpha = \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y-\mu_y) = \mu_x + PH^T(HPH^T+R)^{-1}(y-H\mu_x) \tag{4.37}$$

e da equação 4.32 se tem que a covariância do erro $\hat{P}$ é dada por:

$$\hat{P} = \Upsilon = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx} = P - PH^T(HPH^T+R)^{-1}HP^T$$

Como $P = E[(x-\mu_x)(x-\mu_x)^T] = P^T$, $\hat{P}$ pode ser reescrito como:

$$\hat{P} = \Upsilon = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx} = P - PH^T(HPH^T+R)^{-1}HP \tag{4.38}$$

A equação pode ser simplificada ao se notar que[3]:

$$PH^T(HPH^T+R)^{-1} = (P^{-1} + H^TR^{-1}H)^{-1}H^TR^{-1}$$

portanto, $\hat{P}$ pode ser reescrito como:

$$\begin{aligned}
&\Rightarrow \hat{P} = P - PH^T(HPH^T+R)^{-1}HP = P - (P^{-1}+H^TR^{-1}H)^{-1}H^TR^{-1}HP \Rightarrow \\
&\Rightarrow (P^{-1}+H^TR^{-1}H)\hat{P} = (P^{-1}+H^TR^{-1}H)P - H^TR^{-1}HP \Rightarrow \\
&\Rightarrow (P^{-1}+H^TR^{-1}H)\hat{P} = I + H^TR^{-1}HP - H^TR^{-1}HP \Rightarrow \\
&\Rightarrow \hat{P} = (P^{-1}+H^TR^{-1}H)^{-1}
\end{aligned} \tag{4.39}$$

Este exemplo simples tratando de vetores de variáveis aleatórias é importante pois introduz passagens algébricas que serão exploradas quando se for tratar da estimação de processos estocásticos, que é o problema resolvido pelo filtro de Kalman.

[3] $[P^{-1}+H^TR^{-1}H]PH^T = P^{-1}PH^T + H^TR^{-1}HPH^T = H^TR^{-1}R + H^TR^{-1}HPH^T = H^TR^{-1}(HPH^T+R)$. Ao se pré multiplicar os dois extremos da igualdade por $(P^{-1}+H^TR^{-1}H)^{-1}$ e pós multiplicando por $(HPH^T+R)^{-1}$ tem-se a igualdade apresentada.

4.3 Estimador por projeção ortogonal

O problema de estimação ótima também pode ser tratado sob a ótica de um espaço vetorial de variáveis aleatórias em que é definido um produto interno tal que a norma de qualquer vetor deste espaço seja finita. Este espaço é um espaço de Hilbert[4] e é denotado por $\mathcal{H}$.

Sejam duas variáveis aleatórias x e y pertencentes a $\mathcal{H}$. Por serem elementos do espaço $\mathcal{H}$, x e y podem ser chamados de vetores de $\mathcal{H}$, embora x e y sejam apenas variáveis aleatórias, e não vetores de variáveis aleatórias. Neste ponto deve ficar bem clara a distinção entre um vetor de um espaço de variáveis aleatórias, que é uma variável aleatória, e vetores de variáveis aleatórias, que são coleções de variáveis aleatórias. A princípio se está falando do primeiro grupo, ou seja, de vetores de um espaço de variáveis aleatórias.

O produto interno entre os vetores x e y pode ser definido da seguinte forma:

$$(x, y)_{\mathcal{H}} = E[xy] \tag{4.40}$$

Assim como em qualquer outro espaço, se os vetores x e y são ortogonais, seu produto interno é nulo e pode-se escrever $x \perp y$.

A norma de um vetor no espaço $\mathcal{H}$ pode ser definida como:

$$||x||_{\mathcal{H}} = \sqrt{E[xx]} \tag{4.41}$$

Por definição, neste espaço são contidos vetores de norma finita, ou seja, para que $x \in \mathcal{H}$, x deve satisfazer:

$$||x||_{\mathcal{H}} = \sqrt{E[xx]} < \infty \tag{4.42}$$

Suponha agora que x é um vetor de variáveis aleatórias com n elementos e y um vetor de variáveis aleatórias com l elementos. Para que todos elementos de x sejam ortogonais a todos os elementos de y, a seguinte igualdade deve ser válida:

$$(x_i, y_j)_{\mathcal{H}} = E[x_i y_j] = 0 \tag{4.43}$$

para todos $i = 1 \ldots n$ e $j = 1 \ldots l$. Ou seja, a matriz $E[xy^t]$ deve ser o zero do espaço $\Re^{nXl}$.

Suponha que, dentro do espaço $\mathcal{H}$, exista um subespaço $\mathcal{Y}$ formado por vários vetores y_j. Suponha também que exista um vetor x no espaço $\mathcal{H}$, que não está contido em $\mathcal{Y}$. A variável aleatória $\hat{x}$ que representa a menor projeção de x em $\mathcal{Y}$ é tal que o erro $\tilde{x} = x - \hat{x}$ é perpendicular a $\mathcal{Y}$, ou seja, $\hat{x}$ é uma projeção ortogonal do vetor x em $\mathcal{Y}$.

Se x for um vetor de variáveis aleatórias x_i com n elementos, cada um destes elementos terá uma projeção em um espaço $\mathcal{Y}$ formado por l variáveis aleatórias y_j. Portanto, existe um vetor de variáveis aleatórias $\hat{x} \in \Re^n$, sendo que cada um de seus componentes é a projeção de cada componente do vetor x no subespaço $\mathcal{Y}$. Desta forma, existem termos a_{ij} de uma matriz $A \in \Re^{nxl}$ que representam o componente de cada x_i na direção de cada y_j. Com isto, a projeção de um vetor de variáveis aleatórias

[4]David Hilbert - Matemático alemão nascido em 1862 em Königsberg. Hilbert é considerado um principais matemáticos do século XX tendo contribuído com a teoria dos números, criação de espaços e axiomatização da geometria Euclidiana. Faleceu em 1943 após ver o grupo de estudos de sua universidade - Göttingen - ser dizimado pelo regime nazista.

x, formado por n variáveis aleatórias x_i, no espaço formado por um vetor de variáveis aleatórias y, formado por p variáveis aleatórias y_j, é dada pela seguinte expressão:

$$\hat{x} = Ay \tag{4.44}$$

Como os vetores de variáveis aleatórias envolvidos no problema podem ter médias não nulas, introduz-se uma constante b, de forma que x não é mais formado por elementos que são combinações lineares das variáveis aleatórias y_j, mas sim variedades lineares destas variáveis aleatórias, ou seja:

$$\hat{x} = Ay + b \tag{4.45}$$

Para que $\tilde{x}$ seja ortogonal a qualquer vetor $y \in \mathcal{Y}$, a seguinte expressão deve ser válida:

$$E[\tilde{x}y] = 0 \tag{4.46}$$

para todo $y \in \mathcal{Y}$. Portanto:

$$\begin{aligned}
&E[(x - \hat{x})y^T] = 0 \Rightarrow \\
&\Rightarrow E[(x - Ay - b)y^T] = 0 \Rightarrow \\
&\Rightarrow E[xy^T] - AE[yy^T] - b\mu_y^T = 0 \Rightarrow \\
&\Rightarrow \Sigma_{xy} + \mu_x\mu_y^T - A\Sigma_{yy} - A\mu_y\mu_y^T - b\mu_y^T = 0 \Rightarrow \\
&\Rightarrow (\Sigma_{xy} - A\Sigma_{yy}) + (\mu_x - A\mu_y - b)\mu_y^T = 0
\end{aligned} \tag{4.47}$$

Pode-se notar facilmente que a equação 4.47 é válida se:

$$\Sigma_{xy} - A\Sigma_{yy} = 0 \Rightarrow A = \Sigma_{xy}\Sigma_{yy}^{-1} \tag{4.48}$$

e

$$\mu_x - A\mu_y - b = 0 \Rightarrow b = \mu_x - A\mu_y \tag{4.49}$$

Pode-se notar também que, se os vetores de média μ_x e μ_y forem nulos, o vetor b também será nulo, provando que este termo está envolvido no problema apenas para que se leve em conta as médias dos processos x e y.

Portanto, o estimador $\hat{x}$ que representa a projeção ortogonal de x em $\mathcal{Y}$ é:

$$\begin{aligned}
\hat{x} &= Ay + b = Ay + \mu_x - A\mu_y = \mu_x + A(y - \mu_y) = \\
&= \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y - \mu_y)
\end{aligned} \tag{4.50}$$

que é exatamente igual ao estimador linear de mínima variância, conforme pode ser visto na equação 4.29. Este é um resultado extremamente importante, apresentado na equação 11 do teorema 2 da referência [45]:

O estimador por projeção ortogonal é igual ao estimador linear de mínima variância.

Desta forma, ao se buscar estimadores de mínima variância, pode-se buscar, alternativamente, estimadores por projeção ortogonal. Isto torna os cálculos mais simples, e a partir desta importante propriedade foi feito todo o desenvolvimento do estimador ótimo de estados, batizado como filtro de Kalman.

4.4 Inovações

Antes de se estudar o filtro de Kalman, há ainda um conceito muito importante que deve ser explorado, que é o de **inovações**. Seja um conjunto de vetores $y_1, \ldots, y_N$ de dimensão l. Suponha que exista um conjunto de vetores $\tilde{y}_1, \ldots, \tilde{y}_N$, também de dimensão l, mas linearmente independentes uns dos outros. Sendo assim, a σ-algebra gerada pelos dois conjuntos de vetores será igual e os vetores do segundo conjunto são chamadas de inovações.

Uma ilustração deste conceito é a seguinte: Seja y_1 o vetor [1 0 0] pertencente a $\Re^3$, y_2 o vetor [1 1 0] e y_3 o vetor [1 1 1]. Com o vetor y_1 é possível traçar uma dimensão de $\Re^3$. Como não há mais nada no espaço além do vetor y_1, a inovação trazida por ele é ele mesmo, ou seja, $\tilde{y}_1$ é o vetor [1 0 0]. Ao se adicionar a informação trazida por y_2 fica possível gerar duas dimensões de $\Re^3$. No entanto, y_2 tem um componente dependente de y_1. Sendo assim, define-se $\tilde{y}_2$ como sendo o vetor [0 1 0], ou seja, o vetor apenas com a inovação trazida pelo conhecimento de y_2. Da mesma forma, ao se adicionar o conhecimento de y_3, pode-se traçar todos os vetores de $\Re^3$, embora a única dimensão nova trazida pelo y_3 esteja no terceiro elemento, de forma que $\tilde{y_3}$ é o vetor [0 0 1]. Note que os conjuntos de vetores y_1, y_2 e y_3 formam o mesmo espaço que os vetores $\tilde{y}_1$, $\tilde{y}_2$ e $\tilde{y}_3$, portanto, pode-se afirmar que a mesma σ-algebra é formada pelos dois conjuntos de vetores.

Seja $\mathcal{F}_k = \sigma\{y_i, \quad i = 1, \ldots, k\}$ a σ-álgebra gerada pelos vetores aleatórios gaussianos y_1, y_2,...,y_k. As inovações trazidas por cada vetor y novo são dadas pela subtração entre o valor do vetor novo e o valor que se esperava para o vetor novo, dado que os antigos eram conhecidos, o que algebricamente pode ser escrito como:

$$\begin{array}{rcl} \tilde{y}_1 & = & y_1 - E[y_1|\mathcal{F}_0] = y_1 - E[y_1] \\ \tilde{y}_2 & = & y_2 - E[y_2|\mathcal{F}_1] \\ \vdots & & \vdots \\ \tilde{y}_N & = & y_N - E[y_N|\mathcal{F}_{N-1}] \end{array} \tag{4.51}$$

Como foi visto nas duas seções anteriores, a esperança condicional coincide com a projeção ortogonal no espaço $\mathcal{Y}$ dos vetores y para o caso em que y é um vetor gaussiano. Portanto, pode-se escrever:

$$E[y_k|y_1, y_2, \ldots, y_{k-1}] = a_k + \sum_{i=1}^{k-1} A_{ki} y_i \tag{4.52}$$

Das equações 4.51 e 4.52 tem-se que:

$$\begin{bmatrix} \tilde{y}_1 \\ \tilde{y}_2 \\ \vdots \\ \tilde{y}_k \end{bmatrix} = \begin{bmatrix} I_l & 0 & \ldots & 0 \\ -A_{21} & I_l & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -A_{k1} & -A_{k2} & \ldots & I_l \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} - \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{bmatrix} \tag{4.53}$$

Em que I_l é a matriz identidade de ordem l.

Desta forma, as inovações são também gaussianas, por serem formadas por combinações lineares de vetores gaussianos. Como a matriz que relaciona as inovações $\tilde{y}$ às saídas y é não singular, ou seja, tem inversa, y também pode ser gerado por combinações lineares de $\tilde{y}$, o que demonstra que os dois conjuntos de vetores formam a mesma σ-álgebra.

Como $\tilde{y}_k$ foi definindo de forma a ser independente de qualquer $\tilde{y}_p$, $p < k$, tem-se que $E[\tilde{y}_k|\mathcal{F}_{k-1}] = 0$, o que também implica que $E[\tilde{y}_k] = 0$. Desta forma, a correlação entre $\tilde{y}_k$ e $\tilde{y}_p$ é dada por:

$$E[\tilde{y}_k\tilde{y}_p^T] = E[E[\tilde{y}_k\tilde{y}_p^T|\mathcal{F}_{k-1}]] = E[E[\tilde{y}_k|\mathcal{F}_{k-1}]\tilde{y}_p^T] = 0$$

E pode ser demonstrado de forma análoga que a relação também vale para $p > k$. Desta forma, tem-se que as inovações são independentes, de média nula e descorrelacionadas entre si. Isto mostra definitivamente que os $\tilde{y}$ são inovações do processo y.

4.5 Estimação ótima por projeção ortogonal

Fixados os conceitos da seção anterior, parte-se agora para o problema de estimação ótima de estados por projeção ortogonal. Seja o seguinte modelo em espaço de estado com entradas puramente estocásticas:

$$\begin{cases} x(k+1) &= A(k)x(k) + G(k)u(k) \\ y(k) &= C(k)x(k) \end{cases} \tag{4.54}$$

em que $x \in \Re^n$ é o vetor de estados, $y \in \Re^l$ é o vetor de saídas, as matrizes $A(k)$, $C(k)$ e $G(k)$ são conhecidas, têm dimensões apropriadas e são funções determinísticas do tempo.O vetor $u(k) \in \Re^n$ representa um ruído gaussiano de média nula com a seguinte matriz de covariância:

$$E[u(t)u(s)^T] = I_n\delta(t-s) \tag{4.55}$$

em que I_n é a matriz identidade de ordem n. O estado inicial $x(0)$ é gaussiano, com média $E[x(0)] = \mu_x(0)$, matriz de covariância $\Pi(0)$ e descorrelacionado com os sinais de ruído em qualquer instante de tempo.

Seja $\mathcal{Y}_k = \sigma\{y(0), y(1), \ldots, y(k)\}$ a σ-álgebra gerada pelas saídas até um determinado instante de tempo k. Todas as informações trazidas até um instante de tempo $s < k$ estão contidas na σ-álgebra de k, de forma que $\mathcal{Y}_k$ é uma família crescente de σ-álgebras, pois, a cada novo instante de tempo, novas informações são adicionadas. Com isto, pode ser formulado o problema da estimação, que é o problema encontrar a estimativa de mínima variância $\hat{x}(k+m|k)$ para o vetor de estado $x(k+m)$. Ou seja, o problema de estimação pode ser resumido no problema de se minimizar o seguinte critério J:

$$J = E[||x(k+m) - \hat{x}(k+m|k)||^2] \tag{4.56}$$

Em que $\hat{x}(k+m|k)$ é um elemento da σ-álgebra $\mathcal{Y}_k$. Se $m > 0$, o problema é chamado de predição, pois se está interessado em prever o estado futuro dadas as saídas até o instante de tempo presente, se $m = 0$, o problema é chamado de filtragem, pois o interesse é recuperar o valor do estado

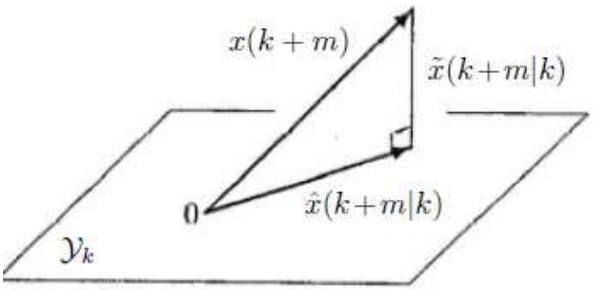

Fig. 4.1: Projeção ortogonal de $x(k+m)$ no espaço $\mathcal{Y}_k$

atual filtrando-o em meio aos ruídos e se $m < 0$, o problema é chamado de refinamento[5], pois a partir de novos dados se quer refinar a estimativa do estado que ocorreu no passado.

Conforme visto anteriormente, a estimativa $\hat{x}(k+m|k)$ que apresenta a menor variância é a esperança condicional α dada por:

$$\hat{x}(k+m|k) = \alpha = E[x(k+m)|\mathcal{Y}_k] \tag{4.57}$$

Se o erro entre a estimativa e o valor real for definido como $\tilde{x}(k+m|k) = x(k+m) - \hat{x}(k+m|k)$, a matriz de covariância do erro é dada por:

$$P(k+m|k) = E[[x(k+m) - \hat{x}(k+m|k)][x(k+m) - \hat{x}(k+m|k)^T]$$

Como foi visto anteriormente, os vetores x e y são gaussianos e $\hat{x}(k+m|k)$ é um vetor contido no espaço formado pelos vetores de observação y, desde o instante inicial até o instante k.

De maneira mais formal, seja $\mathcal{Y}_k$ a variedade formada pelas observações de y conforme apresentado abaixo:

$$\mathcal{Y} = \left\{ b + \sum_{i=1}^{N} A_i y_i \right\} \tag{4.58}$$

Em que $b \in \Re^n$ e $A_i \in \Re^{nXl}$.

Então, como já tratado anteriormente, a estimativa de menor variância $\hat{x}(k+m|k)$ é tal que o erro $\tilde{x}(k+m|k) = x(k+m) - \hat{x}(k+m|k)$ é ortogonal à variedade $\mathcal{Y}_k$. Como provado nas equações 4.30 e 4.31, o estimador é não polarizado.

Na figura 4.1 é ilustrada a interpretação da estimativa como projeção ortogonal do valor real de $x(k+m)$ no espaço $\mathcal{Y}_k$.

[5]O termo em inglês para este tipo de problema é *smoothing*, o que seria traduzido literalmente por alisamento, mas o termo refinamento é mais próprio neste caso

4.6 Algoritmos de predição e filtragem

4.6.1 A função de erro de medida

Seja $e(k)$ o erro entre a saída real e sua estimativa definido como:

$$e(k) = y(k) - E[y(k)|\mathcal{Y}_{k-1}] \tag{4.59}$$

em que $e(0) = y(0) - \mu_y(0)$. Note que e é uma inovação do processo y, conforme definido na equação 4.51.

Como tanto $y(k)$ quanto sua estimativa $E[y(k)|\mathcal{Y}_{k-1}]$ são gaussianos, $e(k)$ também é gaussiano. Ao se aplicar o operador $E[.]$ aos dois lados da equação 4.59 tem-se o seguinte:

$$E[e(k)] = E[y(k)] - E[E[y(k)|\mathcal{Y}_{k-1}]] = E[y(k)] - E[y(k)] = 0 \tag{4.60}$$

Da mesma forma, ao se tomar o valor esperado dos dois lados da equação dado que se conhece $\mathcal{Y}_{k-1}$ tem-se:

$$\begin{aligned} E[e(k)|\mathcal{Y}_{k-1}] &= E[y(k)|\mathcal{Y}_{k-1}] - E[E[y(k)|\mathcal{Y}_{k-1}]|\mathcal{Y}_{k-1}] \\ &= E[y(k)|\mathcal{Y}_{k-1}] - E[y(k)|\mathcal{Y}_{k-1}] = 0 \end{aligned} \tag{4.61}$$

Sejam k e s dois instantes de tempo tais que $k > s$. A esperança condicional $E[e(s)|\mathcal{Y}_{k-1}]$ é igual ao próprio $e(s)$ uma vez que no instante $k-1$ o tempo s já é passado e o valor de $e(s)$ é algo determinístico. Sendo assim, a correlação entre os erros de saída nestes dois instantes de tempo é dada por:

$$E[e(k)e(s)^T] = E[E[e(k)e(s)^T]|\mathcal{Y}_{k-1}] = E[E[e(k)||\mathcal{Y}_{k-1}]e(s)^T] = 0 \tag{4.62}$$

Ou seja, o sinal $e(k)$ é descorrelacionado com $e(s)$ para qualquer $k \neq s$.

Aplicando o segundo termo da equação de estado à definição de $e(k)$, tem-se:

$$\begin{aligned} e(k) &= y(k) - E[y(k)|\mathcal{Y}_{k-1}] = Cx(k) - E[Cx(k)|\mathcal{Y}_{k-1}] = \\ &= Cx(k) - C(k)\hat{x}(k|k-1) = C(k)\tilde{x}(k|k-1) \end{aligned} \tag{4.63}$$

Portanto, a covariância da inovação é dada por:

$$\begin{aligned} E[e(k)e(k)^T] &= E[(C(k)\tilde{x}(k|k-1))(C(k)\tilde{x}(k|k-1))^T] = \\ &= C(k)E[\tilde{x}(k|k-1)\tilde{x}^T(k|k-1)]C(k)^T = \\ &= C(k)P(k|k-1)C(k)^T \end{aligned} \tag{4.64}$$

De forma mais geral, como foi visto que o sinal de erro de medida em um determinado instante de tempo é descorrelacionado com ele mesmo em qualquer outro instante de tempo, pode-se escrever a covariância do processo de inovação como:

$$E[e(k)e(s)^T] = [C(k)P(k|k-1)C(k)^T]\sigma(k,s) \tag{4.65}$$

Sendo que $\sigma(k, s)$ é uma função delta de Kroenecker do tipo $\sigma(k-s)$, ou seja, assume valor 1 para $s = k$ e 0 para $s \neq k$. Como foi demonstrado na equação 4.61, $e(k)$ tem média nula. Portanto, como esta seqüência é gaussiana, ela já é totalmente conhecida pois já se conhecem seus dois primeiros momentos.

4.6.2 Predição de estado um passo a frente

Nesta seção é deduzido o estimador de mínima variância $\hat{x}(k+1|k)$ para o estado $x(k+1)$, dadas as observações da saída $y(k)$ de um determinado modelo até o instante k.

Definindo $e(k)$ como a inovação trazida ao espaço $\mathcal{Y}_{k-1}$ por uma leitura $y(k)$, ou seja:

$$e(k) = y(k) - \hat{E}[y(k)|\mathcal{Y}_{k-1}]$$

tem-se que o espaço $\mathcal{Y}_k$ é a soma direta entre aquilo que já se conhecia, ou seja, $\mathcal{Y}_{k-1}$, e a inovação $e(k)$. Com isto pode-se escrever:

$$\begin{aligned} \hat{x}(k+1|k) &= \hat{E}[x(k+1)|\mathcal{Y}_k] = \hat{E}[x(k+1)|\mathcal{Y}_{k-1} \oplus e(k)] = \\ &= \hat{E}[x(k+1)|\mathcal{Y}_{k-1}] + \hat{E}[x(k+1)|e(k)] \end{aligned} \tag{4.66}$$

Em que $\oplus$ denota a soma direta de subespaços. O significado da expressão acima é que a estimativa $\hat{x}(k+1|k)$ é composta por dois componentes. O primeiro componente está no espaço $\mathcal{Y}_{k-1}$ e o segundo na direção da inovação $e(k)$, trazida com a leitura da saída no instante k. O componente contido em $\mathcal{Y}_{k-1}$ é:

$$\begin{aligned} \hat{E}[x(k+1)|\mathcal{Y}_{k-1}] &= \hat{E}[A(k)x(k) + G(k)u(k)|\mathcal{Y}_{k-1}] = \\ &= A(k)\hat{x}(k|k-1) \end{aligned} \tag{4.67}$$

O termo na direção de $e(k)$ pode ser estimado de forma linear, ou seja, como sendo uma constante que multiplica $e(k)$, conforme equacionado abaixo:

$$\hat{E}[x(k+1)|e(k)] = K(k)e(k) \tag{4.68}$$

Sendo que a constante $K(k)$ pode ser determinada de diversas formas. Como visto anteriormente, se $K(k)$ for definida de forma que a estimativa do estado seja ortogonal ao erro, esta constante implicará na solução de menor variância para o problema de se estimar o estado. A constante que garante esta propriedade é conhecida como **ganho de Kalman**, e será calculada a seguir.

Para que o estimador de estado seja de mínima variância, o erro de da estimativa no subespaço das inovações deve ser ortogonal ao próprio subsepaço, portanto:

$$x(k+1) - \hat{E}[x(k+1)|e(k)] \perp e(k) \Rightarrow x(k+1) - K(k)e(k) \perp e(k)$$

Dois vetores ortogonais têm produto interno nulo. Então, aplicando o produto interno do espaço de Hilbert, tem-se a seguinte relação:

$$
\begin{aligned}
& E[(x(k+1) - K(k)e(k))e(k)^T] = 0 \Rightarrow \\
\Rightarrow \quad & E[x(k+1)e(k)^T] = K(k)E[e(k)e(k)^T] = 0 \Rightarrow \\
\Rightarrow \quad & K(k) = E[x(k+1)e(k)^T](E[e(k)e(k)^T])^{-1}
\end{aligned}
\tag{4.69}
$$

mas o primeiro termo nada mais é que:

$$
\begin{aligned}
E[x(k+1)e(k)^T] &= E[(A(k)x(k) + G(k)u(k))(C(k)\tilde{x}(k|k-1))^T] = \\
&= A(k)E[x(k)\tilde{x}(k|k-1)^T]C(k)^T + \\
&\quad + G(k)E[u(k)\tilde{x}(k|k-1)^T]C(k)^T = \\
&= A(k)E[x(k)\tilde{x}(k|k-1)^T]C(k)^T
\end{aligned}
\tag{4.70}
$$

Uma vez que $u(k)$ e $\tilde{x}(k|k-1)$ são descorrelacionados.

Lembrando que:

$$x(k) = \hat{x}(k|k-1) + \tilde{x}(k|k-1) \qquad \hat{x}(k|k-1) \perp \tilde{x}(k|k-1)$$

temos que:

$$
\begin{aligned}
E[x(k)\tilde{x}(k|k-1)^T] &= E[(\hat{x}(k|k-1) + \tilde{x}(k|k-1))\tilde{x}(k|k-1)^T] = \\
&= E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T] \\
&= P(k|k-1)
\end{aligned}
$$

Portanto:

$$E[x(k+1)e(k)^T] = A(k)P(k|k-1)C(k)^T$$

aplicando esta igualdade e a equação 4.64 à equação 4.69 tem-se que o ganho de Kalman $K(k)$ é dado por:

$$K(k) = A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1} \tag{4.71}$$

Se a matriz $C(k)P(k|k-1)C(k)^T$ for singular, deve-se usar sua pseudoinversa ao invés de sua inversa, que não existe no caso. Desta equação, nota-se que o ganho de Kalman é função das matrizes conhecidas do sistema e da matriz de covariância do erro de estimação de estado $P(k|k-1)$. O cálculo desta covariância é detalhado mais adiante.

Com o valor de $K(k)$ definido, pode-se encontrar o valor da melhor estimativa $\hat{x}(k+1|k)$ para o vetor de estados $x(k+1)$. Da equação 4.66 tem-se que:

$$\begin{aligned}\hat{x}(k+1|k) &= \hat{E}[x(k+1)|\mathcal{Y}_{t-1}] + \hat{E}[x(k+1)|e(k)] = \\ &= A(k)\hat{x}(k|k-1) + K(k)e(k) = \\ &= A(k)\hat{x}(k|k-1) + K(k)[y(k) - C(k)\hat{x}(k|k-1)]\end{aligned} \tag{4.72}$$

Da equação 4.72 nota-se que o ganho de Kalman $K(k)$ é uma ponderação entre o conjunto de informações disponível até o instante k e a nova informação trazida pelo processo de inovação. Uma discussão mais detalhada a respeito desta afirmação pode ser encontrada na referência [33].

Aplicando a primeira das equações de estado à equação acima tem-se que:

$$\begin{aligned}&\hat{x}(k+1|k) = A(k)\hat{x}(k|k-1) + K(k)[y(k) - C(k)\hat{x}(k|k-1)] \Rightarrow \\ \Rightarrow\ & x(k+1) - \tilde{x}(k+1|k) = \\ & = A(k)[x(k) - \tilde{x}(k|k-1)] + K(k)[y(k) - C(k)(x(k) - \tilde{x}(k|k-1))] \Rightarrow \\ \Rightarrow\ & A(k)x(k) + G(k)u(k) - \tilde{x}(k+1|k) = \\ & = A(k)x(k) - A(k)\tilde{x}(k|k-1) + K(k)y(k) - K(k)C(k)x(k) + K(k)C(k)\tilde{x}(k|k-1) \Rightarrow \\ \Rightarrow\ & G(k)u(k) - \tilde{x}(k+1|k) = \\ & = -A(k)\tilde{x}(k|k-1) + K(k)y(k) - K(k)y(k) + K(k)C(k)\tilde{x}(k|k-1) \Rightarrow \\ \Rightarrow\ & \tilde{x}(k+1|k) = [A(k) - K(k)C(k)]\tilde{x}(k|k-1) + G(k)u(k)\end{aligned}$$

O valor esperado do erro $\tilde{x}(k+1|k)$ é o seguinte:

$$E[\tilde{x}(k+1|k)] = [A(k) - K(k)C(k)]E[\tilde{x}(k|k-1)] \tag{4.73}$$

como a condição inicial é $\hat{x}(0) = \mu_x(0)$, então

$$E[\tilde{x}(0)] = E[x(0) - \hat{x}(0)] = \mu_x(0) - \mu_x(0) = 0$$

Como a esperança de $\tilde{x}(k|k-1)$ em qualquer instante k é o produto de uma constante pela esperança no instante de tempo anterior, conforme demonstrado na equação 4.73, e como a esperança do erro de estimaçao é nula para o instante inicial, tem-se que a esperança de $\tilde{x}(k|k-1)$ será nula para qualquer instante de tempo k. Isto significa que o valor esperado para o erro é nulo, ou seja, o estimador de estados é não polarizado. A covariância do erro $\tilde{x}(k+1|k)$, definida por $P(k+1|k)$, é dada por:

$$\begin{aligned} P(k+1|k) &= E[\tilde{x}(k+1|k)\tilde{x}(k+1|k)^T] = \\ &= E[[(A(k)-K(k)C(k))\tilde{x}(k|k-1)+G(k)u(k)]* \\ &\quad *[(A(k)-K(k)C(k))\tilde{x}(k|k-1)+G(k)u(k)]^T] \\ &= (A(k)-K(k)C(k))P(k|k-1)(A(k)-K(k)C(k))^T+ \\ &\quad +G(k)G(k)^T \end{aligned} \tag{4.74}$$

uma vez que o erro de estimação de estado $\tilde{x}(k|k-1)$ e a entrada $u(k)$ são descorrelacionados. No instante inicial, $P(0) = \Pi(0)$, conforme definido anteriormente no início da seção sobre estimação ótima por projeção ortogonal.

Agora é possível não só calcular o estado futuro, mas também a covariância do erro. Quanto menor for a covariância, mais exatas serão as estimativas. Além disto, a covariância do erro é a única matriz envolvida no cálculo do ganho de Kalman que não é uma característica do sistema dinâmico. Desta forma, para se calcular o ganho de Kalman a cada instante de tempo, é necessário que se tenha o valor da matriz $P(k|k-1)$ a cada instante de tempo. Uma das formas de se fazer isto é resolvendo recursivamente a equação 4.74. Para que esta equação dependa apenas das matrizes do modelo conhecido, é necessário que se faça algumas manipulações algébricas, apresentadas a seguir:

$$\begin{aligned} P(k+1|k) &= A(k)P(k|k-1)A(k)^T - A(k)P(k|k-1)C(k)^TK(k)^T- \\ &\quad -K(k)C(k)P(k|k-1)A(k)^T+ \\ &\quad +K(k)C(k)P(k|k-1)C(k)^TK(k)^T + G(k)G(k)^T \end{aligned}$$

substituindo $K(k)$ conforme a equação 4.71 temos:

$$\begin{aligned}
P(k+1|k) &= A(k)P(k|k-1)A(k)^T- \\
&\quad -A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\
&\quad *C(k)P(k|k-1)A(k)^T- \\
&\quad -A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\
&\quad *C(k)P(k|k-1)A(k)^T+ \\
&\quad +A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\
&\quad *C(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\
&\quad *C(k)P(k|k-1)A(k)^T+ \\
&\quad +G(k)G(k)^T
\end{aligned}$$

o segundo, o terceiro e o quarto termos têm o mesmo módulo. Colocando as matrizes $A(k)$ e $A(k)^T$ em evidência em todos os termos, a menos do último, temos finalmente a seguinte expressão:

$$\begin{aligned}
P(k+1|k) &= A(k)[P(k|k-1) - P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\
&\quad *C(k)P(k|k-1)]A(k)^T + G(k)G(k)^T
\end{aligned} \tag{4.75}$$

que é uma equação recursiva de Riccati.

Se o sistema for invariante no tempo, a covariância do erro tenderá a um regime de forma que $P(k+1|k) = P(k|k-1)$. Quando isto ocorrer, a equação 4.75 se torna uma equação álgébrica de Riccati, que tem sua solução explorada em diversas referências como [39], [40], [49], [56] ou [60] e no capítulo 7 desta tese. Na última seção deste capítulo, esta equação é deduzida, assim como todo o problema de estimação de estado de sistemas invariantes no tempo no regime.

A seguir, é feito um resumo dos resultados obtidos nesta seção:

Dado um sistema do tipo

$$\begin{cases} x(k+1) &= A(k)x(k) + G(k)u(k) \\ y(k) &= C(k)x(k) \end{cases}$$

em que as matrizes $A(k)$, $G(k)$ e $C(k)$ são conhecidas e $u(k)$ é um ruído branco de covariância identidade.

Dadas as observações $y(k)$ até o instante k, a estimativa de menor variância $\hat{x}(k+1|k)$ para o estado $x(k+1)$ é a seguinte:

$$\hat{x}(k+1|k) = A(k)\hat{x}(k|k-1) + K(k)e(k)$$

sendo:

$$\hat{x}(0) = E[x(0)]$$

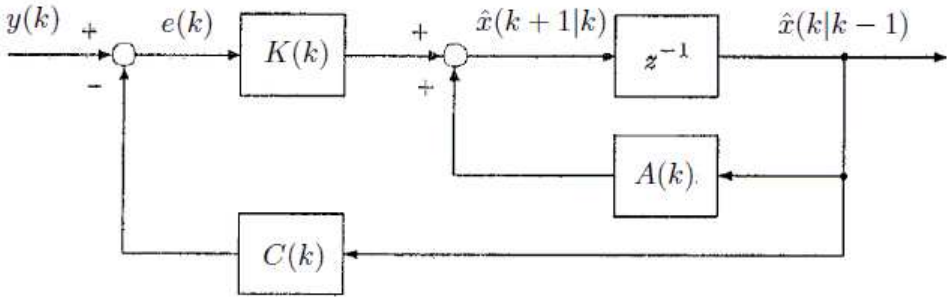

Fig. 4.2: Diagrama de blocos do filtro de Kalman no problema de predição.

em que $e(k)$ *é o processo de inovação definido por:*

$$e(k) = y(k) - C(k)\hat{x}(k|k-1)$$

e $K(k)$ *é o ganho de Kalman, dado pela seguinte equação:*

$$K(k) = A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}$$

em que $P(k|k-1)$ *é a matriz de covariância do erro e obedece à seguinte equação recursiva:*

$$\begin{aligned} P(k+1|k) &= A(k)[P(k|k-1) - P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\ &\quad *C(k)P(k|k-1)]A(k)^T + G(k)G(k)^T \end{aligned}$$

sendo:

$$P(0) = \Pi(0)$$

Na figura 4.2 é apresentado um diagrama de blocos do filtro de Kalman. Da figura nota-se que o filtro tem estrutura similar a um sistema discreto estocástico, sendo que a diferença é que a estimativa de estado é feita por uma realimentação do valor de saída. Com isto, pode-se perceber que o filtro de Kalman nada mais é que um algoritmo recursivo que atualiza as estimativas de estado uma vez que são dadas saídas. Como o algoritmo é recursivo, ele pode ser implementado em um sistema para estimar seus estados em tempo real.

Da definição da inovação $e(k)$, tem-se que o filtro de Kalman pode ser reescrito como:

$$\begin{cases} \hat{x}(k+1) = A(k)\hat{x}(k) + K(k)e(k) \\ y(k) = C(k)\hat{x}(k) + e(k) \end{cases} \tag{4.76}$$

Embora a equação 4.76 tenha um vetor de estados e um ruído diferentes dos da equação 4.54, ambas são representações equivalentes de um mesmo modelo. O modelo apresentado na equação 4.76 é conhecido como modelo inovativo.

4.6.3 Algoritmo de filtragem

Nesta seção é deduzido o algoritmo de filtragem para encontrar a estimativa $\hat{x}(k|k)$ e a matriz de covariância de erro $P(k|k)$ usando raciocínio análogo ao utilizado na seção anterior.

A estimativa do estado $\hat{x}(k|k)$ pode ser vista como a soma da projeção de $x(k)$ no espaço $\mathcal{Y}_{k-1}$, conhecido devido às saídas até o instante $k-1$ com a projeção na direção trazida pela inovação $e(k)$ do instante k, ou seja:

$$\begin{aligned} \hat{x}(k|k) &= \hat{E}[x(k)|\mathcal{Y}_k] = \hat{E}[x(k)|\mathcal{Y}_{k-1} \oplus e(k)] = \\ &= \hat{E}[x(k)|\mathcal{Y}_{k-1}] + \hat{E}[x(k)|e(k)] = \hat{x}(k|k-1) + \hat{E}[x(k)|e(k)] \end{aligned} \tag{4.77}$$

em que:

$$e(k) = y(k) - C(k)\hat{x}(k|k-1) = C(k)x(k) - C(k)\hat{x}(k|k-1) = C(k)\tilde{x}(k|k-1) \tag{4.78}$$

é a inovação no intante k, ou seja, a diferença entre a saída observada no instante k e o valor estimado para a saída dadas as informações até o instante $k-1$.

Da mesma forma como no caso da predição um passo a frente, a estimativa de menor variância é tal que o erro do componente da estimativa devido às inovações é ortogonal ao subespaço definido pelas inovações, ou seja:

$$x(k) - E[x(k)|e(k)] \perp e(k) \Rightarrow x(k) - K_f(k)e(k) \perp e(k) \tag{4.79}$$

Desta forma, o ganho $K_f(k)$ que garante que o estimador seja de mínima variância obedece a seguinte relação:

$$\begin{aligned} & E[(x(k) - K_f(k)e(k))e(k)^T] = 0 \Rightarrow \\ \Rightarrow\ & E[x(k)e(k)^T] - K_f(k)E[e(k)e(k)^T] = 0 \Rightarrow \\ \Rightarrow\ & K_f(k) = E[x(k)e(k)^T](E[e(k)e(k)^T])^{-1} \end{aligned} \tag{4.80}$$

mas[6]

$$\begin{aligned} E[x(k)e(k)^T] &= E[(\hat{x}(k|k-1) + \tilde{x}(k|k-1))(C(k)\tilde{x}(k|k-1))^T] \\ &= E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T]C(k)^T = \\ &= P(k|k-1)C(k)^T \end{aligned}$$

[6]Na expressão abaixo deve-se atentar que, apesar de se estar estimando o ganho ótimo de filtragem, a inovação é definida em função do erro de estimação de estado do problema de predição um passo à frente. Isto é sutil, uma vez que ao se pensar no problema de filtragem poderia se cair no erro de ver a inovação como função do erro do estado estimado $\hat{x}(k|k)$. Isto não é verdade, uma vez que a inovação no instante k é justamente a diferença entre o que foi observado no instante k e o que se esperava daquela variável no instante $k-1$.

O segundo termo da equação do ganho é dado pelo seguinte:

$$\begin{aligned} E[e(k)e(k)^T] &= C(k)E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T]C(k)^T = \\ &= C(k)P(k|k-1)C(k)^T \end{aligned}$$

Portanto:

$$K_f(k) = P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1} \tag{4.81}$$

O estimador ótimo é dado por:

$$\begin{aligned} \hat{x}(k|k) &= \hat{x}(k|k-1) + P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}e(k) = \\ &= \hat{x}(k|k-1) + K_f(k)e(k) \end{aligned} \tag{4.82}$$

o erro de filtragem por:

$$\begin{aligned} \tilde{x}(k|k) &= x(k) - \hat{x}(k|k) = \\ &= x(k) - \hat{x}(k|k-1) - K_f(k)e(k) = \\ &= \tilde{x}(k|k-1) - K_f(k)e(k) \end{aligned} \tag{4.83}$$

e a covariância do erro dada por $P(k|k)$:

$$\begin{aligned} P(k|k) &= E[\tilde{x}(k|k)\tilde{x}(k|k)^T] = \\ &= E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T] - E[\tilde{x}(k|k-1)e(k)^T]K_f(k)^T - \\ &\quad -K_f(k)E[e(k)\tilde{x}(k|k-1)^T] + K_f(k)E[e(k)e(k)^T]K_f(k) = \\ &= E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T] - E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T]C(k)^TK_f(k)^T - \\ &\quad -K_f(k)C(k)E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T] + \\ &\quad +K_f(k)C(k)E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T]C(k)^TK_f(k) = \\ &= P(k|k-1) - P(k|k-1)C(k)^TK_f(k)^T - \\ &\quad -K_f(k)C(k)P(k|k-1) + K_f(k)C(k)P(k|k-1)C(k)^TK_f(k)^T \end{aligned}$$

substituindo o ganho de Kalman de filtragem deduzido na equação 4.81, tem-se o seguinte:

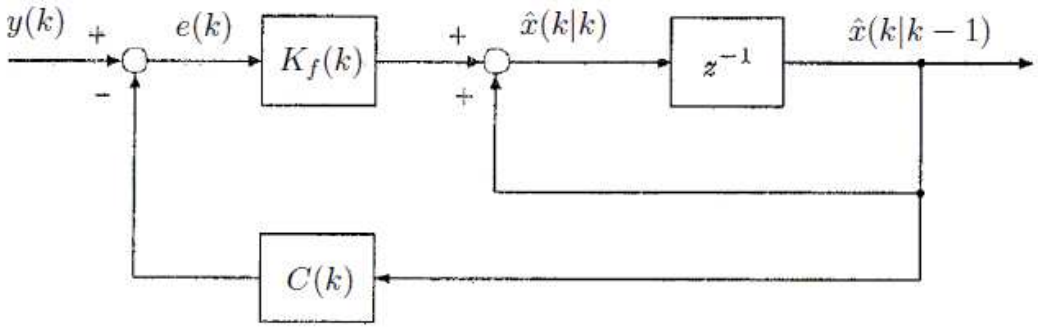

Fig. 4.3: Diagrama de blocos do filtro de Kalman no problema de filtragem.

$$
\begin{aligned}
P(k|k) &= P(k|k-1)- \\
&\quad -P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1)- \\
&\quad -P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1) \\
&\quad +P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1)* \\
&\quad *C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1) = \\
&= P(k|k-1)- \\
&\quad -P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1)
\end{aligned}
\tag{4.84}
$$

Com isto se tem todos os elementos para que se calcule a melhor estimativa para $x(k)$ dadas as observações até o instante k.

Na figura 4.3 é apresentado um diagrama de blocos do filtro de Kalman empregado no problema de filtragem.

Abaixo segue um resumo dos resultados estudados nesta seção:

Dado um sistema do tipo

$$
\begin{cases}
x(k+1) &= A(k)x(k) + G(k)u(k) \\
y(k) &= C(k)x(k)
\end{cases}
$$

em que as matrizes $A(k)$, $G(k)$ e $C(k)$ são conhecidas e $u(k)$ é um ruído branco de covariância identidade.

Dadas as observações $y(k)$ até o instante k, a estimativa de menor variância $\hat{x}(k|k)$ para o estado $x(k)$ é a seguinte:

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K_f(k)e(k)$$

sendo:

$$\hat{x}(0) = E[x(0)]$$

em que $e(k)$ *é o processo de inovação definido por:*

$$e(k) = y(k) - C(k)\hat{x}(k|k-1)$$

e $K_f(k)$ *é o ganho de Kalman de filtragem, dado pela seguinte equação:*

$$K_f(k) = P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}$$

em que $P(k|k-1)$ *é a matriz de covariância do erro e obedece à seguinte equação recursiva:*

$$\begin{aligned} P(k+1|k) &= A(k)[P(k|k-1) - P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\ &\quad *C(k)P(k|k-1)]A(k)^T + G(k)G(k)^T \end{aligned}$$

sendo:

$$P(0) = \Pi(0)$$

e a matriz de covariância do erro de filtragem obedece à seguinte equação:

$$P(k|k) = P(k|k-1) - P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1)$$

4.7 Filtro de Kalman estacionário

Se o sistema sobre o qual se aplica o filtro de Kalman for invariante no tempo, tem-se, a partir do desenvolvimento feito para o sistema variante no tempo, que a estimativa para o vetor de estados no problema de predição um passo a frente é dada por:

$$\hat{x}(k+1|k) = A\hat{x}(k|k-1) + K(k)[y(k) - C\hat{x}(k|k-1)] \tag{4.85}$$

em que a estimativa inicial é $\hat{x}(0) = \mu_x(0)$. O ganho de Kalman é dado por:

$$K(k) = AP(k|k-1)C^T[CP(k|k-1)C^T]^{-1} \tag{4.86}$$

e a covariância do erro $P(k+1|k)$ satisfaz a equação de Riccati e é dada por:

$$\begin{aligned} P(k+1|k) &= AP(k|k-1)A^T - \\ &\quad -AP(k|k-1)C^T[CP(k|k-1)C^T]^{-1}CP(k|k-1)A^T + \\ &\quad +GG^T = \\ &= AP(k|k-1)A^T - AP(k|k-1)C^T[CP(k|k-1)C^T]^{-1} \\ &\quad [CP(k|k-1)C^T][CP(k|k-1)C^T]^{-1}CP(k|k-1)A^T + \\ &\quad +GG^T = \\ &= AP(k|k-1)A^T - K(k)[CP(k|k-1)C^T]K(k)^T + GG^T \end{aligned} \tag{4.87}$$

sendo

$$P(0) = \Pi(0)$$

Supondo que a covariância do erro tenda a um valor constante P quando k é grande o suficiente, tem-se que $P = P(k|k-1) = P(k+1|k)$. Substituindo P na equação 4.87 tem-se:

$$P = APA^T - K[CPC^T]K^T + GG^T$$

Em que o ganho de Kalman estacionário é $K = APC^T(CPC^T)^{-1}$, portanto:

$$\begin{aligned} P &= APA^T - APC^T(CPC^T)^{-1}(CPC^T)(CPC^T)^{-1}CPA^T \\ &= APA^T - APC^T(CPC^T)^{-1}CPA^T \end{aligned} \tag{4.88}$$

uma vez que a matriz (CPC^T) é simétrica. A equação 4.88 é conhecida como equação algébrica de Riccati, e tem sua solução explorada no capítulo 7 desta tese.

Para este caso estacionário, a estimativa do vetor de estados é dada por:

$$\hat{x}(k+1|k) = A\hat{x}(k|k-1) + K[y(k) - C\hat{x}(k|k-1)] = (A - KC)\hat{x}(k) + Ky(k) \tag{4.89}$$

Capítulo 5

Realização e modelagem de séries temporais

Neste capítulo são apresentados os problemas de realização e modelagem de séries temporais discretas no espaço de estado. Também são apresentadas diferentes técnicas conhecidas para se tratar destes dois problemas distintos. No capítulo 2 desta tese, são apresentadas as bases para se entender o presente capítulo. Como pode ser notado ao longo do texto, o equacionamento do problema de realização é muito próximo do equacionamento do filtro de Kalman, tratado no capítulo 4 desta tese. Este problema também pode ser resolvido a partir de sua semelhança com o problema de identificação de sistemas multivariáveis discretos no espaço de estado, tratado no capítulo 3.

Para o entendimento dos problemas estudados neste capítulo, é fundamental diferenciar dois conceitos aparentemente muito próximos. Estes conceitos são a *realização de uma série temporal* e o *problema de realização de uma série temporal*. Séries temporais são sequências de números, ou de vetores, caracterizadas por uma determinada família de covariâncias. Duas sequências diferentes podem ser representações da mesma série temporal, desde que tenham as mesmas covariâncias. A estas duas seqüências diferentes se dá o nome de *realizações da série temporal*. Uma vez que se tenha uma realização de uma determinada série temporal, pode-se querer encontrar um modelo matemático que gere outras realizações da mesma série temporal. Ao problema de encontrar este modelo matemático se dá o nome de *problema de realização de séries temporais*.

Uma vez definidos os conceitos de realização de séries temporais e problema de realização de séries temporais, pode-se definir o conceito de modelagem de séries temporais. Este conceito se resume ao seguinte: supondo que exista uma série temporal e que se conheça uma entrada que, aplicada a um modelo matemático, gera a série, determine qual é este modelo matemático.

Na primeira seção deste capítulo, os métodos de realização de séries temporais serão apresentados e discutidos. A seção seguinte trata do problema de modelagem de séries temporais.

5.1 Realização de séries temporais

O problema de realização de séries temporais no espaço de estado pode ser definido da seguinte maneira:

> *Dado um conjunto de covariâncias de uma determinada série temporal, determinar o modelo matemático em espaço de estado que, quando submetido a uma entrada ruído*

branco, resulta em uma saída com covariâncias o mais próximas o possível das covariâncias da série temporal que se quer realizar.

As técnicas de realização de séries temporais têm em comum o ponto de partida, que é a análise das matrizes de covariância das séries temporais a serem realizadas. A análise de covariâncias da série temporal é algo fundamental, de forma que o problema de realização da série temporal, ou seja, de se encontrar um modelo matemático que, quando submetido a um ruído branco, tenha como saída resultados com covariâncias semelhantes às de uma determinada série temporal, foi enunciado da seguinte forma em [3]:

"Suppose that a linear system is driven by white Gaussian noise and that the variance of the output is known; state the equations that describe the system."

ou ainda, em [35]:

"Given the covariance of a signal process, find a linear system driven by white noise whose output has this covariance."

De fato, se houver boas estimativas para matrizes de covariância de uma série temporal, e se todas essas estimativas forem finitas, é possível encontrar pelo menos um sistema linear que, ao ter como entrada um ruído branco, tem como saída um sinal com a covariância desejada.

Antes da existência da teoria dos modelos em espaço de estado, uma das maneiras de tratar o problema de realização de séries temporais era com o método de fatoração espectral. Este método consiste em encontrar uma função de transferência que, ao ter como entrada um ruído branco, tem como saída o espectro da série temporal com a covariância desejada.

A partir da definição dos modelos em espaço de estado, houve um esforço para se definir maneiras de se provar a existência e de se encontrar modelos no espaço de estado que realizem uma determinada série temporal a partir de sua covariância. A princípio, estes modelos foram desenvolvidos a partir de manipulações diretas da função que gera as matrizes de covariância da série temporal a partir de dois instantes de tempo definidos, ou seja, de funções $f(i,j)$ do tipo:

$$\Lambda(i,j) = f(i,j) = E[y(i)y(j)^T] \tag{5.1}$$

em que $f(i,j)$ é a função geradora de covariâncias da série temporal multivariável $y(k) \in \Re^l$ que, por definição, tem covariância entre os instante i e j a matriz $\Lambda(i,j) \in \Re^{lXl}$.

Posteriormente, foram desenvolvidas técnicas em que o conhecimento da função que gera as matrizes de covariância não seria mais necessário. Nestas técnicas, são usadas apenas as estimativas de covariâncias para determinados atrasos. Isto foi possível a partir do momento em que se observou a semelhança entre as matrizes de covariância e os parâmetros de Markov da modelagem de sistemas determinísticos. Estas técnicas estão resumidas na figura 5.1.

Basicamente, de acordo com a figura 5.1, para se resolver o problema de realização de uma série temporal, parte-se de uma realização desta série. A partir desta realização, encontra-se as covariâncias da série temporal, e tendo as covariâncias, se aplica algum método de realização de séries temporais. Como resultados do método se terá um modelo matemático em espaço de estado e uma covariância para atraso zero de um ruído branco. Ao se encontrar um ruído branco com a covariância para atraso

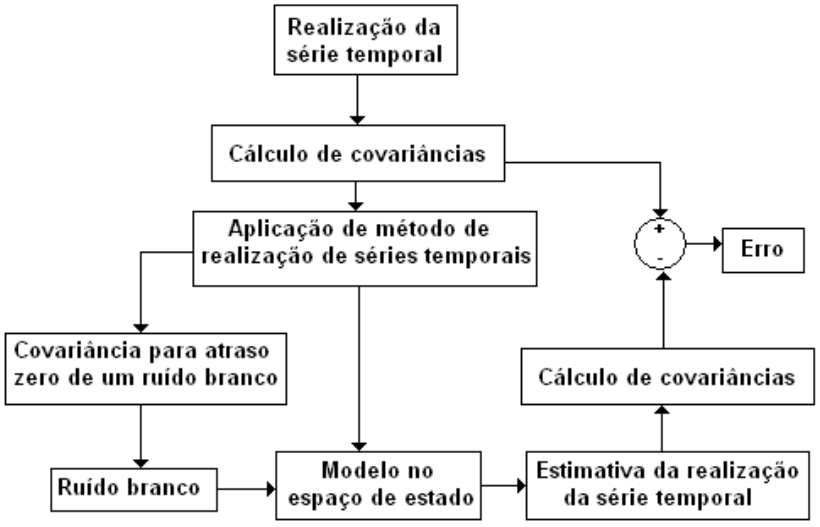

Fig. 5.1: Método de realização de séries temporais

zero obtida e se aplicar este sinal ao modelo, tem-se outra realização da série temporal. A qualidade da solução do problema de realização será melhor quanto menor for a diferença entre as covariâncias da série e as covariâncias da realização obtida com o modelo, ou seja, quanto menor for o erro entre as covariâncias.

Nesta seção, serão apresentados os fundamentos teóricos de algumas das formas de se resolver o problema de realização de séries temporais discretas. A primeira delas, apresentada na seção 5.1.1, é a partir da função semelhante à mostrada na equação 5.1. A base para este tipo de técnica é o filtro de Kalman, explorado no capítulo 4. A outra técnica, que será explorada na seção 5.1.2, é a inspirada na modelagem de sistemas determinísticos, em que a semelhança entre covariâncias e parâmetros de Markov é explorada. A base deste tipo de técnica é introduzida no capítulo 3. Na seção seguinte, é apresentada a técnica de resolução do problema de realização de séries temporais por inequações matriciais lineares, atribuida a Faurre. Na sequência, serão discutidos detalhes do espectro das séries e definidas as funções de transferência que realizam o espectro de uma determinada série temporal. Finalmente, serão mostradas as condições de existência para a solução do problema de realização de séries temporais, concluindo a seção deste capítulo dedicada ao problema de realização de séries temporais.

5.1.1 Realização de séries temporais a partir de funções das covariâncias

Supondo conhecida a função matricial que gera as covariâncias de uma série temporal multivariável a partir de dois instantes de tempo (equação 5.1), é possível que se encontre relações entre esta função e as matrizes do sistema que realiza aquela série temporal. Estas relações foram deduzidas para o caso contínuo em [3]. Nesta referência, é apresentada a relação entre o problema de se encontrar estimativas de mínima variância para o estado e o problema de realização da série temporal.

Em especial, é enfocada a importância da covariância do estado estimado e suas relações com uma determinada covariância de saída.

Poucos anos depois, foi apresentada em [44] a relação entre o problema de estimação de mínima variância do estado e o problema de realização de séries temporais contínuas nas duas direções. No artigo [44] há uma importante discussão a respeito da semelhança entre os problemas e ainda a respeito da não necessidade de se conhecer o modelo em espaço de estado que gera a série temporal para se ter estimativas ótimas do estado. A hipótese de que o modelo que gera a série é conhecido é base para a teoria de estimação de estados [45], embora não seja necessária, como provada em [44].

Seguindo a linha de raciocínio apresentada em [44], a realização de séries temporais discretas a partir da relação entre este problema e o filtro de Kalman, foi apresentada na referência [35], que serve como base para esta seção do capítulo. Nesta referência, é suposto que as matrizes de covariância de uma série temporal podem ser escritas da seguinte forma:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} E(i)F(j) & se \quad i \geq j \\ F(i)^T E(j)^T & se \quad i < j \end{cases} \tag{5.2}$$

sendo $\Lambda(i,j)$ finita e definida positiva para quaisquer i e j.

Esta hipótese é razoável, uma vez que matrizes de covariância são, por definição, definidas positivas e simétricas. Deve-se notar que esta forma de equacionar a covariância de saída é variante no tempo, ou seja, a série temporal é suposta não estacionária. Se houvesse infinitas amostras da série temporal, a covariância poderia ser calculada com exatidão e não seria variante no tempo. No entanto, como o número de amostras disponíveis da série temporal é finito, a covariância não pode ser calculada de forma exata. A maneira de lidar com esta limitação é supor covariâncias variantes no tempo, que implicam em modelos também variantes no tempo.

Realização de séries temporais e o filtro de Kalman

Conforme apresentado no capítulo 4 desta tese, o estimador de estado que minimiza a covariância do erro de saída pode ser encontrado a partir da seguinte situação:

- O modelo em espaço de estado que gera a série temporal a partir do ruído branco é conhecido.
- As matrizes de covariância do ruído branco são conhecidas.
- O objetivo é encontrar a estimativa de mínima variância para o estado a partir das saídas observadas.
- Para estimar o estado encontra-se uma equação linear recursiva que relaciona o estado a ser estimado aos estados passados e ao erro de estimação de saída (inovação) da última iteração, que é totalmente descorrelacionado dos erros anteriores. Ou seja, a inovação é um processo do tipo ruído branco.

No estudo de realização das séries temporais se tem, por outro lado, a seguinte situação:

- Estima-se a covariância de uma determinada série temporal que se quer realizar

- Deseja-se encontrar um modelo que tenha como entrada o ruído branco e que realize aquela covariância.

A princípio, os dois problemas não parecem ter relação entre si, mas ao longo do desenvolvimento deste capítulo a relação entre os dois problemas ficará clara.

Suponha, a princípio, que o sinal $y(k)$ que se quer realizar seja dado pela seguinte equação no espaço de estado:

$$\begin{cases} x(k+1) = A(k+1,k)x(k) + G(k)u(k) \\ y(k) = C(k)x(k) \end{cases} \tag{5.3}$$

em que o vetor $x(k) \in \Re^n$ representa o estado no instante k, $u(k) \in \Re^n$ é um ruído branco com matriz de covariância identidade, $A(k+1,k) \in \Re^{nXn}$ é a matriz de transição do estado k para o estado $k+1$, $G(k) \in \Re^{nXn}$ é uma matriz de transformação e $C(k) \in \Re^{lXn}$ é a matriz que relaciona a saída ao estado. Também é suposto que o vetor de ruído branco é descorrelacionado com o vetor de estado e a matriz de covariância do estado no instante k é definida como $\Pi(k)$.

Este modelo tem algumas peculiaridades que devem ser destacadas. Primeiro, deve-se notar que as matrizes do modelo são variantes no tempo. Desta forma, este modelo não se limita à hipótese de estacionariedade que é adotada nos métodos de modelagem de séries temporais inspirados pelas técnicas de modelagem de sistemas determinísticos. Outra característica importante é que o ruído de entrada é suposto branco e com matriz de covariância identidade. Esta hipótese pode ser feita por causa da presença da matriz de transformação $G(k) \in \Re^{nXn}$, que dá para o ruído branco de covariância unitária um efeito semelhante a um ruído branco de covariância $G(k)G(k)^T$ a cada instante k.

Uma vez que a função de covariância de saída é conhecida e que há um modelo proposto, o objetivo é encontrar uma relação entre as matrizes do modelo e as matrizes de covariância. Para isto, deve-se encontrar a covariância de saída do modelo, que parte da covariância do estado, calculada a seguir:

$$\begin{aligned} \Pi(k+1) &= E[x(k+1)x(k+1)^T] = \\ &= A(k+1,k)E[x(k)x(k)^T]A(k+1,k)^T + G(k)G(k)^T = \\ &= A(k+1,k)\Pi(k)A(k+1,k)^T + G(k)G(k)^T = \end{aligned} \tag{5.4}$$

A covariância da saída do modelo será então

$$\begin{aligned} \Lambda(i,j) &= E[y(i)y(j)^T] = \\ &= C(i)E[x(i)x(j)^T]C(j)^T \end{aligned} \tag{5.5}$$

mas, se $i > j$ temos, da equação de evolução do estado:

$$x(i) = \prod_{k=j}^{i-1} A(k+1,k)x(j) + \sum_{l=j}^{i-1} \prod_{k=l+1}^{i} A(k+1,k)G(l)u(l) \tag{5.6}$$

se $i < j$ temos:

$$x(j) = \prod_{k=i}^{j-1} A(k+1,k)x(i) + \sum_{l=i}^{j-1} \prod_{k=l+1}^{j} A(k+1,k)G(l)u(l) \tag{5.7}$$

definindo:

$$\Phi(i,j) = \prod_{k=j}^{i-1} A(k+1,k) \tag{5.8}$$

como sendo a matriz de transição do estado j para o estado i e voltando à equação da covariância da saída do sistema, lembrando que o estado e o ruído branco de entrada são descorrelacionados, temos, para $i > j$:

$$\begin{aligned} \Lambda(i,j) &= C(i)E[x(i)x(j)^T]C(j)^T \\ &= C(i)\Phi(i,j)\Pi(j)C(j)^T \end{aligned} \tag{5.9}$$

e se $i < j$:

$$\begin{aligned} \Lambda(i,j) &= C(i)E[x(i)x(j)^T]C(j)^T \\ &= C(i)\Pi(i)\Phi(j,i)^T C(j)^T \end{aligned} \tag{5.10}$$

e, obviamente, se $i = j$

$$\begin{aligned} \Lambda(i,j) &= C(i)E[x(i)x(i)^T]C(j)^T \\ &= C(i)\Pi(i)C(i)^T \end{aligned} \tag{5.11}$$

Definindo $\Phi(i,i) = I$, pode-se então escrever a matriz de covariância da saída do sistema como sendo:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} C(i)\Phi(i,j)\Pi(j)C(j)^T & se \quad i \geq j \\ C(i)\Pi(i)\Phi(j,i)^T C(j)^T & se \quad i < j \end{cases} \tag{5.12}$$

comparando esta expressão com a equação 5.2 tem-se que

$$\begin{aligned} E(i) &= C(i)\Phi(i,0) \\ F(i) &= \Phi(0,i)\Pi(i)C(i)^T \end{aligned} \tag{5.13}$$

uma vez que, supondo $i > j$, temos:

$$\begin{aligned}\Phi(i,j) &= \prod_{k=j}^{i-1} A(k+1,k) = \\ &= \prod_{k=0}^{i-1} A(k+1,k) \left[\prod_{k=0}^{j-1} A(k+1,k)\right]^{-1} = \\ &= \Phi(i,0)\Phi(j,0)^{-1} = \\ &= \Phi(i,0)\Phi(0,j)\end{aligned} \tag{5.14}$$

Na passagem acima, foi suposto que as matrizes de transição de estado são inversíveis. Isto sempre pode ser obtido se a ordem n do modelo, que é a dimensão do estado, for respeitada, de forma que a matriz $A(k+1,k)$, e consequentemente as matrizes $\Phi(i,j)$, serão inversíveis.

Para simplificar um pouco a notação, pode-se definir as matrizes $M(k)$ e $N(k)$, tais que:

$$\begin{aligned}M(k) = E(k)\Phi(0,k) = C(k)\Phi(k,0)\Phi(0,k) = C(k) \\ N(k) = \Phi(k,0)F(k) = \Phi(k,0)\Phi(0,k)\Pi(k)C(k)^T = \Pi(k)C(k)^T\end{aligned} \tag{5.15}$$

de forma que a equação de covariância de saída do sistema pode ser reescrita da seguinte maneira:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} M(i)\Phi(i,j)N(j) & se \quad i \geq j \\ N(i)^T\Phi(j,i)^T M(j)^T & se \quad i < j \end{cases} \tag{5.16}$$

As expressões 5.13 (e indiretamente as expressões 5.15), relacionam as matrizes observadas na equação de covariâncias de saída, que é supostamente conhecida, e as matrizes do modelo que se quer encontrar. No entanto, apenas com estas duas equações não é possível determinar a matriz $G(k) \in \Re^{nXn}$ que, como comentado acima, tem relação com a covariância do ruído branco usado como entrada do sistema. Como conclusão, não é possível determinar um modelo semelhante ao da equação 5.3 apenas conhecendo a função geradora de covariâncias na forma da expressão 5.2. O que falta é justamente uma expressão que relacione a covariância de saída conhecida e a covariância de entrada desconhecida. Este problema seria resolvido caso seja tomado um modelo cuja entrada tenha relação com as saídas observadas.

Um modelo em que isto acontece é o de evolução do estado estimado, deduzido na teoria do desenvolvimento do estimador ótimo de estado, também conhecido como filtro de Kalman. O estimador ótimo para o estado no instante k pode ser obtido levando-se em conta os conjuntos de dados entre 0 e vários instantes de tempo, como por exemplo $k+l$, l natural (problema de predição), k (problema de filtragem) ou $k-l$, l natural (problema de refinamento[1]). Em todos estes casos, a entrada da equação do estado estimado será em função da saída observada do sistema. Sendo assim, as covariâncias de estado estimado para todos estes modelos são candidatas a terem relação direta com a covariância da série temporal de saída. Consequentemente, os modelos de evolução de estado resultantes são candidatos a resolverem o problema de realização.

Como ficará claro mais adiante, todas as estimativas de estado implicam em modelos que podem resolver o problema de realização de séries temporais, mas uma delas levará a realizações de fase mínima. A seguir será apresentado o desenvolvimento da solução do problema de realização levando

[1] smoothing

em conta duas estimativas. A primeira é do estado no instante k, dadas as observações até o instante $k-1$ (predição um passo a frente), e a outra é a do estado no instante k, dadas as observações até o próprio instante k (filtragem).

Realização e a predição um passo a frente

Da seção 4.6.2 do capítulo 4 tem-se que, para um modelo descrito pelas equações 5.2, com todas as matrizes conhecidas, a estimativa de menor variância para o estado no instante $k+1$, sendo conhecido o estado no instante k, é dada por:

$$\begin{aligned} \hat{x}(k+1|k) &= A(k)\hat{x}(k|k-1) + K(k)e(k) \\ \hat{x}(0) &= 0 \end{aligned} \tag{5.17}$$

em que

$$K(k) = A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1} \tag{5.18}$$

é o ganho de Kalman sendo:

$$P(k|k-1) = E[\tilde{x}(k+1|k)\tilde{x}(k+1|k)^T] \tag{5.19}$$

a covariância do erro de estimação de estado $\tilde{x}(k|k-1)$, que por sua vez é definido por:

$$\tilde{x}(k+1|k) = x(k+1) - \hat{x}(k+1|k) \tag{5.20}$$

A equação da saída observada é a seguinte:

$$y(k) = C(k)\hat{x}(k|k-1) + e(k) \tag{5.21}$$

em que o processo $e(k)$ é a inovação.

Substituindo a definição de inovação na equação de evolução de estado estimado tem-se o seguinte:

$$\begin{aligned} \hat{x}(k+1|k) &= A(k)\hat{x}(k|k-1) + K(k)[y(k) - C(k)\hat{x}(k|k-1)] = \\ &= [A(k) - K(k)C(k)]\hat{x}(k|k-1) + K(k)y(k) = \\ &= \Phi^*(k+1,k)\hat{x}(k|k-1) + K(k)y(k) \end{aligned} \tag{5.22}$$

em que foi definido $\Phi^*(k+1,k) = [A(k) - K(k)C(k)]$. Desta equação, fica claro que a entrada do modelo inovativo é a saída observada $y(k)$. A partir disto, pode-se inferir que a covariância do processo de saída terá alguma relação com a covariância do estado estimado. Sendo assim, a covariância do estado estimado será calculada.

Observando a ortogonalidade entre estados estimados e o erro de previsão de estado, definindo a covariância do estado estimado como sendo:

$$\Sigma_1(k|k-1) = E[\hat{x}(k|k-1)\hat{x}(k|k-1)^T]$$

e usando a covariância $P(k|k-1)$ do erro de predição de estado, definida na equação 5.19, e a covariância de estado $\Pi(k)$, definida na equação 5.4, tem-se que a seguinte relação é válida:

$$\begin{aligned}\Pi(k) &= E[x(k)x(k)^T] = \\ &= E[(\tilde{x}(k|k-1)+\hat{x}(k|k-1))(\tilde{x}(k|k-1)+\hat{x}(k|k-1)^T)] = \\ &= E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T] + E[\hat{x}(k|k-1)\hat{x}(k|k-1)^T] = \\ &= P(k|k-1)+\Sigma_1(k|k-1)\end{aligned} \tag{5.23}$$

Sendo assim, para o cálculo de $\Pi(k)$ serão calculados os valores de $P(k+1|k)$ e de $\Sigma_1(k|k-1)$.

Da equação 4.75 do capítulo 4 matriz de covariância do erro $P(k|k-1)$ é dada pela seguinte expressão:

$$\begin{aligned}P(k+1|k) &= A(k)[P(k|k-1)-P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1}* \\ &\quad *C(k)P(k|k-1)]A(k)^T+G(k)G(k)^T\end{aligned} \tag{5.24}$$

Substituindo a relação 5.23 em 5.24 tem-se o seguinte:

$$\begin{aligned}\Pi(k+1)-\Sigma_1(k+1|k) &= A(k)[\Pi(k)-\Sigma_1(k|k-1)]A(k)^T- \\ &\quad -A(k)[\Pi(k)-\Sigma_1(k|k-1)]C(k)^T* \\ &\quad *[C(k)[\Pi(k)-\Sigma_1(k|k-1)]C(k)^T]^{-1}* \\ &\quad *C(k)[\Pi(k)-\Sigma_1(k|k-1)]A(k)^T+ \\ &\quad +G(k)G(k)^T\end{aligned} \tag{5.25}$$

rearranjando termos tem-se o seguinte:

$$\begin{aligned}\Pi(k+1)-\Sigma_1(k+1|k) &= A(k)\Pi(k)A(k)^T+G(k)G(k)^T- \\ &\quad -A(k)\Sigma_1(k|k-1)A(k)^T- \\ &\quad -A(k)[\Pi(k)-\Sigma_1(k|k-1)]C(k)^T* \\ &\quad *[C(k)[\Pi(k)-\Sigma_1(k|k-1)]C(k)^T]^{-1}* \\ &\quad *C(k)[\Pi(k)-\Sigma_1(k|k-1)]A(k)^T\end{aligned} \tag{5.26}$$

mas, da equação 5.4 e da definição da matriz de transição de estado, temos que a soma dos dois primeiros termos é igual a $\Pi(k+1)$, portanto:

$$\begin{aligned}\Sigma_1(k+1|k) &= A(k)\Sigma_1(k|k-1)A(k)^T+ \\ &+A(k)[\Pi(k)C(k)^T-\Sigma_1(k|k-1)C(k)^T]* \\ &*[C(k)\Pi(k)C(k)^T-C(k)\Sigma_1(k|k-1)C(k)^T]^{-1}* \\ &*[C(k)\Pi(k)-C(k)\Sigma_1(k|k-1)]A(k)^T\end{aligned} \tag{5.27}$$

Da equação 5.27 nota-se que a covariância do estado estimado obedece a uma equação recursiva que apresenta apenas termos do tipo $C(k)$, $A(k)$ e $C(k)\Pi(k)$. Portanto, se for conhecida a equação que gera as matrizes de covariância de saída na forma 5.2, e se forem estabelecidas as relações apresentadas nas equações 5.13, todos os termos da equação recursiva da evolução do estado estimado são conhecidos. Isto confirma a hipótese levantada anteriormente, em que se afirmava que, se a evolução do estado estimado no modelo inovativo depende da saída, a relação de evolução de sua covariância também dependeria da covariância de saída.

Fazendo a substituição das relacões 5.15 tem-se a seguinte expressão para a evolução da covariância do estado:

$$\begin{aligned}\Sigma_1(k+1|k) &= A(k)\Sigma_1(k|k-1)A(k)^T+ \\ &+A(k)[N(k)-\Sigma_1(k|k-1)M(k)^T]* \\ &*[M(k)N(k)-M(k)\Sigma_1(k|k-1)M(k)^T]^{-1}* \\ &*[N(k)^T-M(k)\Sigma_1(k|k-1)]A(k)^T\end{aligned} \tag{5.28}$$

sendo

$$\Sigma_1(0) = E[\hat{x}(0)\hat{x}(0)^T] = 0$$

de maneira que, se a equação de covariância da série temporal for escrita da forma 5.16, a evolução da covariância do estado estimado é completamente determinada. Deve-se notar que a matriz $A(k)$ é a matriz de transição do estado k para o estado $k+1$, ou seja, $A(k) = \Phi(k+1,k)$.

O ganho de Kalman também pode ser escrito em função dos termos da função de covariância de saída e da matriz de covariância de estado estimado. Para isto, basta tomar a equação 5.18 e fazer as substituições abaixo:

$$\begin{aligned}
K(k) &= A(k)P(k|k-1)C(k)^T[C(k)P(k|k-1)C(k)^T]^{-1} = \\
&= A(k)[\Pi(k) - \Sigma_1(k|k-1)]C(k)^T * \\
&\quad *\{C(k)[\Pi(k) - \Sigma_1(k|k-1)]C(k)^T\}^{-1} = \\
&= A(k)[\Pi(k)C(k)^T - \Sigma_1(k|k-1)C(k)^T] * \\
&\quad *[C(k)\Pi(k)C(k)^T - C(k)\Sigma_1(k|k-1)C(k)^T]^{-1} = \\
&= A(k)[N(k) - \Sigma_1(k|k-1)M(k)^T] * \\
&\quad *[M(k)N(k) - M(k)\Sigma_1(k|k-1)M(k)^T]^{-1}
\end{aligned} \tag{5.29}$$

Desta forma, a partir da covariância do sinal de saída, se tem um modelo que realiza a série temporal. A entrada deste modelo será um ruído branco, com a seguinte covariância para atraso zero:

$$\begin{aligned}
E[e(k)e(k)^T] &= E[(y(k) - C(k)\hat{x}(k|k-1))\,(y(k) - C(k)\hat{x}(k|k-1))^T] = \\
&= C(k)E[(x(k) - \hat{x}(k|k-1))\,(x(k) - \hat{x}(k|k-1))^T]C(k)^T = \\
&= C(k)P(k|k-1)C(k)^T = \\
&= C(k)[\Pi(k) - \Sigma_1(k|k-1)]C(k)^T = \\
&= C(k)\Pi(k)C(k)^T - C(k)\Sigma_1(k|k-1)C(k)^T = \\
&= M(k)N(k) - M(k)\Sigma_1(k|k-1)M(k)^T
\end{aligned} \tag{5.30}$$

portanto, se forem conhecidos $M(k)$ e $N(k)$ a partir da covariância da série temporal que se quer realizar, e com o valor de $\Sigma_1(k)$ calculado a partir da equação 5.28, é possível determinar a covariância da entrada do modelo inovativo para cada instante de tempo.

A demonstração feita acima é resumida a seguir:

Seja $y(k)$ uma série temporal com covariância $\Lambda(i,j)$ semidefinida positiva da seguinte forma:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} M(i)\Phi(i,j)N(j) & se \quad i \geq j \\ N(i)^T\Phi(j,i)^TM(j)^T & se \quad i < j \end{cases}$$

Então, um modelo em espaço de estado que realiza $y(k)$ pode ser escrito da seguinte forma:

$$\begin{cases} \hat{x}(k+1|k) = A(k)\hat{x}(k|k-1) + K(k)e(k), \quad \hat{x}(0) = 0 \\ \\ y(k) = M(k)\hat{x}(k|k-1) + e(k) \end{cases}$$

em que:

$$A(k) = \Phi(k+1,k)$$

$$K(k) = A(k)[N(k) - \Sigma_1(k|k-1)M(k)^T][M(k)N(k) - M(k)\Sigma_1(k|k-1)M(k)^T]^{-1}$$

e a série $e(k)$ *é um ruído branco, cuja covariância para atraso zero é dada por:*

$$E[e(k)e(k)^T] = M(k)N(k) - M(k)\Sigma_1(k|k-1)M(k)^T$$

em que a covariância do estado estimado segue a seguinte equação recursiva:

$$\begin{aligned}\Sigma_1(k+1|k) &= A(k)\Sigma_1(k|k-1)A(k)^T + \\ &+A(k)[N(k) - \Sigma_1(k|k-1)M(k)^T]* \\ &*[M(k)N(k) - M(k)\Sigma_1(k|k-1)M(k)^T]^{-1}* \\ &*[N(k)^T - M(k)\Sigma_1(k|k-1)]A(k)^T\end{aligned}$$

sendo:

$$\Sigma_1(0) = E[\hat{x}(0)\hat{x}(0)^T] = 0$$

Realização e filtragem

Conforme demonstrado na seção 4.6.3 do capítulo 4, a equação de evolução do estado estimado no problema de filtragem é a seguinte:

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K_f(k)e(k), \quad \hat{x}(-1) = 0 \tag{5.31}$$

em que o processo $e(k)$ é a inovação e $K_f(k)$ é o ganho de Kalman para filtragem, dado pela seguinte expressão:

$$K_f(k) = P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1} \tag{5.32}$$

e a covariância do erro de filtragem do estado é dada pela seguinte expressão:

$$P(k|k) = P(k|k-1) - P(k|k-1)C(k)^T(C(k)P(k|k-1)C(k)^T)^{-1}C(k)P(k|k-1) \tag{5.33}$$

Da mesma forma como foi feito para o caso da predição um passo a frente, deve-se encontrar a equação de evolução da covariância do estado estimado para se chegue à relação entre a realização de séries temporais e o algoritmo de filtragem proposto por Kalman. Pode-se chegar a esta equação a partir da equação de covariância do erro de filtragem. Para isto, é preciso reescrever a covariância do erro de filtragem em função da covariância do estado estimado da seguinte maneira:

$$\begin{aligned}\Pi(k) &= E[x(k)x(k)^T] \\ &= E[(\tilde{x}(k|k) + \hat{x}(k|k))(\tilde{x}(k|k) + \hat{x}(k|k))^T] = \\ &= P(k|k) + \Sigma(k|k) \Rightarrow \\ \Rightarrow \quad P(k|k) &= \Pi(k) - \Sigma(k|k)\end{aligned} \tag{5.34}$$

uma vez que o erro é ortogonal ao estado estimado. Analogamente,

$$P(k|k-1) = \Pi(k) - \Sigma(k|k-1) \tag{5.35}$$

substituindo estas relações na equação 5.33 tem-se o seguinte:

$$\begin{aligned} \Pi(k) - \Sigma(k|k) &= \Pi(k) - \Sigma(k|k-1) - (\Pi(k) - \Sigma(k|k-1))C(k)^T * \\ &\quad *(C(k)(\Pi(k) - \Sigma(k|k-1))C(k)^T)^{-1} * \\ &\quad *C(k)(\Pi(k) - \Sigma(k|k-1))) \end{aligned} \tag{5.36}$$

portanto:

$$\begin{aligned} \Sigma(k|k) &= \Sigma(k|k-1) + (\Pi(k) - \Sigma(k|k-1))C(k)^T * \\ &\quad *(C(k)(\Pi(k) - \Sigma(k|k-1))C(k)^T)^{-1} * \\ &\quad *C(k)(\Pi(k) - \Sigma(k|k-1))) \end{aligned} \tag{5.37}$$

Para encontrar a relação de evolução de $\Sigma(k|k)$, é necessário que se relacione esta covariância à covariância $\Sigma(k|k-1)$. Isto pode ser feito ao se observar o seguinte:

$$\hat{x}(k+1|k) = A(k)\hat{x}(k|k) \tag{5.38}$$

de forma que a covariância $\Sigma(k|k)$ se relaciona à covariância $\Sigma(k|k-1)$ da seguinte forma:

$$\begin{aligned} &E[\hat{x}(k+1|k)\hat{x}(k+1|k)^T] = A(k)E[\hat{x}(k|k)\hat{x}(k|k)^T]A(k)^T \Rightarrow \\ &\Rightarrow \Sigma(k+1|k) = A(k)\Sigma(k|k)A(k)^T \end{aligned} \tag{5.39}$$

e consequentemente:

$$\Sigma(k|k-1) = A(k-1)\Sigma(k-1|k-1)A(k-1)^T \tag{5.40}$$

Substituindo a equação 5.40 na equação 5.36 tem-se:

$$\begin{aligned} \Sigma(k|k) &= A(k-1)\Sigma(k-1|k-1)A(k-1)^T + \\ &\quad +(\Pi(k) - A(k-1)\Sigma(k-1|k-1)A(k-1)^T)C(k)^T * \\ &\quad *(C(k)(\Pi(k) - A(k-1)\Sigma(k-1|k-1)A(k-1)^T)C(k)^T)^{-1} * \\ &\quad *C(k)(\Pi(k) - A(k-1)\Sigma(k-1|k-1)A(k-1)^T)) \end{aligned} \tag{5.41}$$

Ao se substituir as expressões de $M(k)$ e $N(k)$ definidos na equação 5.15, que são matrizes conhecidas a partir da covariância da série temporal que se quer realizar, tem-se que a equação de

evolução da covariância do estado estimado é a seguinte:

$$\begin{aligned}\Sigma(k|k) &= A(k-1)\Sigma(k-1|k-1)A(k-1)^T+ \\ &+(N(k)-A(k-1)\Sigma(k-1|k-1)A(k-1)^TM(k)^T)* \\ &*(M(k)N(k)-M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^TM(k)^T)^{-1}* \\ &*(N(k)^T-M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^T)\end{aligned} \tag{5.42}$$

Encontrada a equação de evolução da covariância do estado estimado em função dos termos da covariância da série temporal a ser realizada, é necessário que se reescreva o ganho de Kalman em função da covariância do estado estimado e dos termos da covariância da série temporal a ser realizada. Para isto basta substituir as expressões 5.35, 5.40 e 5.15 na expressão 5.33, o que é feito a seguir:

$$\begin{aligned}K_f(k) &= (\Pi(k)-\Sigma(k|k-1))C(k)^T(C(k)(\Pi(k)-\Sigma(k|k-1))C(k)^T)^{-1}= \\ &= (\Pi(k)-A(k-1)\Sigma(k-1|k-1)A(k-1)^T)C(k)^T* \\ &\quad *(C(k)(\Pi(k)-A(k-1)\Sigma(k-1|k-1)A(k-1)^T)C(k)^T)^{-1}= \\ &= (N(k)-A(k-1)\Sigma(k-1|k-1)A(k-1)^TM(k)^T)* \\ &\quad *(M(k)N(k)-M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^TM(k)^T)^{-1}\end{aligned} \tag{5.43}$$

Desta forma, a partir da covariância do sinal de saída, se tem um modelo que realiza a série temporal. Resta agora calcular a covariância para atraso zero da entrada $e(k)$ a partir dos termos da covariância que se quer realizar e a covariância da estimativa do estado, o que é feito a seguir:

$$\begin{aligned}E[e(k)e(k)^T] &= C(k)E[\tilde{x}(k|k-1)\tilde{x}(k|k-1)^T]C(k)^T= \\ &= C(k)P(k|k-1)C(k)^T= \\ &= C(k)[\Pi(k)-\Sigma(k|k-1)]C(k)^T\end{aligned} \tag{5.44}$$

fazendo a multiplicação e substituindo a expressão 5.39 se tem:

$$\begin{aligned}E[e(k)e(k)^T] &= C(k)\Pi(k)C(k)^T- \\ &\quad -C(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^TC(k)^T=\end{aligned} \tag{5.45}$$

substituindo $M(k)$ e $N(k)$ definidos em 5.15:

$$E[e(k)e(k)^T]=M(k)N(k)-M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^TM(k)^T \tag{5.46}$$

Com isto se tem todas as informações necessárias para se encontrar o modelo que realiza a série temporal.

A demonstração feita acima é resumida a seguir:

Seja $y(k)$ uma série temporal com covariância $\Lambda(i,j)$ semidefinida positiva da seguinte forma:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} M(i)\Phi(i,j)N(j) & se \quad i \geq j \\ N(i)^T\Phi(j,i)^T M(j)^T & se \quad i < j \end{cases}$$

Então, um modelo em espaço de estado que realiza $y(k)$ pode ser escrito da seguinte forma

$$\begin{cases} \hat{x}(k|k) = \hat{x}(k|k-1) + K_f(k)e(k), \quad \hat{x}(0|-1) = 0 \\ \\ y(k) = C(k)\hat{x}(k|k-1) + e(k) \end{cases}$$

em que:

$$\begin{aligned} K_f(k) &= (N(k) - A(k-1)\Sigma(k-1|k-1)A(k-1)^T M(k)^T)* \\ &\quad *(M(k)N(k) - M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^T M(k)^T)^{-1} \end{aligned}$$

e a série $e(k)$ é um ruído branco, cuja covariância para atraso zero é dada por:

$$E[e(k)e(k)^T] = M(k)N(k) - M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^T M(k)^T$$

em que:

$$\begin{aligned} \Sigma(k|k) &= A(k-1)\Sigma(k-1|k-1)A(k-1)^T + \\ &\quad +(N(k) - A(k-1)\Sigma(k-1|k-1)A(k-1)^T M(k)^T)* \\ &\quad *(M(k)N(k) - M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^T M(k)^T)^{-1}* \\ &\quad *(N(k)^T - M(k)A(k-1)\Sigma(k-1|k-1)A(k-1)^T) \end{aligned}$$

sendo:

$$\Sigma(0|0) = 0$$

e

$$A(k) = \Phi(k+1,k)$$

Soluções do problema de realização

Conforme apresentado nas duas seções anteriores, o problema de realização tem pelo menos duas soluções, ou seja, a baseada no algoritmo de predição um passo a frente e a baseada no algoritmo de filtragem. Na verdade, pode-se mostrar que existem infinitas soluções, bastando fazer um procedimento análogo ao feito para solucionar o problema de realização usando o algoritmo de predição um passo a frente, mas ao invés de se partir da predição para apenas um passo a frente, parte-se da estimativa ótima no instante $k+l$, sendo l um número natural qualquer, dadas as saídas observadas até o instante k.

Modelos invariantes no tempo para realização de séries temporais estacionárias

A partir raciocínio desenvolvido acima, é possível também determinar modelos invariantes no tempo para realização de séries temporais estacionárias. Para encontrar os modelos invariantes no tempo, são necessárias estimativas estacionárias das covariâncias da série temporal, que são possíveis de se encontrar a partir de realizações finitas.

Seja uma determinada série temporal estacionária $y(k)$ com a seguinte covariância $\Lambda(i,j)$, separável como produto de duas matrizes $E(i)$ e $F(j)$:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} E(i)F(j) & se \quad i \geq j \\ F(i)^T E(j)^T & se \quad i < j \end{cases} \tag{5.47}$$

Seja agora um modelo estacionário em espaço de estado, com matrizes desconhecidas, cuja saída é a série temporal $y(k)$, da seguinte forma:

$$\begin{cases} x(k+1) = Ax(k) + Gu(k) \\ y(k) = Cx(k) \end{cases} \tag{5.48}$$

em que as matrizes têm dimensões compatíveis e $u(k)$ é um ruído branco com covariância para atraso zero igual à identidade e descorrelacionado com os vetores de estado em qualquer instante de tempo. A covariância do estado deste modelo obedece à seguinte relação:

$$\begin{aligned} \Pi &= E[x(k+1)x(k+1)] \\ &= E[(Ax(k) + Gu(k))(Ax(k) + Gu(k))^T] = \\ &= A\Pi A^T + GG^T \end{aligned} \tag{5.49}$$

e as covariâncias das saídas deste modelo obedecem à seguinte relação:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} CA^{j-i}\Pi C^T & se \quad i \geq j \\ C\Pi(A^{i-j})^T C^T & se \quad i < j \end{cases} \tag{5.50}$$

deve-se notar que a covariância da saída é função apenas da diferença entre os instantes de tempo i e j, e não de seus valores absolutos.

Comparando-se as equações 5.47 e 5.50 e partindo-se do princípio que o modelo apresentado na equação 5.48 de fato tem como saída a série temporal $y(k)$, nota-se são válidas as igualdades $E(k) = CA^k$ e $F(k) = A^{-k}\Pi C^T$. Para simplificar a notação pode-se definir as matrizes $M(k) = E(k)A^{-k} = C = M$ e $N(k) = A^k F(k) = \Pi C^T = N$. Além de simplificar a notação, estas matrizes não dependem do instante de tempo k. A partir destas definições, a covariância do sinal de saída do modelo descrito na equação 5.48 pode ser reescrita da seguinte forma:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} MA^{j-i}N^T & se \quad i \geq j \\ N^T(A^{i-j})^T M^T & se \quad i < j \end{cases} \tag{5.51}$$

Como discutido anteriormente para séries variantes no tempo, um modelo que realiza a série temporal é o de predição ótima um passo a frente. Sendo assim, seja a seguinte predição ótima um passo a frente para o estado de um sistema:

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + Ke(k) \\ \hat{x}(0) &= 0 \end{aligned} \tag{5.52}$$

em que

$$K = APC^T[CPC^T]^{-1} \tag{5.53}$$

é o ganho de Kalman sendo:

$$P = E[\tilde{x}(k)\tilde{x}(k)^T] \tag{5.54}$$

a covariância estacionária do erro de estimação de estado $\tilde{x}(k)$, que por sua vez é definido por:

$$\tilde{x}(k+1) = x(k+1) - \hat{x}(k+1) \tag{5.55}$$

e $e(k)$ é um ruído branco com covariância para atraso zero igual à matriz identidade.

A covariância do estado estimado $\hat{x}$ é definida como Σ, e como pode-se notar da equação de evolução do estado estimado, obedece a seguinte relação:

$$\Sigma = E[\hat{x}(k)\hat{x}(k)^T] = A\Sigma A^T + KK^T \tag{5.56}$$

A equação da saída a partir da previsão ótima de estado é a seguinte:

$$y(k) = C\hat{x}(k) + e(k) \tag{5.57}$$

em que o processo $e(k)$ é a inovação.

Substituindo a definição de inovação na equação de evolução de estado estimado tem-se o seguinte:

$$\begin{aligned} \hat{x}(k+1) &= A\hat{x}(k) + K[y(k) - C\hat{x}(k)] = \\ &= [A - KC]\hat{x}(k) + Ky(k) = \\ &= \Phi^*\hat{x}(k) + Ky(k) \end{aligned} \tag{5.58}$$

em que foi definido $\Phi^* = [A - KC]$. Desta equação, fica claro que a entrada do modelo inovativo é a saída observada $y(k)$. A partir disto, pode-se inferir que a covariância do processo de saída terá alguma relação com a covariância do estado estimado, da mesma forma como foi deduzido para modelos de séries variantes no tempo. Sendo assim, a covariância do estado estimado será calculada.

Observando a ortogonalidade entre estados estimados e o erro de previsão de estado, usando a definição da covariância de estado estimado Σ, usando a covariância P do erro de predição de estado definida na equação 5.54 e a covariância de estado Π, tem-se que a seguinte relação é válida:

$$\begin{aligned}\Pi &= E[x(k)x(k)^T] = \\ &= E[(\tilde{x}(k)+\hat{x}(k))(\tilde{x}(k)+\hat{x}(k)^T)] = \\ &= E[\tilde{x}(k)\tilde{x}(k)^T] + E[\hat{x}(k)\hat{x}(k)^T] = \\ &= P + \Sigma\end{aligned} \tag{5.59}$$

Sendo assim, para o cálculo de Π serão calculados os valores de P e de Σ.

Do filtro de Kalman estacionário, descrito na seção 4.7 desta tese, a matriz de covariância do erro de estado P é dada pela seguinte expressão:

$$P = A[P - PC^T[CPC^T]^{-1}CP]A^T + GG^T \tag{5.60}$$

Substituindo a relação 5.59 em 5.60 se tem o seguinte:

$$\Pi - \Sigma = A[\Pi - \Sigma]A^T - A[\Pi - \Sigma]C^T[C[\Pi - \Sigma]C^T]^{-1}C[\Pi - \Sigma]A^T + GG^T \tag{5.61}$$

rearranjando termos tem-se o seguinte:

$$\begin{aligned}\Pi - \Sigma &= A\Pi A^T + GG^T - A\Sigma A^T - A[\Pi - \Sigma]C^T * \\ &\quad *[C[\Pi - \Sigma]C^T]^{-1}C[\Pi - \Sigma]A^T\end{aligned} \tag{5.62}$$

mas, da equação 5.49 e da definição da matriz de transição de estado, temos que a soma dos dois primeiros termos é igual a Π, portanto:

$$\Sigma = A\Sigma A^T + A[\Pi C^T - \Sigma C^T][C\Pi C^T - C\Sigma C^T]^{-1}[C\Pi - C\Sigma]A^T \tag{5.63}$$

que é uma equação de algébrica de Riccati.

Da equação 5.63, nota-se que a covariância do estado estimado Σ obedece a uma equação recursiva que apresenta apenas termos do tipo C, A e $C\Pi$. Portanto, se for conhecida a equação que gera as matrizes de covariância de saída na forma 5.50, todos os termos da equação recursiva da evolução do estado estimado são conhecidos.

Fazendo a substituição de M e N na equação 5.63 tem-se a seguinte expressão para a evolução da covariância do estado:

$$\Sigma = A\Sigma A^T + A[N - \Sigma M^T][MN - M\Sigma M^T]^{-1}[N^T - M\Sigma]A^T \tag{5.64}$$

Deve-se notar que, se a equação de covariância da série temporal for escrita da forma 5.51, a evolução da covariância do estado estimado apresentada na equação 5.64 é completamente determinada.

O ganho de Kalman também pode ser escrito em função dos termos da função de covariância de saída e da matriz de covariância de estado estimado. Para isto, basta tomar a equação 5.53 e fazer as substituições abaixo:

$$\begin{aligned} K &= APC^T[CPC^T]^{-1} = \\ &= A[\Pi - \Sigma]C^T\{C[\Pi - \Sigma]C^T\}^{-1} = \\ &= A[\Pi C^T - \Sigma C^T][C\Pi C^T - C\Sigma C^T]^{-1} = \\ &= A[N - \Sigma M^T][MN - M\Sigma M^T]^{-1} \end{aligned} \tag{5.65}$$

Desta forma, a partir da covariância do sinal de saída na forma 5.51, se tem um modelo que realiza a série temporal. A entrada deste modelo será um ruído branco com a seguinte covariância para atraso zero:

$$\begin{aligned} E[e(k)e(k)^T] &= E[(y(k) - C\hat{x}(k))\,(y(k) - C\hat{x}(k))^T] = \\ &= CE[(x(k) - \hat{x}(k))\,(x(k) - \hat{x}(k))^T]C^T = \\ &= C[\Pi - \Sigma]C^T = \\ &= C\Pi C^T - C\Sigma C^T = \\ &= MN - M\Sigma M^T \end{aligned} \tag{5.66}$$

portanto, se forem conhecidos M e N a partir da covariância da série temporal que se quer realizar, e com o valor de Σ calculado a partir da equação 5.64, é possível determinar a covariância da entrada a do modelo inovativo para cada instante de tempo.

A demonstração feita acima é resumida a seguir:

Seja $y(k)$ uma série temporal estacionária com covariância $\Lambda(i,j)$ para um determinado intervalo l, semidefinida positiva da seguinte forma:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} MA^{i-j}N & se \quad i \geq j \\ N^T(A^{j-i})^T M^T & se \quad i < j \end{cases}$$

Então, um modelo em espaço de estado que realiza $y(k)$ pode ser escrito da seguinte forma

$$\begin{cases} \hat{x}(k+1) = A\hat{x}(k) + Ke(k), \quad \hat{x}(0) = 0 \\ \\ y(k) = M\hat{x}(k) + e(k) \end{cases}$$

em que:

$$K = A[N - \Sigma M^T][MN - M\Sigma M^T]^{-1}$$

e a série $e(k)$ é um ruído branco, cuja covariância para atraso zero é dada por:

$$E[e(k)e(k)^T] = MN - M\Sigma M^T$$

em que a covariância do estado estimado segue a seguinte equação algébrica de Riccati:

$$\Sigma = A\Sigma A^T + A[N - \Sigma M^T][MN - M\Sigma M^T]^{-1}[N^T - M\Sigma]A^T$$

5.1.2 Realização de séries temporais inspirada na modelagem de sistemas determinísticos

Conforme citado na introdução deste capítulo, além da técnica de realização baseada na função que gera as covariâncias das séries temporais, foi também desenvolvida uma técnica baseada diretamente nas observações da covariância de saída da série temporal que se quer realizar. Esta técnica pôde ser desenvolvida graças à semelhança entre as matrizes de covariância da série temporal e os parâmetros de Markov, definidos na modelagem de sistemas determinísticos no espaço de estado (ver capítulo 3). A grande vantagem desta técnica é que não é necessário que se encontre a função geradora das covariâncias, o que é algo não trivial. A desvantagem da técnica é que os modelos encontrados são invariantes no tempo.

O ponto de partida para o desenvolvimento desta técnica de realização de séries temporais é o modelo que se admite que realiza a série temporal. Dependendo da escolha deste modelo inicial, chega-se a diferentes conclusões e condições de solução do problema. Nesta seção, são apresentados três desenvolvimentos. O primeiro deles parte do modelo de predição ótima de estado. Neste desenvolvimento, se chega a uma condição muito particular, envolvendo uma equação matricial quadrática, demonstrando a dificuldade de se chegar ao modelo de predição ótima a partir deste método. O segundo modelo desenvolvido é o apresentado por Aoki na referência [4]. Com este modelo, é possível encontrar uma equação de Riccati que deve ser satisfeita para que suas matrizes sejam determinadas. Por fim, será apresentado o modelo proporcional da série temporal. O terceiro desenvolvimento levará a uma equação de Lyapunov fundamental na discussão das condições de existência de modelos matemáticos para uma determinada série temporal.

Modelo de predição ótima

Seja uma série temporal estacionária $y(k)$, com uma covariância Λ satisfazendo a seguinte relação:

$$\Lambda(k) = E[y(i)y(i+k)^T] \quad \forall \quad i \tag{5.67}$$

Conforme demonstrado na seção anterior, pode-se considerar que esta série é gerada pelo modelo abaixo:

$$\begin{cases} x(k+1) = Ax(k) + Gu(k) \\ y(k) = Cx(k) \end{cases} \tag{5.68}$$

em que $A \in \Re^{nXn}$ é a matriz de transição de estado, $G \in \Re^{nXm}$ é a matriz que relaciona estado e entradas, $C \in \Re^{lXn}$ é a matriz que relaciona os vetores de saída e de estado, $x(k) \in \Re^n$ é o estado e $u(k)$ é uma entrada ruído branco cuja covariância obedece a seguinte relação:

$$E[u(i)u(i+k)^T] = \begin{cases} I_{lXl} & se \quad k = 0 \\ 0_{lXl} & se \quad k \neq 0 \end{cases} \tag{5.69}$$

Como demonstrado anteriormente, a série temporal pode ser estimada por um modelo de predição ótima em espaço de estado, também definido como modelo inovativo, representado abaixo:

$$\begin{cases} \hat{x}(k+1) = A\hat{x}(k) + Ke(k) \\ y(k) = C\hat{x}(k) + e(k) \end{cases} \tag{5.70}$$

que nada mais é que a junção das equações 5.52 e 5.57 já apresentadas anteriormente, em que $\hat{x}(k) \in \Re^n$ é o estado estimado, $K \in \Re^{nXl}$ é o ganho de Kalman, conforme descrito anteriormente, $C \in \Re^{lXn}$ é uma matriz que relaciona estado e saída e $e(k)$ é um ruído branco que obedece a seguinte equação de covariância:

$$E[e(i)e(i+k)^T] = \begin{cases} I_{lXl} & se \quad k = 0 \\ 0_{lXl} & se \quad k \neq 0 \end{cases} \tag{5.71}$$

em que I_{lXl} e 0_{lXl} são respectivamente a identidade e a matriz nula do espaço $\Re^{lXl}$. Este ruído branco é descorrelacionado com estados reais e estimados.

Conforme demonstrado anteriormente, a covariância da saída do modelo $y(k)$ obedece a seguinte relação:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} CA^{j-i}\Pi C^T & se \quad i \geq j \\ C\Pi(A^{i-j})^T C^T & se \quad i < j \end{cases} \tag{5.72}$$

Definindo $k = j - i$, o que pode ser feito devido à hipótese de estacionariedade da série temporal, a covariância da saída será:

$$\Lambda(i,i+k) = \Lambda(k) = E[y(i)y(i+k)^T] = \begin{cases} CA^k\Pi C^T & se \quad k \geq 0 \\ C\Pi(A^{-k})^T C^T & se \quad k < 0 \end{cases} \tag{5.73}$$

em que $\Pi = E[x(k)x(k)^T]$ é a covariância de estado real do sistema e obedece a seguinte relação:

$$\Pi = A\Pi A^T + GG^T \tag{5.74}$$

conforme deduzido na equação 5.49.

A covariância de saída também pode ser deduzida em função da covariância dos estados estimados $\Sigma(k)$, conforme demonstrado abaixo por indução. A covariância da saída para $k = 0$ em função dos estados estimados é a seguinte:

$$\begin{aligned} \Lambda(0) &= E[y(k)y(k)^T] \\ &= E[(C\hat{x}(k) + e(k))(C\hat{x}(k) + e(k))^T] = \\ &= CE[\hat{x}(k)\hat{x}(k)^T]C^T + E[e(k)e(k)^T] = \\ &= C\Sigma C^T + I_{nXn} \end{aligned} \tag{5.75}$$

a covariância da saída para atraso $i = 1$ é a seguinte:

$$\begin{aligned}
\Lambda(1) &= E[y(k+1)y(k)^T] \\
&= E[(C\hat{x}(k+1)+e(k+1))(C\hat{x}(k)+e(k))^T] = \\
&= E[(C(A\hat{x}(k)+Ke(k))+e(k+1))(C\hat{x}(k)+e(k))^T] = \\
&= C[AE[\hat{x}(k)\hat{x}(k)^T]C^T + KE[e(k)e(k)^T]] = \\
&= C[A\Sigma C^T + K]
\end{aligned} \tag{5.76}$$

a covariância da saída para atraso $i = 2$ é a seguinte:

$$\begin{aligned}
\Lambda(2) &= E[y(k+2)y(k)^T] \\
&= E[(C\hat{x}(k+2)+e(k+2))(C\hat{x}(k)+e(k))^T] = \\
&= E[(C(A\hat{x}(k+1)+Ke(k+1))+e(k+2))(C\hat{x}(k)+e(k))^T] = \\
&= E[(C(A(A\hat{x}(k)+Ke(k))+Ke(k+1))+e(k+2))(C\hat{x}(k)+e(k))^T] = \\
&= C[A^2E[\hat{x}(k)\hat{x}(k)^T]C^T + KE[e(k)e(k)^T]] = \\
&= CA[A\Sigma C^T + K]
\end{aligned} \tag{5.77}$$

e por indução se prova que para o atraso $i \neq 0$, a covariância da saída em função da covariância do estado estimado segue a seguinte relação:

$$\Lambda(i) = E[y(k+i)y(k)^T] = CA^{i-1}[A\Sigma C^T + K] \tag{5.78}$$

A covariância entre o estado estimado no instante $k+1$ e a saída no instante k definida como M_1 é dada pela seguinte relação:

$$\begin{aligned}
M_1 = E[\hat{x}(k+1)y(k)^T] &= E[(A\hat{x}(k)+Ke(k))(C\hat{x}(k)+e(k))^T] = \\
&= AE[\hat{x}(k)\hat{x}(k)^T]C^T + KE[e(k)e(k)^T] = \\
&= A\Sigma C^T + K
\end{aligned} \tag{5.79}$$

Portanto, as covariâncias de saída para atrasos não nulos podem ser escritas em função da covariância M_1 da seguinte forma:

$$\Lambda(i) = E[y(k+i)y(k)^T] = CA^{i-1}M_1 \tag{5.80}$$

e com a equação 5.75, a covariância de saída pode ser escrita da seguinte forma reduzida:

$$\Lambda(i) = E[y(k)y(k+i)^T] = \begin{cases} C\Sigma C^T + I_{nXn} & se \quad i = 0 \\ CA^{i-1}M_1 & se \quad i > 0 \end{cases} \tag{5.81}$$

A partir desta expressão de covariância, pode-se partir para o método de realização. Na próxima seção será apresentado um modelo alternativo proposto por Aoki para o desenvolvimento de seu método de realização de séries temporais

Modelo de Aoki

O modelo do qual Aoki parte para o desenvolvimento de seu método, baseado no método de identificação de sistemas proposto por Ho e Kalman [42] é o mesmo utilizado na predição ótima, ou seja,

$$\begin{cases} \hat{x}(k+1) = A\hat{x}(k) + Ke(k) \\ y(k) = C\hat{x}(k) + e(k) \end{cases} \tag{5.82}$$

em que $\hat{x}(k) \in \Re^n$ é o estado estimado, A é a matriz de transição de estado, $K \in \Re^{nXl}$ é o ganho de Kalman, conforme descrito anteriormente, $C \in \Re^{lXn}$ é uma matriz que relaciona estado e saída e $e(k)$ é um ruído branco. A diferença entre o modelo desenvolvido por Aoki e o modelo puramente de predição ótima é que o ruído branco de entrada não tem uma covariância para atraso zero identidade, mas sim com um valor Δ, ou seja, $e(k)$ é tal que:

$$E[e(i)e(i+k)^T] = \begin{cases} \Delta \quad se \quad k = 0 \\ 0_{lXl} \quad se \quad k \neq 0 \end{cases} \tag{5.83}$$

em que 0_{lXl} é a matriz nula do espaço $\Re^{lXl}$. Este ruído branco é descorrelacionado com estados reais e estimados. Por um desenvolvimento semelhante ao feito na subseção anterior, é possível mostrar que a covariância Σ do estado estimado $\hat{x}(k)$ é dada pela seguinte expressão:

$$\Sigma = A\Sigma A^T + K\Delta K^T \tag{5.84}$$

e que a covariância entre estado estimado um passo a frente e saída, definida como M_2, é dada pela expressão:

$$E[\hat{x}(k+1)y(k)^T] = M_2 = A\Sigma C^T + K\Delta \tag{5.85}$$

e que a covariância de saída deste modelo é dada pela expressão:

$$\Lambda(i) = E[y(k)y(k+i)^T] = \begin{cases} C\Sigma C^T + \Delta \quad se \quad i = 0 \\ CA^{i-1}M_2 \quad se \quad i > 0 \end{cases} \tag{5.86}$$

Esta equação será utilizada mais adiante.

Mais detalhes a respeito do algoritmo proposto por Aoki e dos métodos de subespaços para identificação podem ser encontrados nas referências [6], [11], [5] e [40]. No artigo [59], além de se apresentar o método proposto por Aoki é também apresentada uma proposta para realizar uma série temporal com um método similar ao proposto por Aoki mas com um modelo variante no tempo.

Modelo proporcional

Pode-se também partir da hipótese que uma determinada série temporal $y(k)$ qualquer é simplesmente proporcional a um processo estocástico markoviano, ou seja:

$$\begin{cases} x(k+1) = Ax(k) + Gu(k) \\ y(k) = Cx(k) \end{cases} \tag{5.87}$$

em que $x(k) \in \Re^n$ é o processo estocástico markoviano, $G \in \Re^{nXl}$ é uma matriz estacionária, $C \in \Re^{lXn}$ é a matriz de proporcionalidade entre a série temporal e o processo markoviano e $u(k)$ é um ruído branco de covariância para atraso zero igual a identidade.

Sendo Π a covariância do estado, tem-se a já demonstrada equação de Lyapunov:

$$\Pi = A\Pi A^T + GG^T \tag{5.88}$$

e a covariância da série temporal $y(k)$ é dada por:

$$\Lambda(i) = E[y(k)y(k+i)^T] = \begin{cases} C\Pi C^T & se \quad i = 0 \\ CA^i\Pi C^T & se \quad i > 0 \end{cases} \tag{5.89}$$

Definindo $M_3 = E[x(k+1)y(k)^T] = A\Pi C^T$, pode-se reescrever 5.89 da seguinte maneira:

$$\Lambda(i) = E[y(k)y(k+i)^T] = \begin{cases} C\Pi C^T & se \quad i = 0 \\ CA^{i-1}M_3 & se \quad i > 0 \end{cases} \tag{5.90}$$

A partir desta identidade será demonstrado que é possível determinar as matrizes M, N e A a partir de amostras da covariância $\Lambda(i)$.

Covariâncias e os parâmetros de Markov

O desenvolvimento feito a seguir é válido para os três modelos definidos, ou seja, o modelo de predição ótima, o modelo de Aoki e o modelo proporcional. Definindo $y^+(k+1)$ como a matriz coluna em que são empilhadas as saídas futuras do instante $k+1$ em diante e $y^-(k)$ a matriz coluna de saídas passadas até o instante k, teremos que a matriz de covariância entre as duas matrizes definidas é a seguinte:

$$\begin{aligned} E[y^+(k+1)y^-(k+1)^T] &= E\left[\begin{bmatrix} y(k+1) \\ y(k+2) \\ y(k+3) \\ \vdots \end{bmatrix} \begin{bmatrix} y(k) & y(k-1) & y(k-2) & \dots \end{bmatrix}\right] \\ &= \begin{bmatrix} \Lambda(1) & \Lambda(2) & \Lambda(3) & \cdots \\ \Lambda(2) & \Lambda(3) & \Lambda(4) & \cdots \\ \Lambda(3) & \Lambda(4) & \Lambda(5) & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \end{aligned} \tag{5.91}$$

ou ainda:

$$E[y^+(k+1)y^-(k+1)^T] = \begin{bmatrix} CM_* & CAM_* & CA^2M_* & \cdots \\ CAM_* & CA^2M_* & CA^3M_* & \cdots \\ CA^2M_* & CA^3M_* & CA^4M_* & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{5.92}$$

em que M_* pode ser M_1, M_2 ou M_3.

Definindo a matriz Ω como sendo uma matriz semelhante à matriz de atingibilidade, mas contendo a matriz M_*, ou seja:

$$\Omega = \begin{bmatrix} M_* & AM_* & A^2M_* & \cdots \end{bmatrix} \tag{5.93}$$

e lembrando da matriz de observabilidade da mesma forma que foi definida para o sistema determinístico, ou seja:

$$\mathcal{O} = \begin{bmatrix} C \\ CA \\ CA^2 \\ \vdots \end{bmatrix} \tag{5.94}$$

temos que:

$$E[y^+(k+1)y^-(k+1)^T] = \mathcal{O}\Omega \tag{5.95}$$

Portanto, se a covariância entre saídas passadas e futuras for calculada, é possível que se encontre uma matriz que pode ser decomposta de forma que se encontre as matrizes de observabilidade e a matriz semelhante à de atingibilidade que foi definida como Ω. Esta é a base para o método de realização de séries temporais baseada na modelagem de sistemas determinísticos. Deve-se notar que, a partir desta observação, não é necessário que se conheça uma função que gere as matrizes de covariância da série temporal, ao contrário do que ocorre com o método de realização apresentado anteriormente.

Método de realização

A partir da matriz formada pelos blocos de covariâncias da saída, pode-se obter algumas das matrizes características que realizam o sistema, seguindo procedimento análogo ao usado para se encontrar as matrizes que modelam um sistema determinístico via decomposição da matriz $\mathcal{O}\Omega$. A partir das matrizes calculadas com a decomposição, é possível que se chegue a uma equação de Riccati que tem como solução a matriz de covariância de estados. Com o cálculo desta matriz, é possível que se calcule todas as outras matrizes envolvidas no problema. Este procedimento proposto por Aoki é detalhado a seguir:

Seja a decomposição em valores singulares da matriz de covariâncias de saídas:

$$E[y^+(k+1)y^-(k+1)^T] = U\Sigma V^T = \mathcal{O}\Omega \tag{5.96}$$

portanto, pode-se afirmar que:

$$\mathcal{O} = U\Sigma^{\frac{1}{2}} \quad \Omega = \Sigma^{\frac{1}{2}}V^T \tag{5.97}$$

Neste caso, também é definida uma matriz $E[y^+(k+1)y^-(k+1)^T]_\uparrow$ como sendo a matriz $E[y^+(k+1)y^-(k+1)^T]$ deslocada um bloco linha para cima, ou seja:

$$E[y^+(k+1)y^-(k+1)^T]_\uparrow = \begin{bmatrix} CAM_* & CA^2M_* & CA^3M_* & \cdots \\ CA^2M_* & CA^3M_* & CA^4M_* & \cdots \\ CA^3M_* & CA^4M_* & CA^5M_* & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} = \mathcal{O}A\Omega \tag{5.98}$$

portanto:

$$A = \mathcal{O}^\dagger E[y^+(k+1)y^-(k+1)^T]_\uparrow \Omega^\dagger \tag{5.99}$$

e, da mesma forma que foi feito para os sistemas determinísticos, as matrizes C e M_* são calculadas ao se observar que o primeiro bloco coluna de $E[y^+(k+1)y^-(k+1)^T]$ é igual a $\mathcal{O}M_*$ e que o primeiro bloco linha desta matriz é igual a $C\Omega$, portanto:

$$\begin{aligned} M_* &= \mathcal{O}^\dagger E[y^+(k+1)y^-(k+1)^T](:,1) \\ C &= E[y^+(k+1)y^-(k+1)^T](1,:)\Omega^\dagger \end{aligned} \tag{5.100}$$

sendo que estas matrizes também poderiam ser obtidas dos primeiros blocos das matrizes $\mathcal{O}$ e Ω.

Determinadas as matrizes A, M_* e C, ainda é necessário para que se determine os modelos definidos acima para a série temporal:

- Determinar a matriz K para o modelo de predição ótima apresentado na equação 5.70
- Determinar as matrizes K e Δ para o modelo de Aoki apresentado nas equações 5.82 e 5.83.
- Determinar a matriz G do modelo proporcional apresentado na equação 5.87

Estas questões são desenvolvidas a seguir.

Determinação de K no modelo de predição ótima

Ao se isolar K na equação 5.79 e tem-se a seguinte expressão:

$$K = M_1 - A\Sigma C^T \tag{5.101}$$

ao se substituir esta expressão na equação 5.56 tem-se a seguinte relação:

$$\Sigma = A\Sigma A^T + (M_1 - A\Sigma C^T)(M_1 - A\Sigma C^T)^T \tag{5.102}$$

que é uma equação matricial quadrática que, pelo menos em teoria, permite o cálculo da matriz Σ, uma vez que todas as outras matrizes presentes na equação (A, M_1, C) são conhecidas. Calculada a matriz Σ, o cálculo de K é direto, completando a realização da série temporal.

Além desta equação ser satisfeita, as matrizes C e Σ encontradas devem ser tais que a equação 5.81 também seja válida, ou seja:

$$\Lambda(0) - C\Sigma C^T = I$$

Em resumo, Σ se torna a solução de um sistema de duas equações matriciais, sendo que uma delas é quadrática. Provar a existência desta solução e as condições que devem ser satisfeitas pelas matrizes A, M_1, C e $\Lambda(0)$ para que a solução exista é um trabalho em aberto que fica sugerido por esta tese.

Determinação de K e Δ no modelo de Aoki

Ao se isolar K na equação 5.85 e Δ na primeira relação apresentada na equação 5.86, se tem as seguintes expressões:

$$K = (M_2 - A\Sigma C^T)\Delta^{-1} \tag{5.103}$$

$$\Delta = \Lambda(0) - C\Sigma C^T \tag{5.104}$$

Substituindo-se estas expressões na equação 5.84, encontra-se a seguinte equação de Riccati:

$$\Sigma = A\Sigma A^T + (M_2 - A\Sigma C^T)(\Lambda(0) - C\Sigma C^T)^{-1}(M_2 - A\Sigma C^T)^T \tag{5.105}$$

em que as matrizes A, M_2 e C são conhecidas a partir da decomposição das matrizes bloco de covariância, conforme demonstrado acima. Sendo assim, ao se encontrar um determinado Σ que satisfaz esta equação de Riccati, pode-se voltar às equações 5.103 e 5.104 determinando-se o valor de K e Δ completando o modelo que realiza a série temporal.

A existência de soluções para a equação de Riccati é uma questão que depende das matrizes obtidas na decomposição das matrizes bloco de covariâncias. No entanto, mesmo que as condições não sejam satisfeitas por matrizes determinadas a partir da observação de séries temporais reais, soluções aproximadas podem ser encontradas a partir de um método desenvolvido neste trabalho, conforme apresentado no capítulo 7. De qualquer forma, o problema de encontrar uma solução para a equação de Riccati 5.105 é mais conhecido e explorado que o problema de se determinar uma solução para o sistema que envolve a equação quadrática matricial 5.102.

Determinação de G no modelo proporcional

A determinação de G não é diretamente possível a partir das equações do modelo proporcional e das matrizes C, A e M_3, obtidas pela decomposição da matriz bloco de covariâncias. No entanto, o desenvolvimento do modelo proporcional leva a uma discussão importante a respeito da existência de modelos que realizam uma série temporal.

Dada uma matriz A a partir da decomposição das matrizes bloco de covariâncias, as matrizes Π e G devem ser tais que a equação de Lyapunov 5.88 seja satisfeita, ou seja:

$$\Pi = A\Pi A^T + GG^T$$

Definindo $Q = GG^T$ tem-se que para existir solução para o problema de realização pelo menos as seguintes condições devem ser satisfeitas:

- Deve existir uma matriz Π simétrica, positiva definida.
- Deve existir uma matriz Q simétrica, positiva definida.

- As matrizes Π e Q devem ser tais que satisfaçam a equação de Lyapunov $\Pi = A\Pi A^T + Q$

A primeira condição se deve ao fato de Π ser, por definição, uma matriz de covariância. A segunda condição se deve ao fato de Q ser uma matriz obtida como produto de duas matrizes reais e a terceira condição é a própria equação de Lyapunov. Maiores detalhes a respeito da existência ou não de soluções para o problema de realização de séries temporais serão introduzidos na seção 5.1.6 deste capítulo;

5.1.3 Solução da equação algébrica de Riccati

O método proposto por Aoki [4] para solução da equação algébrica de Riccati é baseado na solução proposta por Vaughan [60] e posteriormente reforçada por Laub [48], conforme será brevemente apresentado a seguir. Esta equação também pode ser resolvida de forma paralela, conforme descrito na tese [55].

Seja F uma matriz nXn definida da seguinte forma:

$$F = (A - M_2\Lambda^{-1}(0)C)^T \tag{5.106}$$

A partir desta definição, a equação de Riccati (5.105) pode ser reescrita da seguinte maneira:

$$\Sigma = F^T\Sigma F - F^T\Sigma C^T(\Lambda(0) + C\Sigma C^T)^{-1}C\Sigma F + M_2\Lambda^{-1}(0)M_2^T \tag{5.107}$$

ou

$$\Sigma = F^T\Sigma F - F^T\Sigma G_1(G_2 + G_1^T\Sigma G_1)^{-1}G_1^T\Sigma F + H \tag{5.108}$$

em que

$$\begin{aligned} G_1 &= C^T \\ G_2 &= \Lambda(0) \\ H &= M_2\Lambda^{-1}(0)M_2^T \end{aligned} \tag{5.109}$$

Sejam G e Z as seguintes matrizes:

$$\begin{aligned} G &= G_1G_2^{-1}G_1^T \\ Z &= \begin{bmatrix} F + GF^{-T}H & -GF^{-T} \\ -F^{-T}H & F^{-T} \end{bmatrix} \end{aligned} \tag{5.110}$$

A matriz Z pode ser reescrita da seguinte maneira, a partir da decomposição de Schur:

$$U^TZU = S \tag{5.111}$$

em que S é uma matriz $2n$X$2n$. Se S é dividida nos seguintes blocos nXn:

$$S = \begin{bmatrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{bmatrix} \tag{5.112}$$

Então $\Sigma = S_{21}S_{11}^{-1}$ é uma solução da equação algébrica de Riccati (para mais detalhes ver [46]).

A matriz Σ deve ser simétrica e com autovalores positivos reais para que seja possível resolver a equação algébrica de Riccati. Isto sempre ocorre caso as seguintes hipóteses sejam válidas:

- $H = H^T \geq 0$
- $G_2 = G_2^T \geq 0$
- $\{F, G_1\}$ é um par estabilizável
- $\{C_H, F\}$ é detectável, em que $C_H^T C_H = H$ e o posto de C_H é igual ao posto de H
- F é inversível

Uma vez que o conjunto de dados coletados para determinar as matrizes constantes da equação de Riccati (ou seja, as matrizes A, M_2, C e $\Lambda(0)$) é finito, algumas das hipóteses acima podem não ser satisfeitas e a solução determinada pelo método decomposição de Schur não é válida. Nesta pesquisa, foi desenvolvido um método para determinar a solução da equação algébrica de Riccati nestes casos. Este método é apresentado no capítulo 7 desta tese.

5.1.4 Realização de séries temporais por LMIs

Outra abordagem para a solução do problema de realização de séries temporais foi proposta por Faurre [31], [32]. Neste método, parte-se do princípio que as séries temporais são estacionárias. A partir desta abordagem é possível entender melhor as questões de ordem do sistema e minimalidade, como será visto a seguir.

Seja uma série temporal estacionária multivariável $y(k)$ de dimensão l, cujas matrizes de covariância sejam função apenas do intervalo entre os instantes de tempo em que as covariâncias são tomadas. A hipótese inicial do método é que é possível realizar a série temporal com um modelo markoviano no espaço de estado, que tem como entradas ruídos brancos descorrelacionados, dado pelas relações abaixo:

$$\begin{cases} \hat{x}(k+1) = A\hat{x}(k) + w(k) \\ y(k) = C\hat{x}(k) + v(k) \end{cases} \tag{5.113}$$

em que $A \in \Re^{nXn}$ e $C \in \Re^{lXn}$ são matrizes do modelo a serem determinadas e $w(k) \in \Re^n$ e $v(k) \in \Re^l$ são ruídos brancos descorrelacionados, cuja covariância é dada pela seguinte expressão:

$$E\left[\begin{bmatrix} w(t) \\ v(t) \end{bmatrix} \begin{bmatrix} w^T(s) & v^T(s) \end{bmatrix}\right] = \begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} \delta(t,s) \tag{5.114}$$

No regime estacionário, a covariância do estado estimado $\hat{x}(k)$, definida como Σ em analogia ao modelo desenvolvido anteriormente, é dada pela seguinte expressão:

$$\Sigma = A\Sigma A^T + Q \tag{5.115}$$

Ao se calcular a covariância da saída do modelo, pode-se mostrar também que, para qualquer l inteiro, a seguinte relação é válida:

$$\Lambda(l) = \begin{cases} CA^{l-1}M_* & l > 0 \\ C\Sigma C^T + R & l = 0 \\ \Lambda^T(-l) & l < 0 \end{cases} \tag{5.116}$$

em que $M_*^T = A\Sigma C^T + S$. Desta forma, as seguintes relações são válidas:

$$\begin{aligned} Q &= \Sigma - A\Sigma A^T \\ R &= \Lambda(0) - C\Sigma C^T \\ S &= M_*^T - A\Sigma C^T \end{aligned} \tag{5.117}$$

Substituindo as relações acima na equação de covariância dos ruídos e levando em conta que uma matriz de covariância por definição é semi definida positiva, chega-se a seguinte inequação matricial linear (LMI da sigla em inglês):

$$\begin{bmatrix} Q & S \\ S^T & R \end{bmatrix} = \mathcal{M}(\Sigma) = \begin{bmatrix} \Sigma - A\Sigma A^T & M_*^T - A\Sigma C^T \\ (M_*^T - A\Sigma C^T)^T & \Lambda(0) - C\Pi C^T \end{bmatrix} \geq 0 \tag{5.118}$$

Na seção anterior foi apresentado como se obter as matrizes A, C, M_* e $\Lambda(0)$ a partir de amostras de uma realização de uma série temporal. Uma vez obtidas estas matrizes, a única variável restante na LMI 5.118 é a matriz Σ. Desta forma, uma vez definida a LMI, suas soluções definidas positivas e simétricas permitem que se realize a série temporal. Deve-se notar que apenas as soluções definidas positivas simétricas são interessantes para a solução do problema, uma vez que Σ é uma matriz de covariância e que, portanto, deve ter estas propriedades.

LMI e equação de Riccati

A LMI obtida com o método desenvolvido por Faurre e apresentada na equação 5.118, tem uma relação estreita com a equação de Riccati, obtida no método desenvolvido por Aoki, que é a mesma obtida pela hipótese de estacionariedade aplicada ao método desenvolvido por Gevers e Kailath, conforme apresentado na equação 5.102. Esta relação fica evidente ao se utilizar a seguinte propriedade de fatoração de matrizes bloco:

$$\begin{bmatrix} X & Y \\ Z & V \end{bmatrix} = \begin{bmatrix} I & YV^{-1} \\ 0 & I \end{bmatrix} \begin{bmatrix} X - YV^{-1}Z & 0 \\ 0 & V \end{bmatrix} \begin{bmatrix} I & 0 \\ V^{-1}Z & I \end{bmatrix} \tag{5.119}$$

A partir desta relação, pode-se fatorar a matriz $\mathcal{M}(\Sigma)$ definida no lado esquerdo da da LMI 5.118 da seguinte maneira:

$$\mathcal{M}(\Sigma) = \begin{bmatrix} I & K \\ 0 & I \end{bmatrix} \begin{bmatrix} -Ric(\Sigma) & 0 \\ 0 & R(\Sigma) \end{bmatrix} \begin{bmatrix} I & 0 \\ K^T & I \end{bmatrix} \tag{5.120}$$

em que:

$$R(\Sigma) = \Lambda(0) - C\Sigma C^T \tag{5.121}$$

que por se tratar de uma matriz de covariância é definida positiva; e

$$K = (M_*^T - A\Sigma C^T)R^{-1}(\Sigma) \tag{5.122}$$

e

$$Ric(\Sigma) = -\Sigma + A\Sigma A^T + (M_* - A\Sigma C^T)(\Lambda(0) - C\Sigma C^T)^{-1}(M_* - A\Sigma C^T)^T \tag{5.123}$$

de onde fica claro que se $M(\Sigma) \geq 0$, conforme definido na LMI de forma que haja solução para o problema de realização, $Ric(\Sigma) \leq 0$.

5.1.5 Espectro de séries temporais

Outra abordagem para o estudo de séries temporais é a partir do espectro dos sinais. Esta abordagem, que é anterior ao desenvolvimento de modelos em espaço de estado para geração de séries, foi desenvolvida principalmente por Wiener e, conforme já citado na introdução deste capítulo, consiste em encontrar uma função de transferência que, quando submetida a um sinal com espectro de ruído branco, tenha como saída um espectro o mais próximo possível do espectro da série temporal a ser realizada.

O principal motivo do desenvolvimento deste tema nesta tese é que a análise do espectro das séries temporais a serem realizadas leva às condições de existência dos modelos que as realizarão. A partir destas condições de existência, pode-se determinar se, para uma dada realização de uma série temporal, é possível encontrar um modelo que realize a série.

Seja um sinal multivariável, discreto, estacionário $y(k) \in \Re^l$, k inteiro, com covariância $\Lambda(\tau) = E[y(k)y(k+\tau)^T]$. Seu espectro é definido como:

$$S_y(z) = \mathcal{Z}[\Lambda(\tau)] = \sum_{\tau=-\infty}^{\infty} \Lambda(\tau) z^{-\tau} \tag{5.124}$$

Partindo da hipótese de que existe um modelo markoviano que gera realizações da série temporal $y(k)$ da forma apresentada na equação 5.48 da seção 5.1.1, ou seja:

$$\begin{cases} x(k+1) = Ax(k) + Gu(k) \\ y(k) = Cx(k) \end{cases} \tag{5.125}$$

pode-se mostrar, como feito anteriormente, que a covariância $\Lambda(i,j)$ da série temporal entre os instantes i e j é dada pela seguinte relação:

$$\Lambda(i,j) = E[y(i)y(j)^T] = \begin{cases} CA^{j-i}\Pi C^T & se \quad i \geq j \\ C\Pi(A^{i-j})^T C^T & se \quad i < j \end{cases} \tag{5.126}$$

em que $\Pi = E[x(k)x(k)^T]$, conforme já definido anteriormente. Como a série temporal por hipótese é estacionária, pode-se definir $\tau = j - i$, e a covariância pode ser reescrita da seguinte forma:

$$\Lambda(\tau) = E[y(i)y(i+\tau)^T] = \begin{cases} CA^{\tau}\Pi C^T & se \quad \tau \leq 0 \\ C\Pi(A^{\tau})^T C^T & se \quad \tau > 0 \end{cases} \tag{5.127}$$

A partir desta covariância, o espectro do sinal gerado pelo modelo apresentado na equação 5.125 é dado pelo seguinte:

$$\begin{aligned} S_y(z) &= C\Pi C^T + C(zI-A)^{-1}A\Pi C^T + C\Pi A^T(z^{-1}I - A^T)^{-1}C^T = \\ &= Z(z) + Z^H(z) \end{aligned} \tag{5.128}$$

em que:

$$Z(z) = \frac{C\Pi C^T}{2} + C(zI-A)^{-1}A\Pi C^T \tag{5.129}$$

e $Z^H(z)$ é a matriz hermitiana de $Z(z)$, ou seja, a complexa conjugada transposta.

Definindo as matrizes $M = C$ e $N = \Pi C^T$ da mesma maneira como foi feito na seção 5.1.1, se tem que a matriz $Z(z)$ pode ser reescrita da seguinte maneira:

$$Z(z) = \frac{MN}{2} + M(zI-A)^{-1}AN \tag{5.130}$$

portanto, se for possível decompor a covariância de uma dada série temporal em seus componentes M, N e A, além de se poder encontrar um modelo ótimo que realiza a série como mostrado na seção 5.1.1, é também possível encontrar diretamente seu espectro $S_y(z)$ e a decomposição deste espectro como soma de uma matriz $Z(z)$ com sua hermitiana.

O método de fatoração espectral proposto por Wiener parte do pressuposto que o espectro de uma série temporal pode ser decomposto como produto da forma $W(z)W^T(z^{-1})$. É possível relacionar a função $W(z)$ às matrizes A, G e C do modelo em espaço de estado 5.125 ao se fazer algumas manipulações algébricas na equação 5.128, conforme demonstrado a seguir:

$$\begin{aligned} S_y(z) &= C\Pi C^T + C(zI-A)^{-1}A\Pi C^T + C\Pi A^T(z^{-1}I - A^T)^{-1}C^T = \\ &= C[\Pi + (zI-A)^{-1}A\Pi + \Pi A^T(z^{-1}I - A^T)^{-1}]C^T \end{aligned} \tag{5.131}$$

introduzindo duas vezes o termo $I = (zI-A)^{-1}(zI-A)$:

$$\begin{aligned} S_y(z) &= C(zI-A)^{-1}(zI-A)* \\ &\quad *[\Pi + (zI-A)^{-1}A\Pi + \Pi A^T(z^{-1}I - A^T)^{-1}]* \\ &\quad *(z^{-1}I - A^T)(z^{-1}I - A^T)^{-1}C^T = \end{aligned} \tag{5.132}$$

rearranjando termos:

$$\begin{aligned} S_y(z) &= C(zI-A)^{-1}* \\ &\quad *[(zI-A)\Pi(z^{-1}I - A^T) + A\Pi(z^{-1}I - A^T) + (zI-A)\Pi A^T]* \\ &\quad *(z^{-1}I - A^T)^{-1}C^T = \end{aligned} \tag{5.133}$$

fazendo as multiplicações dentro das chaves:

$$\begin{aligned} S_y(z) &= C(zI-A)^{-1}* \\ &\quad *[\Pi - z\Pi A^T - A\Pi z^{-1} + A\Pi A^T + A\Pi z^{-1} - A\Pi A^T + z\Pi A^T - A\Pi A^T]* \\ &\quad *(z^{-1}I - A^T)^{-1}C^T \end{aligned} \tag{5.134}$$

e, por fim, cancelando alguns termos:

$$\begin{aligned} S_y(z) &= C(zI-A)[\Pi - A\Pi A^T](z^{-1}I - A^T)^{-1}C^T = \\ &= C(zI-A)^{-1}GG^T(z^{-1}I - A^T)^{-1}C^T = \\ &= W(z)W^T(z^{-1}) \end{aligned} \tag{5.135}$$

em que, por definição,

$$W(z) = C(zI-A)^{-1}G \tag{5.136}$$

5.1.6 Condições de existência para modelos que realizam séries temporais

Seja uma determinada realização de uma série temporal. A partir desta realização, as matrizes de covariância da série temporal podem ser calculadas. A estas matrizes de covariância se aplica o método descrito na seção 5.1.2 e se encontra matrizes A, $M = C$ e $N = \Pi C^T$. Para que se encontre um modelo que realiza a série temporal, as matrizes A, M e N encontradas devem satisfazer as seguintes condições:

- *A matriz A deve ser assintoticamente estável*
- *O par A N deve ser completamente controlável*
- *O par A M deve ser completamente observável*

A primeira condição tem relação com a estabilidade do modelo da série temporal. Se A não for assintoticamente estável, o estado do modelo resultante irá a infinito. A segunda e a terceira condição têm relação com a existência de modelos para realizar sistemas determinísticos. No caso determinísitico, se as condições não forem satisfeitas, não é possível encontrar modelos tenham o mesmo comportamento que o sistema a ser identificado.

A partir das matrizes A, M e N calculadas, se calcula o fator espectral $Z(z)$, definido na equação 5.130, e a partir deste fator se calcula o espectro $S_y(z)$, seguindo a equação 5.128.

O espectro $S_y(z)$ será de fato o espectro de uma série temporal apenas se satisfizer algumas condições, descritas em um teorema enunciado por Faurre em sua tese de doutorado [31] e reproduzido a seguir:

As seguintes afirmações são equivalentes:

- *$S_y(z)$ é o espectro de um processo estocástico.*

- $S_y(z)$ *é o espectro de uma função linear de um processo Markoviano estacionário.*
- $S_y(z)$ *é uma matriz não negativa definida sobre o círculo unitário* $z = e^{j\omega}$, $\omega \in \Re$.
- $S_y(z)$ *é fatorável como um produto da forma* $S_y(z) = W(z)W^T(z^{-1})$ *em que* $W(z)$ *é assintoticamente estável (ou seja, não tem pólos em* $|z| \geq 1$*), de fase mínima (*$det[H(z)] \neq 0$ *em* $|z| > 1$*), quadrada e racional.*
- $S_y(z)$ *é fatorável por uma matriz* $H(z)$ *racional e assintoticamente estável (e não necessariamente quadrada).*
- *Existe uma matriz constante* G *tal que* $S_y(z) = M(zI - A)^{-1}G$
- *Existem duas matrizes* nXn *constantes* Π *e* Q *tais que:*

$$\begin{cases} \Pi = \Pi^T \\ \Pi M^T = N \\ \Pi - A\Pi A^T = Q \end{cases}$$

A prova detalhada deste teorema pode ser encontrada na referência citada.

5.2 Modelagem de séries temporais

Uma vez estudado o problema de realização de séries temporais, o problema de modelagem de séries temporais é facilmente definido. Em linhas gerais este problema pode ser descrito da seguinte maneira:

> *Dada uma realização de uma série temporal, encontre um modelo matemático que, quando submetido ao mesmo sinal dado como entrada para geração da série temporal, resulte em um sinal o mais próximo possível da série temporal que ser quer modelar.*

Neste sentido, a modelagem de séries temporais no espaço de estado se assemelha à modelagem de sistemas determinísticos no espaço de estado, tratada no capítulo 3, uma vez que se tem sinais de saída e entrada e se quer encontrar um modelo linear no espaço de estado que, sendo submetido à entrada, gere um sinal o mais próximo possível da saída. A diferença entre a modelagem de séries temporais e a dos sistemas determinísticos é que, no primeiro caso, se está mais interessado no sinal de saída, enquanto no segundo caso, o maior interesse é no modelo encontrado. Na figura 5.2 é ilustrado um esquema de resolução do problema de modelagem de séries temporais.

Os métodos conhecidos de modelagem de séries temporais também partem do cálculo de covariâncias e da aplicação dos métodos de resolução do problema de realização de séries temporais, conforme pode ser visto na figura 5.2. No entanto, na solução do problema de modelagem, a entrada usada no modelo obtido é supostamente conhecida, e a solução é melhor quanto menor for a diferença entre a realização da qual se partiu para fazer a modelagem e a realização obtida como saída do modelo encontrado.

Deve-se notar que existem pelo menos duas grandes diferenças entre os problemas de realização e modelagem. A primeira delas é o critério de qualidade das soluções dos problemas. No primeiro

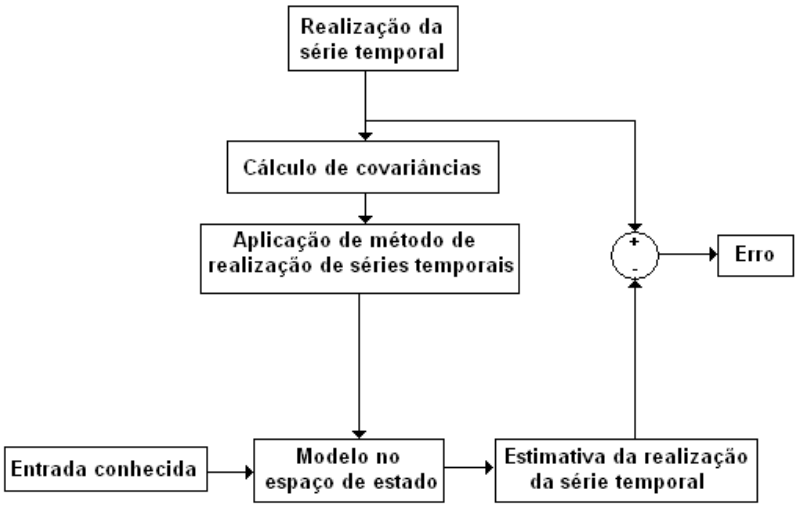

Fig. 5.2: Método de modelagem de séries temporais

caso, a solução do problema de realização será melhor quanto mais suas covariâncias se aproximarem das covariâncias da série temporal que se quer realizar. No segundo caso, a solução do problema de modelagem será melhor quando a saída do modelo for mais próxima da realização da série temporal a ser modelada. A segunda grande diferença é que no problema de realização o sinal de entrada do modelo que gera a solução não é plenamente conhecido. Conhece-se apenas que este sinal é um ruído branco, com uma determinada covariância para atraso zero. Por outro lado, no problema de modelagem, parte-se do pressuposto que o sinal usado para a geração da série temporal que se quer modelar é conhecido e está disponível para ser usado no modelo encontrado.

Na seção anterior foram apresentadas diversas maneiras de se resolver o problema de realização de séries temporais. A partir das técnicas apresentadas e partindo do princípio que o sinal de entrada que gerou a série (supostamente um ruído branco) está disponível, a modelagem de séries temporais se resume simplesmente a aplicar um dos métodos apresentados acima à série temporal de saída e, uma vez encontrado o modelo, aplicar o sinal de entrada conhecido ao modelo encontrado e, com isto, encontrar um sinal de saída estimado. A modelagem será bem sucedida se o sinal de saída encontrado for semelhante no domínio do tempo ao sinal que se quer estimar.

No capítulo 7 desta tese é apresentada uma nova alternativa para modelagem de séries temporais, baseada na aplicação dos algoritmos imuno-inspirados. Com esta alternativa, o cálculo das matrizes de covariância não é necessário. Com isto, mesmo que a realização disponível da série temporal tenha poucas amostras, o que implica em uma estimativa pobre das covariâncias, o resultado da modelagem não é prejudicado.

Capítulo 6

Sistemas Imunológicos Artificiais

Nas últimas décadas foram desenvolvidos inúmeros algoritmos computacionais inspirados na observação de fenômenos naturais, tendo em vista a solução dos mais diversos tipos de problemas. Esta técnica de observação de fenômenos naturais para construção de algoritmos recebe o nome de computação natural. Uma família de técnicas de computação natural é baseada no funcionamento do sistema imunológico de mamíferos. Esta família de técnicas recebe o nome de algoritmos imuno-inspirados.

Os primeiros trabalhos relacionando técnicas de computação a teorias imunológicas focavam nos problemas de aprendizado de máquina e sistemas classificadores. Ao final da década de 90, de Castro e Von Zuben introduziram teorias imunológicas, como o princípio da seleção clonal, a algoritmos de otimização [22] [23], criando o algoritmo *CLONALG*, detalhado em [25]. Pouco tempo depois, De Castro e Timmis introduziram outra teoria imunológica, que é o princípio da rede imunológica, a algoritmos de otimização, criando o algoritmo *opt-aiNet* [20], e ao longo da primeira década do século XXI diversas variantes destes algoritmos têm sido desenvolvidas, como a família *opt-IA* [18] e outras variantes do *opt-aiNet*, resumidas em [27]. Parte destes algoritmos é detalhada na seção 6.2 desta tese e as teorias imunológicas que os motivaram são discutidas na seção 6.1.

O sistema imunológico de mamíferos pode ser usado como inspiração para o desenvolvimento de técnicas de engenharia pois, de acordo com as teorias que tratam deste sistema, ele apresenta várias características que são importantes para a solução de diversos tipos de problemas. Dentre essas características, pode-se destacar as seguintes:

- Unicidade

  O sistema imunológico de cada indivíduo é único, e é desenvolvido a partir de sua interação com o ambiente a sua volta. Quando se trata da solução de problemas esta característica é fundamental, pois cada problema tem suas características particulares.

- Capacidade de reconhecimento

  O sistema imunológico tem a capacidade de reconhecer elementos que foram apresentados anteriormente a ele. Em outras palavras, este sistema apresenta a capacidade de formar uma memória e acessar as posições de sua memória por conteúdo, e não por endereço. Esta capacidade é fundamental na resolução de problemas de reconhecimento de padrões, fazendo do funcionamento dos sistemas imunológicos uma inspiração para o desenvolvimento de ferramentas para solução deste tipo de problema.

- Detecção de novos padrões

  Além da capacidade de reconhecer padrões já apresentados anteriormente, o sistema imunológico tem também a capacidade de detectar e responder a padrões totalmente novos. Desta forma, ao se estudar um problema real em que inovações podem surgir de maneira imprevista, ferramentas baseadas no funcionamento deste sistema se tornam interessantes.

- Detecção distribuída

  O sistema imunológico é capaz de reconhecer várias ameaças simultaneamente. Isto pode ser visto como uma detecção distribuída em todo um espaço de soluções. Desta forma, problemas com múltiplas soluções ou com soluções que variam ao longo do tempo podem ser resolvidos com ferramentas baseadas nos sistemas imunológicos.

- Tolerância a ruído

  A capacidade de reconhecimento de padrões do sistema imunológico vai além da detecção exata. O reconhecimento ocorre mesmo que existam pequenas diferenças entre o padrão apresentado e o padrão previamente conhecido pelo sistema. Desta forma, tendo como base o funcionamento deste sistema, é possível que se crie ferramentas capazes de detectar padrões semelhantes àqueles que fazem parte de uma determinada memória, tolerando ruídos que possam surgir nas informações.

- Capacidade de aprendizado

  Uma das características que torna o sistema imunológico uma inspiração para o desenvolvimento de ferramentas de solução de problemas é sua capacidade de aprendizado. Uma vez que determinado padrão é apresentado ao sistema, ele é reconhecido e passa a integrar o repertório de padrões inerente ao sistema. Esta característica é uma daquelas que faz com que ferramentas baseadas no sistema imunológico sejam empregadas em problemas de reconhecimento de padrões.

Neste capítulo, algumas teorias para o funcionamento do sistema imunológico são apresentadas brevemente e, posteriormente, é descrito como os princípios presentes nestas teorias podem ser empregados na solução de problemas de otimização. Nesta descrição são destacados o algoritmo *CLONALG* em sua versão para otimização [25], o algoritmo *opt-aiNet* [20] e a família de algoritmos *opt-IA* [18]. Por fim, na última seção deste capítulo, são discutidas algumas alterações que foram feitas nos algoritmos imuno-inspirados para a solução de problemas de otimização estudados nesta tese, que são relacionados à identificação de sistemas e realização de séries temporais multivariáveis no espaço de estado.

Nesta tese, não se tem o objetivo de discutir ou de apresentar detalhadamente as teorias de funcionamento do sistema imunológico. O único objetivo que se tem é fornecer ao leitor uma noção básica destas teorias e, principalmente, demonstrar como algoritmos de otimização podem ser desenvolvidos a partir de alguns princípios presentes nestas teorias. Maiores detalhes a respeito deste tema podem ser encontrados nas referências [21], [22] e [23], ou em artigos e livros da área de imunologia.

6.1 Teorias imunológicas

O estudo dos sistemas imunológicos é uma importante linha de pesquisa. Ao longo do avanço dessa linha de pesquisa, diferentes teorias têm sido desenvolvidas para modelar o funcionamento do sistema. Nesta seção, serão apresentadas algumas dessas teorias, tendo como objetivo introduzir alguns princípios que podem ser empregados na solução de problemas de otimização.

6.1.1 Sistema inato e adaptativo

De acordo com algumas teorias, pode-se dividir o sistema imunológico em dois subsistemas que agem de forma complementar. Estes são o sistema inato e o sistema adaptativo. O primeiro deles é aquele que o indivíduo possui desde seu nascimento. Este subsistema não sofre adaptações ao meio em que o indivíduo está. O segundo subsistema é desenvolvido ao longo da vida do indivíduo, e como seu próprio nome sugere, se adapta ao ambiente a que o indivíduo está exposto.

Em geral, o sistema inato age reconhecendo ameaças, que no campo da imunologia são chamadas de antígenos ou patógenos, e permitindo que elas sejam destruídas. O reconhecimento dos antígenos se dá por meio de receptores reconhecedores de padrões (*Pattern Recognition Receptors* ou *PRRs*), que se associam aos padrões moleculares associados aos antígenos (*Pathogen Associated Molecular Patterns* ou *PAMPS*) denunciando sua presença. Uma vez que o sinal de detecção de antígeno é ativado, o sistema inato ativa o sistema adaptativo, dando início à resposta imune adaptativa. Deve-se notar que o sistema inato deve responder apenas a elementos nocivos, e não pode responder a elementos do próprio organismo. Em outras palavras, o sistema inato deve ser capaz de fazer a distinção entre estruturas próprias do organismo e outras não próprias, e reagir apenas ao que é não próprio. Este problema é conhecido em imunologia como problema do próprio-não próprio.

De acordo com algumas teorias imunológicas, o sistema adaptativo tem como principais agentes dois tipos de células, denominadas linfócitos. O primeiro deles é a célula B que, quando ativada, secreta os anticorpos, que são estruturas que se encaixam aos antígenos e sinalizam sua presença. O segundo grupo de linfócitos é conhecido como células T e tem como função principal a regulação da resposta imune. Esta regulação se dá pois este grupo de células tem como funções o incentivo para a produção de células B, realizado pelas células T auxiliares, execução de células nocivas, realizada pelas células T citotóxicas, inibição da ação de células B, o que é feito pelas células T supressoras, dentre outras funções. Todos os linfócitos são criados na medula óssea e as células T são amadurecidas em um orgão conhecido como timo.

6.1.2 Antígenos e Anticorpos

Como já foi brevemente citado, define-se como antígeno uma molécula estranha ao organismo, que normalmente é originária de algum invasor. Para combater os antígenos, é necessário que a presença deles seja notada no organismo. Para que isto aconteça, existem moléculas denominadas anticorpos, que se encaixam ao antígeno sinalizando sua presença. Essas moléculas têm duas regiões: a região constante e a região variável.

Para que um organismo seja bem sucedido no combate aos antígenos, é necessário que seus anticorpos sejam capazes de reconhecer a todos os antígenos a que o indivíduo será exposto ao longo

da vida. Isto pode ocorrer pois, de acordo com teorias imunológicas, os anticorpos têm a capacidade de adaptação aos antígenos por meio de um processo semelhante à seleção natural das espécies. Neste processo, as células B que produzem anticorpos sofrem mutações e as mais bem sucedidas são naturalmente selecionadas e se proliferam. As mutações são introduzidas por dois mecanismos. O primeiro, chamado de recombinação, ocorre na geração do anticorpo. Nesta etapa, essa molécula é sintetizada a partir de combinações aleatórias de informações presentes no genoma da célula B. O segundo mecanismo consiste na mutação em altas taxas das regiões variáveis dos anticorpos. A este mecanismo é dado o nome de hipermutação. A recombinação permite que sejam gerados anticorpos muito diferentes entre si, enquanto o mecanismo de hipermutação permite que haja um ajuste fino para aperfeiçoamento do encaixe ao antígeno.

6.1.3 Reatividade cruzada

Além de contar com mecanismos de mutação de anticorpos para adaptação aos antígenos, a resposta imunológica apresenta também uma característica que, em teoria, permite que um número infinito de antígenos seja reconhecido. Esta característica é a reatividade cruzada, que significa que um determinado anticorpo é capaz de não apenas reconhecer a um antígeno exatamente complementar a ele, mas também a antígenos em que apenas algumas das características foram alteradas com relação ao antígeno exatamente complementar. A diferença entre um antígeno qualquer e o antígeno exatamente complementar a um determinado anticorpo determina a afinidade do anticorpo àquele antígeno. A afinidade é maior quanto menor for esta diferença. Sendo assim, um determinado anticorpo não é capaz de reconhecer apenas a um antígeno, mas a quaisquer antígenos dentro de uma região de um determinado espaço *n*-dimensional definido por todas as n características possíveis de existirem em um antígeno. A este espaço *n*-dimensional é dado o nome de espaço de forma. Neste espaço, cada ponto pode ser um anticorpo ou um antígeno e cada anticorpo tem ao seu redor uma região de reconhecimento. Na figura 6.1 é apresentada uma ilustração do anticorpo e da região ao redor dele em que ainda há reconhecimento. Antígenos que forem representados por pontos dentro daquela região são reconhecidos pelo anticorpo, sendo que quanto maior for a distância entre anticorpo e antígeno menor será a afinidade entre eles.

6.1.4 Seleção clonal

No estudo do sistema imunológico observou-se que, quando um organismo é exposto a um determinado antígeno, o número de anticorpos próprios para o combate daquele antígeno sofre um aumento explosivo. A partir da observação deste fenômeno, Burnet formulou a teoria da seleção clonal [12]. De acordo com esta teoria, quando um organismo é exposto a um antígeno, apenas as células que secretam anticorpos que identificam aquele antígeno são selecionadas para se proliferarem. Desta forma, os esforços do sistema imunológico são bem focados no objetivo de combater os antígenos que o estão ameaçando.

Para modelar o mecanismo que permite o controle do processo de seleção clonal foi formulada a teoria idiotópica, apresentada brevemente mais adiante.

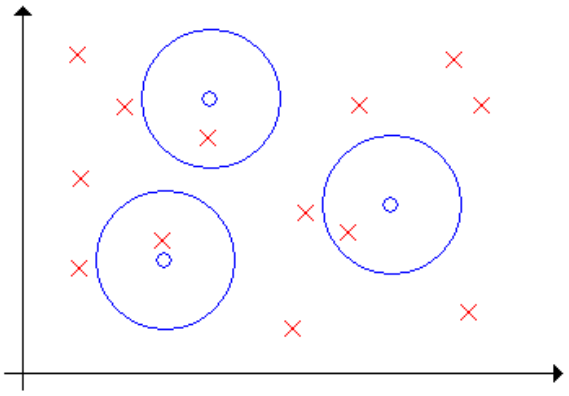

Fig. 6.1: Representação da região de reconhecimento dos anticorpos. Os círculos menores azuis representam os anticorpos, as cruzes vermelhas representam os antígenos e os círculos azuis maiores representam a região de reconhecimento dos anticorpos.

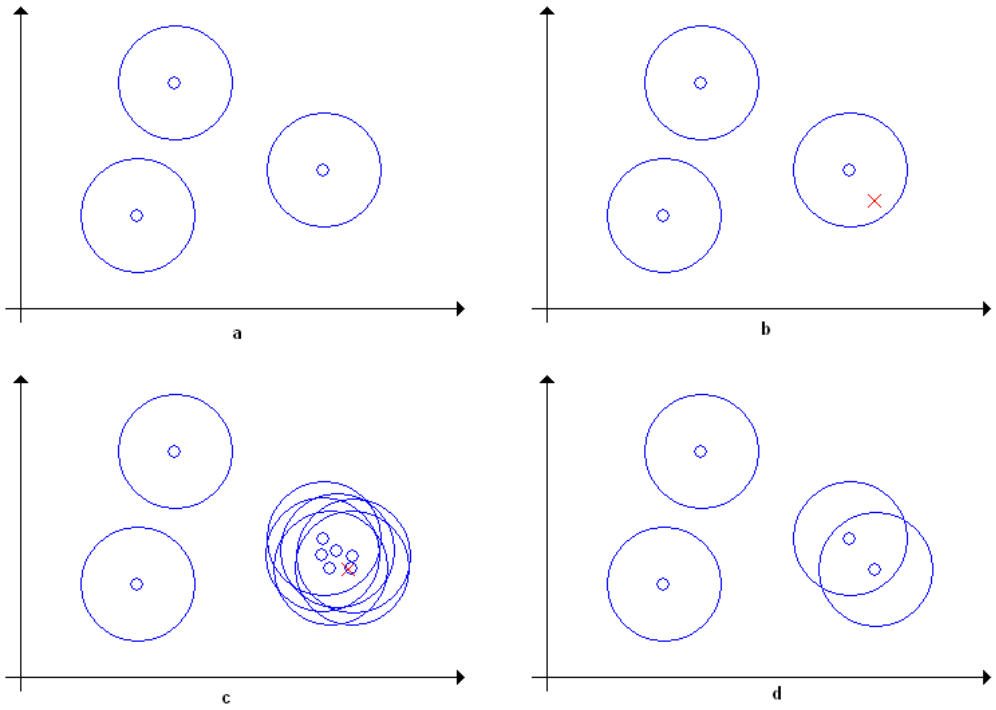

Fig. 6.2: Desenvolvimento da memória do sistema imunológico. (a) Estado inicial das células de memória. (b) Surgimento de um antígeno na região de reconhecimento de uma das células. (c) Clonagem com hipermutação da célula gerando novos indivíduos melhor adaptados ao antígeno. (d) Estado final da memória.

6.1.5 Memória no sistema imunológico

De acordo com algumas teorias imunológicas, durante uma resposta imune, algumas das células B que produziram os anticorpos que mais se adaptaram a um determinado antígeno são selecionadas para amadurecerem e se tornarem células de memória. Estas células têm um tempo de vida bem mais longo que as células normais e ficam inativas no organismo. Quando o organismo é exposto a um antígeno dentro do limiar de afinidade da célula de memória, ela é ativada e passa a se proliferar, produzindo clones capazes de gerar anticorpos que combaterão àquele antígeno. No processo de clonagem ocorrerá a hipermutação, de forma que, se a afinidade entre a célula de memória e o antígeno não for tão alta, os clones terão sua afinidade aperfeiçoada, produzindo anticorpos próprios para o encaixe aos antígenos

Do ponto de vista do espaço de forma, é como se as células de memória fossem pontos do espaço que ficam inativos. Uma vez que algum antígeno surge na região de afinidade, a célula de memória gera clones, que são pontos ao seu redor. Estes pontos vão se proliferando em direção ao antígeno, até que haja uma célula produzindo anticorpos em um ponto suficientemente próximo do antígeno. Quando o antígeno for combatido, o clone mais próximo será selecionado para se tornar uma célula de memória e a próxima resposta do organismo ao mesmo antígeno será imediata. Na figura 6.2 este mecanismo é ilustrado. Conforme pode ser notado na figura, em um tempo infinito é possível que todo o espaço seja mapeado na região de afinidade de algum anticorpo.

A existência de memória no sistema imunológico foi observada ao se notar que indivíduos submetidos a um determinado antígeno apresentam respostas imunológicas mais rápidas caso já tenham sido submetidas àquele antígeno anteriormente. Esta observação levou ao desenvolvimento das vacinas, que têm como princípio de funcionamento a exposição do organismo a versões enfraquecidas dos antígenos para que o sistema imunológico aprenda a lidar com eles, ou seja, para que se criem células capazes de secretar anticorpos contra aquele antígeno. Se o organismo for submetido novamente àquele antígeno, ou a algum outro semelhante a ele, a resposta imunológica será mais rápida e eficaz.

6.1.6 Teoria idiotópica

Em meados do século XX, Niels Jerne desenvolveu uma teoria que explica o auto reconhecimento de células no sistema imunológico, a regulação da resposta imune e a solução do problema do reconhecimento do próprio e do não próprio [43]. Esta teoria é conhecida como teoria idiotópica e tem como base a interação das moléculas dos elementos do sistema imunológico. Esta interação faz com que os elementos formem uma rede imunológica.

De acordo com esta teoria, os antígenos são caracterizados por seus epítopos, que por sua vez são reconhecidos pelos paratopos dos anticorpos. No entanto, os anticorpos também apresentam estruturas similares aos epítopos, e que também podem ser reconhecidos por paratopos. A estas estruturas se dá o nome de idiótopos. Ao conjunto de todos os idiótopos possíveis de existirem é dado o nome de idiótipo.

Desta forma, um determinado antígeno tem seus paratopos reconhecidos pelos epítopos de um determinado grupo de anticorpos uma vez que, pela reatividade cruzada, mais de um anticorpo pode reconhecer um antígeno. Os paratopos destes anticorpos também reconhecem idiótopos de outro grupo de anticorpos, que recebe o nome de imagem do antígeno. O grupo de anticorpos que reconhece

os antígenos por sua vez tem seus idiótopos reconhecidos por paratopos de um segundo grupo de anticorpos, e assim por diante. Desta forma, todos os possíveis antígenos e anticorpos são ligados por uma rede, definida como rede imunológica.

A teoria idiotópica é capaz de explicar a regulação da resposta imune pois explica como os anticorpos podem ser reconhecidos pelo próprio sistema imunológico, levando à proliferação ou à supressão das células B que os produzem. A partir desta teoria também fica claro que a questão da identificação do próprio e do não próprio depende apenas da discriminação entre epítopos e idiótopos.

A regulação dos mecanismos de supressão e de encorajamento à clonagem também pode ser explicada pela teoria da rede imunológica. Esta regulação é proporcionada pelas células T auxiliares, que liberam substâncias que incentivam a propagação das células B que foram capazes de produzir anticorpos bem sucedidos no encaixe a antígenos. Esta propagação é feita pela clonagem destas células, que durante o processo estão sujeitas ao mecanismo de hipermutação, já explicado acima. A medida que as mutações geram células que produzem anticorpos com mais afinidade ao antígeno em questão, estas células passam a ser estimuladas a se proliferarem cada vez mais. As células geradas pela mutação que tiverem afinidade menor que a original são eliminadas pelas células T supressoras em um mecanismo denominado supressão por limiar.

6.1.7 Resumo do funcionamento do sistema imunológico

De forma simplificada, o funcionamento do sistema imunológico pode ser explicado da seguinte maneira: Se um organismo é exposto a algum antígeno, algumas das células B de sua medula óssea produzem anticorpos, que são moléculas cuja função é neutralizar a ação dos antígenos. As células B que produziram os anticorpos com maior afinidade aos antígenos são estimuladas a amadurecer e se tornar ou plasmócitos, ou células B de memória. A função do primeiro grupo de células é amadurecer e gerar clones, enquanto a função do segundo grupo de células é armazenar no organismo a capacidade de reconhecer determinado antígeno. Os clones dos plasmócitos sofrem ligeiras mutações com relação às células originais (hipermutação e adaptação de receptor) e produzem anticorpos diferentes dos gerados pela célula que os originaram. Este mecanismo permite que surjam plasmócitos que produzem anticorpos com maior afinidade aos antígenos. Por outro lado, também são criados plasmócitos que produzem anticorpos com menor afinidade aos antígenos. Os clones que geram anticorpos com maior afinidade ao antígeno a ser combatido são encorajados a se proliferarem ainda mais (seleção clonal), enquanto aqueles que produzem anticorpos de menor afinidade são eliminados pelo organismo (supressão). Após a resposta imune, algumas das células que foram capazes de produzir anticorpos de melhor afinidade são selecionadas para se tornarem células de memória.

De acordo com algumas teorias imunológicas, os mecanismos de reconhecimento das células de maior afinidade ao antígeno, do estímulo a sua reprodução e de supressão de células com baixa afinidade são controlados pelas células T, a partir da reação entre epítopos, idiótopos e paratopos. Além destes mecanismos, algumas novas células B são criadas aleatoriamente pela medula óssea e introduzidas no organismo, mantendo assim a diversidade dos anticorpos que podem ser produzidos.

6.2 Algoritmos imuno-inspirados para otimização

Os princípios observados no sistema imunológico podem ser usados para a resolução de problemas de otimização. Para a solução deste tipo de problema, os anticorpos são as soluções candidatas e o conceito de afinidade entre um anticorpo e um antígeno é substituído pelo valor da função objetivo calculada para cada anticorpo. Alternativamente, também se pode interpretar que os anticorpos são as soluções candidatas e os antígenos são os pontos ótimos do problema.

Nesta seção, três dos diversos algoritmos imuno-inspirados para otimização são apresentados. São eles o algoritmo *CLONALG*, em sua versão para otimização [25], o algoritmo *opt-aiNet* [20] e a família de algoritmos *opt-IA* [18].

6.2.1 CLONALG

O algoritmo *CLONALG* surgiu como uma implementação do princípio da seleção clonal para a solução de problemas de aprendizado de máquina, problemas do caixeiro viajante e problemas de otimização contínua e discreta [22], [25] e [24]. O princípio de funcionamento deste algoritmo aplicado a problemas de otimização é o descrito brevemente a seguir: A princípio, um conjunto de soluções candidatas a resolver o problema de otimização é gerado aleatoriamente. A cada uma das soluções candidatas é aplicada a função objetivo, e o resultado, que é a medida do quanto a solução candidata é adequada para solucionar o problema, é definido como *fitness* da solução. Neste contexto, o *fitness* é um conceito semelhante à afinidade entre antígenos e anticorpos. Após esta etapa, as soluções com maior afinidade são selecionadas e clonadas. O número de clones de cada solução é proporcional[1] ao seu *fitness*. Os clones sofrem mutações durante a etapa de maturação produzindo anticorpos diferentes dos originais. A intensidade da mutação de cada clone é inversamente[2] proporcional ao *fitness*. Na etapa seguinte do algoritmo, novas soluções candidatas são geradas aleatoriamente e introduzidas na população, para evitar a estagnação ao redor de um ótimo local. Para a entrada destas novas soluções candidatas, os anticorpos com menor *fitness* são retirados da população. As etapas são repetidas até que se atinja o critério de parada, que pode ser um número máximo de iterações, um *fitness* mínimo a ser alcançado ou ainda um número máximo de operações. O fluxograma deste algoritmo é apresentado na figura 6.3.

Uma das vantagens trazidas deste tipo de algoritmo imuno-inspirado para solução de problemas de otimização é que há uma tendência de manutenção da diversidade da população, uma vez que novos anticorpos são introduzidos a cada iteração. Outra vantagem é que, diferentemente de técnicas de *fitness sharing*, em que o *fitness* de uma solução é penalizado caso ela esteja em uma região muito povoada do espaço, não é necessário o cálculo da distância entre anticorpos a cada iteração, o que introduziria um grande custo computacional.

6.2.2 Opt-aiNet

Além do princípio da seleção clonal presente no algoritmo *CLONALG*, o algoritmo *opt-aiNet*, proposto em [20] e ilustrado na figura 6.4, também é baseado no princípio da rede imunológica, ci-

[1] Considerando-se o problema de maximização. No caso do problema de minimização o número de clones gerado a partir de cada solução deve ser inversamente proporcional ao *fitness*.

[2] Também considerando o problema de maximização

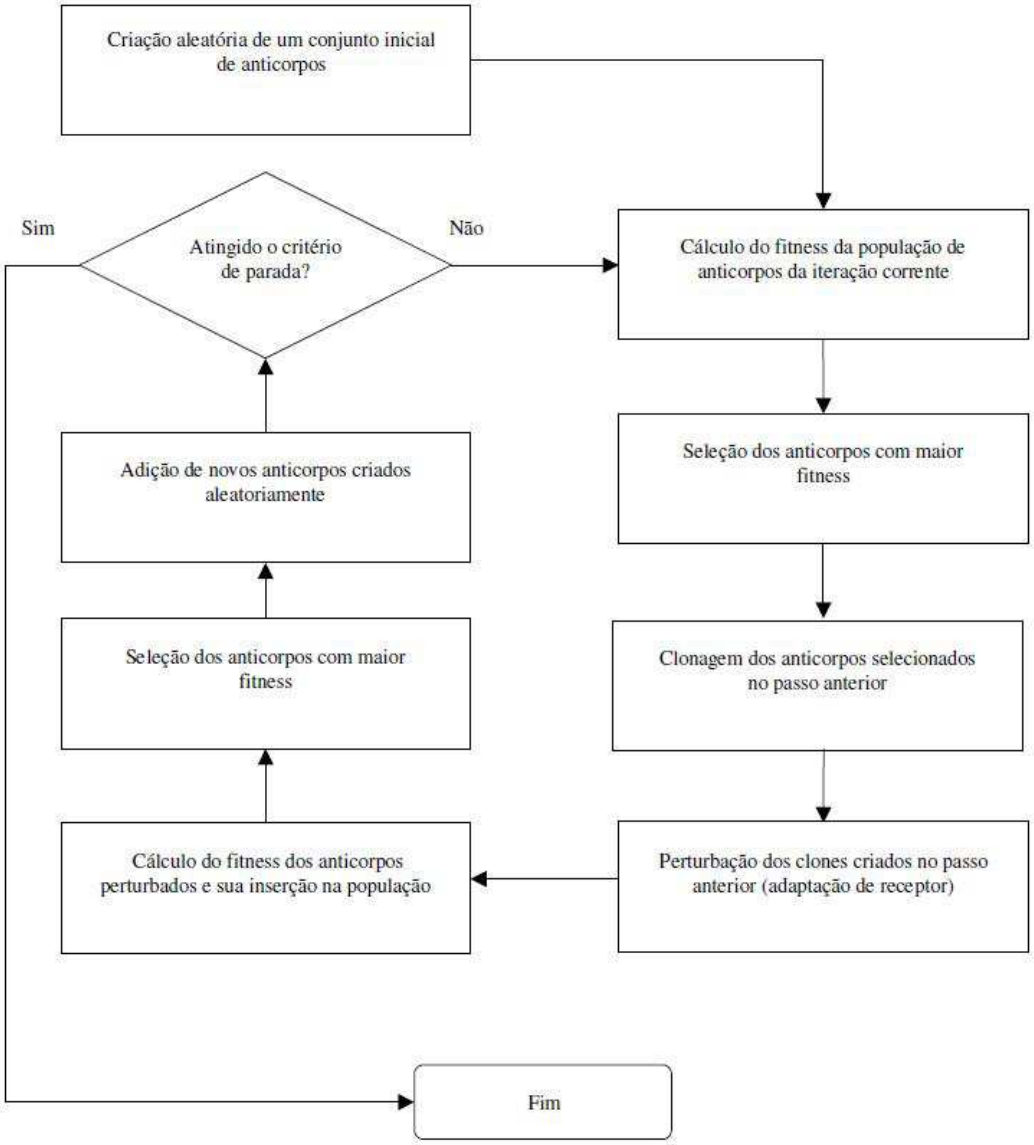

Fig. 6.3: Fluxograma do algoritmo *CLONALG* para otimização.

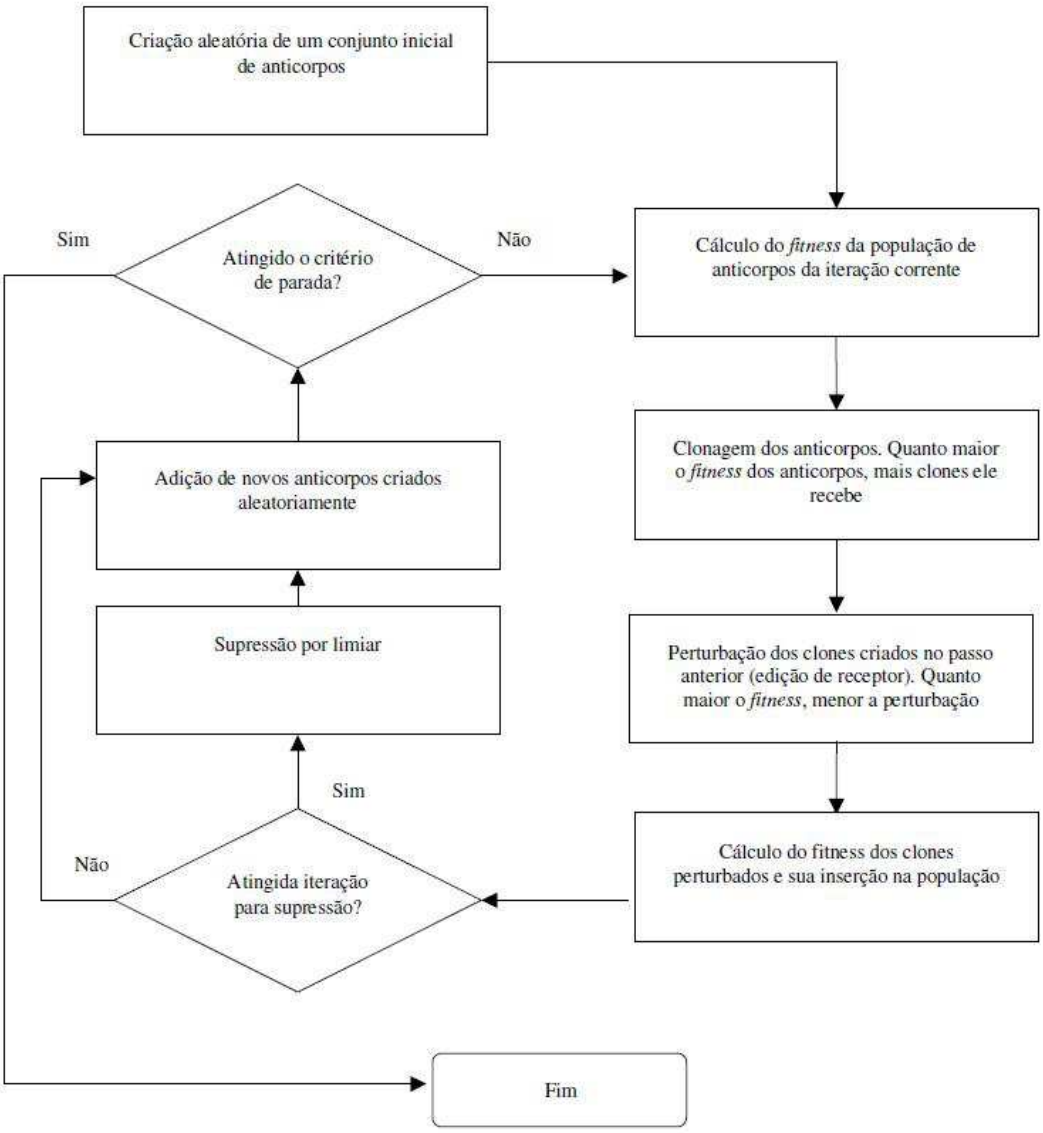

Fig. 6.4: Fluxograma do algoritmo *opt-aiNet*.

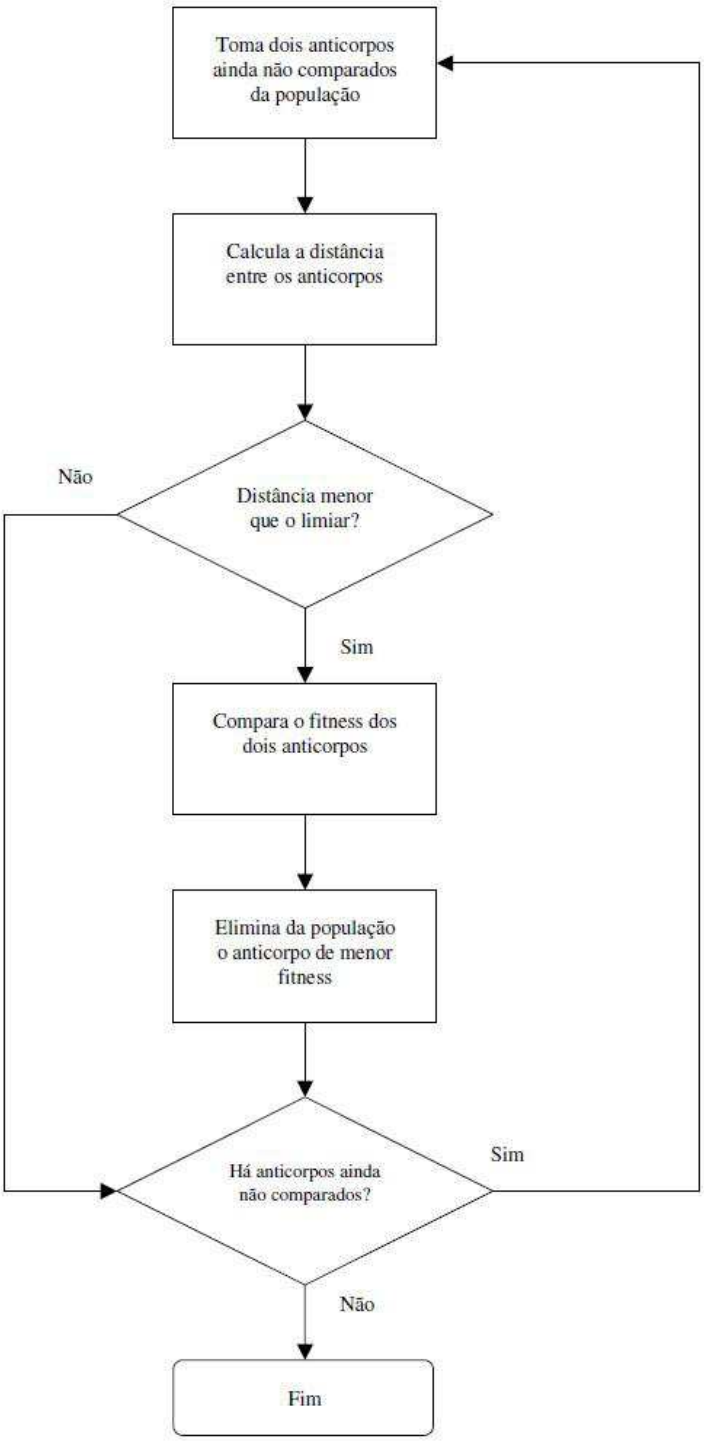

Fig. 6.5: Fluxograma da etapa de supessão por limiar, inclusa no algoritmo *opt-aiNet* (ver figura 6.4)

tado na seção 6.1.6. A grande diferença entre o algoritmo *opt-aiNet* e o *CLONALG* é a etapa de supressão por limiar, detalhada na figura 6.5, que ocorre dentro de um número de iterações definido pelo usuário. Nesta etapa, a distância entre todos os anticorpos da população é calculada e os anticorpos que estiverem em uma região do espaço de soluções em que há outro anticorpo de melhor *fitness* são eliminados, restando apenas o de melhor *fitness*. A região do espaço em que há supressão é definida por uma variável definida como limiar de supressão. Esta variável é igual à distância máxima entre dois anticorpos que implica em supressão. A supressão por limiar reflete o princípio da rede imunológica pois, para sua implementação, parte-se do princípio que os próprios anticorpos são reconhecidos por outros agentes do sistema imunológico, o que permite que se suprima as células que produziram anticorpos de qualidade inferior.

Além da supressão por limiar, outra diferença entre os algoritmos *opt-aiNet* e *CLONALG* é que no primeiro, os anticorpos de menor *fitness* não são retirados da população, como ocorre no segundo . Em outras palavras, o algoritmo *opt-aiNet* não usa um critério de supressão elitista como o *CLONALG*.

A grande vantagem do algoritmo *opt-aiNet* é que o tamanho da população é regulado automaticamente e tende ao número de ótimos locais do problema, desde que o limiar de supressão seja bem escolhido. Além disto, este algoritmo também tende a manter a diversidade da população por dois motivos: o primeiro é a inserção de novos indivíduos aleatoriamente na população, e o segundo é por seu critério de supressão não ser elitista, ou seja, as soluções candidatas eliminadas não são necessariamente as de menor *fitness*, mas sim aquelas que estão explorando uma região do espaço vizinha a um ótimo local e que provavelmente estão na bacia deste mesmo ótimo local.

Na referência [28] é apresentado o resultado de um estudo a respeito da diversidade do algoritmo *opt-aiNet*, comparando a sua performance com um algoritmo *opt-aiNet* modificado para retirada da supressão por limiar e inserção de uma técnica de *fitness sharing*. No artigo também são comparadas duas métricas diferentes para determinação da distância entre dois anticorpos. A conclusão do artigo é que, em geral, o algoritmo *opt-aiNet* apresenta melhor diversidade e qualidade da solução, com um menor custo computacional.

A partir do algoritmo *opt-aiNet* original, várias outras aplicações foram desenvolvidas, conforme pode ser visto de maneira concisa na referência [27]. Estas aplicações variam entre otimização combinatória, otimização em ambientes dinâmicos [26] e também incluem uma proposta de implementação para solução de problemas de otimização genéricos. Nesta tese, o algoritmo *opt-aiNet* foi adaptado para a solução de vários problemas relacionados à identificação de sistemas e à realização de séries temporais multivariáveis no espaço de estado, conforme apresentado no capítulo 7.

Na seção 6.4 é apresentado um exemplo de aplicação do algoritmo *opt-aiNet*. Neste exemplo, alguns detalhes de implementação são discutidos, e os parâmetros de controle do algoritmo são detalhados.

6.2.3 Família Opt-IA

No algoritmo *opt-IA*, proposto inicialmente em [18], os princípios destacados nas teorias imunológicas são aplicados de forma mais simples do que nos algoritmos *CLONALG* e *opt-aiNet*.

A primeira versão deste algoritmo utiliza apenas o princípio de seleção clonal da forma descrita a seguir: O algoritmo é inicializado com uma população de soluções candidatas geradas aleatoriamente. Neste caso as soluções candidatas são simplesmente vetores binários. O *fitness* de cada uma das soluções é calculado e toda a população é clonada. O número de clones é o mesmo para todas as

soluções candidatas. Toda esta população formada por soluções candidatas iniciais e clones sofre mutação e o *fitness* dos anticorpos é novamente calculado. Feito isto, são escolhidos os anticorpos com melhor *fitness* em um mecanismo puramente elitista. O número de soluções escolhidas é igual ao tamanho da população inicial. Este procedimento é repetido até que se encontre uma determinada solução ou até que se atinja um número máximo de iterações. Neste caso deve-se notar que há apenas duas variáveis de controle no algoritmo: o tamanho da população de anticorpos e o número de clones que será gerado para cada anticorpo. Ao longo do artigo [18], os autores seguem resolvendo alguns problemas e discutindo a questão da escolha dos parâmetros de controle do algoritmo.

Nesta proposta inicial do algorimo *opt-IA* não há nenhum mecanismo de supressão por limiar e a escolha da nova população é simplesmente elitista. Desta forma, o algorimo não tem a mesma capacidade que o *opt-aiNet* de escapar de ótimos locais. Além disto, neste algoritmo não se tem o mecanismo da introdução de indivíduos aleatórios a cada iteração, de forma que a diversidade das soluções é pobre relativamente às obtidas no *CLONALG* ou no *opt-aiNet*.

Posteriormente, os mesmos autores que introduziram o algoritmo *opt-IA*, propuseram em [16] algumas modificações que tornaram o algoritmo um pouco mais eficiente. Ao novo algoritmo foi dado o nome de *opt-IMMALG*. As principais alterações introduzidas neste novo algoritmo são as seguintes:

- As soluções candidatas são agora vetores de números reais ao invés de vetores de números binários
- Foi introduzido o *Inversely Proportional Hypermutation Operator* (Operador de hipermutação inversamente proporcional). Esta ferramenta faz com que o grau de mutação dos clones de um determinado anticorpo seja inversamente proporcional ao *fitness* do mesmo. Desta forma, soluções candidatas mais próximas do ponto ótimo sofrem mutações menores que as soluções candidatas que estão mais distantes do ótimo. O grau de mutação de uma solução candidata é equivalente ao número de elementos que sofrerão mutação no vetor que a representa.
- Foi introduzido o operador de envelhecimento *aging*, que é um contador de tempo de vida da solução candidata. Caso este contador seja maior do que um determinado valor, a célula é eliminada da população, independentemente do valor de seu *fitness*. Quando um clone é gerado, o valor de seu contador é definido após a mutação e cálculo de seu *fitness*. Caso o *fitness* do clone após a mutação fique melhor que o da célula que o originou, seu contador recebe zero. Caso contrário, o contador do clone recebe o mesmo valor da célula que o gerou.

Basicamente, todos os passos do *opt-IMMALG* são iguais aos adotados no *opt-IA*. A única diferença significativa é que, antes da escolha dos melhores clones da população, é feita a eliminação das células envelhecidas. O objetivo deste passo é eliminar soluções que estejam ao redor de um ótimo local e privilegiar as células que estiverem mais próximas de outro ponto ótimo. Por outro lado é possível que se penalize células que estejam próximas do ponto ótimo global do problema, diminuindo a velocidade de convergência do algoritmo. Do ponto de vista imunológico, o mecanismo de envelhecimento simula a morte das células B após uma resposta a um antígeno, mas ignora a existência das células de memória.

No artigo em que o *opt-IMMALG* é proposto [16], o desempenho do algoritmo é comparado a outras técnicas de computação natural também usadas para otimização como evolução diferencial,

otimização por enxame de partículas e o algoritmo evolucionário, mostrando que o *opt-IMMALG* é superior aos outros na otimização de diversas funções de diferentes complexidades.

Posteriormente, em [17], os mesmos autores que propuseram o algoritmo *opt-IMMALG* introduziram ainda outras mudanças em seu algoritmo e o compararam a uma versão do *CLONALG* um pouco diferente da proposta em [25].

Nesse trabalho, a alteração feita no *opt-IMMALG* é que a eliminação de células por envelhecimento não é determinística, mas sim estocástica. Ao invés de as células serem eliminadas após um determinado número de iterações, existe uma probabilidade de eliminação de células. Esta probabilidade não depende do valor no contador de iterações da célula e nem de seu *fitness*.

O algoritmo *CLONALG* foi implementado com duas diferentes técnicas de supressão. A primeira é puramente elitista, ou seja, os melhores indivíduos são escolhidos para permanecerem na próxima geração do algoritmo. A segunda é elitista dentre um determinado grupo de clones, ou seja, o melhor indivíduo dentre cada conjunto formado por um anticorpo e seus clones é mantido na população e os outros são eliminados. Com a primeira forma de supressão o algoritmo privilegia apenas um ponto ótimo e com a segunda forma o algoritmo é mais próprio para encontrar ótimos locais. De fato, nos resultados apresentados em [17], é comprovado que a primeira forma é mais eficiente para funções mono-modais (com apenas um ótimo) e a segunda é mais eficiente para funções multi-modais (com mais de um ótimo).

As conclusões do artigo indicam que o desempenho do *opt-IA* é melhor que o do *CLONALG*, no entanto não há nenhuma comparação com o algoritmo *opt-aiNet*.

6.3 Aplicação dos algoritmos imuno-inspirados

Nesta tese, se trata da aplicação de algoritmos imuno-inspirados para a solução de problemas relacionados à identificação de sistemas e realização de séries temporais multivariáveis. Em geral, os algoritmos propostos nesta tese são baseados no algoritmo *opt-aiNet*, devido a suas vantagens destacadas ao longo deste capítulo. Em geral, as alterações feitas no algoritmo *opt-AINet* para a solução dos problemas estudados durante esta pesquisa estão relacionadas à definição dos anticorpos, definição de distância e solução de problemas de otimização com restrições.

Todos os algoritmos apresentados neste capítulo são específicos para problemas sem restrições. Nesta tese, alguns dos problemas tratados apresentam restrições. Por este motivo, nesses casos, as variantes do algoritmo *opt-aiNet* foram implementadas com algumas diferenças com relação ao algoritmo descrito na seção 6.2.2. Basicamente, a principal alteração feita nos algoritmos propostos nesta tese para solução de problemas de otimização com restrições é a introdução da supressão de soluções candidatas infactíveis. Esta supressão se dá em várias etapas do algoritmo. No princípio do algoritmo, ao invés da simples geração aleatória de um conjunto de soluções, é feito um laço gerador de soluções candidatas. Neste laço, toda solução candidata gerada é testada com relação a sua factibilidade. Caso ela seja factível, é armazenada em um conjunto de soluções iniciais. Caso contrário, a solução é descartada. O critério de parada deste laço é que se atinja um determinado número de soluções factíveis, que é o número de indivíduos inicial desejado para o algoritmo. Feito isto, o algoritmo é inicializado normalmente. As mutações que ocorrem nos clones são tais que as soluções permanecem factíveis. Isto também é garantido por um laço que perturba os clones e testa sua factibilidade. Caso os clones perturbados se tornem soluções infactíveis do problema, novas perturbações são adicionadas, seja ao

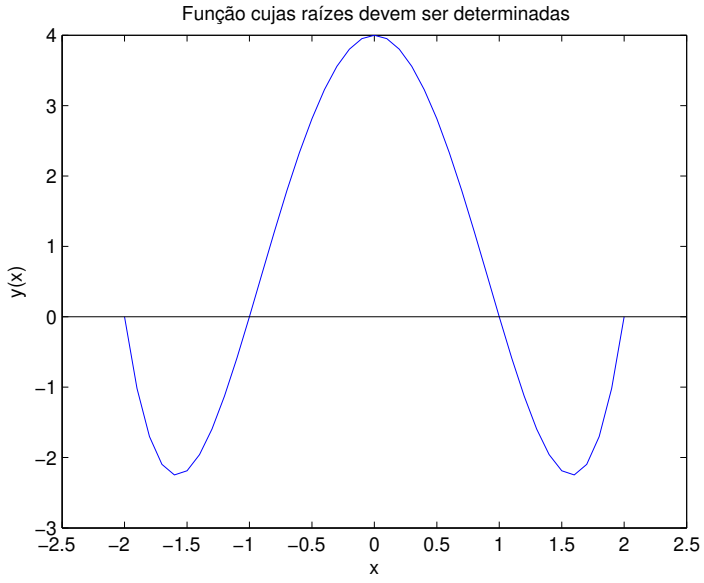

Fig. 6.6: Plotagem da equação 6.1 no domínio $-2 \leq x \leq 2$

clone original, em um processo definido como eliminação de clones infactíveis, seja ao clone perturbado, em um processo definido como travessia de zonas proibidas[3]. Depois, ao término da iteração, são introduzidos novos anticorpos que também são factíveis, o que é garantido também por um laço, a exemplo do que ocorre na geração inicial de anticorpos. Ao longo do capítulo 7 este mecanismo é discutido com maiores detalhes.

Alternativamente, outros métodos de otimização heurística poderiam ser utilizados para a solução dos problemas estudados durante esta pesquisa. No entanto, os algoritmos imuno-inspirados da classe *opt-aiNet* apresentam as vantagens destacadas neste capítulo, implicando em sua escolha como algoritmo base para o desenvolvimento das soluções propostas nesta tese.

Para ilustrar o funcionamento do algoritmo *opt-aiNet*, que é a base das contribuições desta pesquisa, detalhadas no capítulo 7, na seção 6.4 logo abaixo é exemplificado como este algoritmo pode ser aplicado para a solução de um problema simples de determinação das raízes de um polinômio.

6.4 Exemplo de aplicação do algoritmo *opt-aiNet*

Seja o problema de determinação das raízes da equação 6.1, que é plotada na figura 6.6, no domínio $-2 \leq x \leq 2$:

$$y(x) = x^4 - 5x^2 + 4, \quad x \in \Re \tag{6.1}$$

Este problema de determinação de raízes pode ser transformado em um problema de otimização ao se observar o seguinte: Encontrar as raízes de um polinômio $y(x)$ nada mais é que encontrar os

[3] Ver seção 7.3.2 para mais detalhes.

valores $x_r \in \Re$, tais que $y(x_r) = 0$. Desta forma, pode-se definir um problema de maximização tal que a função objetivo seja máxima para as soluções candidatas x tais que $y(x) = 0$. Uma das maneiras de se fazer isto é com a função objetivo $F(x)$ definida na equação 6.2, a partir da qual também pode ser enunciado um problema de otimização.

Função objetivo

$$F(x) = \frac{1}{|y(x)|} = \frac{1}{|x^4 - 5x^2 + 4|}, \quad x \in \Re \tag{6.2}$$

Enunciado do problema de otimização

Maximize a função

$$F(x) = \frac{1}{|x^4 - 5x^2 + 4|}$$

com $x \in \Re$

Uma vez definido o problema de otimização, o algoritmo *opt-aiNet* pode ser aplicado ao se definir os anticorpos, a função de fitness e alguns outros parâmetros de controle do algoritmo. Os anticorpos são as soluções candidatas, ou seja, qualquer $x \in \Re$. A função de fitness é a própria função objetivo do problema de otimização, definida na equação 6.2. Os outros parâmetros do algoritmo *opt-aiNet* são definidos a seguir:

Ordem de grandeza

Conforme discutido na seção 6.2.2, no algoritmo *opt-aiNet* há duas etapas de inserção de indivíduos aleatoriamente. A primeira é na inicialização do algoritmo, e a segunda é feita ao final de cada iteração, para manter a diversidade da população. A geração aleatória de indivíduos pode ser feita a partir de um gerador pseudo-aleatório que gere números entre 0 e 1. No entanto, as soluções ótimas do problema normalmente estão em um domínio maior que o intervalo [0 1], e este domínio pode incluir valores negativos, como no caso do problema aqui tratado, em que há raízes em -1 e -2. Por este motivo, o número aleatório criado pelo gerador pseudo-aleatório deve ser multiplicado por um valor e reescalado para conter também valores negativos. Nesta implementação do algoritmo, o número criado pelo gerador pseudo aleatório no intervalo [0 1] é multiplicado por 2 e subtraído de 1, fazendo com que o intervalo de geração se torne [-1 1]. O número pertencente a este intervalo é multiplicado pela variável *ordem de grandeza*, que reescala o domínio de busca. Neste problema, a variável *ordem de grandeza* é definida como 2, de forma que o intervalo em que soluções aleatórias são geradas é [-2 2].

Critério de parada

O critério de parada do algoritmo pode ser baseado tanto em esforço computacional máximo que se quer empregar, quanto na qualidade da solução que se quer encontrar. No primeiro caso, o algoritmo para após um determinado número de iterações ou de operações completadas. No segundo

caso, o algoritmo pára quando o *fitness* de suas soluções satisfaz a um determinado critério. Para este exemplo, foi escolhido o segundo tipo de critério, e o algoritmo termina sua execução apenas quando o fitness do pior anticorpo da população é maior ou igual a 50.

Número de indivíduos inicial

Este número define quantos anticorpos serão gerados na primeira iteração do problema. No exemplo aqui tratado, sabe-se que o número de soluções ótimas é menor ou igual a 4, por se tratar do problema de determinar as raízes de um polinônio de quarto grau. Mesmo com esta informação, o número de indivíduos inicial escolhido é igual a 10. Esta escolha é proposital para demonstrar a capacidade de controle do número de indivíduos do algoritmo *opt-aiNet*.

Número máximo de clones por anticorpo

Conforme discutido na seção 6.2.2, cada anticorpo recebe um número de clones proporcional[4] à sua qualidade relativa à qualidade de outros indivíduos da população. Uma das maneiras de se implementar isto é definir o número máximo de clones por anticorpo. Na etapa de clonagem, o anticorpo com maior fitness recebe o número máximo de clones definido. Os outros anticorpos recebem um número de clones igual ao arredondamento do número máximo de clones por anticorpo multiplicado pela relação entre seu fitness e o fitness máximo da população naquela iteração. Para o exemplo aqui tratado foi escolhido um número máximo de clones por anticorpo igual a 2.

Grau de perturbação máximo

Conforme discutido na seção 6.2.2, cada clone é perturbado de maneira inversamente proporcional[5] ao seu fitness. Para se implementar isto, define-se um grau de perturbação máximo. Desta forma, cada clone será perturbado pelo produto do grau de perturbação máximo com um número aleatório no intervalo [-1 1] e dividido por seu fitness. Para este problema o grau de perturbação máximo escolhido é 0.25.

Limiar de supressão

No algoritmo *opt-aiNet* é feita a supressão por limiar, detalhada na figura 6.5. Para que esta etapa seja realizada, devem ser feitas duas definições: a primeira é a de distância entre anticorpos. No caso deste problema, como os anticorpos são números reais, a distância é simplesmente definida como o valor absoluto da diferença entre dois anticorpos. A segunda definição é a do limiar de supressão. Este valor é a distância máxima entre dois anticorpos que implica em supressão. Para este problema o limiar de supressão escolhido é 0.5.

[4] no caso do problema de maximização
[5] no caso do problema de maximização

Solução do problema

Definidos o problema de otimização e os parâmetros do algoritmo *opt-aiNet*, o mesmo foi executado e os resultados são apresentados na sequência. Na figura[6] 6.7 é apresentada a primeira população de anticorpos gerada pelo algoritmo. Da figura, nota-se que há 10 anticorpos na população inicial, conforme o número de anticorpos inicial definido no algoritmo. Na figura 6.8 é apresentada a população de anticorpos após a clonagem e perturbação dos clones na primeira iteração. Da figura nota-se que os anticorpos mais próximos de serem raízes da equação geram mais clones, conforme esperado para o algoritmo. Na figura 6.9 é apresentada a população após a supressão por limiar na primeira iteração. Da figura nota-se que os anticorpos mais próximos do ótimo permanecem na população, enquanto os outros são suprimidos.

As populações no início, após clonagem e mutação e após supressão da quinta iteração são apresentadas nas figuras 6.10, 6.11 e 6.12 respectivamente; na nona iteração são apresentadas nas figuras 6.13, 6.14 e 6.15; e na décima terceira iteração são apresentadas nas figuras 6.16, 6.17 e 6.18. Finalmente, na figura 6.19 é apresentada a população do algoritmo na décima sexta iteração, em que foi atingido o critério de parada. Da figura nota-se que, na última iteração, a população é formada por apenas quatro anticorpos, o que é exatamente igual ao número de soluções do problema.

As soluções finais encontradas pelo algoritmo são as seguintes:

$$-1.00000098423437 \quad -2.00000018586194 \quad 0.99999999808179 \quad 1.99875241691700 \tag{6.3}$$

que são muito próximas das soluções corretas (-1, -2, 1 e 2).

Na figura 6.20 é apresentada a evolução do fitness máximo, ou seja, o fitness do melhor anticorpo da iteração corrente, em função do número da iteração. Desta figura nota-se que, quanto mais se aproxima da solução ótima, maior é o incremento do fitness a cada passo. Isto se deve ao processo de perturbação inversamente proporcional ao fitness, que permite um ajuste fino das soluções a medida que se aproxima do ótimo.

Conclusão

Neste exemplo foi ilustrado como a determinação de raízes de um polinômio de quarto grau pode ser resolvida pela definição de um problema de otimização e pela solução deste problema de otimização com o uso do algoritmo *opt-aiNet*. O principal objetivo ao se elaborar este exemplo é a demonstração da capacidade do algoritmo *opt-aiNet* em solucionar problemas de otimização multiobjetivo e de sua capacidade de regulação automática do número de anticorpos da população. Outro motivo que levou à elaboração deste exemplo é detalhar como algumas etapas do algoritmo foram implementadas e quais são seus parâmetros de controle principais. A definição do problema de otimização também é um simples exemplo do princípio utilizado nas propostas feitas nesta tese, detalhadas no capítulo 7.

[6] Nas figuras deste exemplo, apesar de os anticorpos serem elementos no eixo x, para uma melhor visualização eles são representados sobre a curva $y(x)$, definida na equação 6.1.

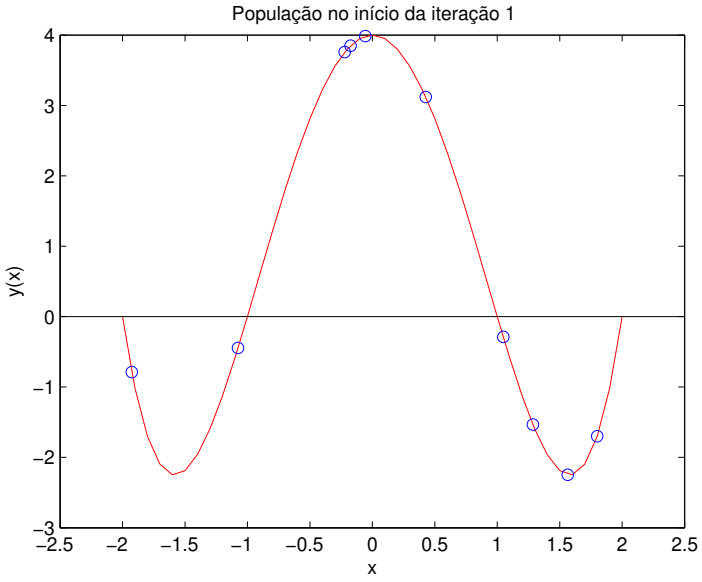

Fig. 6.7: População inicial de anticorpos na solução do exemplo ilustrado na seção 6.4.

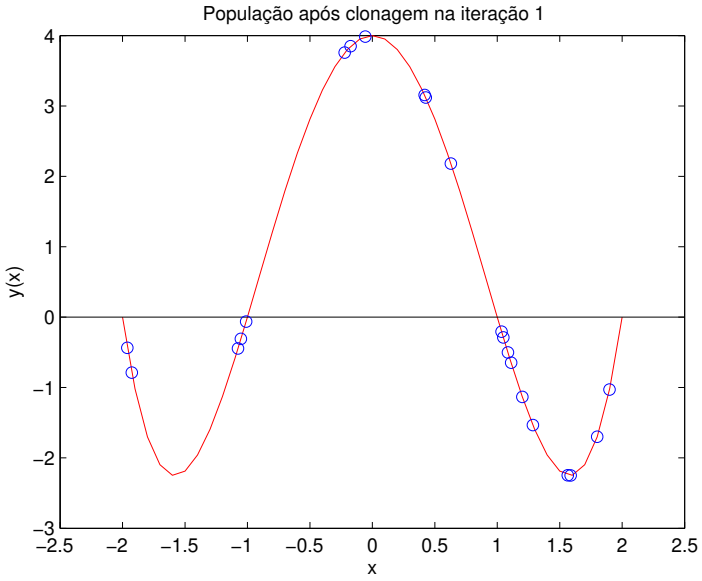

Fig. 6.8: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após clonagem e perturbação na primeira iteração.

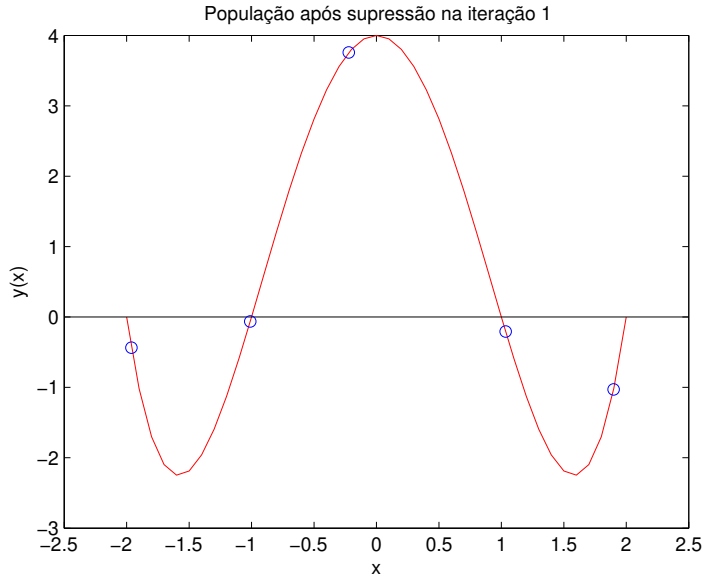

Fig. 6.9: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após supressão por limiar na primeira iteração.

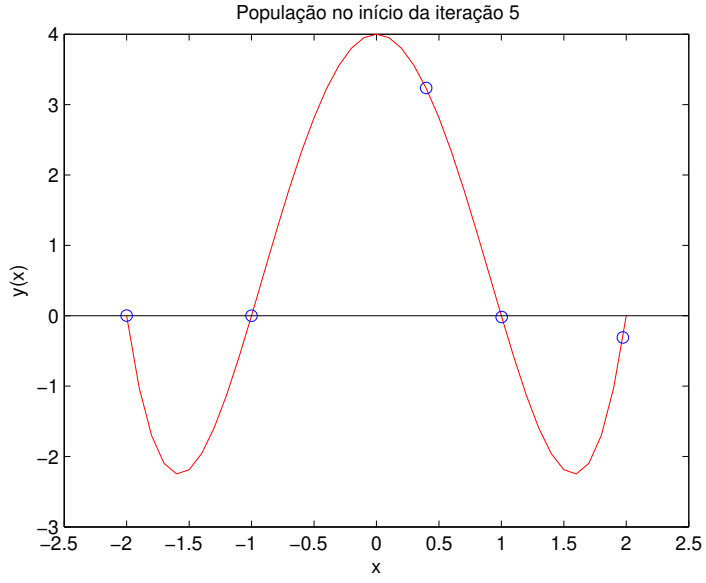

Fig. 6.10: População de anticorpos no início da quinta iteração na solução do exemplo ilustrado na seção 6.4.

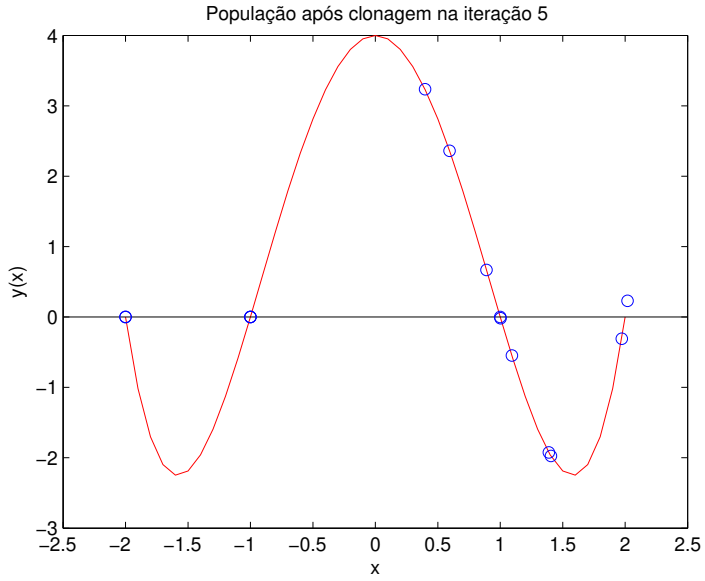

Fig. 6.11: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após clonagem e perturbação na quinta iteração.

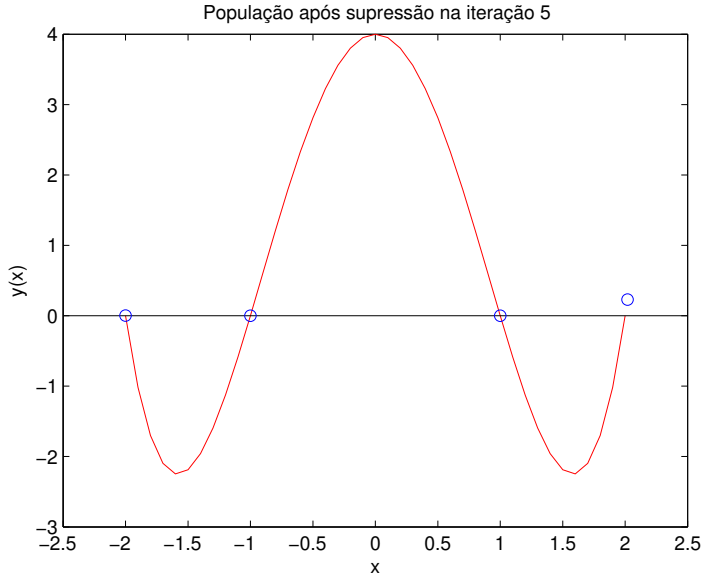

Fig. 6.12: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após supressão por limiar na quinta iteração.

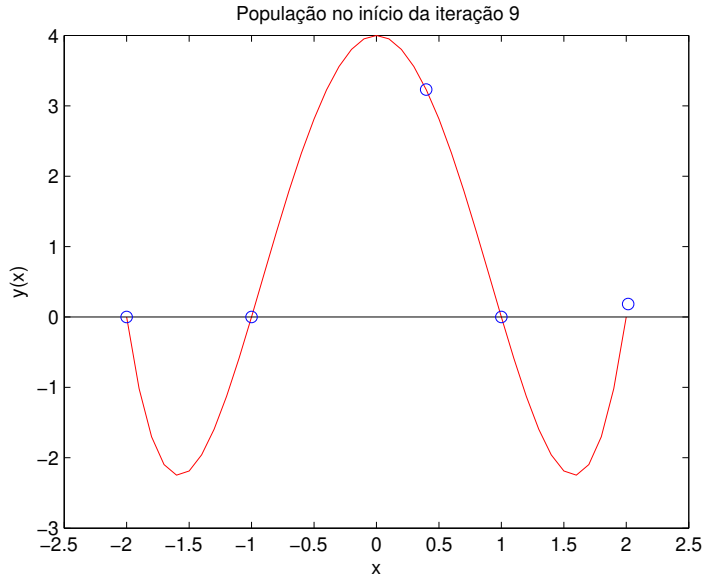

Fig. 6.13: População de anticorpos no início da nona iteração na solução do exemplo ilustrado na seção 6.4.

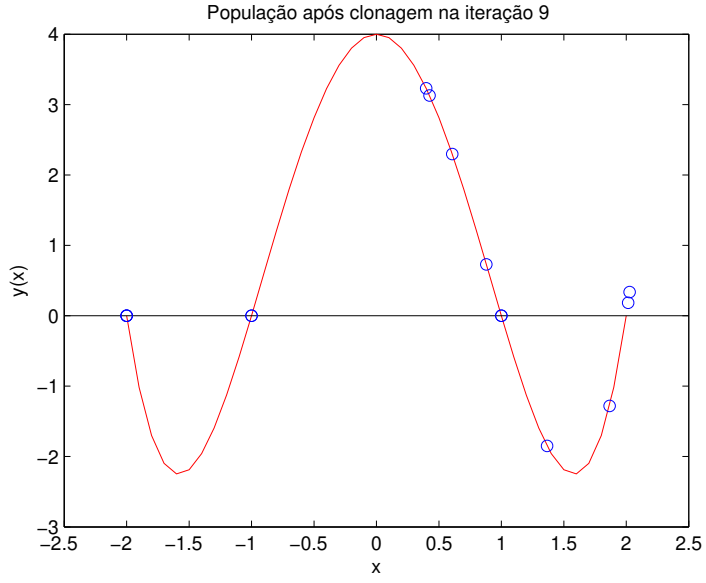

Fig. 6.14: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após clonagem e perturbação na nona iteração.

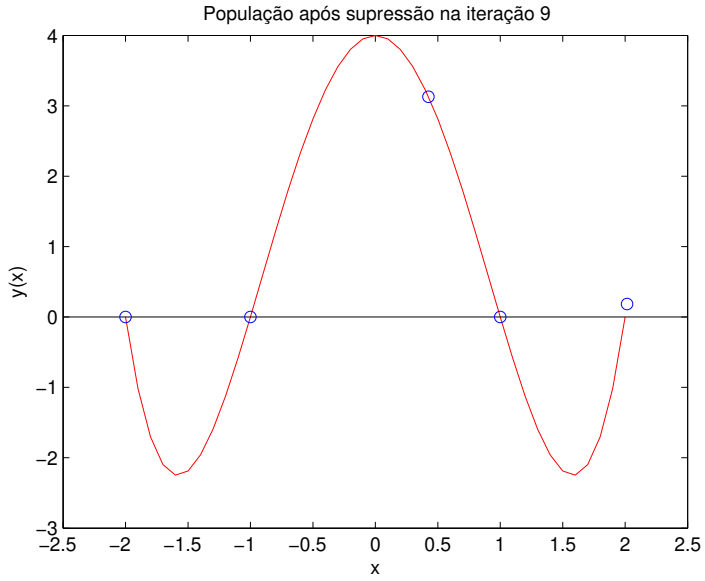

Fig. 6.15: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após supressão por limiar na nona iteração.

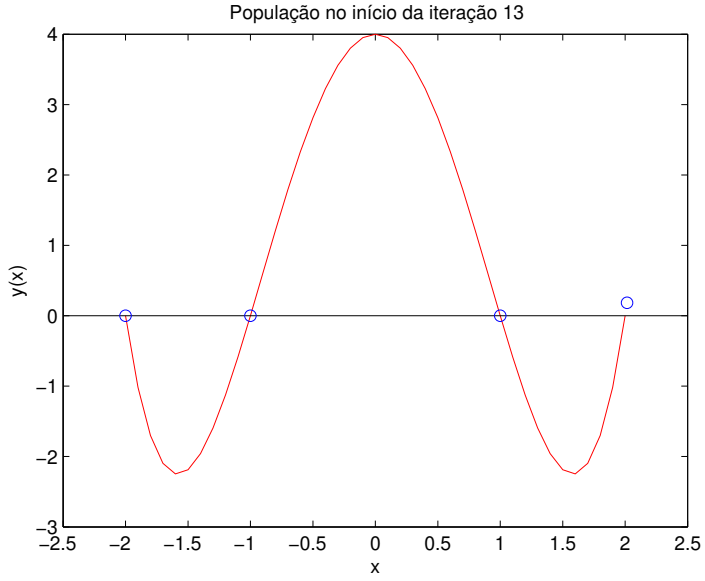

Fig. 6.16: População de anticorpos no início da décima terceira iteração na solução do exemplo ilustrado na seção 6.4.

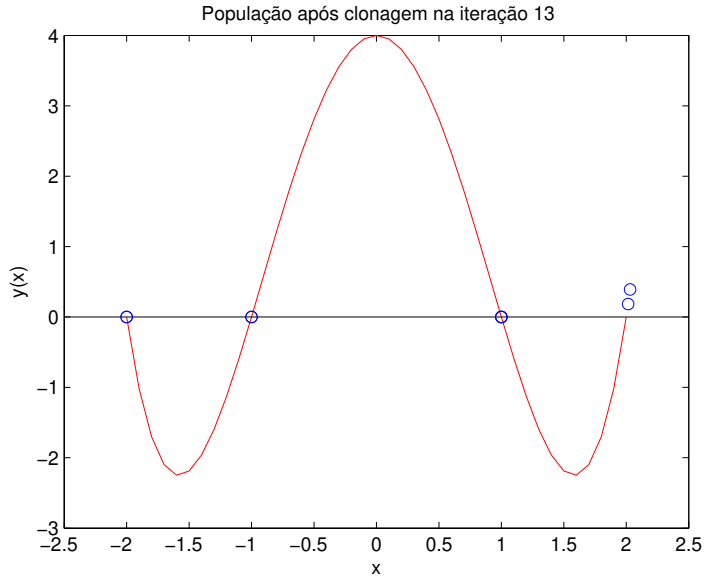

Fig. 6.17: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após clonagem e perturbação na décima terceira iteração.

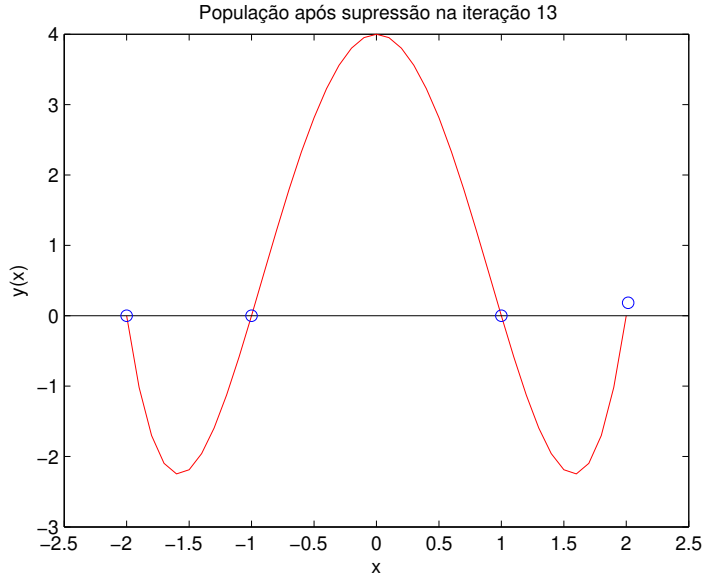

Fig. 6.18: População de anticorpos na solução do exemplo ilustrado na seção 6.4 após supressão por limiar na décima terceira iteração.

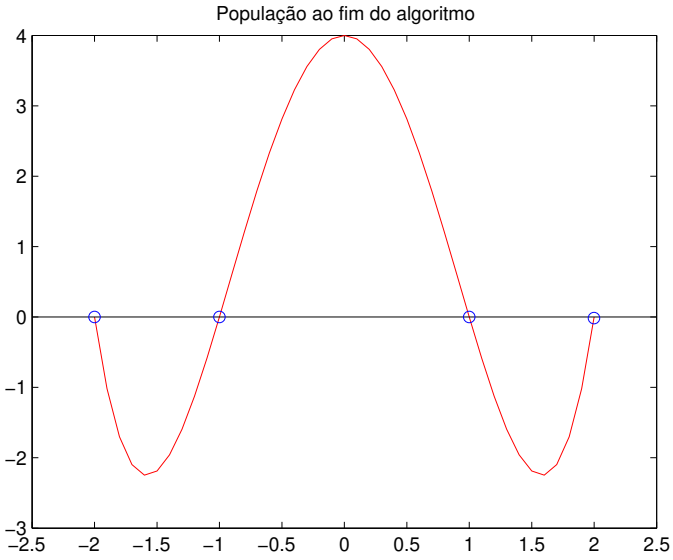

Fig. 6.19: Solução final para o exemplo ilustrado na seção 6.4. Este resultado foi obtido após 16 iterações.

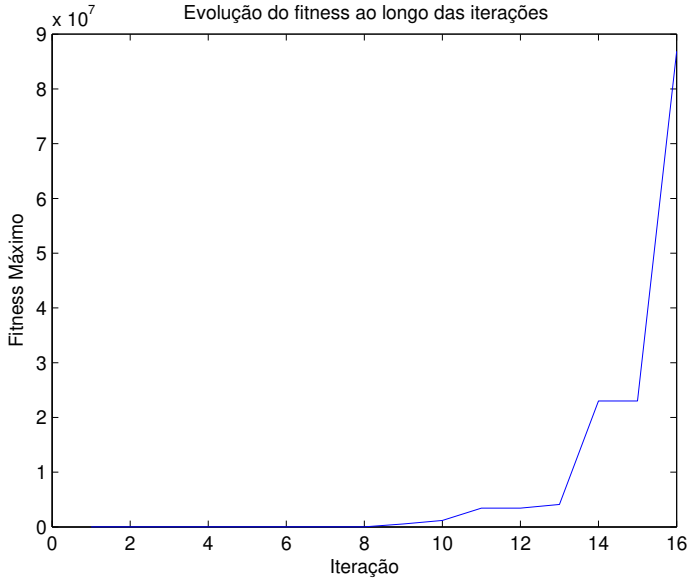

Fig. 6.20: Evolução do fitness máximo em função das iterações para a solução do problema definido na seção 6.4.

Capítulo 7

Contribuições

Neste capítulo, são apresentadas as contribuições propostas nesta tese. Parte destas contribuições está ligada a algumas dificuldades enfrentadas para solução dos problemas de realização e modelagem de séries temporais no espaço de estado. Basicamente, estas dificuldades ocorrem pois o ponto de partida para a resolução dos problemas de realização e modelagem de séries temporais é o cálculo de suas covariâncias. Este cálculo só seria exato se fossem disponíveis infinitas amostras da série temporal, o que não ocorre na realidade. Outra fonte de dificuldade é a geração dos sinais de entrada do modelo que realiza a série temporal, que conforme explicitado no capítulo 5, é um ruído branco. Existem geradores de ruído branco que, ao gerar uma sequência de dados cujo número de amostras tende ao infinito, apresentam covariâncias de uma série temporal que tende ao ruído branco. No entanto, quando se quer gerar uma sequência de dados finita, a covariância do sinal se distancia da covariância típica do ruído branco. Outra contribuição desta tese é uma técnica para a identificação de sistemas lineares multivariáveis variantes no tempo.

Em todas as contribuições feitas nesta tese, os problemas enfrentados são observados sob a ótica de problemas de otimização. Como se tratam de problemas matriciais, em que não são conhecidas relações diretas entre parâmetros e resultado, são utilizadas as técnicas heurísticas de otimização conhecidas como algoritmos imuno-inspirados. Detalhes a respeito destes algoritmos são apresentados no capítulo 6 desta tese.

A primeira contribuição apresentada é um algoritmo para a determinação de ruídos brancos com números de amostras quaisquer. Como visto no capítulo 5, a existência de ruídos brancos é dada como garantida nos processos de realização e modelagem de séries temporais. No entanto, em muitos casos se tem interesse em gerar séries temporais com amostras finitas, o que depende de um ruído branco com o mesmo número de amostras. Caso este número seja pequeno, o sinal gerado por geradores pseudoaleatórios não é um ruído branco, levando a problemas no processo de realização de séries temporais

A segunda contribuição apresentada neste capítulo é um algoritmo para a solução de equações algébricas de Riccati. Dependendo da amostra de série temporal disponível, a fatoração da matriz de covariância leva a matrizes que, quando combinadas na equação de Riccati, ferem as condições de existência de soluções para esta equação, tornando o problema de realização insolúvel. O algoritmo apresentado nesta tese encontra soluções aproximadas para estes casos, permitindo a continuidade do problema de realização de séries temporais.

A terceira contribuição desta tese é voltada ao problema de modelagem de séries temporais. No

caso em que as covariâncias da série temporal encontrada são tais que as condições de existência para modelos que realizam séries temporais são feridas, não é possível resolver o problema de realização, e consequentemente também não é possível resolver o problema de modelagem. Nestes casos é possível determinar modelos aproximados a partir do algoritmo proposto nesta tese.

A quarta contribuição destacada é a identificação de sistemas lineares multivariáveis variantes no tempo. Conforme descrito no capítulo 3, existem métodos para identificação de sistemas variantes no tempo que partem da divisão das amostras de entradas e saídas do sistema em conjuntos de tamanho tal que, dentro destas amostras, não há variação do sistema. A cada um destes conjuntos de amostras é aplicado o método MOESP, também descrito no capítulo 3, e um modelo é determinado para cada conjunto de amostras. No entanto, caso o sistema varie rapidamente, o conjunto de amostras para o qual não há variação significativa do sistema é pequeno e o método MOESP falha ao ser aplicado. Para lidar com este problema, foi proposto um algoritmo detalhado mais adiante.

A cada uma das contribuições indicadas acima é dedicada uma seção deste capítulo.

7.1 Proposta para geração de ruídos brancos

Conforme já discutido no capítulo 5 desta tese, uma maneira de se gerar uma realização de uma série temporal é encontrar o modelo em espaço de estado que realiza a série a partir de algum dos algoritmos discutidos naquele capítulo e então aplicar como entrada a este modelo um ruído branco, com um número de amostras igual ao número de amostras da realização da série temporal que se quer gerar. Em geral, se presume que um ruído branco com número de amostras finito possa ser obtido com o uso de rotinas de geração de números aleatórios, conhecidas como geradores pseudo-aleatórios. No entanto, como será discutido nesta seção, estas rotinas produzem ruídos brancos apenas quando o número de amostras tende a infinito. No caso de um número limitado de amostras, o sinal gerado por estas rotinas não é um ruído branco ideal. Consequentemente, ao se usar um sinal obtido com o uso de um gerador pseudo-aleatório como entrada de um modelo de realização de série temporal, a saída não será de fato uma realização da série temporal.

Nesta seção é apresentado um algoritmo imuno-inspirado proposto para a geração de ruídos brancos com número de amostras finito qualquer. O método proposto nesta seção produz ruídos brancos mais próximos do ruído branco ideal do que rotinas de geração de ruídos brancos disponíveis no MATLAB. Consequentemente, os resultados da realização de uma série temporal usando como entrada o sinal produzido pela rotina proposta nesta seção são melhores do que os resultados obtidos ao se usar um ruído branco gerado pelo gerador pseudo-aleatório do MATLAB.

Nesta seção também é apresentada uma discussão a respeito da relação entre o número de amostras do ruído branco e a proximidade entre o sinal gerado pela rotina proposta e o ruído branco ideal. Esta mesma comparação é feita para o ruído branco gerado pelo MATLAB e a partir disto chega-se a algumas conclusões relativas ao intervalo de número de amostras em que o algoritmo proposto nesta seção produz resultados superiores aos do algoritmo pseudo-aleatório.

Por fim, será apresentada uma discussão a respeito do espectro dos sinais gerados pelo algoritmo proposto nesta seção e o sinal obtido com o gerador pseudo-aleatório do MATLAB. A partir desta discussão será possível notar que existem duas definições de ruído branco, sendo uma delas no domínio do tempo e a outra no domínio da frequência. Conforme será apresentado, para um número finito de amostras, tanto o sinal gerado pela rotina proposta nesta seção quanto o sinal criado pelo gerador

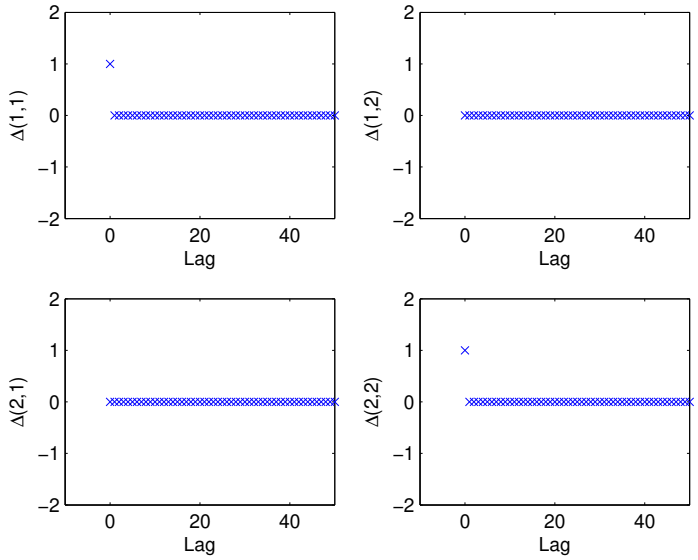

Fig. 7.1: Covariâncias do ruído branco bidimensional ideal para $\Delta = \mathcal{I}_{2x2}$

pseudo-aleatório do MATLAB satisfazem a definição de ruído branco no domínio da frequência. No entanto, a rotina apresentada nesta seção é mais próxima de um ruído branco ideal que a rotina gerada pelo MATLAB quando a definição de ruído branco no domínio do tempo é empregada.

Os resultados apresentados nesta seção em parte estão publicados no artigo [37].

7.1.1 Ruído branco

No domínio do tempo, o ruído branco é uma série temporal (ou processo estocástico) de segunda ordem, que tem média zero e em que todas as amostras tomadas em instantes de tempo diferentes são descorrelacionadas entre si. Algebricamente, isto significa que a série temporal discreta multivariável $e(k) \in \Re^n$ é um ruído branco discreto apenas se o seu vetor de médias $\mu \in \Re^n$ e suas matrizes de covariância $\Delta_e(t) \in \Re^{nXn}$ são os seguintes:

$$\mu = E[e(k)] = \mathbf{0}_n \tag{7.1}$$

$$\Delta_e(t) = E[e(k)e(k+t)^T] \begin{cases} \Delta & t = 0 \\ \mathbf{0}_{nXn} & t \neq 0 \end{cases} \tag{7.2}$$

em que $\mathbf{0}_n$ é o vetor nulo do espaço $\Re^n$, $\Delta \in \Re^{nXn}$ é uma matriz de números reais, $\mathbf{0}_{nXn}$ é a matriz nula do espaço $\Re^{nXn}$, t é um número inteiro definido como atraso e $E[\cdot]$ é o operador de esperança. Na figura 7.1 é plotada a covariância do ruído branco ideal com $n = 2$ e Δ igual à matriz identidade. Em cada um dos sub-gráficos um dos elementos da matriz Δ é plotado, sendo que seu valor é representado no eixo das ordenadas e o valor do atraso (*lag*) é representado no eixo das abscissas.

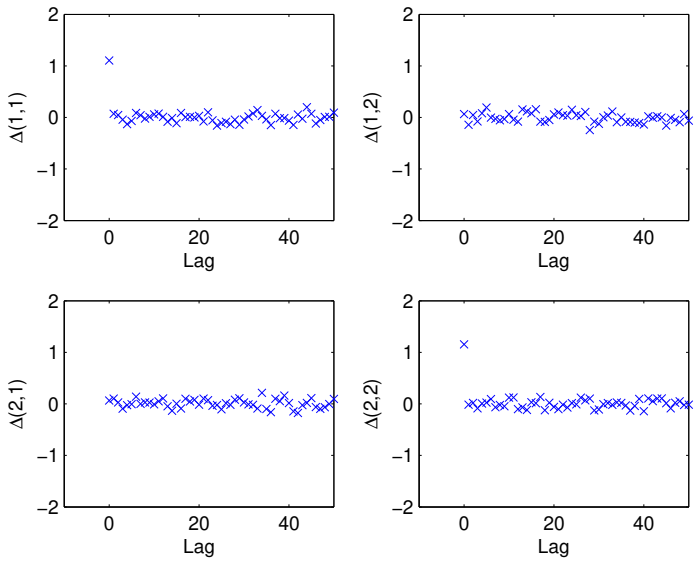

Fig. 7.2: Covariâncias de um ruído branco bidimensional criado com a rotina randn do MATLAB

Este sinal é conhecido como ruído branco pois sua densidade espectral é plana, o que pode ser observado facilmente ao se aplicar a transformada $\mathcal{Z}$ à covariância apresentada na equação 7.2. Desta forma, o sinal é semelhante à luz branca, que também tem densidade espectral plana. O fato de o ruído branco definido no domínio do tempo ter um espectro plano não significa que todo sinal com espectro plano tenha a transformada $\mathcal{Z}$ inversa que satisfaça a definição do ruído branco no domínio do tempo, conforme será discutido mais adiante nesta seção da tese.

O ruído branco é um sinal amplamente utilizado em processamento de sinais e também é o sinal padrão que deve ser usado na realização de séries temporais, conforme já apresentado no capítulo 5. Apesar deste sinal ser usado em diversas aplicações, não é possível gerá-lo exatamente conforme o sinal apresentado na equação 7.2. Sendo assim, ao se usar um sinal criado por um gerador pseudo-aleatório para se encontrar a realização de uma série temporal, os resultados não são muito precisos. Na figura 7.2 são apresentadas as covariâncias de um sinal criado por um gerador pseudo-aleatório existente no MATLAB. Ao se comparar estas covariâncias com as do ruído branco ideal apresentado na figura 7.1, nota-se claramente que este sinal não é um ruído branco ideal no domínio do tempo.

Nas covariâncias apresentadas na figura 7.2, além de se ter elementos de covariância diferentes de zero para atrasos diferentes de zero, tem-se também uma covariância aleatória para o atraso igual a zero. Conforme descrito no capítulo 5, um dos resultados dos algoritmos de realização de séries temporais é a covariância para atraso zero do ruído branco que deve ser dado como entrada no modelo em espaço de estado obtido, para que a saída deste modelo seja de fato uma realização da série temporal que se quer realizar. Sendo assim, para se aplicar na realização de séries temporais um ruído branco criado por um gerador pseudo-aleatório, deve-se forçá-lo a ter média zero e covariância para atraso zero igual à obtida a partir do método de realização de séries temporais.

Para transformar um sinal qualquer em um sinal com média zero e uma determindada covariância Δ para atraso zero, o seguinte procedimento pode ser adotado: Sejam μ_e a média e cov_e a covariância

para atraso zero de um sinal $e_s \in \Re^{nxN}$ gerado, por exemplo, por um gerador pseudo-aleatório. Seja Δ a covariância desejada para este sinal. O primeiro passo é transformar o sinal e_s em um sinal e_{0m} com média zero. Para isto a seguinte operação deve ser feita:

$$e_{0m} = e_s - \mu_e \mathbf{1}_{nXN} \tag{7.3}$$

em que $\mathbf{1}_{nXN}$ é a matriz unitária do espaço $\Re^{nXN}$.

Para transformar o sinal e_{0m} em um sinal e_l com covariância Δ para atraso zero, a seguinte operação deve ser feita:

$$e_l = T e_{0m} \tag{7.4}$$

Em que T pode ser encontrado da seguinte forma:

$$\begin{aligned}
& \Delta = E[e_l e_l^T] \Rightarrow \\
\Rightarrow \quad & \Delta = T E[e_{0m} e_{0m}^T] T^T \Rightarrow \\
\Rightarrow \quad & cov_e \Delta = cov_e T cov_e T^T \Rightarrow \\
\Rightarrow \quad & cov_e \Delta = cov_e T (T cov_e)^T \Rightarrow \\
\Rightarrow \quad & cov_e \Delta = (cov_e T)^2 \Rightarrow \\
\Rightarrow \quad & (cov_e \Delta)^{\frac{1}{2}} = cov_e T \Rightarrow \\
\Rightarrow \quad & T = cov_e^{-1} (cov_e \Delta)^{1/2}
\end{aligned} \tag{7.5}$$

Uma vez que Δ e cov_e são matrizes simétricas.

Mesmo sendo e_l um sinal com média zero e covariância Δ para atraso zero, suas covariâncias para atrasos diferentes de zero não são nulas e portanto este sinal não é um ruído branco no domínio do tempo. Na figura 7.3 são apresentadas as covariâncias de um sinal resultante da aplicação das equações 7.3 e 7.4 ao sinal cujas covariâncias são apresentadas na figura 7.2, para forçá-lo a ter uma covariância Δ para atraso zero igual à matriz identidade do espaço $\Re^{2X2}$ (denotada como $\mathcal{I}_{2x2}$). Da figura nota-se que a covariância para atraso zero de fato é Δ, como desejado. No entanto, as outras covariâncias não são nulas e portanto o sinal cujas covariâncias são apresentadas na figura 7.3 não é um ruído branco no domínio do tempo.

Como o sinal obtido não é um ruído branco ideal no domínio do tempo, o resultado da realização de uma série temporal usando este sinal como entrada não será tão preciso quanto seria caso fosse possível criar um ruído branco ideal no domínio do tempo. Para reduzir este problema, nesta seção é proposto um método para a geração de um sinal o mais próximo possível de um ruído branco ideal no domínio do tempo. Este método é baseado na minimização da distância entre as covariâncias de um sinal e as covariâncias do ruído branco ideal. Para se realizar esta minimização é proposto um método heurístico imuno-inspirado. A seguir, será apresentado como o problema de geração de ruído branco pode ser visto como um problema de otimização e mais adiante o algoritmo proposto para resolver este problema de otimização.

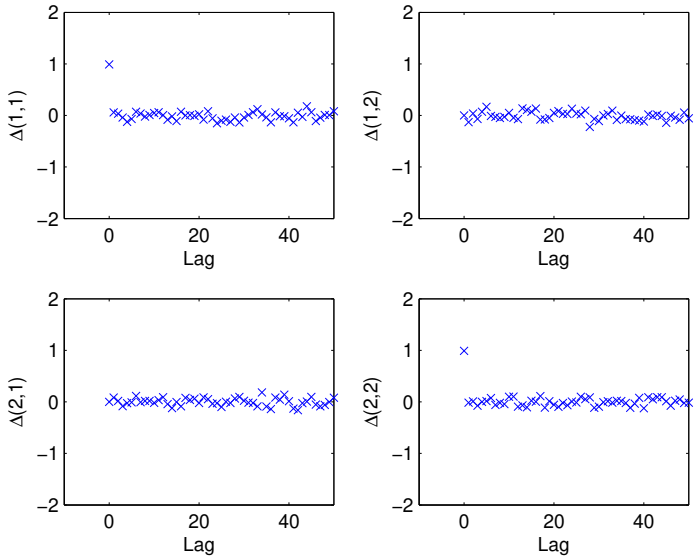

Fig. 7.3: Sinal com covariância $\mathcal{I}_{2x2}$ para atraso zero obtido com a aplicação da equação 7.4 ao sinal apresentado na figura 7.2

7.1.2 Geração de ruído branco vista como um problema de otimização

A geração de ruídos brancos pode ser vista como um problema de otimização ao se interpretar o problema como o de encontrar um sinal cujas covariâncias tenham a menor distância possível às covariâncias do ruído branco ideal. Algebricamente, o problema é encontrar uma sequência $e(k)$ cujas covariâncias satisfaçam a equação 7.2 para uma determinada matriz Δ. Os detalhes deste problema de otimização são apresentados a seguir:

Soluções candidatas

As soluções candidatas são as sequências de vetores $e_{est}(k) \in \Re^n$, em que n é a dimensão do ruído branco que se quer gerar e $k = 1 \ldots N$, em que N é o número de amostras que se quer gerar para o ruído branco.

Função objetivo

Conforme já discutido, deve-se encontrar a solução candidata que cujas covariâncias tenham a menor distância para as covariâncias do ruído branco ideal. Sendo assim, deve-se definir um critério de distância entre covariâncias, o que é feito a seguir:

Seja a matriz $\Delta_{est}(t) \in \Re^{nXn}$ a covariância da solução candidata e_{est} para o atraso t, sendo $t = 1 \ldots N$, ou seja, apenas para atrasos diferentes de zero, conforme definido a seguir:

$$\Delta_{est}(t) = E[e_{est}(k)e_{est}(k+t)^T] \tag{7.6}$$

A distância entre cada matriz $\Delta_{est}(t)$ e a matriz $\mathbf{0}_{nXn}$ é um escalar definido como $d(t)$ e segue a equação abaixo:

$$d(t) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\Delta_{est}(t)_{(i,j)}\right)^2} \tag{7.7}$$

em que $\Delta_{est}(t)_{(i,j)}$ é o termo da linha i e coluna j da matriz $\Delta_{est}(t)$. A definição desta distância entre matrizes é baseada na distância euclidiana entre dois vetores, que é a raiz da soma dos quadrados das diferenças entre cada termo dos dois vetores. Como neste caso uma das matrizes é nula, a diferença entre os termos é simplesmente o termo da matriz Δ_{est}.

Definida a distância, basta definir agora a função objetivo do problema. Seja $w(t)$ um peso qualquer, definido para o instante t. A função objetivo $f(e_{est}(k))$ a ser maximizada é então definida da seguinte forma:

$$f(e_{est}(k)) = \frac{\sum_{t=1}^{N} w(t)}{\sum_{t=1}^{N} w(t)d(t)} \tag{7.8}$$

Esta função objetivo é maior a medida que as covariâncias de e_{est} para atrasos diferentes de zero se aproximam de $\mathbf{0}_{nXn}$. O vetor de pesos é introduzido para aumentar a influência de covariâncias de atrasos mais próximos de zero no cálculo das distâncias, uma vez que covariâncias para atrasos próximos de zero têm mais influência nos resultados da realização de séries temporais. A somatória dos pesos foi introduzida no numerador da função objetivo para permitir a comparação dos resultados de funções objetivos com diferentes vetores de pesos e até mesmo com diferentes números de amostras. Em linhas gerais, esta função objetivo nada mais é que o inverso de uma média ponderada das distâncias das covariâncias das soluções candidatas para atrasos diferentes de zero e o zero do espaço em que estas covariâncias são definidas.

Enunciado do problema de otimização

Tomando por base as definições acima, o problema de otimização que leva à obtenção de um ruído branco é o seguinte:

> *Sejam as sequências de vetores n-dimensionais $e_{est}(k)$, $k = 1 \ldots N$, em que N é o número de amostras e n é a dimensão do ruído branco que se quer gerar. Determine a sequência $e_{est}(k)$ que maximiza a seguinte função objetivo:*
>
> $$f(e_{est}(k)) = \frac{\sum_{t=1}^{N} w(t)}{\sum_{t=1}^{N} w(t)d(t)}$$
>
> *em que $w(t)$ é uma função de pesos, que pode ser escolhida de maneira conveniente, e $d(t)$ é a distância entre a covariância da solução candidata $e_{est}(k)$ e o zero do espaço em que as covariâncias estão definidas, ou seja:*
>
> $$d(t) = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left(\Delta_{est}(t)_{(i,j)}\right)^2}$$

e $\Delta_{est}(t)_{(i,j)}$ é o termo na i-ésima linha e j-ésima coluna da matriz $\Delta_{est}(t)$, que é a covariância para atraso t da solução candidata $e_{est}(k)$, conforme abaixo:

$$\Delta_{est}(t) = E[e_{est}(k)e_{est}(k+t)^T]$$

7.1.3 Algoritmo proposto

O algoritmo proposto para solução do problema de otimização apresentado na seção anterior é baseado no algoritmo *opt-aiNet*, apresentado no capítulo 6 desta tese. Para aplicação deste algoritmo são necessárias algumas definições, feitas a seguir:

Anticorpos

Ao se aplicar os algoritmos imuno-inspirados para solução de um problema de otimização, os anticorpos devem ter uma relação direta as soluções candidatas. No caso deste problema, como as soluções candidatas são sequências de vetores de dimensão n com N amostras, os anticorpos escolhidos são matrizes $e_{est} \in \Re^{nXN}$ em que N é o número de amostras que se quer gerar de ruído branco e n é a dimensão deste ruído branco. A k-ésima coluna da matriz e_{est} representa a amostra da série temporal candidata no instante de tempo $k-1$.

Função de fitness

A função de fitness escolhida é a própria função objetivo do problema, definida na equação 7.8.

Peculiaridades de implementação

Devido à natureza do problema, o algoritmo proposto apresenta algumas diferenças com relação ao algoritmo *opt-aiNet* original. Estas diferenças são descritas a seguir:

Inicialmente, após a geração aleatória da população inicial, é feita uma etapa de tratamento dos anticorpos com as equações 7.3 e 7.4, de forma que suas médias são forçadas a serem nulas e suas covariâncias para atraso zero são forçadas a serem uma matriz Δ escolhida pelo usuário. Este tratamento numérico é repetido na etapa em que novos anticorpos aleatórios são introduzidos na população, ao final de cada iteração. De forma semelhante, após cada clonagem, os anticorpos gerados pelos clones perturbados são forçados a terem média zero e covariância para atraso zero igual a Δ.

Com os procedimentos destacados acima, garante-se que todos anticorpos na população representarão sequências com média nula e covariância para atraso zero igual a um determinado Δ estipulado. Consequentemente, o objeto de otimização se torna apenas a distância entre as covariâncias para atrasos diferentes de zero e a matriz nula $\mathbf{0}_{nXn}$. Como as covariâncias para atrasos mais próximos de zero são melhor estimadas, uma vez que há mais amostras disponíveis para o cálculo destes valores, a distância das covariâncias para atrasos mais próximos de zero têm sua importância destacada na função objetivo a ser otimizada devido à introdução do vetor de pesos w conveniente para este objetivo.

Uma outra alternativa de implementação do algoritmo seria incluir na função objetivo também a distância entre a covariância para atraso zero das soluções candidatas e a matriz Δ que representa a covariância desejada para atraso zero. Neste caso, não seria necessário aplicar a equação 7.4 aos anticorpos na geração da população inicial, durante a etapa de inserção de indivíduos aleatórios e aos

clones perturbados. Esta variação chegou a ser implementada, mas os resultados não foram tão bons quanto a alternativa descrita nos parágrafos acima, pois não se garante que o valor da covariância de atraso zero do indivíduo de maior *fitness* seja exatamente igual à matriz Δ desejada. Desta forma, o resultado da realização das séries temporais usando os indivíduos criados com esta alternativa não é tão bom quanto o encontrado com a alternativa anterior.

O fluxograma do algoritmo proposto é apresentado na figura 7.4

7.1.4 Exemplo de aplicação do ruído branco à realização de séries temporais

O algoritmo proposto foi implementado e os ruídos brancos encontrados foram aplicados na realização de uma série temporal usando o método proposto por Aoki, descrito no capítulo 5. Para gerar a realização da série temporal que servirá como benchmark, um ruído branco foi aplicado a modelo em espaço de estado descrito na equação 5.82, tendo as seguintes matrizes:

$$A = \begin{bmatrix} 0.2128 & 0.1360 & 0.1979 & -0.0836 \\ 0.1808 & 0.4420 & -0.3279 & 0.2344 \\ -0.5182 & 0.1728 & -0.5488 & -0.3083 \\ 0.2252 & -0.0541 & -0.4679 & 0.8290 \end{bmatrix} \tag{7.9}$$

$$C = \begin{bmatrix} 0.6557 & -0.2502 & -0.5188 & -0.1229 \\ 0.6532 & -0.1583 & -0.055 & -0.2497 \end{bmatrix} \tag{7.10}$$

$$K = \begin{bmatrix} -0.0016 & 0.2209 \\ -1.6146 & -1.0061 \\ -1.2287 & -0.4531 \\ 0.2047 & 1.3995 \end{bmatrix} \tag{7.11}$$

Após a geração da realização da série temporal a partir do modelo acima, o procedimento de realização proposto por Aoki descrito no capítulo 5 foi aplicado e as seguintes matrizes foram estimadas:

$$A_{est} = \begin{bmatrix} 0.5116 & 0.5091 & -0.2145 & 0.1607 \\ 0.1397 & 0.3565 & 0.6143 & 0.0786 \\ 0.1967 & -0.4372 & 0.0632 & 0.8832 \\ 0.2177 & -0.2489 & -0.5926 & -0.2344 \end{bmatrix} \tag{7.12}$$

$$C_{est} = \begin{bmatrix} -0.8870 & 0.5243 & -0.1399 & 0.2808 \\ -0.8603 & -0.0319 & -0.1230 & 0.0332 \end{bmatrix} \tag{7.13}$$

$$K_{est} = \begin{bmatrix} 0.1426 & -0.4116 \\ 0.2779 & 0.0443 \\ 0.0265 & 0.0875 \\ 0.2648 & -0.0092 \end{bmatrix} \tag{7.14}$$

Apesar de as matrizes A_{est} e C_{est} encontradas serem diferentes das matrizes A e C originais, os parâmetros de Markov do sistema original e do sistema estimado são da mesma ordem de grandeza, conforme pode ser constatado com os resultados abaixo. Nas equações 7.15, 7.16 e 7.17 são apresentados os primeiros três parâmetros de Markov do sistema original e nas equações 7.18, 7.19 e 7.20 são

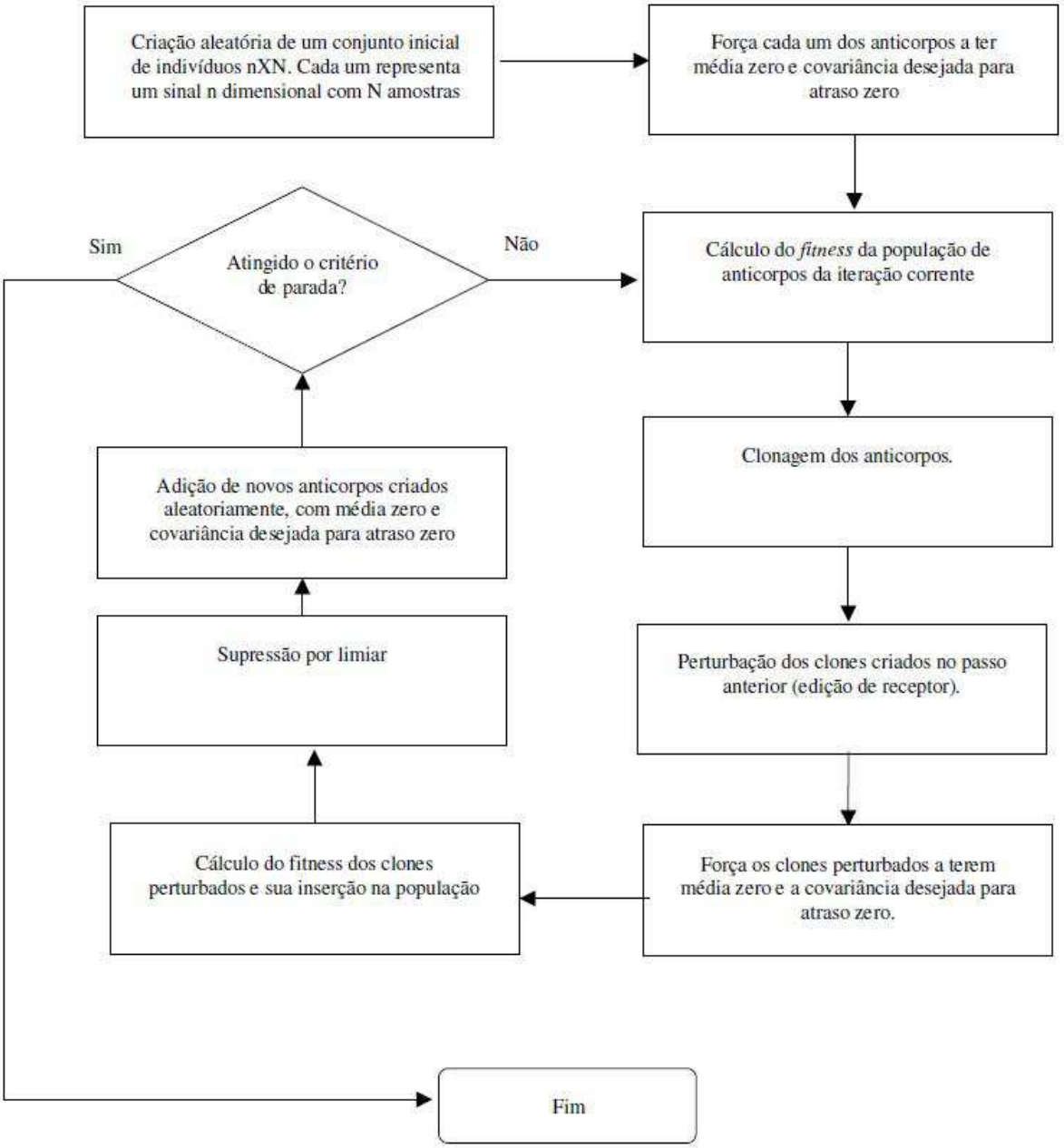

Fig. 7.4: Fluxograma do algoritmo imuno-inspirado para geração de ruídos brancos

apresentados os três primeiros parâmetros de Markov do sistema encontrado com o método proposto por Aoki:

$$CM = \begin{bmatrix} 0.7114 & 1.0748 \\ 0.0672 & 1.1113 \end{bmatrix} \tag{7.15}$$

$$CAM = \begin{bmatrix} -0.3805 & 0.5667 \\ -0.4447 & 0.4711 \end{bmatrix} \tag{7.16}$$

$$CA^2M = \begin{bmatrix} -0.0494 & 0.0459 \\ -0.1878 & 0.1962 \end{bmatrix} \tag{7.17}$$

$$C_{est}M_{est} = \begin{bmatrix} 0.6011 & 1.1042 \\ 0.1181 & 0.9888 \end{bmatrix} \tag{7.18}$$

$$C_{est}A_{est}M_{est} = \begin{bmatrix} -0.2964 & 0.3623 \\ -0.3340 & 0.4727 \end{bmatrix} \tag{7.19}$$

$$C_{est}A_{est}^2M_{est} = \begin{bmatrix} -0.0713 & 0.2320 \\ -0.1668 & 0.3341 \end{bmatrix} \tag{7.20}$$

A matriz de covariância para atraso zero Δ_{est} do ruído branco que deve ser aplicado ao sistema formado pelas matrizes A_{est}, K_{est} e C_{est} para a geração da série temporal determinada pelo método proposto por Aoki é a seguinte:

$$\Delta_{est} = \begin{bmatrix} 2.3747 & 0.6450 \\ 0.6450 & 2.1173 \end{bmatrix} \tag{7.21}$$

Encontrado o modelo em espaço de estado descrito acima, duas abordagens foram adotadas para a criação de um ruído branco a ser aplicado como entrada no modelo para a realização da série temporal. A primeira delas foi a simples aplicação das equações 7.3 e 7.4 a uma sequência de números obtida com um gerador pseudo aleatório, forçando esta sequência a ter média zero e a covariância para atraso zero Δ_{est} descrita acima. A segunda delas foi a aplicação do algoritmo proposto nesta tese. A variação em função do atraso dos quatro elementos das matrizes de covariância dos dois ruídos brancos gerados são apresentados respectivamente nas figuras 7.5 e 7.6. A partir das figuras nota-se que o ruído branco gerado com o método proposto nesta tese é mais próximo do ruído branco ideal (ver figura 7.1) que o sinal criado com o gerador pseudo-aleatório.

Os sinais apresentados nas figuras 7.5 e 7.6 foram aplicados como entrada no modelo formado pelas matrizes A_{est}, K_{est} e C_{est} encontradas com o método de realização de séries temporais. As covariâncias das saídas dos modelos foram calculadas e plotadas juntamente com as covariâncias da série temporal original. Nas figuras 7.7, 7.8, 7.9 e 7.10 são apresentados os termos das covariâncias da série temporal de benchmark e de saída do modelo estimado tendo como entrada o ruído branco gerado com a rotina randn do MATLAB. Nas figuras 7.11, 7.12, 7.13 e 7.14 são apresentados os termos de covariância da série temporal de benchmark e os termos de covariância da série temporal obtida ao se aplicar ao modelo estimado o ruído branco obtido com o método proposto nesta tese. A partir das figuras apresentadas nota-se que os resultados obtidos com o uso da entrada definida pelo método proposto nesta tese são melhores que os resultados obtidos com o uso de um ruído branco criado por um gerador pseudo-aleatório.

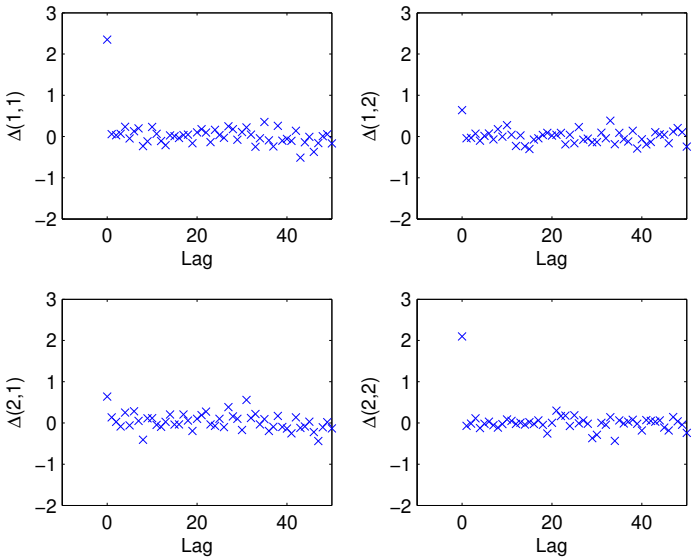

Fig. 7.5: Covariâncias do ruído branco gerado com o gerador pseudo-aleatório

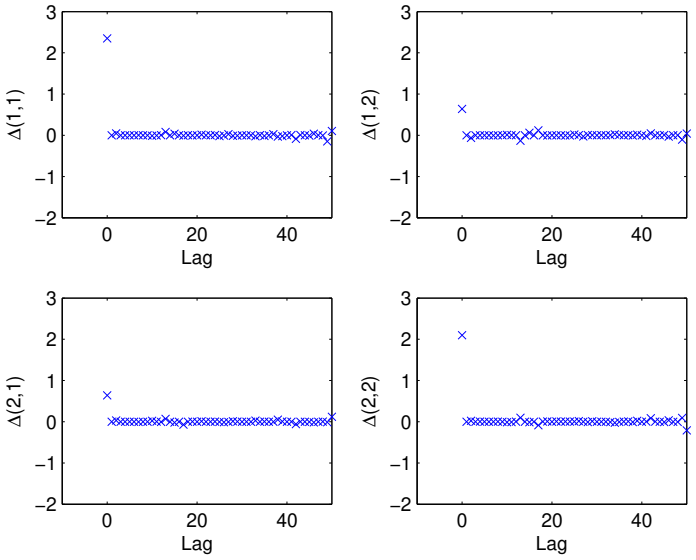

Fig. 7.6: I Covariâncias do ruído branco gerado com o algoritmo proposto nesta tese

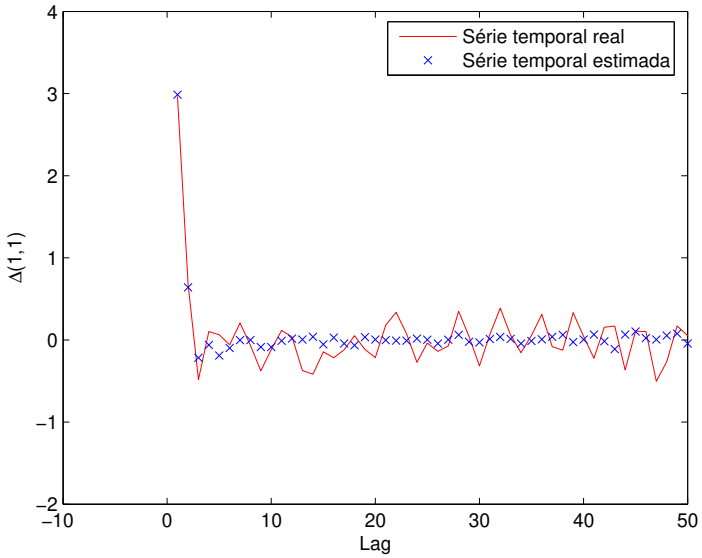

Fig. 7.7: Termo (1,1) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado por um gerador pseudo-aleatório no modelo encontrado com o método Aoki.

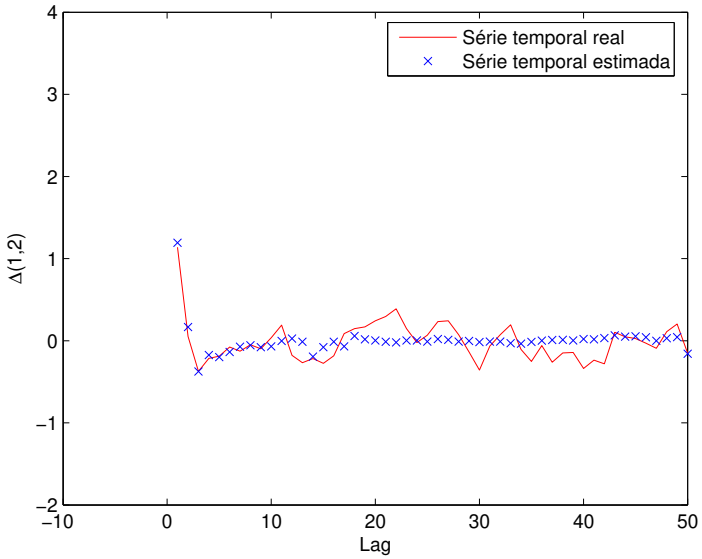

Fig. 7.8: Termo (1,2) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado por um gerador pseudo-aleatório no modelo encontrado com o método Aoki.

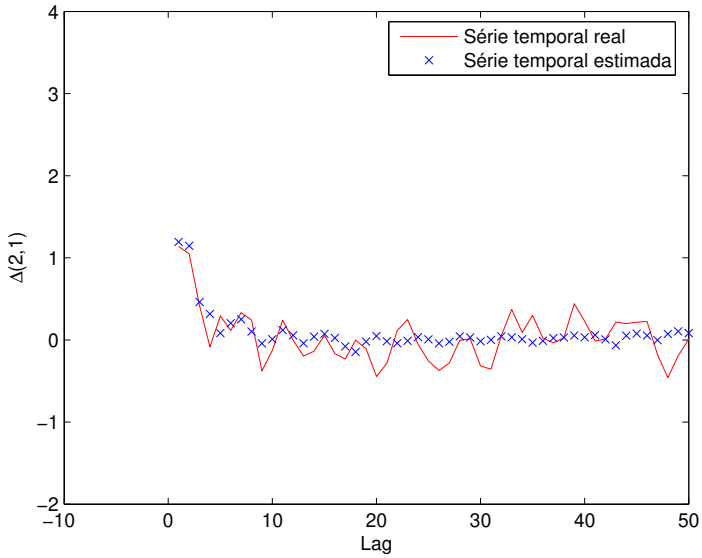

Fig. 7.9: Termo (2,1) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado por um gerador pseudo-aleatório no modelo encontrado com o método Aoki.

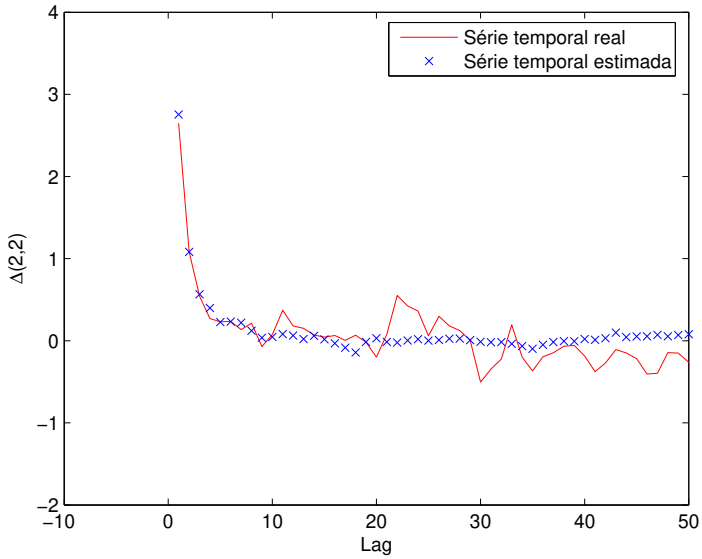

Fig. 7.10: Termo (2,2) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado por um gerador pseudo-aleatório no modelo encontrado com o método Aoki.

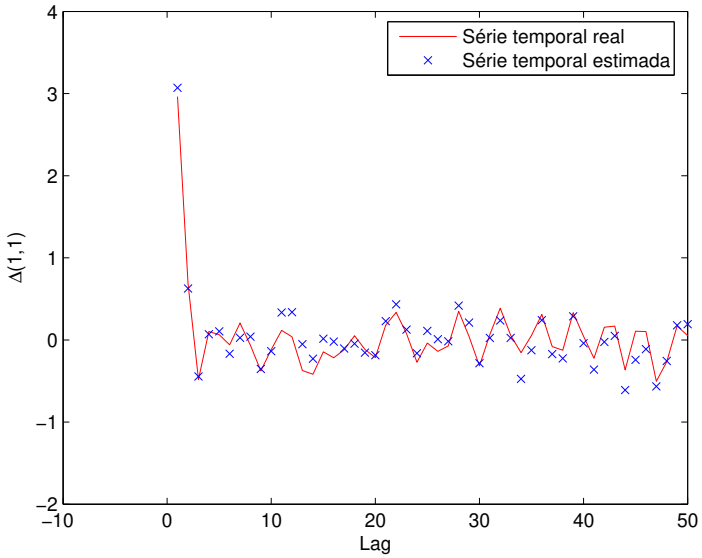

Fig. 7.11: Termo (1,1) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado pelo algoritmo proposto nesta tese no modelo encontrado com o método Aoki.

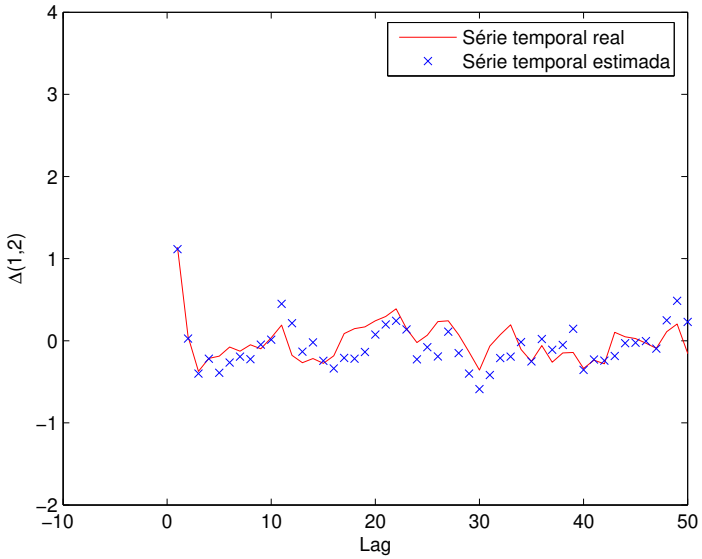

Fig. 7.12: Termo (1,2) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado pelo algoritmo proposto nesta tese no modelo encontrado com o método Aoki.

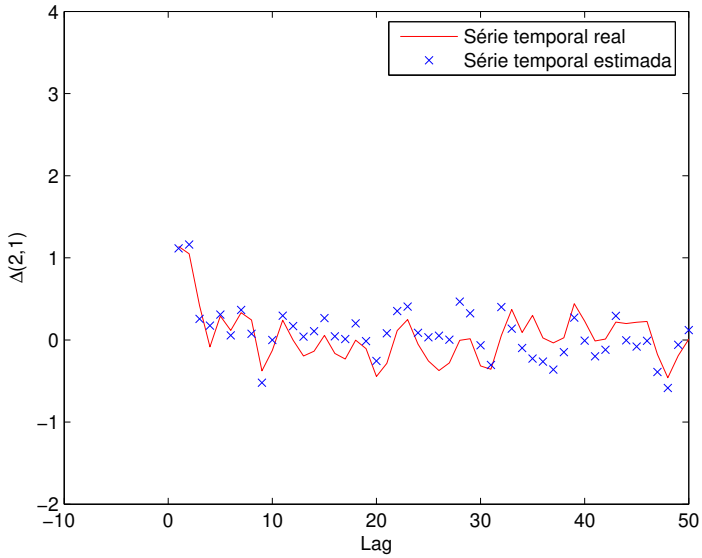

Fig. 7.13: Termo (2,1) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado pelo algoritmo proposto nesta tese no modelo encontrado com o método Aoki.

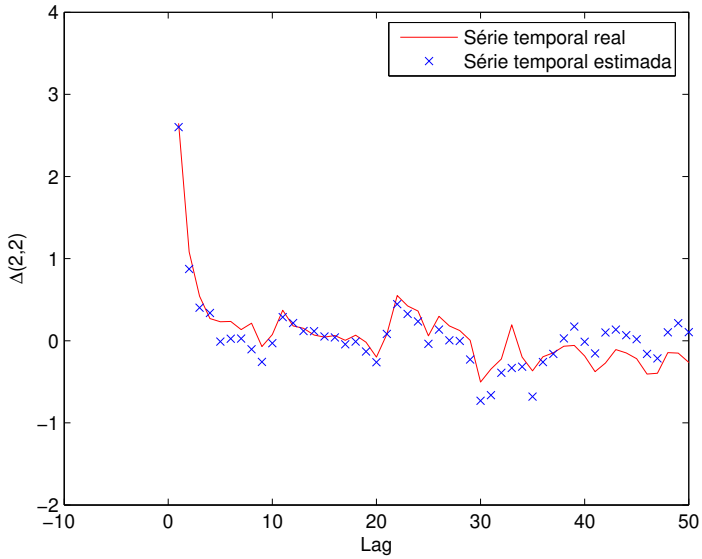

Fig. 7.14: Termo (2,2) das covariâncias da série temporal a ser realizada e da realização obtida ao se aplicar um ruído branco criado pelo algoritmo proposto nesta tese no modelo encontrado com o método Aoki.

7.1.5 Influência do número de amostras no método proposto

Para avaliar a influência do número de amostras desejado para o ruído branco no *fitness* do melhor indivíduo obtido pelo método proposto neste artigo, este método e a rotina randn do MATLAB foram executados para diferentes números de amostras e os *fitness* foram comparados.

Para cada diferente número de amostras N para o qual o algoritmo imuno-inspirado foi executado, foi necessário variar o limiar de supressão, uma vez que a dimensão do espaço de busca (que é nxN conforme discutido anteriormente) varia em função do número de amostras. Caso o limiar de supressão não fosse alterado, o número de anticorpos na população tenderia a um valor desnecessariamente grande, aumentando o esforço computacional necessário a cada iteração.

Para ajustar o limiar de supressão em função do número de amostras, um método automático de regulação é proposto. Este método funciona da seguinte forma: o algoritmo é inicializado com o menor número de amostras para o qual se quer realizar o cálculo. A cada variação de número de amostras, o número de indivíduos na primeira iteração do algoritmo imuno-inspirado é comparado a um número máximo de indivíduos desejado. Caso o número de indivíduos na primeira iteração seja maior que o número máximo, o algoritmo imuno-inspirado é reinicializado para aquele mesmo número de amostras, mas o limiar de supressão é acrescido de um valor fixo. Isto é repetido até que se tenha um número de indivíduos após a primeira iteração menor que o valor máximo de indivíduos. Desta forma, se torna possível executar o algoritmo imuno-inspirado sem necessidade de ajustes manuais em função da variação do espaço de busca.

Na figura 7.15 o melhor *fitness* obtido pelo método proposto nesta tese é plotado em função do número de amostras. Na figura também é apresentado o *fitness* da solução obtida com a rotina randn do MATLAB em função do número de amostras. A partir da figura, observa-se que a medida que o número de amostras aumenta, o *fitness* obtido com o algoritmo imuno-inspirado diminui devido ao aumento do espaço de busca, uma vez que em um espaço maior a obtenção de uma solução ótima é mais custosa. Por outro lado, o *fitness* da rotina randn do MATLAB cresce cada vez mais, uma vez que, a medida que o número de amostras tende a infinito, se tem melhores estimativas da covariância das séries e o algoritmo pseudo-aleatório é tal que sua covariância tende à do ruído branco quando o número de amostras tende a infinito.

A partir da figura 7.15 nota-se que há um limite de número de amostras para o qual o algoritmo proposto nesta tese é vantajoso.

7.1.6 Análise espectral

O espectro dos ruídos brancos obtidos com a rotina randn do MATLAB e com o algoritmo proposto neste artigo foram calculados para todos os números de amostras discutidos na seção anterior e em todos os casos o espectro é plano e igual à matriz identidade do espaço $\Re^{2x2}$. Na figura 7.16 são apresentados os espectros para o experimento em que o número de amostras é igual a 100.

Com este resultado, nota-se que mesmo não sendo possível determinar um ruído branco ideal no domínio do tempo, os sinais gerados pelos dois algoritmos são ruídos brancos no domínio da frequência, ou seja, têm um espectro plano.

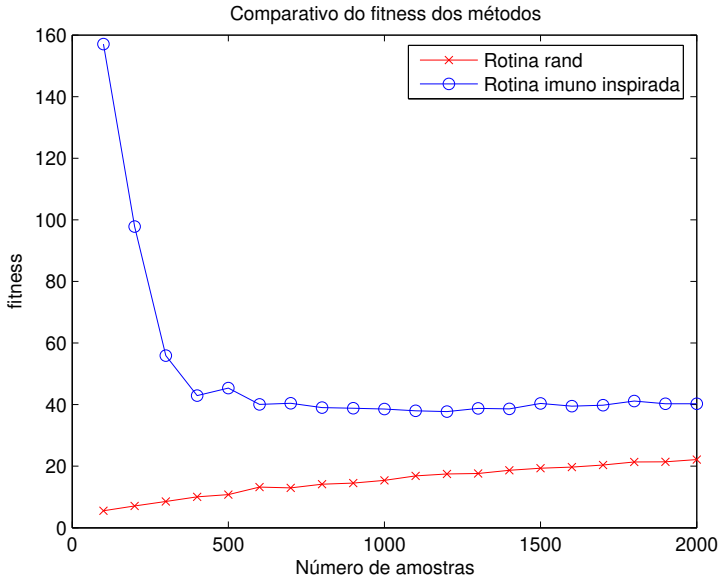

Fig. 7.15: *Fitness* máximo obtido em função do número de amostras para o algoritmo proposto neste artigo e para a rotina randn do MATLAB

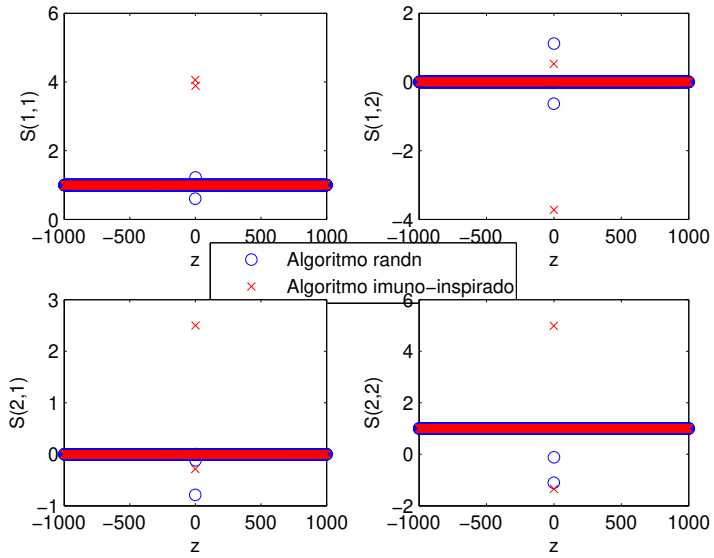

Fig. 7.16: Espectro do ruído branco para número de amostras igual a 100.

7.1.7 Conclusões

Nesta seção foi proposto um método imuno-inspirado para a geração de sequências de números o mais próximas o possível de um ruído branco no domínio do tempo. Este objetivo é atingido ao se interpretar o problema de geração de ruídos brancos como um problema de otimização e ao se aplicar um algoritmo imuno-inspirado para resolver este problema. No domínio do tempo, ou seja, ao se observar as covariâncias, os resultados obtidos com o algoritmo proposto nesta seção são melhores que os resultados obtidos pela rotina randn do MATLAB quando o número de amostras é limitado a um certo número. Consequentemente, ao se aplicar o sinal obtido nesta seção como entrada de um modelo que gera a realização de uma série temporal, os resultados são melhores que os obtidos ao se aplicar como entrada no modelo um sinal gerado pelo gerador pseudo-aleatório randn do MATLAB.

Também foi demonstrado que o algoritmo proposto tem desempenho melhor do que a rotina randn para um certo número de amostras. Para um número de amostras tendendo a ∞, o algoritmo proposto tem algumas limitações devidas ao aumento do espaço de busca.

A partir da análise espectral dos sinais gerados pelo algoritmo proposto nesta seção e por um gerador pseudo-aleatório, nota-se que ambos têm espectro plano, ou seja, ambos são ruídos brancos no domínio da frequência. Isto leva à conclusão de que a definição de ruído branco depende do domínio que se está tratando e que um ruído branco no domínio da frequência não necessariamente é um ruído branco no domínio do tempo. Como para a realização de séries temporais é necessário o uso de um ruído branco com número de amostras limitadas, no domínio do tempo, não é qualquer ruído branco no domínio da frequência que implica em bons resultados na realização de séries temporais.

7.2 Proposta para solução da equação de Riccati

Como visto no capítulo 5, o método proposto por Aoki leva a uma equação algébrica de Riccati 5.105, cuja solução Σ é fundamental para a conclusão da resolução do problema de realização de séries temporais. No entanto, em alguns casos, a amostra disponível da série temporal leva a matrizes A, M_2 e C tais que a solução da equação algébrica de Riccati encontrada não é possível. Nestes casos, para determinar soluções aproximadas Σ, o problema de solução de equação de Riccati foi reformulado como um problema de otimização, que foi solucionado com um algoritmo imuno-inspirado.

7.2.1 Solução da equação de Riccati como um problema de otimização

O problema de otimização relacionado à solução da equação de Riccati tem suas características destacadas a seguir

Soluções candidatas

As soluções candidatas são as matrizes $\Sigma \in \Re^{nXn}$ definidas positivas e simétricas, em que n é a ordem do modelo que se quer estimar.

Função objetivo

Sejam A, M_2, C e $\Lambda(0)$ as matrizes resultantes das decomposições das matrizes de covariâncias envolvidas nos primeiros passos da resolução do problema de realização de séries temporais pelo método Aoki. A função objetivo é o valor absoluto da equação algébrica de Riccati 5.105 com todos os elementos não nulos isolados de um dos lados da igualdade, ou seja:

$$f(\Sigma) = |\Sigma - A\Sigma A^T - (M_2 - A\Sigma C^T)(\Lambda(0) - C\Sigma C^T)^{-1}(M_2 - A\Sigma C^T)^T| \tag{7.22}$$

Se for encontrado um Σ tal que a função $f(\Sigma)$ definida acima seja igual ao zero do espaço $\Re^{nXn}$, este Σ é uma solução exata da equação algébrica de Riccati. Caso não exista solução exata, o valor de Σ que minimiza a função acima é uma aproximação da solução da equação algébrica de Riccati.

Restrições

As restrições impostas ao problema se devem ao fato de a matriz Σ ser uma matriz de covariância, e que portanto, deve ser definida positiva e simétrica. Outra restrição tem relação com a matriz Δ, que é calculada a partir de Σ. Esta é uma matriz de autocovariância, portanto todos seus elementos devem ser positivos.

Enunciado do problema de otimização

Resumindo a discussão acima, o problema de solução da equação de Riccati pode ser transformado no seguinte problema de otimização:

Dadas as matrizes A, M_2, C e $\Lambda(0)$, encontre a matriz Σ que minimiza

$$f(\Sigma) = |\Sigma - A\Sigma A^T - (M_2 - A\Sigma C^T)(\Lambda(0) - C\Sigma C^T)^{-1}(M_2 - A\Sigma C^T)^T|$$

Sujeita a:

$$\Sigma > 0$$

$$\Sigma = \Sigma^T$$

$$\Delta_{jk} = (\Lambda_0 - C\Sigma C^T)_{jk} > 0$$

em que Δ_{jk} é o elemento na j-ésima linha e k-ésima coluna da matriz Δ.

7.2.2 Algoritmo proposto

As principais definições a respeito do algoritmo imuno-inspirado proposto para a solução do problema de otimização que leva à solução da equação algébrica de Riccati são destacadas nesta seção.

Anticorpos

Neste caso os anticorpos são simplesmente matrizes $\Sigma \in \Re^{nXn}$, que são as soluções candidatas do problema de otimização.

Função de fitness

A função de fitness para este problema deve ser tal que, quanto menor a distância da função objetivo $f(\Sigma)$ definida na equação 7.22, aplicada a um determinado anticorpo Σ, ao zero do espaço $\Re^{nXn}$, maior é o valor do fitness daquele anticorpo. Isto é feito para que o problema possa ser tratado como um problema de maximização. Com isto, a função de fitness é definida da seguinte maneira:

$$fitness(\Sigma) = \frac{1}{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n} f(\Sigma)_{(i,j)}^2}} \tag{7.23}$$

em que $f(\Sigma)_{(i,j)}$ é o termo da linha i e da coluna j da matriz $f(\Sigma)$, que é o resultado da função objetivo definida na equação 7.22 aplicada a um determinado anticorpo Σ.

Peculiaridades de implementação

Como o problema de otimização a ser resolvido tem restrições, o algoritmo *opt-aiNet* foi alterado resultando em um algoritmo denominado Imuno-Riccati. As principais características do novo algoritmo proposto que o diferem de um algoritmo imuno-inspirado comum são as seguintes:

- Criação de soluções candidatas factíveis:

  Toda solução candidata no conjunto inicial de anticorpos deve ser factível. Para garantir que isto ocorra, um laço gera anticorpos e testa sua factibilidade, aplicando ao anticorpo as restrições do problema. Caso o anticorpo seja uma solução factível, ele é mantido na população. Caso contrário, ele é descartado. Este laço é mantido até que se encontre um número de soluções factíveis igual ao número de anticorpos inicial desejado.

- Mutações:

  As soluções Σ devem ser sempre simétricas. Para manter a simetria das soluções ao se aplicar as mutações, todas as matrizes de mutação geradas são simétricas. Desta maneira se garante que a restrição $\Sigma = \Sigma^T$ é sempre satisfeita.

- Supressão de clones não factíveis:

  Para garantir a factibilidade das soluções exploradas pelo algoritmo, após o processo de clonagem, todos os anticorpos são submetidos à verificação de sua factibilidade. Caso a perturbaçao dos clones tenha implicado em criação de um anticorpo não factível, outra perturbação é gerada e adicionada ao clone, até que se chegue a um clone perturbado factível.

- Inserção de novos anticorpos:

  A exemplo do que é feito na criação do conjunto de anticorpos inicial, os novos anticorpos inseridos ao final de cada laço do algoritmo são testados para verificar a factibilidade. São inseridos na população apenas novos indivíduos factíveis.

- Critério de parada:

  O critério de parada do algoritmo é o erro admissível na solução. Neste caso específico, o algoritmo é encerrado quando a distância euclidiana entre o melhor anticorpo da população e o zero do espaço em que o anticorpo é definido é menor que um resíduo dado como parâmetro de entrada ao algoritmo.

Na figura 7.17 é apresentado um fluxograma do algoritmo imuno-riccati contendo as características destacadas acima.

Para se ter uma boa performance do algoritmo imuno-inspirado é necessário que se faça um ajuste correto de seus parâmetros de controle, que são listados a seguir:

- Ordem de grandeza

  As soluções candidatas são escolhidas como matrizes aleatórias do espaço $\Re^{nXn}$. No entanto, as soluções desejadas para a equação de Riccati estão limitadas a um subespaço de $\Re^{nXn}$. Sendo assim, as soluções candidatas podem ser limitadas a um hipercubo do espaço $\Re^{nXn}$, que tem a dimensão definida pela variável ordem de grandeza. Se a ordem de grandeza é muito alta, o tempo de convergência da solução também será muito alto, uma vez que haverá um grande espaço de busca. Por outro lado, se este número for muito baixo, é possível que a solução do problema de otimização esteja fora do hipercubo delimitado e não será encontrada pelo algoritmo.

- Grau de perturbação

  A mutação dos anticorpos consiste em somá-los a uma determinada perturbação. Neste problema específico, as perturbações são matrizes simétricas do espaço $\Re^{nXn}$ com elementos escolhidos aleatoriamente a partir de uma distribuição uniforme. A amplitude desta distribuição uniforme é definida como sendo o grau de perturbação. Se esta variável for relativamente grande, os clones podem cair em uma região do espaço que atraia para um ponto ótimo diferente daquele explorado pelo anticorpo que o deu origem, levando à não exploração de um ótimo local. Por outro lado, se esta variável for muito pequena, o clone será muito próximo do anticorpo que o gerou, levando a um maior número de iterações necessárias para se chegar ao ótimo da região explorada.

- Taxa de decaimento do grau de perturbação

  A medida que o algoritmo vai evoluindo, espera-se que os anticorpos estejam cada vez mais próximos dos pontos ótimos a que se quer chegar. Desta forma, é necessário que o grau de perturbação diminua ao longo das iterações para que os clones fiquem cada vez mais próximos dos anticorpos que os geraram e sejam capazes de explorar detalhadamente as regiões próximas dos ótimos locais do espaço. Sendo assim, é definida a variável taxa de decaimento do grau de perturbação, que indica o quanto o grau de perturbação deve diminuir a cada iteração. Se este valor for muito baixo, o grau de perturbação levará muitas iterações para diminuir, levando a uma exploração grosseira do espaço de busca por um maior número de iterações. Por outro lado, se esta variável tiver um valor muito grande, o grau de perturbação ficará muito pequeno em iterações em que ainda não se atingiu os arredores dos ótimos locais, fazendo com que o algoritmo leve mais iterações para atingir esta região, o deixando em geral mais lento.

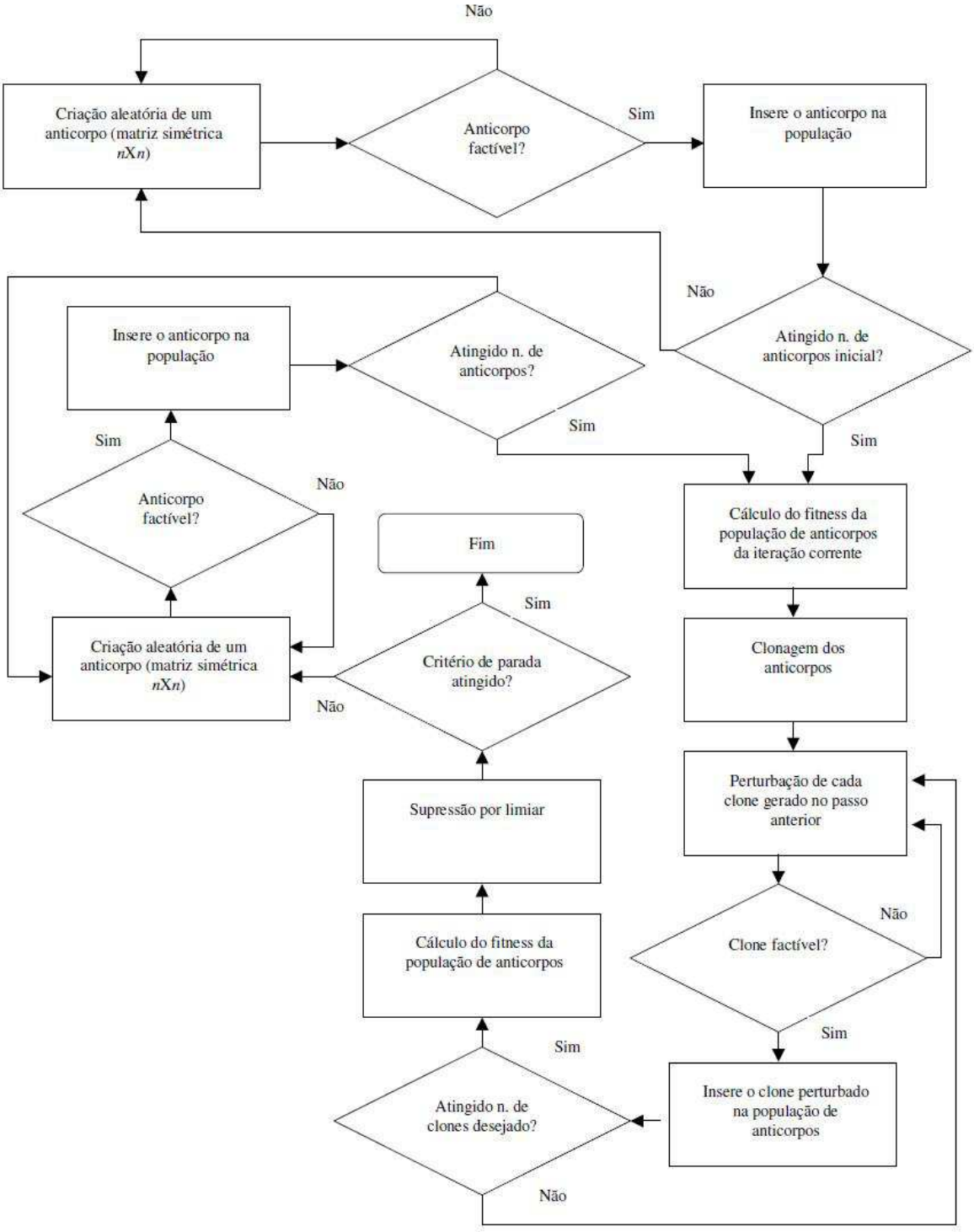

Fig. 7.17: Fluxograma do algoritmo Imuno-Riccati

- Limiar de supressão

  Esta variável define qual é a máxima distância entre dois anticorpos que faz com que o de menor *fitness* seja eliminado da população. No problema discutido, a distância entre dois anticorpos é definida como a distância euclidiana d no espaço $\Re^{nXn}$. Sejam A e B duas matrizes nXn. A distância euclidiana d entre as matrizes A e B é definida como:

$$d = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}(a_{ij} - b_{ij})^2} \tag{7.24}$$

  em que a_{ij} e b_{ij} são os termos de A e B.

  Se o limiar de supressão for relativamente alto, o anticorpo que estiver próximo de um ótimo local pode eliminar outro anticorpo que esteja próximo de outro ótimo local, fazendo com que nem todos ótimos locais sejam encontrados. Por outro lado, se este valor for relativamente pequeno, haverá um excesso de anticorpos explorando um mesmo ótimo local, levando a uma execução mais lenta do algoritmo.

- Menor *fitness* de parada

  Este parâmetro tem relação direta com o critério de parada. Trata-se do valor mínimo admissível para o *fitness* da solução que permite que o algoritmo pare, ou seja, esta variável indica exatamente a partir de qual *fitness* se está satisfeito com a solução encontrada. Se este valor for muito baixo, soluções menos exigentes serão obtidas. Se, por outro lado, este valor for extremamente alto, o *fitness* desejado pode não ser nunca atingido, levando a um loop infinito no algoritmo imuno-inspirado. Isto ocorre pois, no caso estudado nesta tese, a equação de Riccati proposta não tem solução exata, conforme ficará claro mais adiante. No caso em que se busca a solução para uma equação algébrica de Riccati que tem solução exata, não há limite superior para a escolha do menor *fitness* de parada.

7.2.3 O problema de identificação

A série temporal usada para testar a eficácia do método proposto foi gerada a partir de um sinal $e(k)$ aplicado a um sistema MIMO linear, invariante no tempo e estável com a estrutura apresentada na equação 5.82 e com as seguintes matrizes:

$$A = \begin{bmatrix} 0.2128 & 0.1360 & 0.1979 & -0.0836 \\ 0.1808 & 0.4420 & -0.3279 & 0.2344 \\ -0.5182 & 0.1728 & -0.5488 & -0.3083 \\ 0.2252 & -0.0541 & -0.4679 & 0.8290 \end{bmatrix} \tag{7.25}$$

$$K = \begin{bmatrix} -0.0016 & 0.2209 \\ -1.6146 & -1.0061 \\ -1.2287 & -0.4531 \\ 0.2047 & 1.3995 \end{bmatrix} \tag{7.26}$$

$$C = \begin{bmatrix} 0.6557 & -0.2502 & -0.5188 & -0.1229 \\ 0.6532 & -0.1583 & -0.0550 & -0.2497 \end{bmatrix} \tag{7.27}$$

O sinal $e(k)$ foi gerado com a função *rand* do MATLAB e tinha como matriz de covariância uma matriz identidade. A série temporal gerada continha 10000 amostras e o estado inicial $x(0)$ também foi gerado com a função *rand*.

A série temporal multivariável $y(k)$ foi gerada e a matriz M_2 da série temporal real foi calculada a partir das séries $x(k)$ e $y(k)$, resultando no seguinte:

$$M_2 = \begin{bmatrix} -0.4110 & 0.5009 \\ -2.6696 & -2.4799 \\ -0.7965 & 0.0353 \\ -0.1128 & -1.5383 \end{bmatrix} \tag{7.28}$$

Desta forma, os três primeiros parâmetros de Markov da série temporal são os seguintes:

$$CM_2 = \begin{bmatrix} 0.8255 & 1.1196 \\ 0.2261 & 1.1019 \end{bmatrix} \tag{7.29}$$

$$CAM_2 = \begin{bmatrix} -0.2942 & 0.5318 \\ -0.3247 & 0.4297 \end{bmatrix} \tag{7.30}$$

$$CA^2M_2 = \begin{bmatrix} 0.0111 & 0.0406 \\ -0.0927 & 0.1785 \end{bmatrix} \tag{7.31}$$

No método MOESP de identificação de sistemas lineares determinísticos multivariáveis, os parâmetros de Markov são fundamentais para a comparação entre o sistema real e o modelo obtido [56]. No método Aoki, os parâmetros de Markov também são importantes, mas a semelhança entre os parâmetros de Markov da série temporal e do modelo não garante que o modelo de fato realiza a série temporal conforme será visto mais adiante.

7.2.4 Resultados e discussão

O método proposto por Aoki foi aplicado à série temporal gerada com o *benchmark* definido nas equações 7.25, 7.26 e 7.27, para se calcular as matrizes A_{est}, C_{est} e M_{est} do modelo. A partir destas matrizes os parâmetros de Markov do modelo calculados foram os seguintes:

$$C_{est}M_{est} = \begin{bmatrix} 0.8403 & 1.1390 \\ 0.2278 & 1.1360 \end{bmatrix} \tag{7.32}$$

$$C_{est}A_{est}M_{est} = \begin{bmatrix} -0.2897 & 0.5876 \\ -0.3229 & 0.4517 \end{bmatrix} \tag{7.33}$$

$$C_{est}A_{est}^2M_{est} = \begin{bmatrix} -0.0048 & 0.0456 \\ -0.1114 & 0.1519 \end{bmatrix} \tag{7.34}$$

que são semelhantes aos da série temporal original, apresentados nas equações 7.29, 7.30 e 7.31.

Cálculo de Σ pelo método da decomposição de Schur

Ao se calcular uma estimativa para a matriz Σ pela decomposição de Schur chega-se ao seguinte resultado:

$$\Sigma_{est} = \begin{bmatrix} 4.6942 & 5.2360 & -7.2721 & 6.7035 \\ -3.8527 & -3.6529 & 13.9343 & -24.5357 \\ 1.2947 & 2.5558 & 4.9647 & -16.7178 \\ -0.6972 & -0.2044 & 5.7368 & -13.1206 \end{bmatrix} \tag{7.35}$$

Os autovalores desta matriz são os seguintes:

$$-6.8246 \quad -2.0669 \quad 0.3088 \quad 1.4681 \tag{7.36}$$

Nota-se diretamente que Σ_{est} não é uma matriz de covariância, uma vez que ela não é simétrica e tem autovalores negativos. Consequentemente, a matriz K_{est} calculada a partir da matriz Σ_{est} não completa a tripla de matrizes A, K, C que realiza a série temporal $y(k)$. Isto ocorre pois as matrizes A, M_2, C e $\Lambda(0)$ resultantes do método Aoki não satisfazem as hipóteses necessárias para a aplicação do método de Schur, o que por sua vez ocorre devido a estimativas pobres de covariâncias da série, resultantes por sua vez do número limitado de amostras da série temporal.

A matriz Δ_{est} encontrada a partir de Σ_{est} é a seguinte:

$$\Delta_{est} = \begin{bmatrix} -2.1061 & 1.5982 \\ -0.8539 & 0.4564 \end{bmatrix} \tag{7.37}$$

Esta matriz não é simétrica, como se espera de uma matriz de covariância. Além disso, esta matriz tem elementos negativos, o que não poderia ocorrer em uma matriz de auto-covariância. Consequentemente, não é possível gerar uma série temporal $e(k)$ para testar se o modelo encontrado de fato realiza a série temporal $y(k)$.

Deve-se notar que, apesar de os parâmetros de Markov da série temporal original e do modelo serem bastante parecidos, a solução encontrada ao se utilizar este método não realiza a série temporal $y(k)$, devido a inconsistências no resultado encontrado na solução da equação de Riccati. Esta foi a motivação para o desenvolvimento do método apresentado nesta tese, cujos resultados são apresentados a seguir.

Escolha dos parâmetros de controle do algoritmo Imuno-Riccati

Para a execução do algoritmo Imuno-Riccati, foi necessária a escolha dos parâmetros de controle do algoritmo imuno-inspirado. A escolha de cada um destes parâmetros e a justificativa da escolha é apresentada abaixo:

- Ordem de grandeza

  Com todas as outras variáveis fixas, foram feitos vários testes em que se variava a ordem de grandeza e o número de iterações necessárias para se atingir o critério de parada era observado. Os parâmetros fixos são apresentados na tabela 7.1 e os resultados do experimento são apresentados na tabela 7.2.

Tab. 7.1: Parâmetros fixos no experimento de variação da ordem de grandeza

Parâmetro	Valor
Número de anticorpos inicial	100
Grau de perturbação	10
Taxa de decaimento do grau de perturbação	0.999
Limiar de supressão	10
Menor *fitness* de parada	10^6

Tab. 7.2: Resultados do experimento de variação da ordem de grandeza

Ordem de grandeza	Iterações necessárias para encontrar a população inicial	Iterações necessárias para encontrar o *fitness* máximo	*Fitness* máximo
0.01	7736	16220	1202717.04
0.1	7736	16315	1000406.83
1	7736	15967	1057288.26
10	355165	16728	1001002.87
20	5771178	17036	1017090
40	198562109	>60000	<270.66

Tab. 7.3: Parâmetros fixos no experimento de variação do grau de perturbação

Parâmetro	Valor
Tamanho da população inicial	100
Ordem de grandeza	1
Taxa de decaimento do grau de perturbação	0.999
Limiar de supressão	10
Menor *fitness* de parada	10^6

Tab. 7.4: Resultados do experimento com variação do grau de perturbação

Grau de perturbação	Iterações necessárias para encontrar a população inicial	Iterações necessárias para encontrar o *fitness* máximo	*Fitness* máximo
0.01	7736	9215	1047765.47
0.1	7736	11449	1090723
1	7736	13975	1086008.88
10	7736	15967	1057288.26
100	7736	18592	1139270.50

Com a ordem de grandeza igual a 40, o número de iterações foi suficiente para que o grau de perturbação fosse a zero, sem que o *fitness* necessário para parada do algoritmo fosse atingido. Por este motivo, a simulação teve seu encerramento forçado após horas de processamento computacional. Outra dificuldade encontrada quando a ordem de grandeza foi muito grande com relação ao limiar de supressão é que o número de anticorpos na população é muito alto e aumentava a cada iteração. Isto se deve à grande área explorada no espaço de busca, que era muito maior que a área de supressão ao redor de cada anticorpo, implicando na super população de anticorpos, que por sua vez levou a um grande tempo consumido a cada iteração do algoritmo. Para ordens de grandeza menores que 40, o número de iterações para se atingir o critério de parada foi praticamente constante, uma vez que a diversidade da população foi garantida pelo grau de perturbação, que gerou clones por todo espaço, apesar do pequeno espaço explorado nas primeiras iterações do algoritmo em alguns casos. Estes clones fizeram com que o espaço de busca fosse bem explorado levando a ótimos que satisfizeram o critério de parada do algoritmo.

- Grau de perturbação

  Ao se variar o grau de perturbação deixando todos os outros parâmetros fixos com os valores apresentados na tabela 7.3, se obteve os resultados apresentados na tabela 7.4.

Tab. 7.5: Parâmetros fixos no experimento de variação da taxa de decaimento do grau de perturbação

Parâmetro	Valor
Tamanho da população inicial	100
Ordem de grandeza	1
Grau de perturbação	0.01
Limiar de supressão	10
Menor *fitness* de parada	10^6

A partir da tabela 7.4, nota-se que, ao se aumentar o grau de perturbação, o número de iterações necessárias para encontrar o *fitness* máximo também aumenta. Isto ocorre pois valores muito altos de grau de perturbação fazem com que a aproximação do ótimo seja grosseira, fazendo o algoritmo mais lento. Como esta variável é diminuida ao longo das iterações pela sua taxa de decaimento, mesmo que o valor inicial seja muito grande, após algumas iterações se chega a valores razoáveis, fazendo com que o ótimo seja alcançado mesmo se for feita uma escolha inicial ruim desta variável.

A pequena influência desta variável no fato de se atingir ou não o ótimo evidencia uma boa escolha da variável ordem de grandeza, uma vez que se esta fosse mal escolhida, seria necessário que os anticorpos saissem do hipercubo definido por esta variável e procurassem o ótimo ao longo do espaço, o que só seria conseguido com um ajuste inicial grosseiro do grau de perturbação, uma vez que um grau de perturbação inicial pequeno não permitiria que os anticorpos chegassem muito longe do hipercubo definido inicialmente pela ordem de grandeza. Para comprovar isto, foi feito um teste com grau de perturbação igual a 0,01 e todos os outros parâmetros conforme apresentado na tabela 7.3, a menos da ordem de grandeza, que foi escolhida como sendo 10. Neste teste, o algoritmo precisou de 120000 iterações para encontrar uma solução de *fitness* 1,36, ou seja, com este número de iterações o critério de parada estava longe de ser atingido.

Da tabela 7.4 também é possível notar que a variável grau de perturbação não tem relação nenhuma com o número de iterações necessárias para se encontrar a população inicial, o que é óbvio, uma vez que esta variável não tem relação com o passo do algoritmo em que a população inicial é determinada.

A partir dos resultados da tabela 7.4, o grau de perturbação escolhido para ser usado é 0,01.

- Taxa de decaimento do grau de perturbação

  Ao se variar a taxa de decaimento do grau de perturbação mantendo todas os outros parâmetros constantes com os valores apresentados na tabela 7.5, obteve-se os resultados apresentados na tabela 7.6. Como este parâmetro não tem relação com o número de iterações necessárias para se atingir a população inicial, esta variável não é listada na tabela 7.6.

  A partir dos resultados apresentados na tabela 7.6 nota-se que, uma vez que um ajuste fino é necessário para se encontrar a solução do problema, uma taxa de decaimento do grau de

Tab. 7.6: Resultados do experimento de variação da taxa de decaimento do grau de perturbação

Taxa de decaimento do grau de perturbação	Iterações necessárias para encontrar o *fitness* máximo	*Fitness* máximo
0.99	> 100000	23.37
0.999	9215	1047765.47
0.9999	86225	1021814.56

Tab. 7.7: Parâmetros fixos no experimento de variação do limiar de supressão

Parâmetro	Valor
Tamanho da população inicial	100
Ordem de grandeza	1
Grau de perturbação	0.01
Taxa de decaimento do grau de perturbação	0.999
Menor *fitness* de parada	10^6

perturbação próxima de 1 implica em um grande número de iterações para se encontrar o grau de perturbação necessário para se determinar a solução ótima. Por outro lado, quando a taxa de decaimento do grau de perturbação é mais distante de 1, o grau de perturbação rapidamente vai a zero, fazendo com que o algoritmo não alcance o critério de parada estabelecido. A partir dos resultados apresentados na tabela 7.6, a taxa de decaimento do grau de perturbação escolhida para o algoritmo foi de 0,999.

- Limiar de supressão

  Ao se variar o limiar de supressão fazendo todos os outros parâmetros de controle do algoritmo constantes com os valores apresentados na tabela 7.7, os resultados obtidos são listados na tabela 7.8.

  A partir da tabela 7.8 pode-se verificar que, se o limiar de supressão é pequeno, o número de anticorpos da população é muito grande. Isto tende a deixar o número de iterações menor, mas também consome mais recursos computacionais para realizar cada iteração. Como no problema estudado se está interessado em apenas uma solução, um número grande de anticorpos não é desejável. Sendo assim, foi escolhido para o algoritmo final o limar de supressão igual a 10.

Resultados da solução da equação de Riccati

Ao se aplicar o algoritmo Imuno-Ricatti, a matriz Σ calculada é a seguinte:

Tab. 7.8: Resultados do experimento de variação do limiar de supressão

Limiar de Supressão	Iterações necessárias para encontrar o *fitness* máximo	Número de anticorpos na iteração 200	*Fitness* máximo
0.1	-	315	-
1	9311	3	1085817.52
10	9215	1	1047765.47

$$\Sigma_{est} = \begin{bmatrix} 2.1586 & 1.0855 & -0.8312 & -1.3626 \\ 1.0855 & 4.3481 & -1.4196 & -1.5944 \\ -0.8312 & -1.4196 & 1.0980 & 1.1194 \\ -1.3626 & -1.5944 & 1.1194 & 1.3646 \end{bmatrix} \tag{7.38}$$

Que é uma matriz simétrica, conforme esperado. Os autovalores desta matriz são os seguintes:

$$0.5407 \quad 1.8284 \quad 4.8344 \quad 7.6663 \tag{7.39}$$

Que são positivos, de acordo com o que é esperado para uma matriz de covariância.

A matriz Δ_{est} encontrada a partir deste Σ_{est} é a seguinte:

$$\Delta_{est} = \begin{bmatrix} 0.9185 & 0 \\ 0 & 1.0886 \end{bmatrix} \tag{7.40}$$

Que é uma matriz simétrica de covariância de um processo estocástico descorrelacionado, como é esperado.

A matriz K_{est} calculada com Σ_{est} é a seguinte:

$$K_{est} = \begin{bmatrix} -0.3806 & 0.0716 \\ 1.3178 & 1.0467 \\ 0.0876 & -1.0547 \\ 0.2322 & -0.2080 \end{bmatrix} \tag{7.41}$$

e a série temporal obtida ao se entar com um ruído branco com covariância Δ_{est} no sistema dado pela tripla $\{A_{est}\ C_{est}\ K_{est}\}$ e com o mesmo estado inicial $x(0)$ é muito semelhante à série temporal original, como pode ser visto nas figuras 7.18 e 7.19. Neste caso, a realização da série temporal estimada pôde ser comparada à realização da série que se quer realizar pois a entrada utilizada para gerar a segunda foi também utilizada para gerar a primeira.

7.2.5 Conclusão

Nesta seção um novo método de solução da equação algébrica de Riccati foi proposto. A partir deste novo método, uma solução aproximada da equação de Riccati foi encontrada, possibilitando a

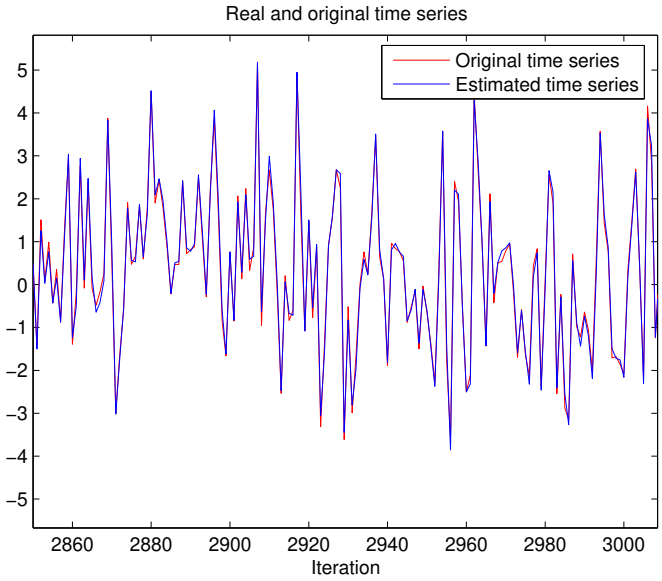

Fig. 7.18: Comparação entre dados reais e estimados para o primeiro elemento do vetor $y(k)$.

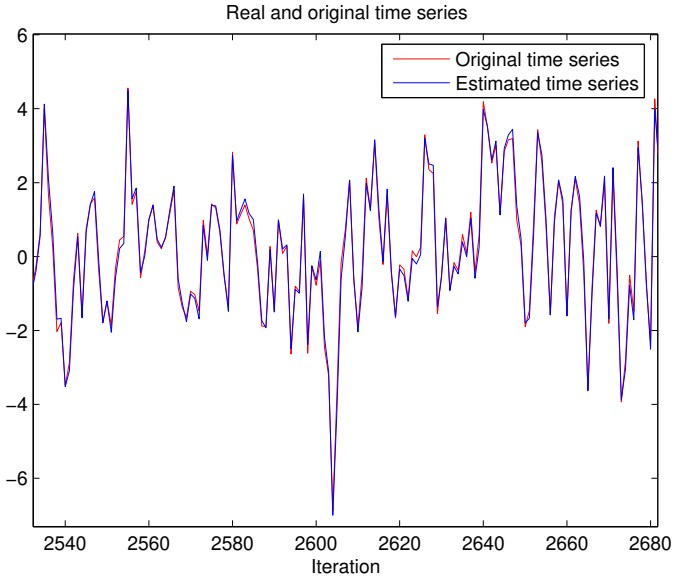

Fig. 7.19: Comparação entre dados reais e estimados para o segundo elemento do vetor $y(k)$.

modelagem de uma série temporal multivariável. A solução obtida com o novo método não poderia ter sido encontrada com o método conhecido até então. Os resultados encontrados com esta solução apresentaram grande exatidão, e as séries temporais geradas com o modelo são bastante similares às que se queria realizar.

Nesta seção também foi deixada clara a influência da escolha de parâmetros de controle do algoritmo imuno-inspirado em seu desempenho. Escolhas ruins de parâmetros podem fazer com que não haja resultados, ou que o número de iterações seja grande o suficiente para inviabilizar a solução por este tipo de ferramenta.

7.3 Propostas para modelagem de séries temporais

Nesta seção é apresentada uma nova proposta para a modelagem de séries temporais discretas no espaço de estado. Esta proposta é baseada na interpretação do problema de modelagem de séries temporais como um problema de otimização. Para resolver o problema de otimização, foram usadas três alternativas de algoritmos imuno-inspirados desenvolvidas para este fim. Para demonstrar a capacidade das alternativas propostas, ao final da seção é apresentado um exemplo de aplicação do método.

A motivação para o desenvolvimento desta técnica é que, para todos os métodos de realização de séries temporais apresentados no capítulo 5, é necessário que se calcule a covariância de saída das séries temporais, que pode ser bem estimada caso haja um número grande o suficiente de amostras. No entanto, no caso de séries temporais reais, muitas vezes o número de amostras disponível é pequeno, o que leva a estimativas pobres das covariâncias de saídas, implicando em resultados finais ruins. Como a modelagem de séries temporais parte das técnicas de realização de séries temporais, conforme apresentado na seção 5.2, este procedimento é também prejudicado pela pobreza de informações nas covariâncias disponíveis. No método de modelagem proposto nesta seção não é necessário o cálculo da covariância de saída, sendo que apenas os dados da série temporal são levados em conta. Por uma questão de simplicidade, o modelo utilizado no método apresentado nesta tese é o da equação 5.82.

7.3.1 Modelagem de séries temporais vista como um problema de otimização

A proposta de modelagem de séries temporais apresentada nesta seção tem como princípio a interpretação do problema como sendo de otimização. Nesta abordagem, a função objetivo que se quer minimizar é o erro entre a série temporal real e a estimada usando uma determinada tripla de matrizes $\{A_{est}, C_{est}, K_{est}\}$, que formam uma estimativa de modelo das séries conforme equação 5.82. Deve-se notar que o objeto de otimização, que é a tripla de matrizes do modelo, não está diretamente relacionado à função objetivo a ser analisada, uma vez que a função objetivo opera na série temporal resultante da aplicação do ruído branco ao modelo definido pela tripla, e não diretamente sobre a tripla. Desta forma, não é possível a aplicação de métodos de otimização não linear baseados em gradientes da função objetivo, pois não há relação direta entre o gradiente da função objetivo em um ponto e a tripla de matrizes. Sendo assim, foi necessário propor alternativas para a solução do problema, como os algoritmos imuno-inspirados. O problema de otimização também está sujeito a algumas restrições que serão detalhadas a seguir, logo após a descrição da função objetivo.

Função objetivo

Seja $mod_e(k)$ uma função que para cada k, $1 \leq k \leq N$, representa módulo da diferença entre o valor da série temporal a ser modelada no instante k e a saída no instante k do modelo semelhante ao mostrado na equação 5.82, tendo como matrizes a tripla $\{A_{est}, C_{est}, K_{est}\}$. A função objetivo $F(mod_e)$ a ser maximizada é a seguinte:

$$F(mod_e) = \frac{N}{\sum_{k=1}^{N} mod_e(k)} \tag{7.42}$$

em que N é o número total de amostras da série temporal a ser modelada.

A variável N foi colocada no numerador para que a função objetivo pudesse ser comparada para casos com diferentes números de amostras. No denominador da função objetivo está a soma dos erros, de forma que quanto menor for esta soma, maior será a função objetivo. Como o problema é de maximização, a solução será aquela em que houver o menor erro.

No contexto dos algoritmos imuno-inspirados usados para a solução deste problema, o valor de F para uma determinada tripla $\{A_{est}, C_{est}, K_{est}\}$ também é chamado de *fitness* da solução.

Restrições

Para ser uma solução do problema, a tripla $\{A_{est}, C_{est}, K_{est}\}$ deve satisfazer as seguintes restrições: A matriz A_{est} deve implicar em um modelo estável, o que no caso discreto significa que seus autovalores devem estar dentro do círculo unitário. Além disso, as matrizes C_{est} e K_{est} devem ser tais que as saídas do modelo sejam razoavelmente próximas da série temporal, ou seja, devem ser tais que não levem o erro a infinito.

Além destas restrições, em uma das alternativas propostas foram também colocadas as restrições de atingibilidade e observabilidade, conforme será apresentado mais adiante. Estas restrições refletem as descritas na seção 5.1.6 desta tese.

Enunciado do problema de otimização

Levando em conta as características destacadas acima, o problema de otimização relacionado à modelagem das séries temporais pode ser enunciado da seguinte forma:

Dada a uma realização de uma série temporal $y(k)$, encontre a tripla de matrizes $\{A_{est}$, C_{est}, $K_{est}\}$ que minimiza

$$F(mod_e) = \frac{N}{\sum_{k=1}^{N} mod_e(k)}$$

em que

$$mod_e(k) = |y(k) - y_{est}(k)|$$

e

$$\begin{cases} \hat{x}(k+1) = A_{est}\hat{x}(k) + K_{est}e(k) \\ y_{est}(k) = C_{est}\hat{x}(k) + e(k) \end{cases}$$

sendo $\hat{x}(0)$ igual ao estado inicial usado para gerar a série a ser modelada e $e(k)$, $0 \leq k \leq N$, a mesma entrada usada para gerar a série a ser modelada.

Sujeito a:

$$|eig(A_{est})| < 1$$

e

$$|y(k) - y_{est}(k)| < \infty$$

As restrições de observabilidade e atingibilidade também foram incluídas para um dos casos estudados.

7.3.2 Alternativas propostas

Para a solução deste problema, foram propostas algumas variações aos algoritmo *opt-aiNet*. A primeira destas variações surgiu devido à presença de restrições no problema estudado. O algoritmo *opt-aiNet* foi desenvolvido para problemas sem restrições e o problema de otimização relacionado à modelagem de séries temporais tem restrições, conforme discutido logo acima. A segunda variação surgiu da necessidade de se explorar o espaço de soluções buscando regiões de soluções factíveis atravessando zonas de soluções não factíveis. Por fim, em uma tentativa de restringir um pouco mais o espaço de busca, foram inseridas no problema de otimização duas novas restrições relacionadas à natureza do modelo gerador das séries temporais. As alternativas propostas para a solução do problema são detalhadas a seguir.

Algoritmos imunológicos para otimização com restrições

A princípio, como o problema de otimização estudado tem restrições, o método utilizado foi semelhante ao *opt-aiNet*, a menos que, na etapa inicial de geração de indivíduos, as restrições do problema foram levadas em conta, de forma que, ao invés da simples geração aleatória de indivíduos, foi feito um laço garantindo que indivíduos fossem gerados até que se chegasse a um número desejável de indivíduos iniciais satisfazendo as restrições do problema. Este procedimento também foi adotado na etapa de inserção de novos indivíduos ao final de cada iteração do algoritmo.

Outra diferença entre o método explorado inicialmente e o algoritmo *opt-aiNet* é que, na etapa da clonagem dos anticorpos, os clones gerados são verificados quanto a sua factibilidade. Se os clones implicam em soluções factíveis, eles são mantidos na população. Caso contrário, uma nova perturbação é gerada e somada ao clone original. Isto é repetido até que se chegue a um clone perturbado factível. A partir destas diferenças garante-se que o algoritmo sempre gera soluções factíveis do problema estudado. A abordagem adotada é semelhante à usada na seção 7.2. Na figura 7.20 é apresentado o fluxograma da abordagem proposta para a modelagem de séries temporais.

Travessia de zonas proibidas

Ao se adotar o procedimento descrito acima, foi verificado que existe a possibilidade de alguns clones ficarem presos em regiões factíveis cercadas por regiões infactíveis. Isto é possível pois, caso uma perturbação gere um clone perturbado infactível, ela é descartada e uma nova perturbação é gerada, até que se chegue a uma que faça o clone perturbado ser factível. Desta forma, se o clone estiver cercado por uma região infactível, as perturbações são geradas até que se chegue a uma que não

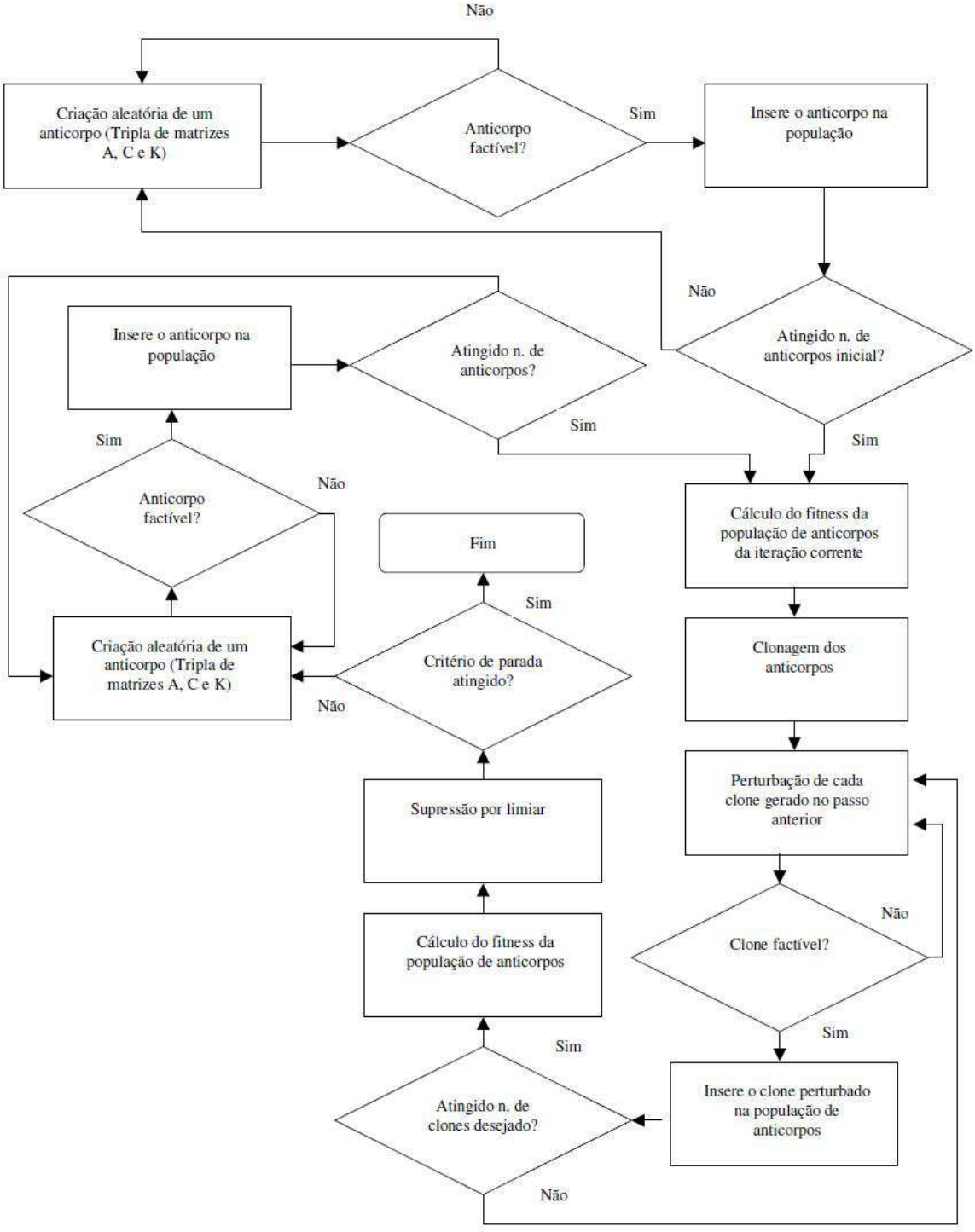

Fig. 7.20: Fluxograma do algoritmo imuno-inspirado para modelagem de séries temporais

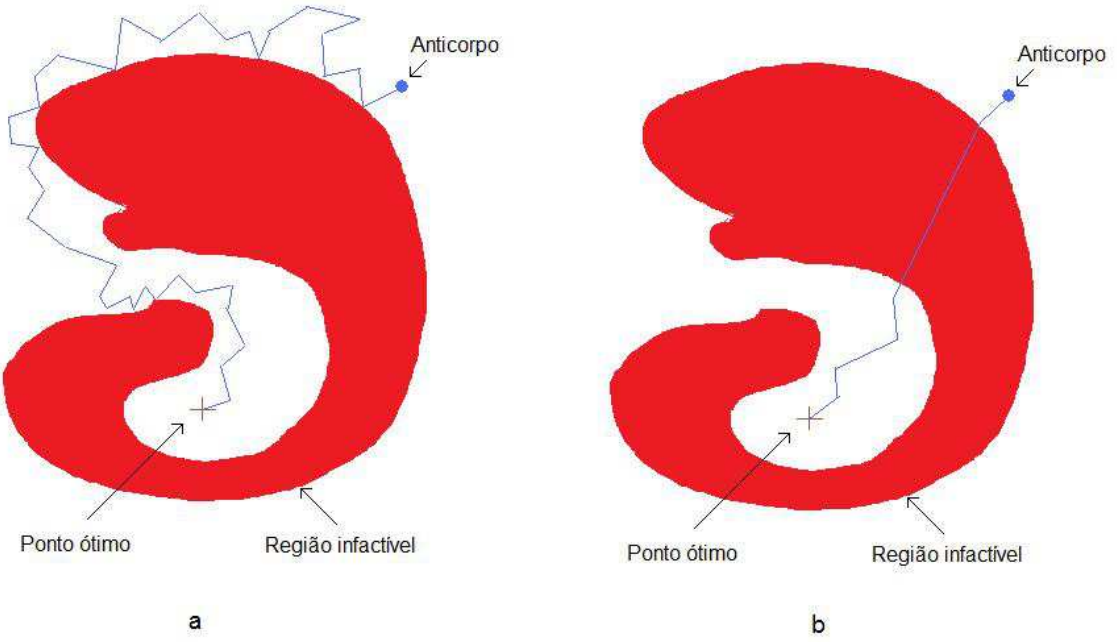

Fig. 7.21: Situação em que o ponto ótimo está cercado por regiões infactíveis. Na situação a) é apresentado o comportamento do algoritmo original. Na situação b) é apresentado o comportamento do algoritmo com travessia de zonas proibidas

seria grande o suficiente para fazê-lo ultrapassar a fronteira de factibilidade, enclausurando o clone em uma determinada região do espaço. Também foi verificado que a solução ótima do problema poderia estar em uma região cercada de regiões infactíveis, dificultando a chegada dos clones até ela.

Para solucionar esta possível falha do algoritmo, foi adotado um novo método de clonagem em que, caso um clone perturbado caia em uma região infactível, é gerada uma nova perturbação, que é somada ao clone perturbado, e não mais ao clone original. Desta forma, mesmo que um clone esteja em uma região confinada por zonas proibidas (regiões infactíveis), ele seria perturbado até chegar a uma região factível adjacente. Desta maneira, também se resolve o problema no caso de a solução ótima estar cercada por zonas proibidas, uma vez que ela pode ser alcançada por clones de anticorpos que estão em regiões factíveis adjacentes. Todos os outros passos do algoritmo apresentado na figura 7.20 foram mantidos.

Na figura 7.21 é apresentado um caso em que o ponto ótimo está cercado por uma região infactível. Neste caso, o comportamento do algoritmo original é apresentado à esquerda. O anticorpo circunda a região infactível até encontrar o ótimo; o comportamento do algoritmo com travessia de zonas proibidas é apresentado à direita. O anticorpo atravessa a região infactível alcançando o ponto ótimo em um número menor de passos.

Restrições de Atingibilidade e observabilidade

A possibilidade de travessia de zonas proibidas pelos clones permitiu que fossem inseridas novas restrições ao problema, reduzindo o espaço de busca. As novas restrições inseridas foram as de atingibilidade e observabilidade. Para verificar se os anticorpos formados pela tripla $\{A_{est}, C_{est}, K_{est}\}$ implicam em um sistema observável, basta tomar as matrizes A_{est} e C_{est} estimadas e calcular a matriz de observabilidade e seu posto. Se o posto da matriz de observabilidade for igual à ordem do sistema, o mesmo é observável. Para a verificação da restrição de atingibilidade é necessário o cálculo da matriz M_{est}, o que depende dos estados e das saídas. O cálculo desta matriz foi feito na rotina de cálculo do *fitness* da solução, em que os estados e as saídas estão disponíveis. A partir da matriz M_{est} e da matriz A_{est}, é feito o cálculo da matriz de atingibilidade e de seu posto, determinando se o sistema é ou não observável.

7.3.3 Exemplo de aplicação

Para testar os algoritmos propostos, uma série temporal com 200 amostras foi gerada pela aplicação de um ruído branco a um sistema formado pelas seguintes matrizes:

$$A = \begin{bmatrix} 0.873847 & 0.166155 \\ -0.166155 & 0.426153 \end{bmatrix} \tag{7.43}$$

$$C = \begin{bmatrix} 1.231862 & -0.719364 \end{bmatrix} \tag{7.44}$$

$$K = \begin{bmatrix} 1.231862 & 0.719364 \end{bmatrix} \tag{7.45}$$

Esta série temporal, também chamada de série temporal real, foi dada como entrada ao algoritmo proposto por Aoki, resultando em um modelo que foi usado para gerar uma série temporal estimada, que é mostrada na figura 7.22 juntamente com a série temporal real. O *fitness* da série temporal estimada pelo método de Aoki foi calculado com a equação 7.42, resultando em 0.7765. Da figura 7.22 nota-se que a dinâmica da série temporal estimada não acompanha a da série temporal real. Provavelmente isto se deve ao número relativamente pequeno de amostras utilizado, de forma que as covariâncias da série temporal real não são bem estimadas, implicando em uma estimativa ruim para o modelo, que leva a uma série temporal estimada relativamente distante da real.

As três alternativas para modelagem de séries temporais propostas nesta seção da tese foram implementadas e foram comparadas com a série temporal real. Os parâmetros de entrada do algoritmo imuno-inspirado adotados são apresentados na tabela 7.9. Foram observados o número de iterações até se chegar ao critério de parada, o *fitness* máximo atingido e o número de indivíduos na última iteração. Os resultados são apresentados na tabela 7.10, em que a alternativa 1 é a sem travessia de zonas proibidas, a alternativa 2 é aquela em que foi implementada a travessia de zonas proibidas e a alternativa 3 é aquela em que foram introduzidas as restrições de observabilidade e atingibilidade. Na figura 7.23 é mostrada a série temporal original e a estimada na alternativa que apresentou o maior *fitness* máximo. Da figura, nota-se que a modelagem da série temporal pelo método proposto nesta seção foi bem sucedida, ou seja, a série temporal estimada é muito próxima da real, e ao se comparar as figuras 7.22 e 7.23 nota-se que a série temporal obtida com o método proposto nesta seção é mais

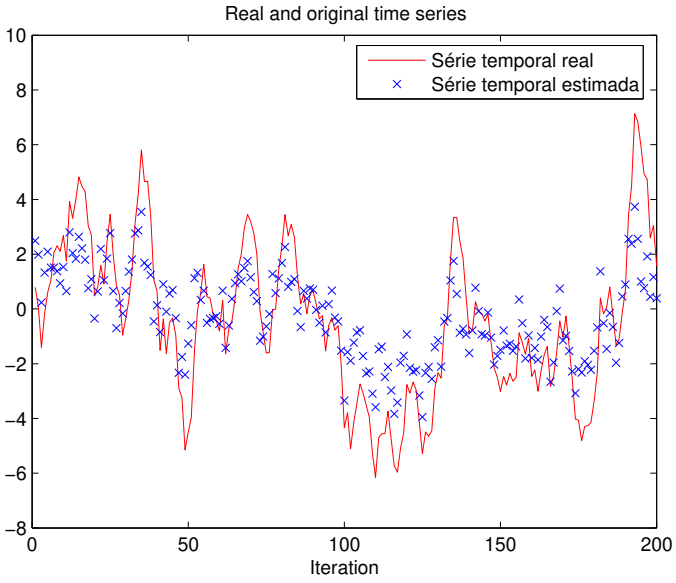

Fig. 7.22: Comparação entre a série temporal real e a estimada pelo método proposto por Aoki

próxima da série temporal real do que a série temporal obtida usando o algoritmo proposto por Aoki. De fato, mesmo o menor *fitness* máximo das alternativas imuno-inspiradas propostas é maior que o obtido ao se fazer a modelagem da série com o método proposto por Aoki.

Da tabela 7.10 nota-se que o maior *fitness* foi observado para a alternativa 3. A inserção das restrições de atingibilidade e observabilidade fez com que o algoritmo chegasse a uma solução melhor que nos outros casos. Isto se justifica pela redução do espaço de busca, que permitiu que o algoritmo encontrasse melhores soluções. O número de iterações para atingir o critério de parada também foi menor na alternativa 3, o que também se justifica pelo menor espaço de busca. Na alternativa 3 também foi observado o maior número de indivíduos na população final. Isto indica que seria possível aumentar o limiar de supressão do algoritmo para melhorar sua performance. Isto não foi feito para não prejudicar a comparação desta proposta com as outras duas.

A travessia de zonas proibidas implementada na alternativa 2 levou a um maior número de iterações para se chegar ao critério de parada e o *fitness* máximo obtido foi menor que o dos outros casos. Nesta alternativa, os clones que sofreram mutações tinham a possibilidade de ficarem distantes do anticorpo que os gerou. Ao contrário do que era esperado, aparentemente isto prejudicou a capacidade de refinamento da solução do algoritmo. No entanto a travessia de zonas proibidas foi interessante quando aliada às restrições de observabilidade e atingibilidade, levando aos bons resultados da alternativa 3.

A alternativa 1, apesar de apresentar um *fitness* máximo menor que o observado na alternativa 3, também levou a um resultado final razoável. Apesar de o número de iterações para se atingir o critério de parada ter sido um pouco maior que o observado na alternativa 3, a alternativa 1 levou menos tempo para ser executada, devido ao menor número de indivíduos na população.

Tab. 7.9: Parâmetros utilizados nos algoritmos imuno-inspirados propostos para modelagem de séries temporais

Parâmetro	Valor
Número de indivíduos inicial	20
Número de clones inicial	5
Ordem de grandeza (A)	1
Ordem de grandeza (C e K)	4
Grau de perturbação (A)	0.5
Grau de perturbação (C e K)	2
Decaimento da perturbação (A)	0.999
Decaimento da perturbação (C e K)	0.999

Tab. 7.10: Resultados na última iteração

Alternativas	1	2	3
Fitness máximo	1.2537	1.2501	1.2638
Número de iterações	14203	40123	13749
Número de indivíduos	30	33	186

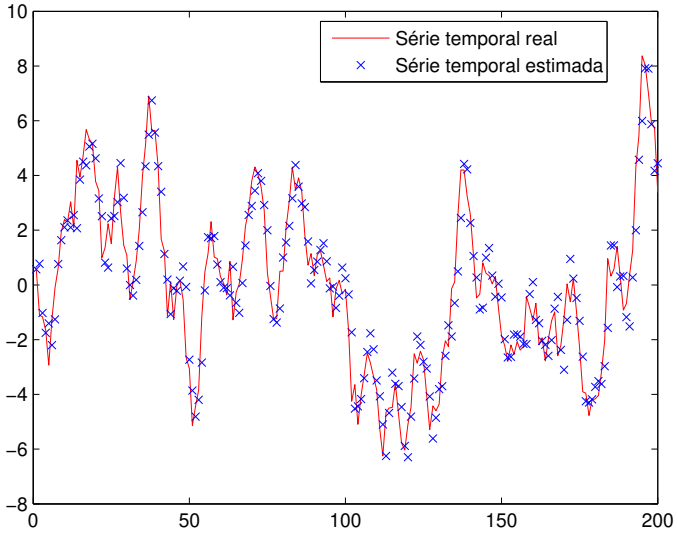

Fig. 7.23: Comparação entre série real e estimada pelo algoritmo imuno-inspirado

7.3.4 Conclusão

A partir dos resultados observados conclui-se que o método proposto para a modelagem de séries temporais leva a resultados melhores do que os obtidos com um método amplamente divulgado de modelagem de séries temporais para o caso estudado em que poucas amostras estão disponíveis. A grande vantagem do método proposto é que não é necessária a estimação das matrizes de covariância das séries temporais, que não é acurada quando se tem um número de amostras limitado.

Além dos bons resultados da primeira alternativa imuno-inspirada, as outras alternativas propostas levaram ao aumento de performance e de qualidade da solução, como pode ser visto na tabela 7.10. Apesar de nesta seção as alternativas serem dedicadas ao problema de modelagem de séries temporais, algumas delas podem ser usadas para a solução de outros problemas de otimização com restrições em que se queira utilizar os algoritmos imuno-inspirados.

7.4 Identificação de sistemas variantes no tempo

Conforme já discutido no capítulo 3 desta tese, existe um algoritmo de identificação de sistemas multivariáveis variantes no tempo no espaço de estado denominado MOESP-VAR. Este algoritmo é baseado na repartição das amostras de entrada e saída do sistema [56], [57]. De forma resumida, o algoritmo funciona da seguinte maneira: Em um primeiro passo são definidas janelas temporais. Cada janela contém um subconjunto de entradas e saídas do sistema a ser identificado. O número de dados em cada janela é definido de maneira que o sistema não sofra mudanças significativas durante todo o intervalo de tempo definido para aquela janela. Feito isto, o algoritmo MOESP é aplicado para cada janela e as matrizes A, B, C e D são estimadas para cada uma delas.

O algoritmo MOESP-VAR pode ser aplicado caso haja um grande número de dados disponível para cada janela, ou seja, caso as variações do sistema não sejam muito rápidas, garantindo que haja um conjunto de entradas e saídas suficientemente grande para a identificação via método MOESP em cada um dos intervalos. No caso em que as variações do sistema são rápidas, o método MOESP-VAR não implica em respostas muito precisas. Ao longo desta pesquisa foi desenvolvido um algoritmo imuno-inspirado para lidar com estes casos. Este algoritmo é baseado na interpretação do problema de identificação de séries temporais multivariáveis como um problema de otimização, conforme será detalhado a seguir.

7.4.1 Identificação de sistemas multivariáveis variantes no tempo como um problema de otimização

O problema de identificação de sistemas multivariáveis variantes no tempo pode ser definido como um problema de otimização se as matrizes do modelo em espaço de estado que representa o sistema durante um determinado intervalo de tempo forem interpretadas como um ponto no espaço definido por todas as possíveis quádruplas $\{A, B, C, D\}$ (ver equação 3.9). Ao se considerar todos os possíveis modelos como pontos de um espaço, o objetivo é determinar o ponto que minimiza o erro entre a saída do sistema a ser identificado e a saída do modelo, quando submetido à mesma entrada aplicada no sistema. Este ponto no espaço de quádruplas representa o sistema. Assim que o sistema sofrer uma variação, outro ponto no espaço de quádruplas minimizará o erro entre saída real

e estimada. No entanto, como se parte da hipótese de que o sistema sofre variações graduais ao longo do tempo, este novo ponto estará próximo do ponto anterior. Desta forma, um algoritmo heurístico poderá ser capaz de encontrar este novo ponto no espaço.

De uma maneira mais formal, nesta contribuição a identificação de sistemas multivariáveis variantes no tempo é tratada como um problema de otimização da seguinte maneira: a cada instante de tempo k um conjunto com N_w amostras de entradas e saídas do sistema é tomada. Este conjunto, definido como janela, contém as entradas entre o instante de tempo $k - \lfloor N_w/2 \rfloor$ e $k + \lfloor N_w/2 \rfloor$ e as saídas entre $k - \lfloor N_w/2 \rfloor + 1$ e $k + \lfloor N_w/2 \rfloor + 1$, em que $\lfloor * \rfloor$ denota a parte inteira do número real $*$. O problema de otimização é estimar a quádrupla de matrizes $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ que, quando substituída no modelo em espaço de estado apresentado na equação 3.9, excitado pelas entradas contidas na janela, minimiza a diferença entre as saídas do modelo e as saídas do sistema contidas na janela. Este problema de otimização é resolvido para todas as amostras de entrada e saída disponíveis. Desta forma, a cada instante k é determinado um modelo no espaço de estado que representa o sistema. Os detalhes do problema de otimização são apresentados a seguir:

Espaço de busca

O espaço de busca do problema de otimização é a união dos espaços em que as matrizes A, B, C e D são definidas. Ou seja, o espaço é $\Re^{nXn} \cup \Re^{nXm} \cup \Re^{lXn} \cup \Re^{lXm}$ em que n, m e l são respectivamente a ordem, a dimensão da entrada e a dimensão da saída do sistema, conforme definido no capítulo 3. Conforme será demonstrado adiante, este espaço de busca é grande e tem uma topologia mal comportada.

Soluções candidatas

As soluções candidatas do problema de otimização são as quádruplas $\{A_{est}, B_{est}, C_{est}, D_{est}\}$, que são pontos do espaço de busca definido anteriormente. Estas quádruplas definem modelos em espaço de estado de acordo com a estrutura definida na equação 3.9. Cada um destes modelos em espaço de estado, quando submetido a uma entrada $u(k)$, produz uma saída estimada $y_{est}(k)$, que é comparada à saída real do sistema pela função objetivo, descrita logo a seguir. No contexto dos algoritmos imuno-inspirados as soluções candidatas $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ são definidas como anticorpos.

Função objetivo

Para uma determinada janela, seja mod_e uma matriz em que cada elemento representa o módulo da diferença vetorial l-dimensional entre a saída do sistema e a saída do modelo definido por uma determinada solução candidata $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ para cada amostra contida na janela, ou seja:

$$mod_e = \begin{bmatrix} |y(k - \lfloor N_w/2 \rfloor + 1) - y_{est}(k - \lfloor N_w/2 \rfloor + 1)| \\ |y(k - \lfloor N_w/2 \rfloor + 2) - y_{est}(k - \lfloor N_w/2 \rfloor + 1)| \\ \vdots \\ |y(k-1) - y_{est}(k-1)| \\ |y(k) - y_{est}(k)| \\ |y(k+1) - y_{est}(k+1)| \\ \vdots \\ |y(k + \lfloor N_w/2 \rfloor + 1) - y_{est}(k + \lfloor N_w/2 \rfloor + 1)| \end{bmatrix} \tag{7.46}$$

A função objetivo a ser maximizada é a seguinte:

$$F(mod_e) = \frac{N_w}{\sum_{i=1}^{N_w} mod_e(i)} \tag{7.47}$$

A variável N_w está no numerador da função objetivo para permitir a comparação entre soluções com diferentes números de amostras em cada janela. No denominador da função objetivo está a soma dos erros, desta forma o resultado da função objetivo será maior quanto menor for a diferença entre saída do sistema e saída do modelo estimado. No contexto dos algoritmos imuno-inspirados para otimização, o valor F calculado para uma determinada solução candidata $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ é definido como o *fitness* da solução.

Restrições

As soluções candidatas $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ devem satisfazer as seguintes restrições: A matriz A_{est} deve implicar em um modelo estável, o que no contexto dos sistemas discretos implica que seus autovalores devem estar dentro do círculo unitário. As matrizes B_{est} e C_{est} devem ser tais que o erro não vá a infinito. As soluções candidatas também devem implicar em sistemas observáveis e atingíveis. Para isto a matriz de observabilidade formada pelas matrizes C_{est} e A_{est} (ver equação 3.22) e a matriz de atingibilidade formada pelas matrizes A_{est} e B_{est} da solução candidata (ver equação 3.25), devem ter posto igual a n, que é a ordem do modelo.

Enunciado do problema de otimização

O problema de otimização a ser solucionado para cada janela definida no problema de identificação de sistemas multivariáveis variantes no tempo no espaço de estado é enunciado a seguir.

> *Dadas as amostras de entrada* $u(i)$, $k - \lfloor N_w/2 \rfloor \leq i \leq k + \lfloor N_w/2 \rfloor$ *e saída* $y(i)$, $k - \lfloor N_w/2 \rfloor + 1 \leq i \leq k + \lfloor N_w/2 \rfloor + 1$, *de um sistema variante no tempo, encontre a quádrupla* $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ *que minimiza*
>
> $$F(mod_e) = \frac{N_w}{\sum_{i=1}^{N_w} mod_e(i)}$$

sendo N_w o número de elementos em uma janela,

$$mod_e = \begin{bmatrix} |y(k-\lfloor N_w/2 \rfloor + 1) - y_{est}(k-\lfloor N_w/2 \rfloor + 1)| \\ |y(k-\lfloor N_w/2 \rfloor + 2) - y_{est}(k-\lfloor N_w/2 \rfloor + 1)| \\ \vdots \\ |y(k-1) - y_{est}(k-1)| \\ |y(k) - y_{est}(k)| \\ |y(k+1) - y_{est}(k+1)| \\ \vdots \\ |y(k+\lfloor N_w/2 \rfloor + 1) - y_{est}(k+\lfloor N_w/2 \rfloor + 1)| \end{bmatrix}$$

e

$$\begin{cases} x_{est}(k+1) = A_{est}x_{est}(k) + B_{est}u(k) \\ y_{est}(k) = C_{est}x_{est}(k) + D_{est}u(k) \end{cases}$$

em que $x_{est}(0)$ é um vetor qualquer de dimensão n.

Sujeito a:

$$|eig(A_{est})| < 1$$
$$|y(k) - y_{est}(k)| < \infty$$
$$rank\mathcal{O} = n$$
$$rank\mathcal{C} = n$$

Topologia do espaço de busca

Conforme citado anteriormente, o espaço de busca do problema de otimização relacionado à identificação de sistemas multivariáveis variantes no tempo é grande e tem uma topologia desfavorável a soluções via algoritmos heurísticos. Para ilustrar esta dificuldade é proposto um exemplo similar mais simples. Seja:

$$C = [\ 1.231862 \quad -0.719364\ ]$$

uma matriz 2x1 e x_0 o seguinte vetor 2x1:

$$x_0 = \begin{bmatrix} 0.3601 \\ 0.2725 \end{bmatrix}$$

Seja y o produto entre C e x_0, ou seja:

$$y_0 = Cx_0 = 0.2476$$

Para i e j variando entre -2 e 2 em intervalos de 0.01, um conjunto de matrizes $C_{est}(i,j)$ é definido da seguinte maneira:

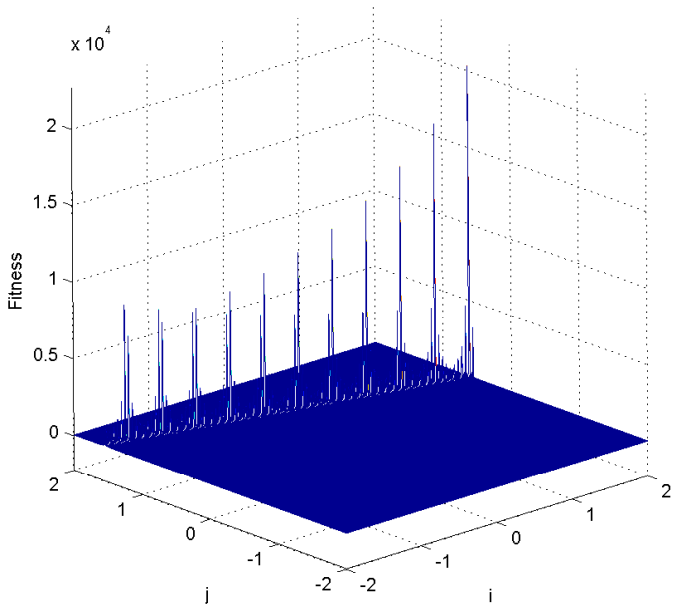

Fig. 7.24: Resultados do cálculo do *fitness* F_s para cada $C_{est}(i,j)$

$$C_{est}(i,j) = [i \quad j]$$

Para cada uma das matrizes C_{est} o número $y_{0est}(i,j) = C_{est}(i,j)x_0$ é calculado e um valor de *fitness* é atribuído a cada C_{est} de acordo com a seguinte equação, que é equivalente à equação 7.47:

$$F_s(C_{est}(i,j)) = \frac{1}{|y_0 - y_{0est}(i,j)|}$$

Os resultados de F_s para cada $C_{est}(i,j)$ são plotados na figura 7.24. Da figura fica claro que a superfície do problema não é bem comportada, uma vez que contém picos altos e abruptos, ou seja, com derivadas muito grandes. Consequentemente, se um algoritmo puramente heurístico for utilizado para a resolução do problema de otimização que é a determinação do C_{est} que melhor representa a matriz C, pelo menos uma solução candidata aleatória deve cair na bacia de um dos picos. Como estas bacias são pequenas, a probabilidade de isto ocorrer é baixa, portanto um algoritmo puramente heurístico teria difuldades de resolver este problema.

O problema estudado nesta tese é ainda mais complexo que o apresentado no exemplo acima, uma vez que a solução candidata é formada por quatro matrizes com dimensões maiores que da matriz exemplificada. Para lidar com este problema, o algoritmo proposto nesta tese tem uma etapa de inicialização das soluções candidatas, conforme apresentado a seguir. Esta etapa leva as soluções candidatas para uma região próxima das soluções ótimas que devem ser encontradas.

7.4.2 Algoritmo proposto

Como apresentado na seção anterior, o problema de otimização relacionado à identificação de sistemas multivariáveis variantes no tempo tem restrições e um espaço de busca que não é favorável para

o uso de algoritmos heurísticos devido aos seus picos altos de pequenas bacias. Por outro lado, ao se considerar que as variações do sistema são contínuas ao longo do tempo, uma vez que uma solução do problema de otimização é encontrada, é relativamente simples seguir pequenas variações desta solução por meio de pequenas perturbações aplicadas sobre a solução determinada anteriormente. Além disto, se um algoritmo imuno-inspirado for utilizado e se o limiar de supressão for bem escolhido, uma solução que maximiza a função objetivo em um momento passado do sistema é mantida na população. Desta forma, caso o sistema volte a se comportar como naquele instante de tempo passado, a solução ótima já faz parte da população de soluções e não precisa ser buscada novamente.

A analogia entre o problema de otimização tratado nesta seção da tese e o sistema imunológico é a seguinte: Os anticorpos são as quádruplas estimadas $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ e os antígenos são as matrizes do sistema variante no tempo a cada instante k, ou seja, a quádrupla $\{A(k), B(k), C(k), D(k)\}$. Se o sistema sofre uma variação temporal, ou em outras palavras, se o antígeno sofre uma mutação, a quádrupla $\{A_{est}, B_{est}, C_{est}, D_{est}\}$ sofrerá mutações até que se mininize o erro entre a saída do modelo e a saída do sistema. Em outras palavras, as células B que secretam anticorpos sofrerão mutação até serem capazes de produzir anticorpos que combatam a mutação do antígeno. Caso o sistema sofra outra variação, a quádrupla estimada será alterada novamente para minimizar o erro, mas a solução anterior será mantida na população de soluções candidatas. Caso o sistema volte a uma situação em que estava anteriormente, o sistema imunológico artificial já aprendeu como lidar com aquela situação e o antígeno correto será aplicado, eliminando a necessidade de novas iterações para busca do ótimo. Esta é uma grande vantagem em se utilizar algoritmos imuno-inspirados para solução deste tipo de problema, uma vez que este tipo de algoritmo tem a propriedade de manter soluções anteriores na população sem que isto implique em grande esforço computacional.

Em linhas gerais, o algoritmo proposto nesta seção segue os passos listados abaixo:

1. Uma janela de inicialização é definida contendo N_{ini} amostras de entradas e saídas entre os instantes $k = 0$ e $k = N_{ini} - 1$.

2. O algoritmo MOESP é aplicado às entradas e às saídas contidas na janela de inicialização e uma quádrupla inicial $\{A_{ini}, B_{ini}, C_{ini}, D_{ini}\}$ é determinada.

3. A quádrupla $\{A_{ini}, B_{ini}, C_{ini}, D_{ini}\}$ é clonada implicando na criação de $N_{clonesini}$ clones e cada um deles é perturbado, ou seja, cada matriz que faz parte da quádrupla que representa cada clone é somada a uma matriz de perturbação com as mesmas dimensões. Estes anticorpos obtidos com a perturbação dos clones definem a população inicial.

4. Para cada instante $k \geq N_{ini}$ um conjunto de amostras contendo as entradas entre os instantes $k - floor(N_w)/2$ e $k + floor(N_w/2)$ e as saídas entre $k - \lfloor N_w/2 \rfloor + 1$ e $k + \lfloor N_w/2 \rfloor + 1$ é tomado e definido como janela.

5. O *fitness* de cada anticorpo é calculado levando em conta os dados de entrada e saída da janela atual.

6. Cada anticorpo da população é clonado e perturbado. O número de clones criado para cada anticorpo é proporcional ao seu *fitness*. Caso um clone perturbado não satisfaça as restrições do problema, uma nova matriz de perturbações é somada ao clone perturbado. O processo é

repetido até que um clone factível seja encontrado. Os clones factíveis obtidos são adicionados à população.

7. A distância entre cada um dos anticorpos da população é calculada. Esta distância é definida como a soma das distâncias euclidianas entre cada matriz da quádrupla que define os anticorpos.

8. Os anticorpos que estiverem muito próximos de outro com um *fitness* melhor são eliminados da população. Este conceito de proximidade é definido como uma distância menor que uma variável de entrada do algoritmo definida como limiar de supressão.

9. Os *fitness* dos anticorpos são calculados tendo como base os dados da janela atual. Caso o melhor *fitness* da população seja melhor que um determinado *fitness* requerido F_{req}, o algoritmo volta ao passo 4. Caso contrário o algoritmo volta ao passo 5 e a solução referente a janela atual é refinada. Caso o melhor *fitness* da população seja maior que F_{req} e as amostras de entrada e saída tiverem chegado ao fim, o algoritmo é finalizado.

Na figura 7.25 é apresentado o fluxograma simplificado do algoritmo proposto.

Caso o sistema não sofra variações entre os instantes de tempo $k - \lfloor N_w/2 \rfloor$ e $k + \lfloor N_w/2 \rfloor + 1$, a mesma quádrupla que apresenta o *fitness* maior que F_{req} para os dados da janela definida entre $k - \lfloor N_w/2 \rfloor$ e $k + \lfloor N_w/2 \rfloor$ também apresentará um *fitness* maior que F_{req} para os dados da janela definida entre $k-\lfloor N_w/2 \rfloor+1$ e $k+\lfloor N_w/2 \rfloor+1$ e apenas uma iteração será necessária nesta nova janela. Caso o sistema sofra uma variação pequena entre estas janelas, pequenas perturbações na solução obtida para a primeira janela serão necessárias para determinar uma boa solução para a segunda janela, ou seja, o algoritmo precisará de um número relativamente pequeno de iterações para determinar a solução ótima quando for submetido à nova janela. No caso de uma variação muito grande do sistema, maior será o número de iterações necessário para encontrar a solução ótima para a nova janela. A quantidade de variação do sistema pode ser medida pela derivada de seus parâmetros de Markov, conforme apresentado no exemplo de aplicação a seguir.

7.4.3 Exemplo de aplicação

Para testar o algoritmo proposto nesta seção da tese, um ruído branco tridimensional com $N = 1000$ amostras foi aplicado como entrada a um sistema linear variante no tempo com estrutura conforme equação 3.9 e com as seguintes matrizes:

$$A(k) = \begin{bmatrix} 0.2128 & h(k) & 0.1979 & -0.0836 \\ 0.1808 & 0.4420 & -0.3279 & 0.2344 \\ -0.5182 & 0.1728 & -0.5488 & -0.3083 \\ 0.2252 & -0.0541 & -0.4679 & 0.8290 \end{bmatrix} \tag{7.48}$$

em que:

$$h(k) = \begin{cases} 0.1360 & k < 200 \\ 0.1360 - \sin(\frac{k-200}{200}) & 200 \leq k \leq 828 \\ 0.1360 & k > 828 \end{cases} \tag{7.49}$$

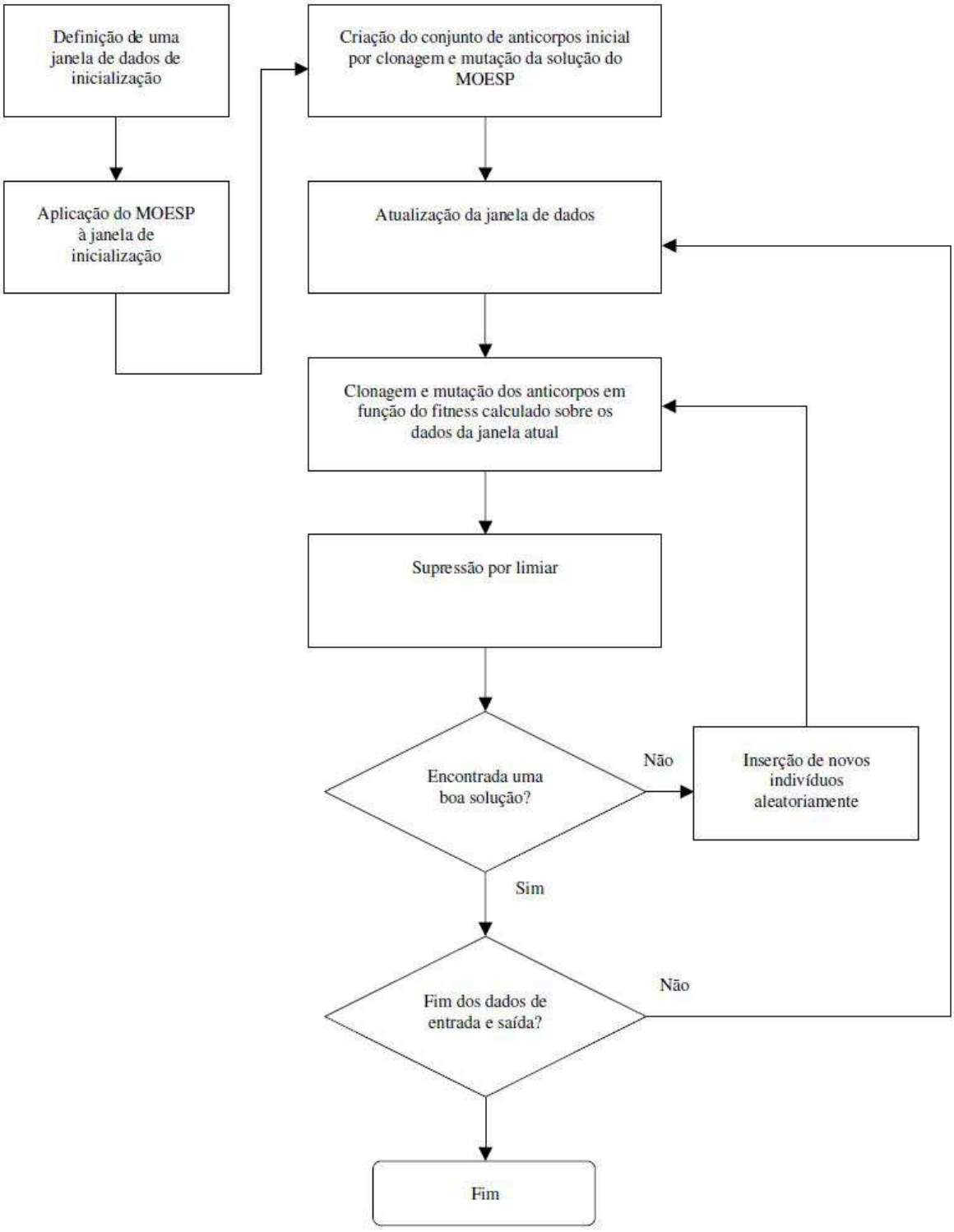

Fig. 7.25: Fluxograma do algoritmo imuno-inspirado para identificação de sistemas multivariáveis variantes no tempo.

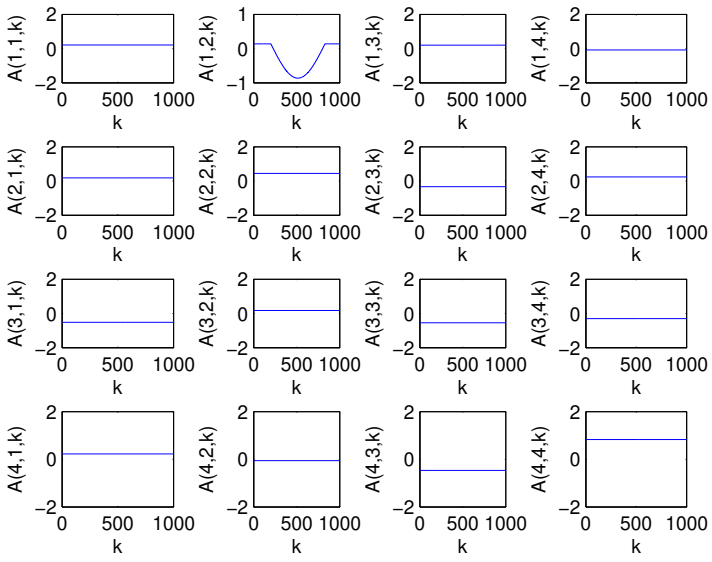

Fig. 7.26: Variação de cada elemento da matriz $A(k)$ em função de k para o sistema variante no tempo a ser identificado. Cada quadro representa um elemento da matriz.

$$B(k) = \begin{bmatrix} -0.0101 & 0.0317 & -0.9347 \\ -0.0600 & 0.5621 & 0.1657 \\ -0.3310 & -0.3712 & -0.5846 \\ -0.2655 & 0.4255 & 0.2204 \end{bmatrix} \tag{7.50}$$

$$C(k) = \begin{bmatrix} 0.6557 & -0.2502 & -0.5188 & -0.1229 \\ 0.6532 & -0.1583 & -0.055 & -0.2497 \end{bmatrix} \tag{7.51}$$

$$D(k) = \begin{bmatrix} -0.4326 & 0.1253 & -1.1465 \\ -1.6656 & 0.2877 & 1.1909 \end{bmatrix} \tag{7.52}$$

A variação de cada elemento da matriz $A(k)$ ao longo do tempo é ilustrada na figura 7.26. Apesar de apenas um dos elementos de apenas uma das matrizes do sistema a ser identificado sofra variações, os parâmetros de Markov do sistema são profundamente afetados, conforme pode ser visto nas figuras 7.27, 7.28, 7.29 e 7.30, em que as variações dos quatro primeiros parâmetros de Markov, que são respectivamente $C(k)B(k)$, $C(k)A(k)B(k)$, $C(k)A^2(k)B(k)$ e $C(k)A^3(k)B(k)$, são apresentadas.

O algoritmo proposto nesta seção da tese foi aplicado ao conjunto de entradas e saídas do sistema variante no tempo com $N_{ini} = 40$ e $N_w = 19$. O resultado do algoritmo é um sistema variante no tempo com a seguinte estrutura:

$$\begin{cases} x_{est}(k+1) = A_{est}(k)x(k) + B_{est}(k)u(k) \\ y(k) = C_{est}(k)x(k) + D_{est}(k)u(k) \end{cases} \tag{7.53}$$

em que as variações das matrizes $A_{est}(k)$, $B_{est}(k)$, $C_{est}(k)$ e $D_{est}(k)$ são apresentadas respectivamente nas figuras 7.31, 7.32, 7.33 e 7.34. Como pode ser visto das figuras, as matrizes determinadas

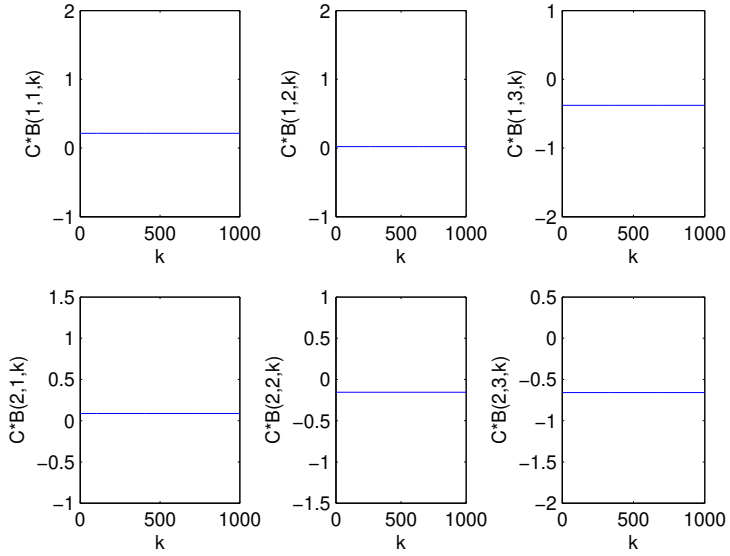

Fig. 7.27: Variação do parâmetro de Markov $C(k)B(k)$ do sistema a ser identificado em função de k.

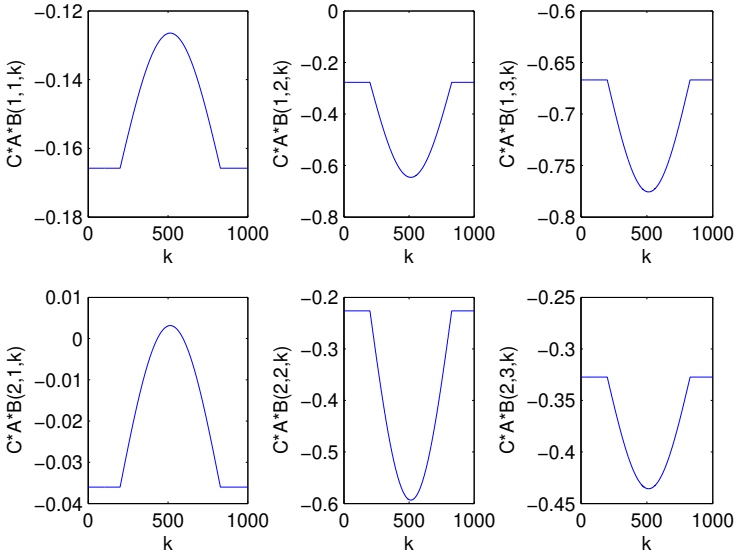

Fig. 7.28: Variação do parâmetro de Markov $C(k)A(k)B(k)$ do sistema a ser identificado em função de k.

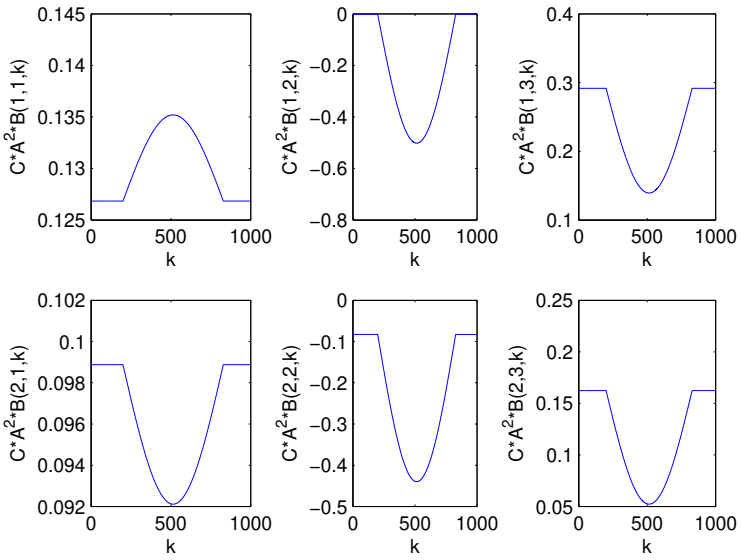

Fig. 7.29: Variação do parâmetro de Markov $C(k)A(k)^2B(k)$ do sistema a ser identificado em função de k.

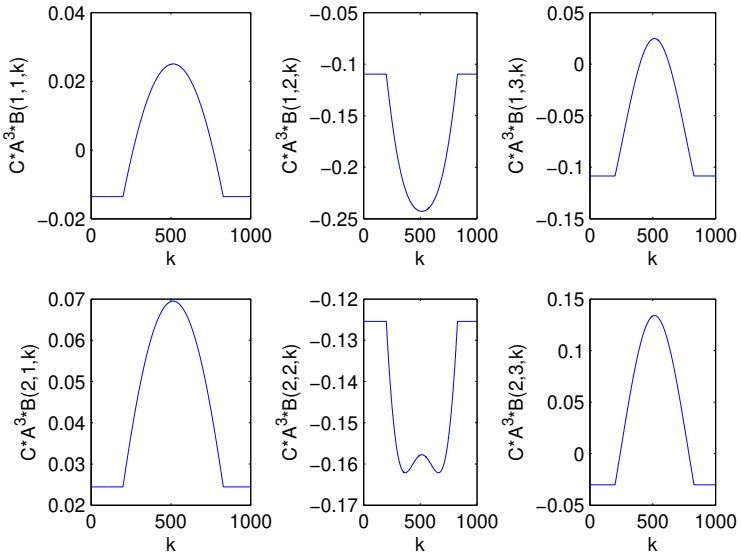

Fig. 7.30: Variação do parâmetro de Markov $C(k)A(k)^3B(k)$ do sistema a ser identificado em função de k.

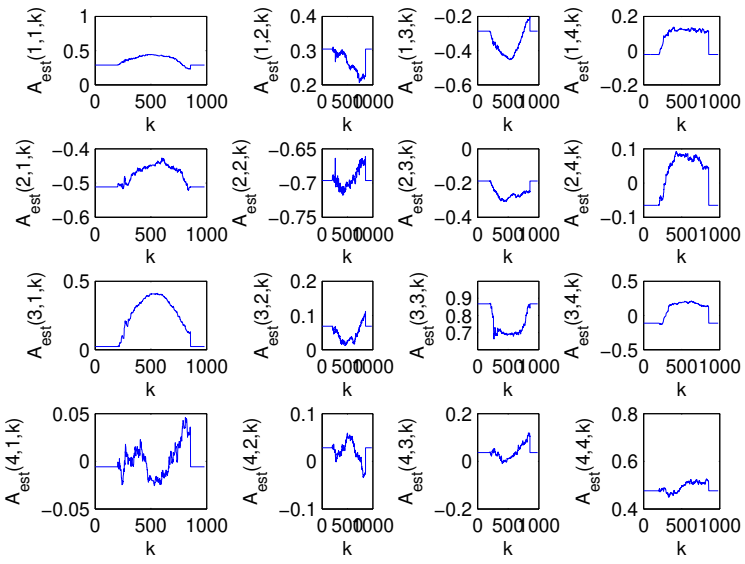

Fig. 7.31: Variação de cada elemento da matriz $A_{est}(k)$ em função de k.

com o algoritmo proposto são diferentes das matrizes do sistema a ser identificado. Além disto, as matrizes $B_{est}(k)$, $C_{est}(k)$ e $D_{est}(k)$ apresentadas nas figuras 7.32, 7.33 e 7.34 são variantes no tempo, diferentemente das matrizes $B(k)$, $C(k)$ e $D(k)$ do sistema a ser identificado.

Apesar de as matrizes estimadas com o algoritmo proposto serem diferentes das matrizes do sistema a ser identificado, os parâmetros de Markov variantes no tempo do modelo encontrado são similares aos parâmetros de Markov do sistema a ser identificado, conforme pode ser visto nas figuras 7.35, 7.36, 7.37 e 7.38, em que os quatro primeiros parâmetros de Markov do modelo e do sistema a ser identificado são plotados em função do instante de tempo k. A semelhança entre parâmetros de Markov demonstra que, mesmo com diferentes matrizes, a resposta ao impulso do modelo estimado é próxima da resposta ao impulso do sistema a ser identificado, portanto o modelo encontrado é uma boa aproximação para o sistema.

A mesma entrada usada para gerar as saídas do sistema a ser identificado foram aplicadas ao modelo variante no tempo obtido com o algoritmo proposto nesta seção. Para comparação esta entrada também foi aplicada ao modelo definido pela quadrupla $\{A_{ini}, B_{ini}, C_{ini}, D_{ini}\}$ obtida na etapa de inicialização do algoritmo pelo método MOESP e também a um modelo variante no tempo obtido com o algoritmo MOESP-VAR com janelas do mesmo tamanho que as usadas para o algoritmo proposto nesta seção, ou seja, 19.

As saídas em alguns instantes de tempo do sistema a ser identificado, do modelo variante no tempo obtido com o algoritmo imuno-inspirado proposto nesta tese, do modelo definido pela quádrupla de inicialização $\{A_{ini}, B_{ini}, C_{ini}, D_{ini}\}$ e do modelo variante no tempo obtido com o algoritmo MOESP-VAR são apresentadas nas figuras 7.39, 7.40 e 7.41.

Na figura 7.39 são apresentadas as saídas em um intervalo de tempo em torno do qual o sistema não varia. Desta figura nota-se que as saídas do modelo obtido com o algoritmo proposto nesta tese e do modelo obtido com a quádrupla de inicialização são próximas da saída do sistema a ser

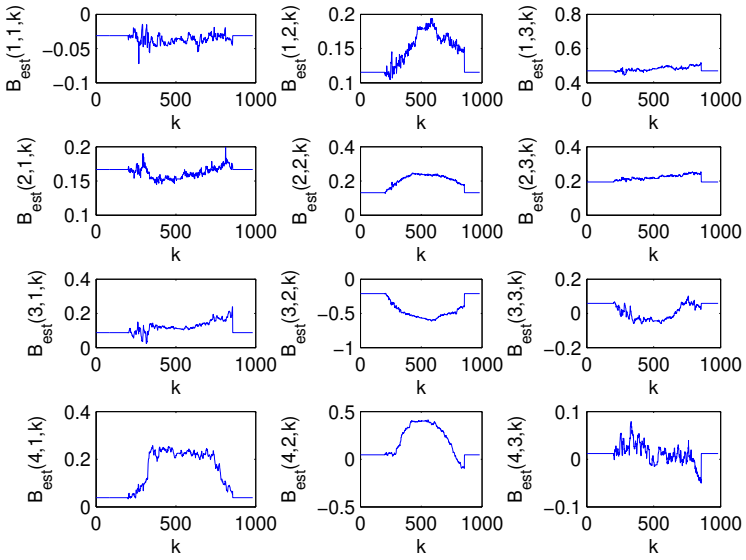

Fig. 7.32: Variação de cada elemento da matriz $B_{est}(k)$ em função de k.

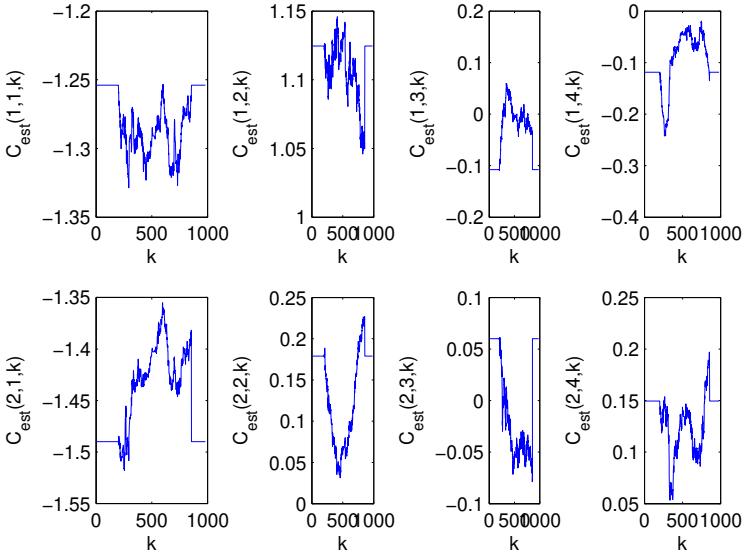

Fig. 7.33: Variação de cada elemento da matriz $C_{est}(k)$ em função de k.

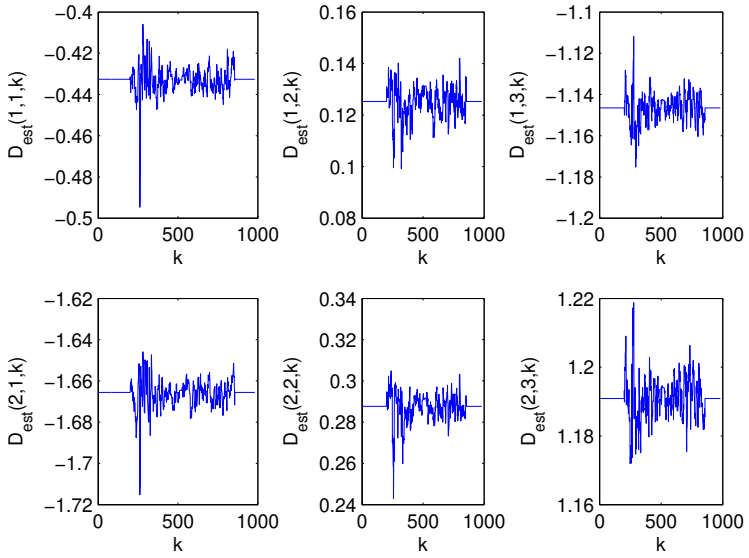

Fig. 7.34: Variação de cada elemento da matriz $D_{est}(k)$ em função de k.

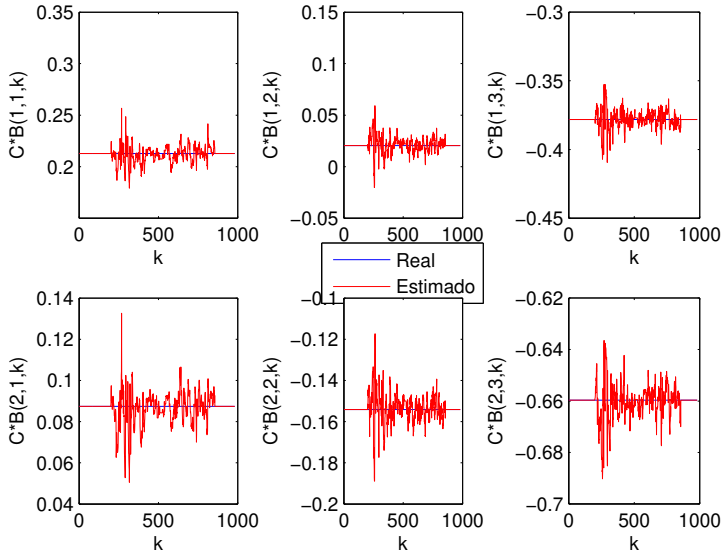

Fig. 7.35: Parâmetros de Markov do sistema a ser identificado e do modelo estimado. O parâmetro de Markov $C(k)B(k)$ do sistema a ser identificado é plotado em linha contínua azul e o parâmetro de Markov $C_{est}(k)B_{est}(k)$ do modelo estimado é plotado em cruzes vermelhas.

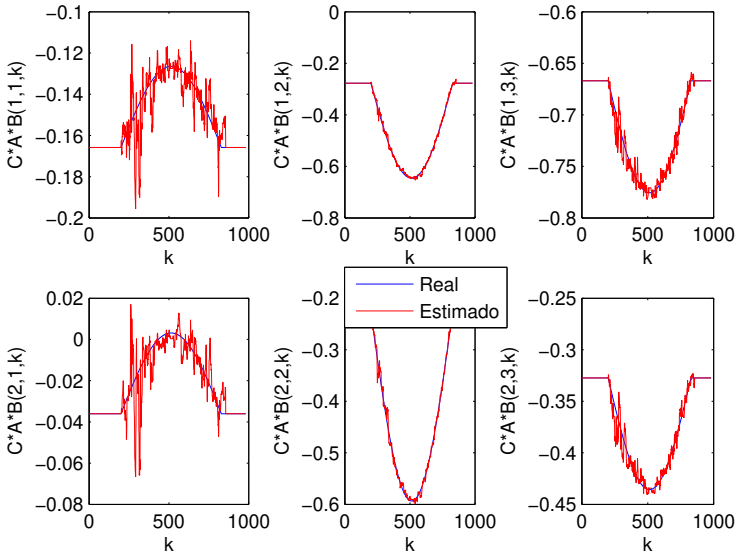

Fig. 7.36: Parâmetros de Markov do sistema a ser identificado e do modelo estimado. O parâmetro de Markov $C(k)A(k)B(k)$ do sistema a ser identificado é plotado em linha contínua azul e o parâmetro de Markov $C_{est}(k)A_{est}(k)B_{est}(k)$ do modelo estimado é plotado em cruzes vermelhas.

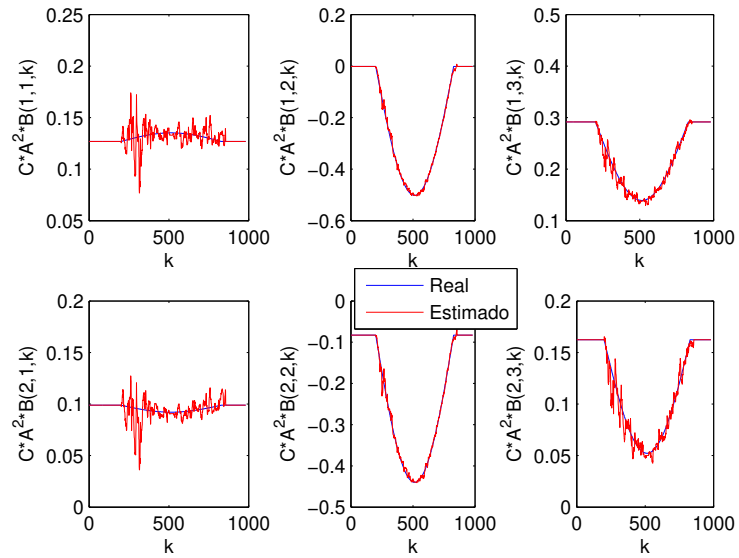

Fig. 7.37: Parâmetros de Markov do sistema a ser identificado e do modelo estimado. O parâmetro de Markov $C(k)A^2(k)B(k)$ do sistema a ser identificado é plotado em linha contínua azul e o parâmetro de Markov $C_{est}(k)A^2_{est}(k)B_{est}(k)$ do modelo estimado é plotado em cruzes vermelhas.

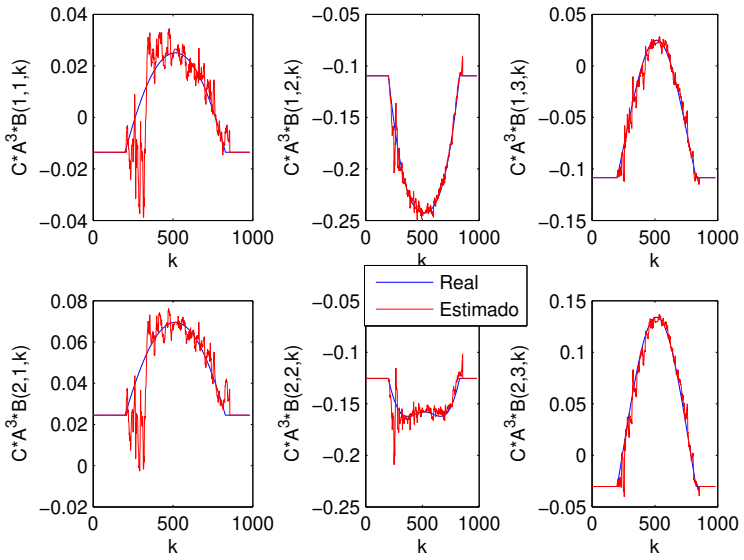

Fig. 7.38: Parâmetros de Markov do sistema a ser identificado e do modelo estimado. O parâmetro de Markov $C(k)A^3(k)B(k)$ do sistema a ser identificado é plotado em linha contínua azul e o parâmetro de Markov $C_{est}(k)A^3_{est}(k)B_{est}(k)$ do modelo estimado é plotado em cruzes vermelhas.

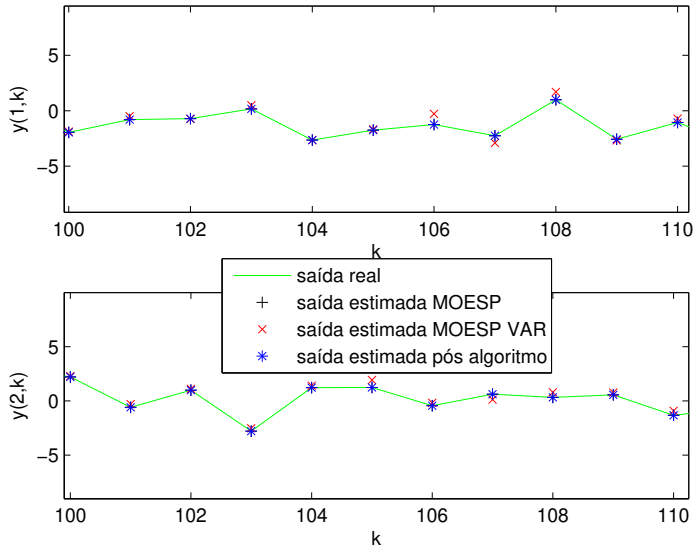

Fig. 7.39: Saídas no intervalo $100 < k < 110$.

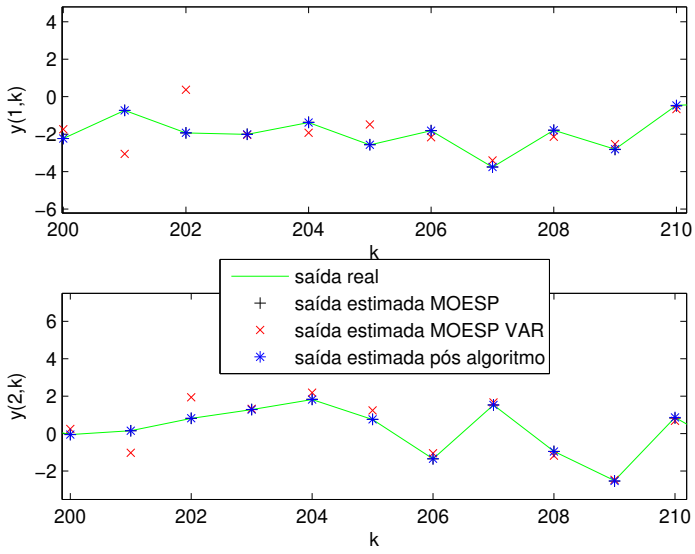

Fig. 7.40: Saídas no intervalo $200 < k < 210$.

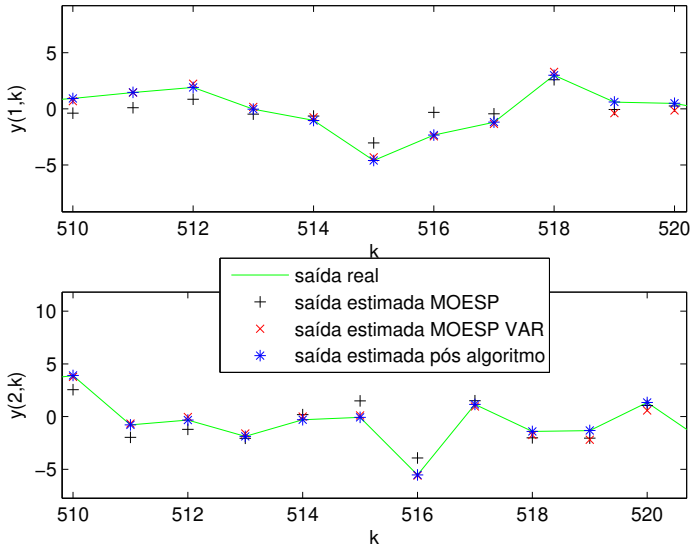

Fig. 7.41: Saídas no intervalo $510 < k < 520$.

Tab. 7.11: *Fitness* calculados levando em consideração todas as saídas do experimento para os métodos MOESP, MOESP-VAR e Imuno-VAR (método proposto nesta tese).

MOESP	MOESP-VAR	Imuno-VAR
2.7129	1.5562	12.7997

identificado. Isto é esperado uma vez que neste intervalo de tempo o sistema a ser identificado tem as mesmas matrizes que no intervalo de tempo utilizado para aplicação da inicialização MOESP. As saídas do modelo obtido pelo algoritmo MOESP-VAR em alguns casos são diferentes das saídas do sistema a ser identificado. Isto ocorre pois o número de amostras em cada janela é pequeno para garantir exatidão na decomposição de matrizes necessária no método MOESP.

Na figura 7.40 são apresentadas as saídas em torno de um instante de tempo em que o sistema sofre a mais brusca variação neste experimento. Da figura nota-se facilmente que as saídas do modelo definido pelas matrizes de inicialização ainda demonstram acurácia, uma vez que o sistema está em uma condição próxima daquela em que a inicialização foi feita. As saídas obtidas com o algoritmo proposto nesta tese também têm acurácia. As saídas obtidas com o algoritmo MOESP-VAR não têm acurácia pois neste ponto o sistema sofre uma variação brusca e as amostras de entrada e saída não satisfazem a hipótese de não variação do sistema dentro de uma janela.

Na figura 7.41 são apresentadas as saídas em um instante de tempo em que o sistema está distante do ponto em que a inicialização foi feita. Como é de se esperar, as saídas do modelo obtido durante a inicialização estão distantes das saídas reais do sistema a ser estimado. As saídas obtidas com o algoritmo MOESP-VAR são próximas das saídas do sistema pois o mesmo não está sofrendo grandes variações neste instante, e as saídas obtidas com o algoritmo proposto nesta tese estão próximas das saídas reais.

Em resumo, o modelo obtido com o método proposto nesta tese reproduz as saídas do sistema mesmo que este seja submetido a uma variação brusca - ponto em que o MOESP-VAR falha - e mesmo que ele esteja distante da região em que foi inicializado - ponto em que o MOESP puro falha.

Para comparar analiticamente os resultados obtidos, o *fitness* definido na equação 7.47 foi calculado para cada um dos três métodos levando em consideração todas as 1000 saídas do experimento. Os resultados são apresentados na tabela 7.11. Da tabela nota-se que o modelo obtido com o algoritmo proposto nesta tese tem *fitness* maior que os modelos obtidos com o MOESP e com o MOESP-VAR. O motivo da performance não tão boa do MOESP é que o sistema é variante no tempo, portanto uma identificação com um modelo invariante no tempo é claramente inadequada. O MOESP-VAR por sua vez não tem um resultado tão preciso devido ao tamanho da janela que foi definido. Para verificar este resultado a identificação do sistema foi refeita com o algoritmo MOESP-VAR mas com uma janela de tamanho 41. O *fitness* obtido foi de 3.2264, que é maior que o obtido com o modelo encontrado pelo método MOESP, mas ainda distante do resultado obtido com o modelo determinado pelo algoritmo proposto nesta tese.

Com o comportamento do algoritmo proposto nesta tese também é possível determinar se um sistema é ou não variante no tempo, e também a intensidade desta variação. Isto é possível ao se observar o número de iterações necessárias para alcançar o *fitness* mínimo requerido F_{req} para fim do algoritmo a cada janela. Caso o número de iterações seja pequeno, o sistema sofreu uma pequena

variação. Por outro lado, se o número de iterações for grande, o sistema está sofrendo uma grande variação. Na figura 7.42 é apresentada a variação do número de iterações ao longo das janelas. Da figura é claro que o sistema começa a variar a partir de $k > 200$, como de fato ocorre. Mais que isto, ao se comparar a figura 7.42 às figuras 7.27, 7.28, 7.29 e 7.30 nota-se que o número de iterações é proporcional a taxa de variação do sistema, ou seja, às derivadas dos parâmetros de Markov do sistema.

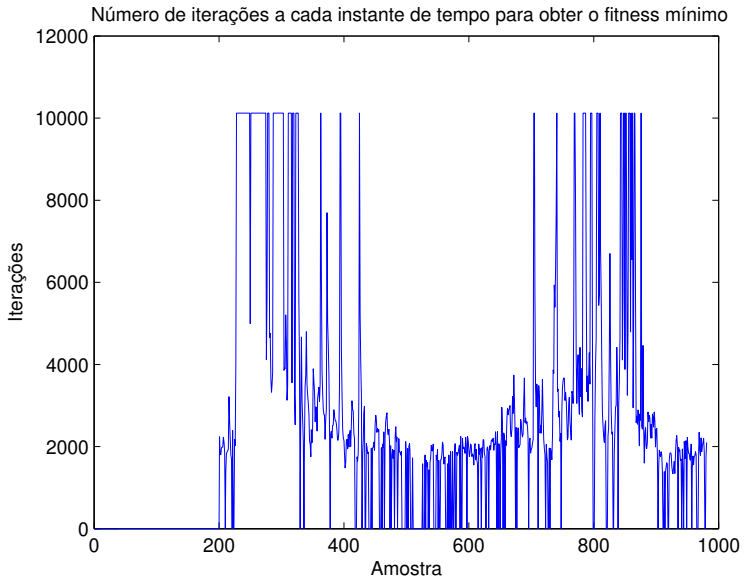

Fig. 7.42: Número de iterações para se alcançar o *fitness* mínimo requerido a cada janela.

Na figura 7.42 o número de iterações máximo não passa de 10000 pois o algoritmo foi programado para limitar o número máximo de iterações por janela a este número. Esta limitação permitiu que modelos um pouco piores que os desejados fossem adotados em regiões críticas do sistema. No entanto isto não prejudicou o resultado geral do algoritmo, conforme apresentado.

7.4.4 Conclusão

A partir dos resultados apresentados na seção anterior, a conclusão é que o algoritmo imuno-inspirado proposto nesta tese é capaz de seguir as variações de um sistema multivariável variante no tempo. Se o algoritmo MOESP fosse utilizado para modelar o sistema variante no tempo como um sistema invariante no tempo, as saídas do modelo em um instante de tempo em que o sistema estiver distante do ponto em que estava quando as matrizes do MOESP foram estimadas não serão próximas das saídas do sistema a ser identificado. O algoritmo proposto nesta tese tem também a propriedade de seguir variações do sistema mais rápidas que aquelas que podem ser detectadas pelo método MOESP-VAR.

Além da evidência apontada pela semelhança entre a saída do sistema a ser identificado e do modelo determinado pelo algoritmo proposto nesta tese, foi demonstrado que os parâmetros de Markov

do sistema são muito semelhantes aos parâmetros de Markov do modelo obtido com o algoritmo proposto nesta tese. Sendo assim, a resposta ao impulso do modelo variante no tempo obtido com este algoritmo é semelhante à do sistema, indicando que a identificação foi bem sucedida.

A técnica utilizada nesta tese para identificação do sistema variante no tempo, que é a interpretação do modelo no espaço de estado como um ponto em um espaço de quádruplas de matrizes e a determinação do ponto que minimiza o erro entre a saída do modelo e a saída do sistema a ser identificado por meio da solução de um problema de otimização, foi fundamental para o estabelecimento da técnica de solução final do problema, que foi o uso do algoritmo imuno-inspirado. O método proposto pode ser usado para identificar sistemas variantes no tempo reais e, uma vez que é recursivo, pode também ser usado para identificação em tempo real de sistemas variantes no tempo.

Capítulo 8

Conclusão e trabalhos futuros

Nesta tese, são tratados alguns problemas relacionados à identificação de sistemas multivariáveis discretos no espaço de estado, à realização de séries temporais multivariáveis discretas no espaço de estado e à modelagem de séries temporais multivariáveis discretas no espaço de estado. A metodologia de solução dos problemas foi sua transformação em problemas de otimização e sua solução por algoritmos imuno-inspirados.

A conclusão principal do estudo que resultou nesta tese é que, em geral, os métodos de otimização heurística implementados para a solução dos problemas, baseados em algoritmos imuno-inspirados, foram capazes de resolver os problemas de otimização criados e que, com isto, os problemas tratados, que não poderiam ser resolvidos por métodos conhecidos, foram solucionados.

A partir das contribuições propostas nesta tese, abre-se a possibilidade de estudos adicionais, como o estudo da performance de outros algoritmos de otimização heurística, como algoritmos genéticos, algoritmos de otimização por enxame de partículas, dentre outros, para solução dos problemas de otimização propostos. Os resultados obtidos com os outros algoritmos podem ser comparados aos resultados apresentados nesta tese, de forma que se encontre qual é a melhor alternativa para a solução de cada um dos problemas de otimização definidos. Além da solução dos problemas de otimização com outros tipos de algoritmos, também pode ser objeto de pesquisas futuras a aplicação de variações de algoritmos imuno-inspirados para a solução dos problemas aqui tratados, como a implementação da distância de linha, proposta em [29], ou da aplicação da etapa de supressão em apenas algumas das iterações, e não em todas, como foi feito nos algoritmos apresentados nesta tese.

Além dos problemas solucionados nesta tese, outros problemas envolvidos com a identificação de sistemas, realização de séries temporais e modelagem de séries temporais podem ter propostas de soluções por meio de sua transformação em problemas de otimização. Como exemplos, pode-se propor um método de realização de séries temporais imuno-inspirado ao se minimizar a distância entre as covariâncias de soluções candidatas e as covariâncias da série temporal que se quer realizar. Isto seria muito semelhante ao que foi feito no método de geração de ruído branco proposto nesta tese, com a diferença que, ao invés de se minimizar a distância entre as covariâncias para atrasos não nulos a zero, se minimizaria a distância entre as covariâncias para quaisquer atrasos e a covariância que se quer realizar. Ainda nesta linha, pode-se também determinar o modelo em espaço de estado que realiza uma determinada série temporal ao se encontrar as matrizes de modelo que, quando sujeito a um ruído branco, tem uma saída cuja covariância seja tal que a diferença entre ela e a covariância da série temporal que se quer realizar seja mínima.

Outra proposta que pode ser implementada é a solução das LMIs envolvidas no problema de realização de séries temporais através da definição de um problema de otimização e solução por métodos heurísticos. Este problema poderia ser tratado semelhantemente ao que foi feito com a equação de Riccati mas, neste caso, como se trata de uma desigualdade, o algoritmo deve ter algum mecanismo para guardar todas as soluções que satisfazem a desigualdade. Também seria interessante o uso de um critério para valorizar as soluções na fronteira de desigualdade, de forma que os limites da desigualdade possam ser definidos. Uma maneira de verificar se o algoritmo foi bem sucedido é comparar suas soluções com os limites superior e inferior determinados por Faurre.

Ainda na linha de realização de séries temporais, um problema de otimização pode ser proposto para determinação do valor de Σ no problema de realização por predição ótima. Também pode ser proposto um método algébrico para solução do sistema formado pela equação matricial quadrática e a equação matricial linear deste problema.

Na linha de modelagem de séries temporais, um trabalho futuro que pode ser realizado é a modelagem de séries temporais variantes no tempo. Este trabalho seria muito semelhante à identificação de sistemas variantes no tempo tratada nesta tese.

Na linha da geração de ruídos brancos, pode-se determinar um método imuno-inspirado que seja capaz de adicionar a uma sequência já existente algumas amostras adicionais, de forma a manter a propriedade de ruído branco do sinal. Este método pode ser implementado de forma recursiva, de maneira que, no caso de um problema em que o número de amostras total não é conhecido a priori, o ruído branco pudesse ser gerado de acordo com a demanda.

Com relação aos algoritmos imuno inspirados, as propostas apresentadas nesta tese para solução de problemas de otimização com restrições, com ou sem travessia de zonas proibidas, podem ter seus resultados comparados quando aplicadas a problemas de otimização com restrições conhecidos na literatura. Desta maneira, pode-se avaliar em quais tipos de problema cada uma destas técnicas é mais própria para ser usada.

Nesta tese o foco foi dado nos problemas de identificação, realização e modelagem discretas. Uma outra linha de pesquisa que pode ser desenvolvida é a adaptação e avaliação das técnicas aqui descritas para problemas em tempo contínuo.

Em resumo, a existência de algoritmos heurísticos de otimização permite a solução de problemas de otimização com funções objetivo não muito diretas, e que portanto, seriam dificilmente formulados como problemas de otimização clássica. Desta forma, problemas que antes dificilmente seriam tratados como problemas de otimização podem ser definidos como tal, e as soluções obtidas em alguns casos como os tratados nesta tese, são melhores que as obtidas com métodos clássicos.

Referências Bibliográficas

[1] AGUIRRE, L. A. *Introdução à identificação de sistemas*, 2 ed. Editora UFMG, 2004.

[2] ANDERSON, B. D. O., AND MOORE, J. B. *Optimal filtering*. Dover Publications, 2005.

[3] ANDERSON, B. D. O., MOORE, J. B., AND LOO, S. G. Spectral factorization of time varying covariance functions. *IEEE Transactions on information theory* (September 1969), 550–557.

[4] AOKI, M. *State Space Modeling of Time Series*. Springer-Verlag, 1987.

[5] BARRETO, G. *Modelagem computacional distribuída e paralela de sistemas e de séries temporais multivariáveis no espaço de estado*. PhD thesis, Unicamp, 2002.

[6] BARRETO, G., AND BOTTURA, C. P. Revisitando os fundamentos de identificação multivariável no espaço de estados ii - idéias básicas para o método de subespaços. In *Proceedings of 2nd DINCON* (August 2003).

[7] BIRKHOFF, G. D. Proof of the ergodig theorem. *Proceedings of the National Academy of Sciences of the United States of America 17* (December 1931), 656–660.

[8] BOLDRINI, J. L., COSTA, S. I. R., RIBEIRO, V. L. F. F., AND WETZLER, H. G. *Álgebra Linear*, 2 ed. Ed. Harper & Row do Brasil Ltda., 1978.

[9] BOTTURA, C. P. Elementos da teoria de probabilidades (notas de aula). UNICAMP, 1977.

[10] BOTTURA, C. P. *Análise linear de sistemas*. Ed. Guanabara Dois, 1982.

[11] BOTTURA, C. P., AND BARRETO, G. Revisitando os fundamentos de identificação multivariável no espaço de estados i - realização de estado e operador de hankel. In *Proceedings of 2nd DINCON* (August 2003).

[12] BURNET, F. M. Clonal selection and after. In *Theoretical Immunology*, G. I. Bell, A. S. Perelson, and G. H. Pimgley Jr, Eds., pp. 63–85.

[13] CÁCERES, A. F. T. *Identificação e controle estocásticos descentralizados de sistemas interconectados multivariáveis no espaço de estado*. PhD thesis, Unicamp, Julho 2005.

[14] CALLIOLI, C. A., DOMINGUES, H. H., AND COSTA, R. C. F. *Álgebra Linear e Aplicações*. Ed. Atual, 1995.

[15] CLAVIJO, D. G. Métodos de subespaços para identificação de sistemas: propostas de alterações, implementações e avaliações. Master's thesis, UNICAMP, 2008.

[16] CUTELLO, V., NARIZI, G., NICOSIA, G., AND PAVONE, M. Real coded clonal selection algorithm for global numerical optimization using a new inversely proportional hypermutation operator. *21st Annual ACM Symposium on Applied Computing, SAC 2006* (2006), 950–954.

[17] CUTELLO, V., NARZISI, G., NICOSIA, G., AND PAVONE, M. Clonal selection algorithms: A comparative case study usign effective mutation potentials. *4th Intl. Conference on Artificial Immune Systems ICARIS 2005* (2005), 13–28.

[18] CUTELLO, V., AND NICOSIA, G. An immunological approach to combinatorial optimization problems. *Advances in artifical intelligence IBERAMIA* (2002), 361–370.

[19] DA FONSECA NETO, J. V. *Alocação computacional inteligente de autoestruturas para controle multivariável*. PhD thesis, UNICAMP, março 2000.

[20] DE CASTRO, L. N., AND TIMMIS, J. An artificial immune network for multimodal function optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation (CEC), Hawai, USA* (2002), pp. 699–704.

[21] DE CASTRO, L. N., AND TIMMIS, J. *Artificial Immune Systems - A new computational intelligence approach*. Springer Verlag, 2002.

[22] DE CASTRO, L. N., AND VON ZUBEN, F. J. Artificial immune systems - part i - basic theory and appllications. Tech. rep., DCA - Unicamp, December 1999.

[23] DE CASTRO, L. N., AND VON ZUBEN, F. J. Artificial immune systems - part ii - a survey of applications. Tech. rep., DCA - Unicamp, February 2000.

[24] DE CASTRO, L. N., AND VON ZUBEN, F. J. The clonal selection algorithm with engineering applications. In *Workshop proceedings of the GECCO 2000* (July 2000), pp. 36–37.

[25] DE CASTRO, L. N., AND VON ZUBEN, F. J. Learning and optimization using the clonal selection principle. *IEEE Transactions on evolutionary computation 6* (June 2002), 239–251.

[26] DE FRANÇA, F. O. Algoritmos bio-inspirados aplicados à otimização dinâmica. Master's thesis, UNICAMP, Dezembro 2005.

[27] DE FRANÇA, F. O., COELHO, G. P., CASTRO, P. A. D., AND VON ZUBEN, F. J. Conceptual and practical aspects of the ainet family of algorithms. *International Journal of Natural Computing Research 1* (January-March 2010), 1–35.

[28] DE FRANÇA, F. O., COELHO, G. P., AND VON ZUBEN, F. J. On the diversity mechanisms of opt-ainet: a comparative study with fitness sharing. *Proceedings of the 2010 IEEE World Congress on Computational Inteligence* (July 2010), 3523–3530.

[29] DE FRANÇA, F. O., VON ZUBEN, F. J., AND DE CASTRO, L. N. An artificial immune network for multimodal function optimization on dynamic environments. *Proceedings of the 2005 conference on genetic and evolutionary computation, ACM* (2005), 289–296.

[30] DOOB, J. I. *Stochastic Processes*. Willey, New York, 1953.

[31] FAURRE, P. L. *Representation of stochastic processes*. PhD thesis, Stanford University, February 1967.

[32] FAURRE, P. L. Stochastic realization algorithms. In *System Identification: Advances and Case Studies* (1976), R. Mehra and D. Lainiotis, Eds., pp. 1–25.

[33] G. WELCH, G. B. An introduction to the kalman filter.

[34] GEROMEL, J. C., AND PALHARES, A. G. B. *Análise Linear de Sistemas Dinâmicos*. Editora Edgard Blücher LTDA, São Paulo, 2004.

[35] GEVERS, M. R., AND KAILATH, T. An innovations approach to least squares estimation part vi - discrete-time innovations representations and recursive estimation. *IEEE Transactions on automatic control* (December 1973), 588–600.

[36] GIESBRECHT, M. Modelagem computacional do aquecimento de um motor de indução monofásico aplicado a máquinas de lavar roupas durante a etapa de agitação. Master's thesis, Unicamp, Novembro 2007.

[37] GIESBRECHT, M., AND BOTTURA, C. . P. An immuno inspired approach to generate white noise. *Proceedings of the Fourth International Workshop on Advanced Computational Intelligence* (October 2011), 742–749.

[38] GIESBRECHT, M., AND BOTTURA, C. . P. Immuno inspired approaches to model discrete time series at state space. *Proceedings of the Fourth International Workshop on Advanced Computational Intelligence* (October 2011), 750–756.

[39] GIESBRECHT, M., AND BOTTURA, C. P. An immuno-inspired approach to find the steady state solution of riccati equations not solvable by schur method. *Proceedings of the 2010 IEEE World Congress on Computational Inteligence* (July 2010).

[40] GIESBRECHT, M., AND BOTTURA, C. P. Uma proposta imuno-inspirada para a solução algébrica da equação de riccati no problema de identificação de séries temporais no espaço de estado. *Anais do XVIII Congresso Brasileiro de Automática* (Setembro 2010).

[41] GIESBRECHT, M., AND BOTTURA, C. P. Uma proposta imuno inspirada para a modelagem de séries temporais discretas no espaço de estado. In *Anais do X Simpósio Brasileiro de Automação Inteligente* (Universidade Federal de São João del-Rei, São João del-Rei, MG, Brasil, setembro 2011), pp. 462–467.

[42] HO, B. L., AND KALMAN, R. E. Effective construction of linear state-variable models from input-output functions. *Regelungstechnik - zeitschrift für steuern, regeln und automatisieren* (1966), 545–548.

[43] JERNE, N. K. Towards a network theory of the immune system. *Annales d'immunologie* (Jan 1974), 373–389.

[44] KAILATH, T., AND GEESEY, R. A. An innovations approach to least squares estimation part iv - recursive estimation given lumped covariance functions. *IEEE Transactions on automatic control* (December 1971), 720–727.

[45] KALMAN, R. E. A new approach to linear filtering and prediction problems. *Transactions of the ASME-Journal of Basic Engineering* (1960).

[46] KATAYAMA, T. *Subspace Methods for System Identification: a Realization Approach*. Springer Verlag, Leipzig, 2005.

[47] KLOEDEN, P. E., AND PLATEN, E. *Numerical Solution of Stochastic Differential Equations*. Springer, 1992.

[48] LAUB, A. J. A schur method for solving algebraic riccati equations. *IEEE Transactions on automatic control AC24*, 6 (December 1979), 913–921.

[49] LAUB, A. J. Numerical aspects of solving algebraic riccati equations. In *Proceedings of IEEE Conference on decision and control* (1983), IEEE, pp. 184–186.

[50] LIPSCHUTZ, S. *Teoria y problemas de Algebra Lineal*. Libros McGraw-Hill, 1971.

[51] OKSENDAL, B. K. *Stochastic differential equations - An introduction with applications*, 6 ed. Springer, 2003.

[52] PARZEN, E. *Stochastic Processes*. Holden Day INC, San Francisco, London, Amsterdan, 1962.

[53] SERRA, G. L. D. O. *Propostas de metodologias para identificação e controle inteligentes*. PhD thesis, UNICAMP, 2005.

[54] STRANG, G. *Introduction to Linear Algebra*. Wellesley-Cambridge Press, 1993.

[55] TAMARIZ, A. D. R. Uma nova proposta para solução computacional da equação algebrica de riccati em formas sequencial e paralela. Master's thesis, UNICAMP, 1999.

[56] TAMARIZ, A. D. R. *Modelagem computacional de dados e controle inteligente no espaço de estado*. PhD thesis, Unicamp, Julho 2005.

[57] TAMARIZ., A. D. R., BOTTURA, C. P., AND BARRETO, G. Iterative moesp type algorithm for discrete time variant system identification. *Proceedings of the 13th Mediterranean Conference on Control and Automation (MED 2005)* (June 2005).

[58] TER HAAR, D. *Elements of statistical mechanics*, 3rd ed. Butterworth-Heinemann Ltd, Oxford, 1995.

[59] TOBAR, J., BOTTURA, C. P., AND GIESBRECHT, M. Computational modeling of multivariable non-stationary time series in the state space by the aoki var algorithm. *IAENG International Journal of Computer Science 37*, November (2010).

[60] VAUGHAN, D. R. A nonrecursive algebraic solution for the discrete riccati equation. *IEEE Transactions on automatic control 15*, 5 (October 1970), 597–599.

[61] VERHAEGEN, M., AND DEWILDE, P. Subspace model identification - part 2 : Analysis of the elementary output-error state-space model identification algorithms. *International Journal of Control 56*, 5 (November 1992), 1211–1241.

[62] VERHAEGEN, M., AND DEWILDE, P. Subspace model identification - part i : The output-error state-space model identification class of algorithms. *International Journal of Control 56*, 5 (November 1992), 1187–1210.

[63] VON NEUMANN, J. Proof of the quasi-ergodic hypothesis. *Proceedings of the National Academy of Sciences of the United States of America 18*, 1 (January 1932), 70–82.