

UNIVERSIDADE FEDERAL DO CEARÁ
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

JOSÉ MARIA PIRES DE MENEZES JÚNIOR

REDES NEURAS DINÂMICAS PARA PREDIÇÃO E
MODELAGEM NÃO-LINEAR DE SÉRIES TEMPORAIS

FORTALEZA

2006

JOSÉ MARIA PIRES DE MENEZES JÚNIOR

**REDES NEURAS DINÂMICAS PARA PREDIÇÃO E
MODELAGEM NÃO-LINEAR DE SÉRIES TEMPORAIS**

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Engenharia de Teleinformática, da Universidade Federal do Ceará, como parte dos requisitos exigidos para obtenção do grau de Mestre em Engenharia de Teleinformática.

Orientador: Prof. Dr. Guilherme de Alencar Barreto

FORTALEZA

2006

Ficha catalográfica elaborada pela bibliotecária Aline Vieira

M511r

Menezes Júnior, José Maria Pires de

Redes neurais dinâmicas para predição e modelagem não-linear de séries temporais / José Maria Pires de Menezes Júnior

134 f.; il.

Orientador: Prof. Dr. Guilherme de Alencar Barreto

Mestrado em Engenharia de Teleinformática

Universidade Federal do Ceará, Fortaleza, 2006.

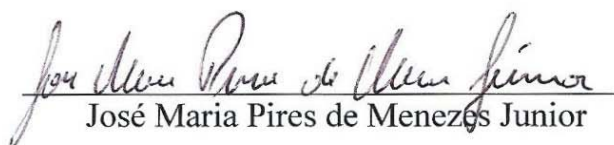
1. Tráfego 2. Caos 3. Recorrente I. Barreto, Guilherme de Alencar II. Universidade Federal do Ceará III. Título

CDD 621.3

José Maria Pires de Menezes Junior

Redes Neurais Dinâmicas para Predição e Modelagem Não-Linear de Séries Temporais


Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia de Teleinformática e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia de Teleinformática da Universidade Federal do Ceará.


José Maria Pires de Menezes Junior

Banca Examinadora:


Prof. Guilherme de Alencar Barreto, Dr.


Prof. Paulo-César Cortez, Dr.


Prof. Danielo Gonçalves Gomes, Dr.


Prof. Adrião Duarte Dória Neto, Dr.

Fortaleza, 14 de Julho de 2006

*Dedico este trabalho aos meus pais
José Maria e Rosa Virgínia
pelo constante apoio, incentivo e
admiração.*

Agradecimentos

A Deus, acima de tudo.

Ao meu orientador, Prof. Guilherme de Alencar Barreto, a quem sou grato pela orientação, paciência e confiança depositada.

Aos meus irmãos, pela ajuda em todas as horas.

Aos colegas de laboratório, por estarem sempre prontos a ajudar, proporcionando excelente ambiente de trabalho.

Ao Prof. João César Moura Mota, pelo apoio durante esta jornada.

Aos professores e funcionários do Departamento de Engenharia de Teleinformática que de forma direta ou indireta participaram do desenvolvimento deste trabalho.

À FUNCAP (Fundação Cearense de Amparo à Pesquisa) pelo suporte financeiro.

Em especial à Ana Valéria, minha namorada, pelo amor, carinho, incentivo, admiração e apoio incondicional.

Resumo

Neste trabalho, redes neurais dinâmicas são avaliadas como modelos não-lineares eficientes para predição de séries temporais complexas. Entre as arquiteturas avaliadas estão as redes FTDNN, Elman e NARX. A capacidade preditiva destas redes são testadas em tarefas de predição de um-passo-adiante e múltiplos-passos-adiante. Para este fim, são usadas as seguintes séries temporais: série laser caótico, série caótica Mackey-Glass, além de séries de tráfego de rede de computadores com características auto-similares.

O uso da rede NARX em predição de séries temporais é uma contribuição desta dissertação. Esta rede possui uma arquitetura neural recorrente usada originalmente para identificação entrada-saída de sistemas não-lineares. A entrada da rede NARX é formada por duas janelas deslizantes (*sliding time window*), uma que desliza sobre o sinal de entrada e outra que desliza sobre sinal de saída. Quando aplicada para predição caótica de séries temporais, a rede NARX é projetada geralmente como um modelo autoregressivo não-linear (NAR), eliminando a janela de atraso da saída. Neste trabalho, é proposta uma estratégia simples, porém eficiente, para permitir que a rede NARX explore inteiramente as janelas de tempo da entrada e da saída, a fim de melhorar sua capacidade preditiva. Os resultados obtidos mostram que a abordagem proposta tem desempenho superior ao desempenho apresentado por preditores baseados nas redes FTDNN e Elman.

Palavras-chave: Redes Neurais Artificiais, Redes Neurais Dinâmicas, Modelos NARX, Sistemas Caóticos, Séries Temporais, Modelagem de Tráfego.

Abstract

This work evaluates the use of dynamic neural networks as efficient nonlinear tools for time series prediction and modeling. Among the evaluated architectures, we list the following: FTDNN, Elman, NARX neural networks, whose predictive performances are tested in one-step-ahead, multi-step-ahead and dynamic modeling tasks. To this end, the following well-known time series were used for benchmarking purposes: laser time series of the Santa Fe competition, chaotic time series generated from Mackey-Glass maps, as well as auto-similar traffic-like time series.

In particular, the application of the NARX network in time series prediction is a contribution of this dissertation. The NARX network model is a recurrent neural architecture commonly used for input-output identification of nonlinear systems. The input of the NARX network is formed by two tapped-delay lines, one sliding over the input signal and the other one over the output signal. When applied to chaotic time series prediction, the NARX network is usually designed as a plain nonlinear autoregressive (NAR) model by eliminating the output's delay line. In this paper, we propose a simple but efficient strategy to allow the NARX network to fully exploit input and output delay lines to improve its prediction performance. We use the laser data of the Santa Fe Competition to evaluate the proposed approach in multi-step-ahead prediction tasks. The results show that the proposed approach consistently outperforms standard neural network based predictors, such as the FTDNN and Elman networks.

Keywords: Artificial Neural Networks, Dynamic Neural Networks, NARX models, Chaotic Systems, Time Series, Traffic Models.

Lista de Figuras

2.1	(a) mapa logístico para estado caótico; (b) autocorrelação para o mapa logístico com $a = 4$	17
2.2	reconstrução do atrator de Hénon. (a) atrator original; (b) atrator reconstruído para $\tau = 2$ e $n_d = 2$; (c) $\tau = 2$ e $n_d = 3$; (d) $\tau = 1$ e $n_d = 2$	18
2.3	atrator de Lorenz.	19
2.4	série de Caótica de Lorenz: (a) informação mútua para o cálculo do atraso de imersão; (b) método de CAO para o cálculo da dimensão de imersão.	20
2.5	diagramas de recorrência: (a) mapa logístico, série periódica; (b) mapa logístico, série caótica; (c) ruído branco aleatório.	22
2.6	séries temporais geradas a partir do mapa logístico para duas condições iniciais diferentes, linha cheia, $x(0) = 0,1$ e linha pontilhada, $x(0) = 0,1001$	25
3.1	esboço dos pontos de ocorrência e tempo entre-chegadas.	31
3.2	realização típica de um processo de contagem.	33
3.3	dinâmica do mapa intermitente simples.	49
3.4	dinâmica do mapa intermitente duplo.	50
4.1	(a) neurônios da camada escondida; (b) neurônios de saída.	55
4.2	arquitetura genérica de uma rede neural dinâmica construída a partir de uma rede neural estática por meio de mecanismos externos de memória de curta duração.	62
4.3	exemplo de atrasadores formando uma janela de tempo de comprimento na entrada de uma rede neural.	63
4.4	arquitetura genérica de uma rede FTDNN de uma camada escondida.	64
4.5	sinapse representada como um filtro FIR.	65
4.6	neurônio formado por sinapses do tipo FIR.	65

4.7	arquitetura da rede de Elman aplicada ao problema de predição não-linear de séries temporais.	69
4.8	arquitetura da rede de Jordan aplicada ao problema de predição não-linear de séries temporais.	71
4.9	rede NARX com d_u entradas e d_y atrasos da saída.	74
4.10	rede NARX com modo paralelo.	76
4.11	rede NARX com modo série-paralelo.	77
5.1	série caótica de Mackey-Glass: (a) informação mútua para o cálculo do atraso de imersão; (b) método de Cao para o cálculo da dimensão de imersão.	82
5.2	(a) série caótica de Mackey-Glass; (c) série caótica do Laser.	82
5.3	série caótica do Laser: (a) informação mútua para o cálculo do atraso de imersão; (b) método de Cao para o cálculo da dimensão de imersão.	83
5.4	(a) série Bellcore; (b) série Tráfego de Vídeo VBR.	84
5.5	(a) preditor sem realimentação; (b) preditor recursivo, com realimentação.	86
5.6	NMSE versus ordem do regressor de saída (d_y) dos modelos NARX: (a) predição UPA; (b) predição KPA.	88
5.7	predição KPA para a série do laser caótico utilizando a rede NARX-SP. . .	90
5.8	predição KPA para a série caótica do Laser. (a) Elman; (b) FTDNN. . . .	91
5.9	NMSE versus horizonte de predição para a rede FTDNN, Elman, NARX-P e NARX-SP.	91
5.10	diagrama de recorrência: (a) série original; (b) NARX-SP; (c) FTDNN; (d) Elman.	92
5.11	NMSE versus d_y : (a) UPA; (b) KPA, $K = 12$	93
5.12	NMSE versus números de épocas de treinamento : (a) UPA; (b) KPA, $K = 12$	94
5.13	predição recursiva da série Bellcore, valores estimados (linha sólida) e valores exatos (linha tracejada): (a) resposta para predição um-passo-adiante; (b) resposta para predição recursiva.	94

5.14	avaliação da sensibilidade da rede neural: (a) dimensão de imersão e (b) número de épocas de treinamento.	95
5.15	informação mútua da série de tráfego de vídeo VBR.	96
5.16	predição recursiva obtidas pelas redes (a) FTDNN e (b) Elman.	97
5.17	predição recursiva obtidas pela rede NARX-SP.	98
A.1	(a) nó estável, (b) foco estável, (c) nó instável, (d) foco instável, (e) ponto de sela e (f) centro.	104

Lista de Tabelas

5.1	Predição k-passos-adiante	87
A.1	Classificação do Estado de Equilíbrio de um Sistema de Segunda Ordem .	103

Lista de Símbolos

f	função para variável contínua
F	função para variável discreta
\mathfrak{R}	conjunto dos números reais
x	variável escalar, i.e., $x \in \mathfrak{R}$
t	índice indicativo de tempo contínuo
n	índice indicativo de tempo discreto
Δt	diferença de tempo
τ	atraso de imersão
d_E	dimensão de imersão
\hat{x}	variável escalar predita
d	dimensão do atrator
ρ	distância entre dois vetores
ϵ^2	erro médio quadrático
σ^2	variância amostral
\mathbf{x}	variável vetorial, i.e., $\mathbf{x} \in \mathfrak{R}^n$
\mathbf{w}_i	vetor de pesos associados ao neurônio i em uma rede neural
\mathbf{m}_k	vetor de pesos associados ao neurônio k em uma rede neural
η	taxa de aprendizagem das redes neurais artificiais
\mathbf{X}	matriz de dados
\mathbf{d}	vetor de saídas desejadas para redes supervisionadas
$\hat{\mathbf{F}}[\cdot]$	mapeamento entre entrada e saída estimado pelas redes supervisionadas
w_{ij}	peso associado à ligação entre entrada j e neurônio i da camada intermediária de uma rede supervisionada
m_{ki}	peso associado à ligação entre neurônio i da camada intermediária e saída k de uma rede supervisionada
ϵ_{med}	erro quadrático médio por época de treinamento da rede supervisionadas
$\phi(\cdot)$	função de ativação de um neurônio de rede supervisionada
θ_i	limiar de ativação de um neurônio i de rede supervisionada
u_i	ativação do neurônio i da camada escondida de uma rede supervisionada
y_i	saída do neurônio i na camada de saída de uma rede supervisionada

δ_k	gradiente local do neurônio k em uma rede neural
δ_i	gradiente local do neurônio i em uma rede neural
p	dimensão do vetor de entrada de uma rede neural
m	dimensão do vetor de saída de uma rede neural
q_1	número de neurônios da primeira camada escondida
q_2	número de neurônios da primeira camada escondida
L	ordem do filtro FIR
s_{ij}	saída da j -ésima sinapse do neurônio i da rede FIR-MLP
M	número de parâmetros ajustáveis numa rede neural
C	unidade de contexto em uma rede neural recorrente
d_u	ordem de memória de entrada de uma modelo NARX
d_y	ordem de memória de saída de uma modelo NARX

Lista de Siglas

RNAs	Redes Neurais Artificiais
RNRs	Redes Neurais Recorrentes
NARX	<i>Nonlinear AutoRegressive model with eXogenous inputs</i>
NARX-P	NARX com Modo Paralelo
NARX-SP	NARX com Modo Série Paralelo
NAR	<i>Nonlinear AutoRegressive</i>
ARMA	<i>Auto-Regressivos Médias Móveis</i>
ARIMA	<i>Auto-Regressivos Integrado Médias Móveis</i>
AR	<i>Auto-Regressivos</i>
MQ	<i>Mínimos Quadrados</i>
MA	<i>Médias Móveis</i>
FAC	Função de Autocorrelação
FCAC	Função de Coeficiente de Autocorrelação
SVD	<i>Singular Value Decomposition</i>
MPE	<i>Mean Prediction Error</i>
MSE	<i>Mean-Squared Error</i>
NMSE	<i>Normalized Mean-Squared Error</i>
MLP	<i>MultiLayer Perceptron</i>
STM	<i>Short-Term Memory</i>
FTDNN	<i>Focused Time Delay Neural Network</i>
FIR-MLP	<i>Finite Impulse Response Multilayer Perceptron</i>
FIR	<i>Filtro de Resposta ao Impulso de Duração Finita</i>
MLE	<i>Maximum Likelihood-Type Estimates</i>
WAN	<i>Wide Area Network</i>
MMPP	<i>Markov-Modulated Poisson Process</i>
LAN	<i>Local Area Network</i>
ISDN	<i>Integrated Services Digital Network</i>
VBR	<i>Variable Bit Rate</i>
WWW	<i>World Wide Web</i>

MPEG	<i>Moving Pictures Experts Group</i>
GoP	<i>Group of Pictures</i>
UPA	Um-Passo-Adiante
KPA	K-Passos-Adiante

Sumário

Resumo	iii
Abstract	iv
Lista de Figuras	vii
Lista de Tabelas	viii
Lista de Símbolos	ix
Lista de Siglas	xi
<hr/>	
1 INTRODUÇÃO	1
1.1 Introdução	1
1.2 Motivação	4
1.3 Objetivos da Dissertação	5
1.4 Produção Científica	5
1.5 Organização Geral do Restante da Dissertação	6
2 FERRAMENTAS PARA ANÁLISE DE SISTEMAS CAÓTICOS	8
2.1 Introdução	8
2.2 Sistemas Dinâmicos Não-Lineares: Uma Breve Introdução	9
2.2.1 Conceitos Básicos	10
2.3 Reconstrução do Espaço de Estados	12

2.3.1	Estimação da Dimensão de Imersão	14
2.3.2	Estimação do Atraso de Imersão	15
2.4	Exemplos de Sistemas Dinâmicos Não-Lineares	16
2.5	Caracterização Elementar de Sistemas Caóticos	21
2.5.1	Limitado em Amplitude	21
2.5.2	Diagramas de Recorrência	21
2.5.3	Determinismo em Séries Temporais	23
2.5.4	Sensibilidade às Condições Iniciais	24
2.6	Caos e Fractais	27
2.7	Conclusão	29
3	MODELAGEM DE TRÁFEGO DE REDES	30
3.1	Introdução	30
3.2	Modelos de Tráfego	31
3.2.1	Processos de Poisson	33
3.3	Cadeias de Markov	35
3.3.1	Modulação via Processos de Markov	36
3.4	Modelos Lineares de Box-Jenkins	37
3.4.1	Modelos Autoregressivos	38
3.4.2	Modelos de Médias Móveis	39
3.4.3	Modelos Autoregressivos e de Médias Móveis	39
3.4.4	Modelos Auto-Regressivos Integrado de Médias Móveis	40
3.5	Processos Auto-Similares	41
3.5.1	Descrição de Processos com Auto-Similaridade	42
3.5.1.1	Definições e Propriedades	43
3.5.1.2	Parâmetro de Hurst	45
3.5.1.3	Estimação do Parâmetro de Hurst	45

3.6	Mapas Caóticos como Fonte de Tráfego	46
3.6.1	Mapa Intermitente Simples	48
3.6.2	Mapa Intermitente Duplo	50
3.7	Conclusão	51
4	REDES NEURAIS SUPERVISIONADAS DINÂMICAS	52
4.1	Introdução	52
4.2	Redes Neurais Estáticas	53
4.2.1	Rede Perceptron Multicamadas	54
4.2.1.1	Algoritmo de Retropropagação do Erro	55
4.3	Redes Neurais Dinâmicas Não-Recorrentes	61
4.3.1	Rede MLP com Atrasadores na Entrada	62
4.3.2	Rede MLP com Neurônios do Tipo FIR	64
4.4	Redes Neurais Dinâmicas Recorrentes	66
4.4.1	Tipos de Conexão de Realimentação	67
4.4.2	Redes Recorrentes Simples	68
4.4.2.1	Rede Recorrente de Elman	69
4.4.2.2	Rede Recorrente de Jordan	70
4.5	Rede Dinâmica NARX	72
4.5.1	Rede NARX para Predição de Séries Temporais	75
4.6	Conclusão	79
5	RESULTADOS	80
5.1	Introdução	80
5.2	Estudo de Caso I - Sistemas Caóticos	81
5.2.1	Série Caótica de Mackey-Glass	81
5.2.2	Série do Laser Caótico	82

5.3	Estudo de Caso II - Séries Temporais de Tráfego de Redes	83
5.3.1	Tráfego de Internet - Série Bellcore	84
5.3.2	Tráfego de Vídeo MPEG VBR	84
5.4	Metodologia de Avaliação	85
5.5	Simulações e Resultados	87
5.5.1	Conclusão	97
6	CONCLUSÕES E PERSPECTIVAS	99
	Apêndice A – Estabilidade de Estados de Equilíbrio	102
	Apêndice B – Expoentes de Lyapunov	105
	Apêndice C – Método de Cao	107
C.1	Definições Preliminares	107
C.2	Cálculo da Dimensão de Imersão pelo Método de Cao	108
	Referências	110

1 INTRODUÇÃO

1.1 Introdução

Redes neurais artificiais (RNAs) têm sido utilizadas com sucesso em problemas de predição e modelagem de série temporais de dinâmica complexa, tais como predição de séries temporais financeiras (DABLEMONT et al., 2003), previsão de vazão de rios (ATIYA et al., 1999), modelagem de séries temporais biomédicas (COYLE et al., 2005) e predição de tráfego de rede (ATIYA et al., 2005; DOULAMIS et al., 2003), para mencionar apenas algumas destas aplicações. Geralmente, modelos de RNA têm melhor desempenho que as técnicas lineares tradicionais, tais como os modelos Box-Jenkins (BOX et al., 1994), quando as séries temporais são ruidosas e não-lineares. Nestes casos, as habilidades de generalização e aproximação universal de funções de RNA justificam seu melhor desempenho preditivo.

Antes do advento da teoria do caos e da geometria fractal, por volta da década de 1960, o comportamento irregular observado em certos sistemas determinísticos não-lineares era tipicamente modelado como estocástico, isto é, tal comportamento era definido com aleatório e imprevisível (KUGIUMTZIS et al., 1994). Em outras palavras, tal comportamento irregular era atribuído a alguma entrada aleatória, externa ao sistema. Segundo uma das premissas da teoria do caos, entradas aleatórias deixaram de ser a única fonte possível de irregularidades em um sistema. Sistemas não-lineares caóticos podem também gerar sinais que se assemelham a sinais estocásticos, mas que foram gerados, contudo, por equações puramente determinísticas. Desta forma, técnicas lineares convencionais têm cedido cada vez mais espaço para técnicas não-lineares que conseguem capturar, com mais eficiência, a dinâmica de sistemas complexos.

Em predição de séries temporais não-lineares (e.g. caóticas), modelos de RNA comumente são usados como preditores de um-passo-adiante, estimando somente o próximo valor de uma série temporal, sem realimentar o valor de saída predito para a entrada da rede. Para predição de horizonte mais amplo, faz-se necessário um procedimento conhecido como predição múltiplos-passos-adiante, em que a saída do modelo deve ser

realimentada para a entrada de forma recursiva até atingir o instante futuro desejado.

Se o horizonte de predição tende ao infinito, em algum momento futuro, a entrada do modelo começa a ser composta somente de valores previamente estimados da série temporal. Neste caso, a tarefa de predição múltiplos-passos-adiante torna-se uma tarefa de *modelagem dinâmica*, em que o modelo de RNA age como um sistema autônomo, tentando recursivamente emular o comportamento dinâmico do sistema que gerou a série temporal não-linear (HAYKIN; PRINCIPE, 1998). A predição múltiplos-passos-adiante e a modelagem dinâmica são mais difíceis de lidar do que a simples predição de um único passo. Estas são tarefas complexas em que os modelos de RNA desempenham um importante papel, em particular, aqueles relacionados à arquiteturas neurais recorrentes (PRINCIPE et al., 2000).

É importante destacar que a minimização do erro de predição um-passo-adiante não implica necessariamente que as predições múltiplos-passos-adiante sejam boas, nem que a RNA seja um modelo preciso da dinâmica do sistema não-linear em análise. Isto é particularmente verdadeiro na predição de séries temporais oriundas de sistemas com sensível dependência às condições iniciais, tais como sistemas caóticos. Muitos trabalhos reconhecem este problema e sugerem que a validação de preditores neurais seja feita por meio da minimização do erro de predição múltiplos-passos-adiante, mesmo que tais preditores tenham sido treinados para minimizar o erro um-passo-adiante (HAYKIN; PRINCIPE, 1998; LILLEKJENDLIE et al., 1994; ABARBANEL et al., 1993).

RNAs recorrentes têm laços de realimentação local e/ou global em sua estrutura. Da mesma forma que redes *feedforward* sem realimentação podem facilmente ser adaptadas para processar séries temporais através de uma entrada com linha de atrasos com derivações (regressor) e treinada pelo algoritmo *backpropagation*, elas podem também ser facilmente convertidas em arquiteturas recorrentes simples por realimentações de ativações dos neurônios das camadas escondidas ou de saída, dando origem às redes de Elman e de Jordan, respectivamente (KOLEN; KREMER, 2001).

Redes neurais recorrentes (RNRs) são capazes de representar mapas dinâmicos não-lineares arbitrários (NARENDRA; PARTHASARATHY, 1990), tal como os comumente encontradas em tarefas de predição de série temporais não-lineares. Young & Chan (1993) e Lendasse et al. (2004) mostram que redes recorrentes têm desempenho melhor do que técnicas lineares tradicionais e redes neurais sem realimentação.

No entanto, aprender a executar tarefas em que as dependências temporais presentes nos sinais de entrada/saída durem longos períodos pode ser bastante difícil usando regras de aprendizagem baseadas em otimização pelo método do gradiente descendente (BENGIO

et al., 1994). Lin et al. (1998) demonstram que aprender dependências de longo prazo com técnicas de gradiente-descendente é mais eficiente numa classe de RNAs conhecida como modelos NARX (*Nonlinear AutoRegressive model with eXogenous inputs*) do que em modelos recorrentes baseados na rede MLP (*MultiLayer Perceptron*), quando aplicado à identificação de sistema de entrada-saída não-lineares. Tem sido mostrado também que as redes NARX, além de apresentarem um bom desempenho para aprender as dependências de longa duração, possuem convergência mais rápida e generalizam melhor do que outras redes recorrentes. Isto ocorre porque o vetor de entrada dos modelos NARX são construídos por meio de uma linha de atrasos com derivação deslizada sobre o sinal de entrada, junto com uma linha de atrasos com derivação formada pelas realimentações do sinal de saída da rede (LIN et al., 1996).

Apesar das vantagens previamente mencionadas da rede NARX como uma ferramenta de modelagem de sistemas entrada-saída dinâmicos, percebe-se que sua aplicação em predição de séries temporais não explora todo o poder computacional da rede NARX. Neste tipo de aplicação, a linha de atrasos sobre o sinal de saída é eliminada, reduzindo a rede NARX a um modelo autoregressivo não-linear (NAR, *Nonlinear Autoregressive*). Tomando como ponto de partida este uso limitado da rede NARX, propõem-se nesta dissertação estratégias simples que permitam que a arquitetura recorrente da rede NARX seja plenamente explorada em tarefas de predição e modelagem dinâmica de série temporais não-lineares.

Nesta dissertação, para avaliação dos modelos NARX na tarefa de predição e modelagem de séries temporais, utiliza-se primeiramente duas séries caóticas: a série caótica de Mackey-Glass e a série caótica do Laser. Em seguida, os modelos NARX são avaliados na tarefa de predição de séries temporais de tráfego de redes de computadores. O tráfego atual de redes de alta velocidade possui características fractais, irregulares e não-estacionárias, sendo de difícil tratamento por meio de modelos lineares ou modelos que não levem em conta as dependências de longo prazo. Desta forma, existe a necessidade de construir modelos que melhor se apliquem a este tipo de tráfego.

Pode-se citar alguns dos principais trabalhos dentro desta linha de pesquisa. O trabalho de Luz (2003) mostra um método de previsão aplicado a gerência pró-ativa de redes de computadores através da modelagem pelo método linear de Box e Jenkins. A pesquisa de Silva et al. (2001) procura abordar a modelagem estocástica do tráfego de redes de comunicações. Silva (2004) propõe e implementa um mecanismo para prever possíveis congestionamentos, através da identificação da componente de tendência em séries temporais

que representam o tráfego de redes, utilizando a transformada de *wavelet* discreta. De particular interesse para esta dissertação estão os trabalhos de Yousefi'zadeh (1997, 2002) e Yousefi'zadeh & Jonckheere (2005), em que o autor investiga a aplicação de mapas caóticos determinísticos e redes neurais *feedforward* para modelar os padrões de tráfego auto-similar agregado e de origem em pacotes das redes de computadores.

Por fim, pode-se citar os trabalhos em que são feitas modelagem e predição de dados de tráfego de vídeo VBR (*Variable Bit Rate*) MPEG. Primeiramente destaca-se Doulamis et al. (2003), que apresentam um esquema de redes neurais *feedforward* para modelar fontes de vídeo VBR MPEG-2. Bhattacharya et al. (2003) fornecem uma aproximação para o desenvolvimento de um preditor do tráfego de vídeo em tempo real com codificação MPEG para o uso em horizontes de predição um-passo-adiante e vários passos-adiante, cujo preditor projetado consiste de redes neurais recorrentes e redes neurais *feedforward*. Liang (2004) investiga a predição do tráfego de vídeo VBR com dependências de logo alcance e de tempo real, utilizando para isso preditor de tráfego baseado em redes neurais *feedforward*.

1.2 Motivação

As características e dificuldades na predição e modelagem de séries temporais expostas na seção anterior e a revisão bibliográfica levantada nesta dissertação indicam certos aspectos que necessitam de maior atenção, a saber:

- dificuldade dos modelos lineares em extrair o comportamento complexo e irregular de algumas séries temporais, pois equações lineares somente conduzem a soluções periódicas ou exponencialmente decrescentes;
- entendimento das características e invariâncias dos sistemas não-lineares caóticos, já que estes sistemas descrevem características complexas presentes em muitas séries temporais reais;
- existe uma grande dificuldade entre as técnicas de predição e de modelagem de tráfego em redes de comunicações ao lidar com dependências temporais longas, necessitando-se assim de modelos específicos que tratem deste problema;
- RNAs vêm se destacando na modelagem e na previsão de séries temporais, com ênfase para redes dinâmicas, pois estas redes podem capturar informações temporais úteis de sistemas não-lineares melhores do que os modelos tradicionais;

- modelos NARX pertencem a uma classe de redes neurais recorrentes que tem desempenho melhor que redes do tipo MLP quando empregadas em problemas com dependências temporais longas e tem as mesmas facilidades de construção das redes MLP.

Os tópicos supracitados serviram de motivação para o desenvolvimento do presente trabalho que, por sua vez, busca propor estratégias para o melhor modelamento e predição de séries temporais com dependências temporais longas. Objetivos específicos deste trabalho estão detalhados a seguir.

1.3 Objetivos da Dissertação

Tendo em vista os tópicos mencionados nas seções anteriores, concernentes à modelagem e predição de séries temporais e às motivações deste trabalho, os principais objetivos desta dissertação são listados abaixo:

- estudo de ferramentas oriundas da teoria de sistemas dinâmicos não-lineares, caóticos e fractais para aplicação em diversos tipos de séries temporais;
- aplicação de arquiteturas de redes neurais artificiais (RNAs) dinâmicas, recorrentes e não-recorrentes, na modelagem e predição não-linear de séries temporais;
- comparação de desempenho de várias arquiteturas de redes neurais dinâmicas, enfatizando suas vantagens e limitações em relação aos modelos clássicos de análise de séries temporais;
- propor uma abordagem para utilização da rede NARX em problemas de séries temporais com dependência temporal longa.

1.4 Produção Científica

Ao longo do desenvolvimento desta dissertação os seguintes artigos científicos foram publicados:

- **José M. Menezes Jr.** & Guilherme A. Barreto (2006), “On the Prediction of Chaotic Time Series Using the NARX Recurrent Neural Network: A New Approach”, aceito para publicação na *9th Experimental Chaos Conference (ECC'2006)*, 29/05/2006 - 02/06/2006, São José dos Campos-SP.

- **José M. Menezes Jr.** & Guilherme A. Barreto (2006), “On Recurrent Neural Networks for Self-Similar Traffic Prediction: A Performance Evaluation”, aceito para *6th International Telecommunications Symposium (ITS’2006)*, Fortaleza-CE.
- **José M. Menezes Jr.** & Guilherme A. Barreto (2006), “A New Look at Nonlinear Time Series Prediction with NARX Recurrent Neural Network”, aceito para *IX Brazilian Neural Networks Symposium (SBRN’2006)*, Ribeirão Preto-SP.

1.5 Organização Geral do Restante da Dissertação

O restante desta dissertação está organizado em cinco capítulos. Um breve comentário sobre cada um deles é feito a seguir.

No Capítulo 2 é feito um estudo dos sistemas caóticos, apresentando as definições dos principais termos da área. Também são discutidas diversas técnicas para a caracterização destes sistemas complexos, mostrando as principais diferenças em relação aos sistemas estocásticos.

O Capítulo 3 apresenta uma breve descrição das principais técnicas de modelagem de tráfego de rede, desde as mais convencionais, em que são feitas poucas suposições sobre a natureza tráfego, passando pela modelagem estocástica mais refinada em que se teve uma maior atenção na dependência temporal do tráfego, e por fim é apresentada a abordagem moderna para tráfego de redes, que tem como principais características a auto-similaridade, a dependência de longa duração e as propriedades fractais.

No Capítulo 4 é feita uma introdução às arquiteturas de redes neurais aplicadas a problemas de predição e modelagem de séries temporais. Em particular é dada ênfase para as redes neurais dinâmicas e recorrentes, que pela própria forma como são construídas, têm uma melhor capacidade de tratar elementos temporais. Desta forma, este capítulo apresenta as principais características das RNAs para a aplicação em predição e modelagem de séries temporais.

O Capítulo 5 apresenta resultados obtidos a partir da aplicação das técnicas de análise discutidas no Capítulo 2 e das arquiteturas de redes neurais dinâmicas apresentadas no Capítulo 4 na predição e modelagem de séries temporais. Para a extração de tais resultados, são utilizadas séries temporais caóticas artificiais, séries temporais caóticas reais e séries de tráfego de rede de computadores. Em particular, este capítulo tem o interesse em observar o desempenho das RNAs estudadas e da abordagem proposta no trato de

séries que apresentem dependência temporal longa.

No Capítulo 6 são apresentadas as principais conclusões da dissertação, sendo apontadas também as principais perspectivas desta dissertação.

2 FERRAMENTAS PARA ANÁLISE DE SISTEMAS CAÓTICOS

2.1 Introdução

O estudo sistemático de fenômenos não-lineares, em particular os caóticos, tem sua origem por volta da década de 1960. Uma possível razão para este interesse tardio reside no fato do cenário de análise de sistemas ser dominado por técnicas lineares, seja na Matemática Aplicada ou na Engenharia. Além disso, a maioria das ferramentas de análise de sistemas caóticos dependem do intenso uso dos computadores digitais, que por sua vez se tornaram do uso mais difundido somente a partir do final da década de 70.

Em consequência do baixo poder computacional dos primeiros computadores, o comportamento irregular de certos sistemas determinísticos não-lineares não era avaliado em sua totalidade e quando tal comportamento era manifestado em observações, era explicado tipicamente como estocástico (KUGIUMTZIS et al., 1994). Isto é, todo comportamento irregular de um sistema atribuía-se à alguma entrada externa aleatória ao sistema. Mais recentemente, a teoria do caos advoga que entradas aleatórias não são as únicas fontes possíveis de irregularidade na saída de um sistema. Desta forma, sistemas dinâmicos não-lineares de pequena ordem podem produzir sinais muito irregulares, a partir de equações não-lineares puramente determinísticas (KANTZ; SCHREIBER, 1997).

Um exemplo de um sistema não-linear bastante simples, com apenas um parâmetro, conhecido como mapa logístico, produz uma série temporal, cuja função de autocorrelação se assemelha a de uma seqüência de ruído branco (KUGIUMTZIS et al., 1994), quando na verdade corresponde a uma série temporal caótica. Devido a esta curiosa, porém falsa, semelhança com processos estocásticos lineares, muitas séries temporais caóticas costumam ser tratadas a partir de modelos lineares convencionais, tal como o modelo autoregressivo com médias móveis (*autoregressive moving average*, ARMA). Contudo, tais modelos têm se mostrado inadequados para a análise e predição de sistemas caóticos, pois

não capturam a dinâmica não-linear subjacente à série temporal de interesse. Posto de maneira mais formal, isto se deve ao fato de que modelos lineares conduzem somente a soluções exponencialmente decrescentes ou periodicamente oscilantes, chamadas genericamente de pontos ou soluções de equilíbrio. Sistemas caóticos apresentam outras possíveis soluções ou comportamentos que só são obtidos quando se usa as ferramentas e modelos não-lineares adequados.

Algumas ferramentas de análise de sistemas caóticos são descritas neste capítulo, enquanto que modelos não-lineares baseados em redes neurais artificiais são descritos no Capítulo 4. Os principais artigos que serviram de referência para o estudo das ferramentas descritas neste capítulo são Abarbanel et al. (1993), Kugiumtzis et al. (1994), Lillekjendlie et al. (1994), Schreiber (1999). Dentre os livros consultados destacam-se os de Monteiro (2006), Kantz & Schreiber (1997), Glass & Mackey (1997) e Kaplan & Glass (1995).

2.2 Sistemas Dinâmicos Não-Lineares: Uma Breve Introdução

O estudo de sistemas dinâmicos caóticos pode ser dividido em três áreas fundamentais: (i) identificação do comportamento caótico, (ii) modelagem e previsão da dinâmica de sistemas caóticos, e por fim, (iii) controle de sistemas caóticos (KUGIUMTZIS et al., 1994).

A primeira área tem como principal objetivo classificar um certo sistema com comportamento (dinâmica) irregular como sendo um sistema caótico ou como um sistema estocástico; ao mesmo tempo, fornece estimativas de graus de liberdade e de complexidade do sistema caótico (KUGIUMTZIS et al., 1994). A segunda área se divide ainda em duas sub-áreas: reconstrução da dinâmica do sistema caótico, a partir de observações, e caracterização da dinâmica reconstruída¹. A terceira relaciona-se com a capacidade de obter uma resposta específica para um certo sistema dinâmico, tornando-o sensível, porém estável, a sinais de entrada semelhantes aqueles observados quando o sistema está com comportamento caótico. O objetivo é fazer com que o sistema possa ser levado a produzir a resposta desejada sem muito esforço.

Esta dissertação está concentrada nas duas primeiras áreas mencionadas no parágrafo anterior. Em particular, este trabalho almeja utilizar modelos não-lineares baseados em redes neurais artificiais para identificar e reconstruir a dinâmica de sistemas caóticos. A

¹Por caracterização entende-se o ato de calcular certas grandezas invariantes que identificam o processo como caótico.

reconstrução dinâmica por sua vez é avaliada por várias ferramentas, principalmente as que se baseiam na análise de sinais gerados por sistemas dinâmicos caóticos. Tais sinais, por serem observados (amostrados) em intervalos de tempo discreto, são comumente chamados de séries temporais caóticas.

Uma série temporal é representada de forma genérica como uma seqüência finita de valores de uma certa variável $x \in \mathbb{R}$, $\{x(1), x(2), \dots, x(N)\}$ ou $\{x(n)\}_{n=1}^N$, em que N representa a quantidade de amostras observadas. Uma série temporal caótica pode ser entendida grosseiramente como a parte mensurável da saída de um sistema dinâmico não-linear, cujo comportamento caótico se deseja compreender. Esta série temporal é, assim, uma das poucas (se não a única!) fonte de informação disponível sobre o sistema dinâmico de interesse e é a partir dela que o comportamento caótico do sistema deve ser inferido. Assim, quando se fala em analisar uma série temporal caótica se está, na verdade, tentando entender como o comportamento irregular de tal série temporal reflete a dinâmica do sistema caótico que lhe dá origem.

Ferramentas de análise de séries temporais discutidos nesta dissertação são oriundas, em sua maioria, da teoria de sistemas dinâmicos (KAPLAN; GLASS, 1995; KANTZ; SCHREIBER, 1997), sendo por isto necessário apresentar certos conceitos básicos desta teoria antes de iniciar a análise de séries temporais propriamente dita.

2.2.1 Conceitos Básicos

O espaço de estados ou espaço de fase de um sistema dinâmico é definido como o espaço formado pelas variáveis dependentes x_i , $i = 1, \dots, m$, associadas a um dado sistema dinâmico. Desta forma, um ponto no espaço de estados, corresponde ao vetor de variáveis estados, $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_m]^T$, $\mathbf{x} \in \mathfrak{R}^m$. De modo geral, o espaço de estados forma um conjunto aberto no \mathfrak{R}^m . Todavia, em certos casos, a topologia do espaço pode estar restrita a uma superfície geométrica de forma particular, tais como cilíndrica ou toroidal. Topologicamente, diz-se que este espaço é uma variedade (*manifold*) (SAVI, 2004).

Pode-se descrever a evolução temporal da variável \mathbf{x} no espaço de estados ou por um mapa² m -dimensional ou por um sistema de m equações diferenciais ordinárias de primeira ordem. Tanto mapas discretos, quanto sistemas de equações diferenciais servem para descrever matematicamente como o vetor de estado varia com o passar do tempo, ou

²Em português, o termo *map* costuma ser também traduzido como *mapeamento* ou *aplicação*. O primeiro termo é usado nas Engenharias de Controle e de Telecomunicações, enquanto o último termo é muito usado por Matemáticos.

seja, qual é a dinâmica do sistema. Para simplificar, considera-se que o espaço de estados é um espaço vetorial de dimensão finita \mathfrak{R}^m .

No caso da dinâmica do sistema ser descrita por mapas, o tempo é uma variável discreta, sendo denotada por n . Assim, a dinâmica é descrita como

$$\mathbf{x}(n+1) = \mathbf{F}(\mathbf{x}(n)), \quad (2.1)$$

em que $\mathbf{F}(\cdot)$ é uma função não-linear de seu argumento. Esta equação relaciona matematicamente o estado futuro do sistema com o estado atual. Equações deste tipo, que relacionam grandezas em instantes de tempo discreto, são chamadas genericamente de equações a diferenças-finitas.

Caso a dinâmica seja representada por equações diferenciais, tem-se que a variável tempo é uma grandeza contínua, denotada por t . Neste caso, a dinâmica da variável de estado x_i é descrita por

$$\frac{d}{dt}x_i(t) = f_i(x_i(t)), \quad i = 1, 2, \dots, m. \quad (2.2)$$

Pode-se escrever este sistema de equações em uma forma compacta utilizando-se notação vetorial, dado por

$$\frac{d}{dt}\mathbf{x}(t) = \mathbf{f}(\mathbf{x}(t)), \quad (2.3)$$

em que $\mathbf{f}(\cdot)$ também é uma função não-linear de seu argumento. Pode-se dizer que $\mathbf{x}(t)$ é um caminho no espaço de estados percorrido com velocidade $\frac{d}{dt}\mathbf{x}(t)$, que coincide, em cada ponto, com o campo de velocidades $\mathbf{f}(\mathbf{x}(t))$. Esta forma de representação é usualmente referida como fluxo. Se $\mathbf{f}(\cdot)$ é explicitamente dependente de t , ou seja, $d\mathbf{x}(t)/dt = \mathbf{f}(t, \mathbf{x}(t))$, pode-se chamar o sistema dinâmico de não-autônomo; caso contrário, ele é autônomo, como na Equação (2.3) (MONTEIRO, 2006).

Uma seqüência de pontos $\mathbf{x}(n)$ ou $\mathbf{x}(t)$ obtidos a partir da solução das Equações (2.1) e (2.3) é chamada trajetória de um sistema dinâmico, sendo $\mathbf{x}(0)$ sua condição inicial. Uma trajetória pode evoluir rumo ao infinito com o tempo, sendo chamada por isto de solução instável, ou permanece restrita a uma área (subespaço) para sempre. Se a trajetória convergir para um único estado (ponto) no espaço de estados, tal que $\mathbf{x}^* = \mathbf{F}(\mathbf{x}^*)$, o ponto \mathbf{x}^* é chamado de ponto de equilíbrio. Outra possível solução de equilíbrio, muito comum em sistemas dinâmicos, é conhecida como ciclo-limite, em que, em vez de um único ponto de equilíbrio, tem-se uma trajetória (conjunto de pontos) que se repete periodicamente. Pontos de equilíbrio e ciclos-limites são chamados genericamente de atratores. Um resumo dos possíveis soluções de equilíbrio para sistemas não-lineares é

apresentado no Apêndice A.

A expressão matemática de $\mathbf{F}(\cdot)$ ou $\mathbf{f}(\cdot)$, seus parâmetros associados e as condições iniciais são os fatores que decidem qual é o comportamento assintótico resultante para uma certa trajetória. Um conjunto de condições iniciais que conduzem ao mesmo atrator define a bacia de atração daquele atrator (KUGIUMTZIS et al., 1994). Sistemas dinâmicos que apresentam comportamento caótico não possuem pontos de equilíbrio ou ciclos-limites, muito embora as trajetórias sempre convirjam para uma “região limitada” do espaço de estados, independente da condição inicial, de tal forma que os pontos da trajetórias nunca se repetem. Dá-se o nome de atrator estranho à trajetória desenhada no espaço de estados por um sistema dinâmico caótico.

A formulação de tempo discreto é mais conveniente para processamento em computadores digitais, o que resulta na geração de várias séries temporais, uma para cada variável x_i , $i = 1, \dots, m$. Um sistema dinâmico de tempo contínuo, descrito como na Equação (2.3), pode ser facilmente discretizado e transformado em um mapa discreto. Por exemplo, usando a equação de Euler para aproximação da derivada de primeira ordem chega-se ao seguinte resultado:

$$\frac{d\mathbf{x}(t)}{dt} \approx \frac{\mathbf{x}(t + \Delta t) - \mathbf{x}(t)}{\Delta t} \Rightarrow \mathbf{x}(t + \Delta t) \approx \mathbf{x}(t) + \Delta t \cdot \mathbf{f}(\mathbf{x}(t)), \quad (2.4)$$

em $0 < \Delta t \leq 1$ é chamado de passo de amostragem e define o grau de discretização da equação. De modo geral, quanto menor for Δt , menor é a diferença entre valores consecutivos $\mathbf{x}(t)$ e $\mathbf{x}(t + \Delta t)$ e melhor é a aproximação do fluxo pelo mapa discreto equivalente.

2.3 Reconstrução do Espaço de Estados

A idéia básica da reconstrução do espaço de estado está calcada no fato de que a série temporal de uma certa variável de estado x_i contém informações sobre as outras variáveis de estado não-observáveis, podendo ser usadas para prever o vetor de estado atual $\mathbf{x}(n)$. Ao processo de previsão do vetor de estados, a partir de uma única série temporal, dá-se o nome de reconstrução do espaço de estados (KAPLAN; GLASS, 1995; KANTZ; SCHREIBER, 1997; SCHREIBER, 1999).

A reconstrução do espaço de estado está baseada no Teorema da Imersão de Takens (*Takens' embedding theorem*) (TAKENS, 1981). Este teorema permite reconstruir um espaço de estado d_E -dimensional similar ao espaço de estado original, a partir de uma única

variável de estado, que é a variável medida. Este espaço reconstruído deve preservar as propriedades invariantes do sistema dinâmico subjacente (SAVI, 2004).

De modo geral, o teorema de Takens é posto da seguinte maneira. Seja uma série temporal de tamanho N (suficientemente grande) e livre de ruído, $\{x(1), x(2), \dots, x(N)\}$, obtida a partir de uma das variáveis de um sistema dinâmico determinístico. O espaço de estados deste sistema pode ser exatamente reconstruído por um grupo de vetores, chamados coordenadas de atraso, montados a partir de amostras atrasadas daquela série temporal da seguinte forma

$$\mathbf{x}(n) = [x(n) \quad x(n - \tau) \quad x(n - 2\tau) \quad \dots \quad x(n - (d_E - 1)\tau)]^T, \quad (2.5)$$

em que $x(n)$ é a amostra da série temporal no tempo n , d_E é chamada de dimensão de imersão (*embedding dimension*) e τ é chamado de atraso de imersão (*embedding delay*). Uma idéia semelhante ao teorema de Takens é proposta originalmente no trabalho de Whitney (1936), tal que costuma-se referir a ela também como “Teorema de Whitney” porque ele é o primeiro a provar que uma variedade suave (*smooth manifold*) de dimensão n pode ser imersa em \mathfrak{R}^{2n+1} .

O teorema de Takens é um importante teorema porque implica na seguinte constatação: se as suposições gerais do teorema são satisfeitas, existe uma função $g(\cdot)$, tal que, $x(n + 1) = g(\mathbf{x}(n))$. Ou seja, se as coordenadas de atraso $\mathbf{x}(n)$, montadas como na Equação (2.5), reconstroem com exatidão o espaço de estados, então existe uma função $g(\cdot)$ que gera a variável de estado $x(n + 1)$ com exatidão. Contudo, como esta função é geralmente desconhecida, o problema de reconstrução do espaço de estados pode intuitivamente ser colocado como um problema de predição de séries temporais, no qual o objetivo é determinar os valores futuros da variável observada, ou seja,

$$\hat{x}(n + 1) = \hat{g}(\mathbf{x}(n)), \quad (2.6)$$

em que $\hat{x}(n+1)$ é uma estimativa do valor exato de $x(n+1)$ e $\hat{g}(\cdot)$ denota uma aproximação da função $g(\cdot)$. Assim, conclui-se que um bom modelo computacional para a aproximação $\hat{g}(\cdot)$, resulta em uma reconstrução fidedigna do espaço de estados, pois os valores preditos para $\hat{x}(n + 1)$ são próximos dos valores exatos.

2.3.1 Estimação da Dimensão de Imersão

A dimensão de imersão d_E do espaço de estados reconstruído é um importante parâmetro a ser determinado. Geralmente ela é diferente da dimensão exata (e desconhecida) do espaço de estados, $m = [d] + 1$, em que $[d]$ significa a parte inteira da dimensão fractal do atrator d . Takens (1981) mostra ser suficiente que $d_E \geq 2[d] + 1$. O teorema garante que o atrator imerso no espaço de estado d_E -dimensional é desdobrado (*unfolded*) sem qualquer auto-interseções. A condição $d_E \geq 2[d] + 1$ é suficiente mas não é necessária, e um atrator pode ser reconstruído também na prática, com uma dimensão de imersão tão baixa quanto $[d] + 1$ (KUGIUMTZIS et al., 1994). Nos próximos parágrafos são descritos métodos para estimar a dimensão de imersão d_E , a partir de uma série temporal com ou sem ruído.

Cálculo de invariantes geométricos. Este método baseia-se na tentativa de encontrar um valor assintótico de alguma invariante geométrica (e.g. dimensão de correlação) do sistema dinâmico em função do valor da dimensão de imersão. Assim, quando o invariante geométrico calculado estabilizar em um determinado valor, o valor escolhido para a dimensão de imersão é o menor valor para o qual aquele invariante estabiliza.

Decomposição em valores singulares. Este método é baseado na diagonalização da matriz de covariância dos vetores de reconstrução, identificando os seus autovalores. O número de autovalores não-nulos é um valor estimado da dimensão mínima de imersão.

Método dos falsos vizinhos (*False Neighbors*). Este método baseia-se no fato de que em um atrator bem reconstruído não deve haver cruzamento de uma trajetória consigo mesma; ou seja, pontos não devem se repetir, uma vez que a dinâmica é caótica. Assim, avalia-se um vizinho como “verdadeiro” ou “falso” apenas em virtude da projeção do sistema em uma determinada dimensão. Desta forma, um falso vizinho é um ponto do sinal que só corresponde a um vizinho devido a observação das órbitas em um espaço muito pequeno, $D < d_E$. Quando o espaço está imerso em uma dimensão $D > d_E$, todos os pontos vizinhos de todas as órbitas são vizinhos verdadeiros.

Método de Cao (1997). Este método é uma extensão da técnica anterior, sendo voltada para aplicações em séries temporais estocásticas ou determinísticas. Este método também é pouco sensível ao tamanho da série em questão. O procedimento consiste em explorar a estrutura geométrica do atrator à medida que se aumenta o valor de d_E , a partir de 1. Se d_E é muito pequeno, o atrator apresenta auto-interseções da trajetória do atrator no espaço de estados. Nestes casos, pontos próximos no atrator são, ou vizi-

nhos exatos devido à dinâmica do sistema, ou falsos vizinhos devido às auto-intersecções. Em dimensões maiores, em que as auto-intersecções são desfeitas, os falsos vizinhos são revelados visto que eles vão se distanciando. O objetivo do método de Cao é encontrar um limiar mínimo para d_E , tal que não existam falsos vizinhos no atrator reconstruído a partir desta dimensão de imersão.

Nesta dissertação adota-se o método de Cao, pois, o mesmo leva a resultados melhores no processo de predição não-linear associado. Devido a sua importância para esta dissertação, o método de Cao está descrito em maiores detalhes no Apêndice C.

2.3.2 Estimação do Atraso de Imersão

Embora Takens (1981) não tenha considerado este parâmetro relevante na sua formulação original, em séries temporais reais, que não estão livres de ruído (muito pelo contrário!), ele se torna em um parâmetro da maior importância. Para τ demasiado pequeno, coordenadas de atraso $\mathbf{x}(n)$ consecutivas tornam-se similares, de tal forma que o atrator reconstruído é esticado ao longo de uma diagonal e obscurecido facilmente pelo ruído. Assim, é desejável uma escolha de τ que mantenha coordenadas de atrasos consecutivas mais independentes entre si. Por outro lado, valores demasiado grandes causam perda de informação contida nos dados, tal que dois vetores, temporalmente próximos, tornam-se bastante afastados, dando origem a incertezas na reconstrução (KUGIUMTZIS et al., 1994).

Uma das principais ferramentas para a estimação de independência entre termos é a função de autocorrelação temporal (FAC), cuja expressão, para um sinal de média zero, é dada por

$$R_X(k) = \frac{\sum_{n=1}^{N-k} x(n)x(n+k)}{N-k}, \quad (2.7)$$

em que o parâmetro $k \geq 0$ é separação temporal (*lag*) entre as amostras. A FAC é uma medida quantitativa da dependência temporal entre amostras sucessivas de uma série temporal, propriedade esta associada com a presença de “memória” no sistema. Uma série temporal, em que $R_X(k) \neq 0$ para $k = 0$, e $R_X(k) \approx 0$ para $k > 0$, é típica de sistemas sem memória, de modo que tal sequência é chamada genericamente de ruído branco.

Uma formulação alternativa da FAC, chamada de função coeficiente de autocorrelação (FCAC), divide a Equação (2.7) pela variância amostral $\sigma_X^2 = R_X(0)$ da série, resultando

na seguinte expressão

$$\rho_X(k) = \frac{R_X(k)}{R_X(0)} \approx \frac{\sum_{n=1}^{N-k} x(n)x(n+k)}{\sum_{n=1}^N x^2(n)}, \quad (2.8)$$

tal que, neste caso, o maior valor de $\rho_X(k)$ é 1, obtido para $k = 0$.

Uma escolha comum para τ é o atraso (*lag*) para o qual a FAC atinge seu primeiro valor nulo. Por este método, as coordenadas de atraso passam a ser linearmente não-correlacionadas. Outra regra semelhante consiste em escolher o atraso de imersão como o *lag* no qual a FAC decai para $1/e = 0,37$ (KANTZ; SCHREIBER, 1997). Williams (1997) sugere outro método para a escolha do atraso de imersão mínimo, como sendo o *lag* seguinte ao ponto em que a FAC pára de diminuir; ou seja, no primeiro mínimo da FAC.

Uma objeção aos procedimentos mencionados anteriormente é que a estimação do atraso de imersão através da FAC é baseada em estatísticas lineares, não levando em conta correlações não-lineares (KANTZ; SCHREIBER, 1997). Fraser & Swinney (1986) sugere uma escolha para τ mais adequada ao problema de modelagem de sistemas dinâmicos, baseado em um critério de medida de independência mais geral, tal como como a informação ganha em bits sobre $x(n + \tau)$ dada a medida de $x(n)$. Em suma, esta medida é conhecida como informação mútua e o primeiro mínimo no gráfico desta grandeza, em função de τ , é freqüentemente sugerida como uma boa estimativa para τ (KUGIUMTZIS et al., 1994). Nesta dissertação este é o critério adotado para determinar o atraso de imersão.

A expressão para o cálculo da informação mútua é baseada na entropia de Shannon. Dentro de um intervalo de dados de uma série temporal, é criado um histograma para a distribuição de probabilidade dos dados. Denota-se por p_i a probabilidade que o sinal assuma um valor dentro do i th caixa (*bin*) do histograma e assumi-se p_{ij} ser a probabilidade que $x(n)$ esteja na caixa i e $x(n + \tau)$ esteja na caixa j . Então a informação mútua para um atraso no tempo τ é definido como,

$$I(\tau) = \sum_{i,j} p_{ij}(\tau) \ln p_{ij}(\tau) - 2 \sum_i p_i \ln p_i. \quad (2.9)$$

2.4 Exemplos de Sistemas Dinâmicos Não-Lineares

Um dos mais simples e conhecidos mapas dinâmicos não-lineares que podem apresentar comportamento caótico é chamado de mapa logístico ou mapa quadrático, sendo

descrito pela seguinte equação

$$x(n+1) = ax(n)[1 - x(n)], \quad (2.10)$$

em que o parâmetro $a > 0$ é uma constante a ser escolhida em função do comportamento desejado. Para $1 \leq a \leq 4$, a trajetória da variável de estado x produz valores restritos ao intervalo $[0, 1]$ para condições iniciais no mesmo intervalo (KUGIUMTZIS et al., 1994).

Para valores de a entre 0 e 3, na Equação (2.10), o estado assintótico de $\{x(n)\}$ consiste em apenas um ponto de equilíbrio. Para $3 < a < 3,57$, a solução assintótica consiste de ciclos-limites de diferentes periodicidades. Para valores de $3,57 < a \leq 4$, o sistema passa a apresentar comportamento caótico (KAPLAN; GLASS, 1995). Na Figura 2.1(a) é mostrada uma realização do mapa logístico para $a = 4$, em que pontos sucessivos são ligados por linhas retas para facilitar a visualização.

Apenas a visualização da série não é suficiente para caracterizá-la como estocástica ou caótica, fazendo-se necessária a utilização de medidas auxiliares. Uma metrica útil é a função de autocorrelação (FAC), que no caso de uma série estocástica, um ruído branco, é não-nula somente quando o distanciamento (atraso) entre as amplitudes é zero ($k = 0$). Para a seqüência gerada pelo mapa logístico, mostrada na Figura 2.1(b), é também não-nula somente quando o distanciamento entre as amplitudes é zero, de tal forma que a seqüência produzida pode ser facilmente confundida com ruído branco. Esta característica permitiu que o mapa logístico fosse usado por muito tempo como um gerador de números aleatórios em computadores (KUGIUMTZIS et al., 1994).

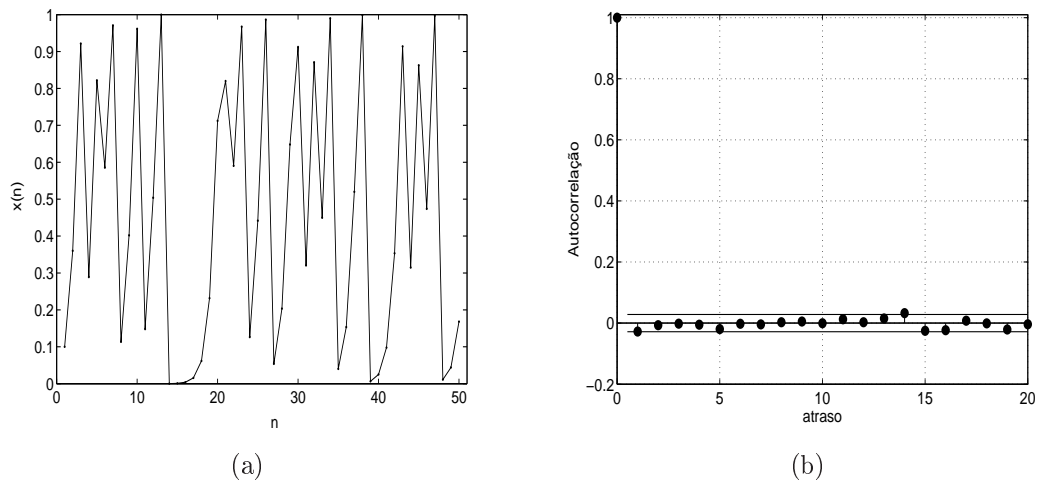


Figura 2.1: (a) mapa logístico para estado caótico; (b) autocorrelação para o mapa logístico com $a = 4$.

Como segundo exemplo de sistema caótico tem-se o mapa de Hénon

$$\begin{aligned} s_1(n+1) &= s_2(n) + 1 - as_1(n)^2, \\ s_2(n+1) &= bs_1(n). \end{aligned} \quad (2.11)$$

Com o valor de $b = 0$, este mapa se reduz ao sistema caótico apresentado anteriormente, o mapa logístico. Para comportamento caótico, este sistema possui uma pequena faixa de valores para a e b , sendo que os valores mais usuais para produzir um sistema caótico são $a = 1,4$ e $b = 0,3$ (KANTZ; SCHREIBER, 1997).

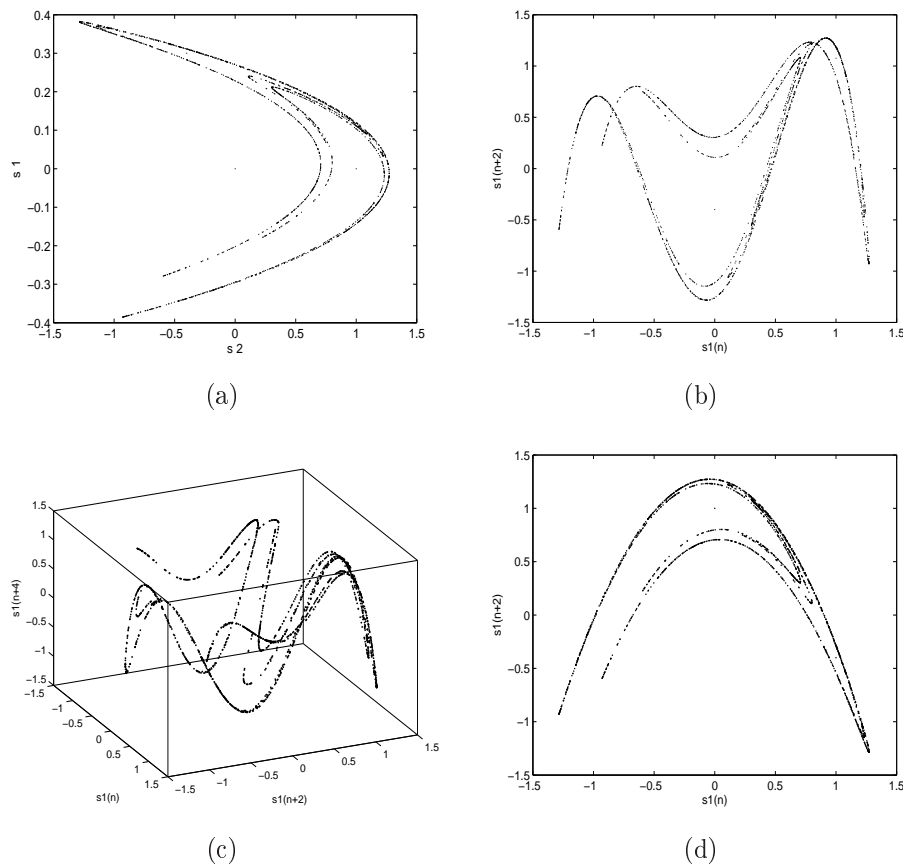


Figura 2.2: reconstrução do atrator de Hénon. (a) atrator original; (b) atrator reconstruído para $\tau = 2$ e $n_d = 2$; (c) $\tau = 2$ e $n_d = 3$; (d) $\tau = 1$ e $n_d = 2$.

Na Figura 2.2(a) é mostrado o atrator original do mapa com os valores discutidos anteriormente. Na Figura 2.2(b) é reconstruído o atrator da medida s_1 , usando uma dimensão de imersão $d_E = 2$ e um atraso de imersão $\tau = 2$, resultando em coordenadas de atraso $\mathbf{x}(n) = [x(n) \quad x(n+2)]^T$. Observa-se aqui que existem intercessões na reconstrução do atrator e que desaparecem quando se aumenta o valor da dimensão de imersão para 3, Figura 2.2(c). Também é interessante demonstrar que para uma escolha de coordenadas tal como $\mathbf{x}(n) = [x(n) \quad x(n+1)]^T$, isto é, $d_E = 2$ e $\tau = 1$, o atrator também é reconstruído

sem intercessões, como mostrado na Figura 2.2(d).

A terceira série temporal caótica apresentada neste trabalho é oriunda da variável $x(t)$ do sistema de equações de Lorenz (1963)

$$\begin{aligned}\frac{dx(t)}{dt} &= a(y(t) - x(t)), \\ \frac{dy(t)}{dt} &= bx(t) - y(t) - x(t)z(t), \\ \frac{dz(t)}{dt} &= y(t)x(t) - cz(t),\end{aligned}\tag{2.12}$$

em que a , b e c são constantes reais. Este sistema de equações é considerado como o primeiro a revelar a presença de caos em sistemas dinâmicos dissipativos³. Para a série usada nesta dissertação utiliza-se $a = 10$, $b = 28$ e $c = 8/3$. A série temporal caótica de Lorenz é gerada a partir da discretização do sistema de equações (2.12), usando a equação de Euler (Seção 2.2). É adotado $\Delta t = 0,01$ e as primeiras amostras geradas são descartadas por causa do efeito transitório. Na Figura 2.3, cuja forma lembra as asas de uma borboleta, observa-se a evolução das coordenadas $x(t)$, $y(t)$ e $z(t)$ num gráfico tridimensional, mostrando a dinâmica do sistema para os parâmetros citados acima.

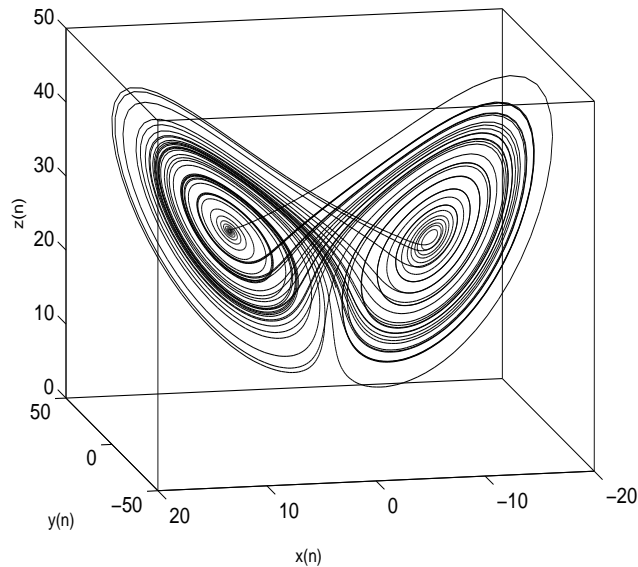


Figura 2.3: atrator de Lorenz.

O sistema de equações (2.12) modela as variações temporais no gradiente de temperatura de um fluido, tal como a atmosfera, aquecido por baixo. A variável x está relacionada com a velocidade do fluxo de fluido em convecção. Se $x > 0$, o fluido circula

³Sistema no qual o volume se contrai com o passar do tempo, ao contrário de sistema conservativo, que durante sua evolução temporal, há preservação de volume no espaço de estados.

no sentido horário, caso contrário, o fluido circula no sentido anti-horário. A variável y é proporcional à diferença de temperatura entre o fluido ascendente e o fluido descendente. A variável z é uma medida do grau de não-linearidade do gradiente de temperatura. Desde que o parâmetro c , chamado de *número de Rayleigh*, seja alto o suficiente, o sistema de Lorenz exibe caos e sensibilidade às condições iniciais. Assim, Lorenz concluiu que o “clima” é inerentemente imprevisível, a longo prazo.

Pela evolução temporal de apenas uma variável das equações de Lorenz, a variável $x(t)$, pode-se reconstruir a dinâmica do sistema através do Teorema da Imersão Takens. Pelos métodos de estimação das coordenadas de atraso discutidos na seção anterior, a Figura 2.4 mostra os valores 3 e 4 sugeridos como boas estimativas, respectivamente para o valor da dimensão de imersão e atraso de imersão da série de Lorenz. Sendo a dimensão do atrator de Lorenz $d = 2,06$ (ABARBANEL et al., 1993), uma condição suficiente para o teorema de imersão ($n_E \geq 2[d] + 1$) é adotar $n_E = 5$. Entretanto, tal como o valor encontrado pelo método de Cao, como também o valor de $n_E = 3$ encontrado pelo método dos falsos vizinhos encontrado em Abarbanel et al. (1993), coincido com o número de variáveis do sistema são condições mínimas que recuperam também as invariâncias do sistema na sua reconstrução.

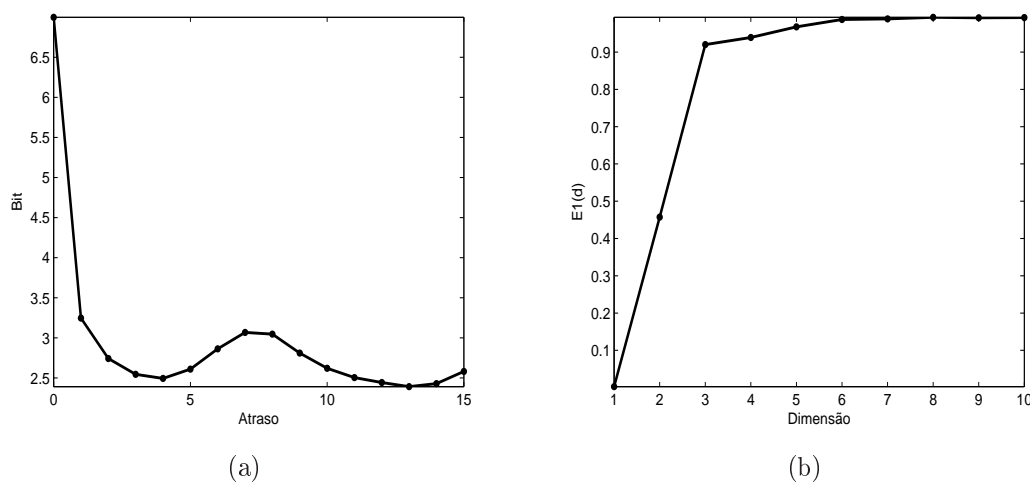


Figura 2.4: série de Caótica de Lorenz: (a) informação mútua para o cálculo do atraso de imersão; (b) método de CAO para o cálculo da dimensão de imersão.

Os exemplos anteriores ilustram como séries geradas por sistemas dinâmicos caóticos podem ter características próprias de sinais estocásticos, apresentarem aperiodicidade e possuírem sensível dependência das condições iniciais. Assim, uma análise baseada apenas na FAC ou uma análise visual é incapaz de caracterizar uma série temporal caótica, sendo necessária algumas outras ferramentas que são descritas a seguir.

2.5 Caracterização Elementar de Sistemas Caóticos

Sistemas caóticos são definidos qualitativamente como sistemas não-lineares, limitados em amplitude, possuidores de dinâmicas determinísticas que são aperiódicas e com uma alta dependência às condições iniciais. Nesta seção são apresentadas algumas técnicas de análise de séries temporais caóticas que permitem investigar cada uma destas características.

2.5.1 Limitado em Amplitude

Sistemas dinâmicos são ditos limitados (*bounded*) em amplitude se eles permanecem restritos a um volume finito do espaço de estados; ou seja, não se aproximam de ∞ ou $-\infty$ à medida que o tempo passa. Do ponto de vista prático, a definição de limitado como “permanecer em um volume finito” não é muito útil quando se tem um conjunto finito de dados, uma vez que qualquer medida de uma grandeza feita está sempre numa faixa finita, pois, a massa e energia do universo são finitos⁴. Assim, o conceito de limitado deve sempre estar associado aos valores máximos e mínimos aceitáveis para uma determinada aplicação.

2.5.2 Diagramas de Recorrência

Um sistema dinâmico é aperiódico quando um mesmo estado nunca é repetido. O comportamento caótico é inerentemente aperiódico. Esta porém é uma condição necessária, mas não suficiente. Lembre-se contudo que uma série estocástica também pode ser aperiódica, e que uma série temporal caótica pode se assemelhar a uma série periódica (KAPLAN; GLASS, 1995). Isto posto, o método a seguir pode ser utilizado para avaliar o grau de periodicidade de uma série temporal.

Retomando a Equação (2.5), sejam dois pontos $\mathbf{x}(i)$ e $\mathbf{x}(j)$ no espaço de imersão d_E -dimensional, cada ponto representando o estado do sistema nos instantes i e j , respectivamente. Pode-se calcular a distância entre estes dois pontos por

$$\rho_{i,j} = \|\mathbf{x}(i) - \mathbf{x}(j)\|, \quad (2.13)$$

em que $\|\cdot\|$ denota a distância euclidiana entre os dois vetores.

Se a série temporal for periódica com período T , então $\rho_{i,j} = 0$ quando $|i - j| = nT$,

⁴Infinito é um conceito matemático e não um conceito físico (KAPLAN; GLASS, 1995).

para $n = 0, 1, 2, \dots$. Por outro lado, para séries temporais aperiódicas, $\rho_{i,j}$ não mostra o mesmo padrão. Seja r um valor de referência ou limiar para a distância, tal que se desejava anotar quando a seguinte condição é verificada $\|\mathbf{x}(i) - \mathbf{x}(j)\| < r$. Pode-se fazer isto construindo um gráfico, no qual i é o eixo das abscissas horizontal e j é o eixo das ordenadas, em que um ponto é marcado na coordenada (i, j) quando $\|\mathbf{x}(i) - \mathbf{x}(j)\| < r$. Este gráfico é conhecido como Diagrama de Recorrência (*recurrence plot*) porque ele descreve como uma trajetória reconstruída tende a se repetir (KAPLAN; GLASS, 1995).

Na Figura 2.5 tem-se os diagramas de recorrência para três séries temporais, sendo que duas são geradas a partir do mapa logístico para diferentes valores do parâmetro a e a outra é uma seqüência de ruído branco aleatório. A partir das séries são gerados coordenadas de atrasos de dimensão $d_E = 2$, segundo a Equação (2.5).

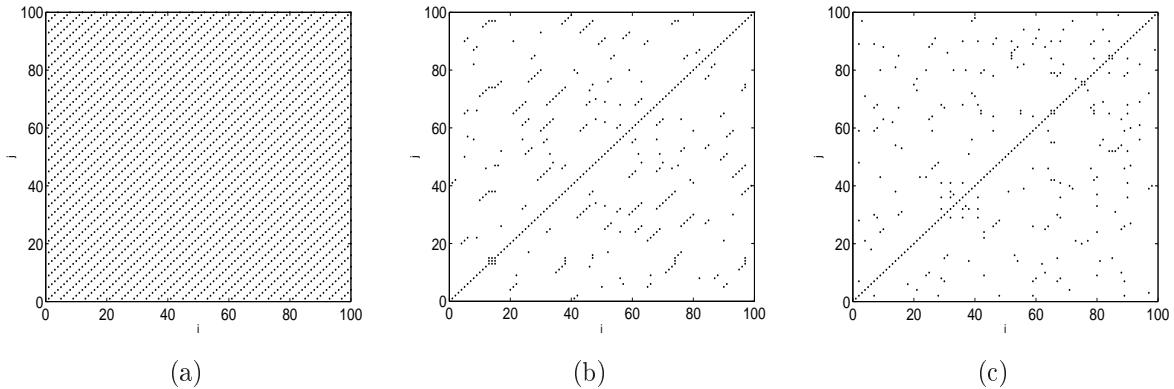


Figura 2.5: diagramas de recorrência: (a) mapa logístico, série periódica; (b) mapa logístico, série caótica; (c) ruído branco aleatório.

Para uma série periódica, o diagrama de recorrência mostrado na Figura 2.5a, formado pelo mapa logístico com $a=3,52$, $d_E=2$ e $r=0,01$, tem listras orientadas num ângulo de 45° e distanciadas entre si de 4 unidades de tempo, tanto ao longo do eixo vertical quanto no eixo horizontal. Já para a série temporal caótica, formada pelo mapa logístico com $a=4$, $d_E=2$ e $r=0,03$, o diagrama de recorrência mostrado na Figura 2.5b tem um estrutura mais complicada, algumas vezes com rastros de trajetórias com periodicidade e outras vezes não. E por último, para diagramas gerados para séries estocásticas aleatórias, no caso da Figura 2.5c, formado pelo ruído branco aleatório com $d_E=2$ e $r=0,3$, os padrões encontrados nos dois casos anteriores não são verificados.

2.5.3 Determinismo em Séries Temporais

Grosso modo, sistemas dinâmicos são ditos determinísticos quando nenhum termo aleatório governa a dinâmica do sistema e sua evolução temporal pode ser determinada com acurácia. Pode-se determinar se o sistema (ou série temporal) é determinístico construindo-se um modelo da dinâmica e verificando se as previsões feitas a partir deste modelo são precisas. Se as previsões são exatas, então o sistema é completamente determinístico. Mesmo que as previsões não sejam exatas, mas aproximadas, pode-se dizer que o sistema possui um componente determinístico. Se as previsões são ruins, então o sistema não é determinístico (KAPLAN; GLASS, 1995).

Pode-se construir um modelo dinâmico de inúmeras formas. No momento, não se está interessado em detalhar estes modelos, mas apenas apresentar a idéia de determinismo, bastando para isto explicitar algumas maneiras de quantificar a qualidade de uma previsão. Suponha que um certo modelo matemático implementa a função $\hat{g}(\cdot)$ mostrada na Equação (2.6). Esta função, por sua vez, permite gerar estimativas de valores futuros $\hat{x}(n+1)$ da série temporal. Uma vez tendo-se feito várias previsões, o primeiro passo é calcular o erro médio de previsão (*Mean Prediction Error*, MPE), que equivale ao conhecido erro quadrático médio (*Mean Square Error*, MSE)

$$\varepsilon^2 = \frac{\sum_{n=1}^T (x(n) - \hat{x}(n))^2}{T}, \quad (2.14)$$

em que T é o comprimento da seqüência de amostras preditas. Valores elevados para ε significam que as previsões são ruins e o modelo pode não ser determinístico⁵. Da mesma forma, valores pequenos de ε sugerem que o sistema é determinístico.

O valor de ε , na Equação (2.14), é um número absoluto, ou seja, por si só não diz se o erro está alto ou baixo. Para decidir o quanto um erro de previsão é elevado ou não, deve-se compará-lo com algum valor de referência. Isto é necessário para que modelos distintos possam ser comparados entre si. Para este fim, uma forma bastante utilizada para avaliar a precisão de um modelo é por meio do MSE Normalizado (*Normalized MSE*, NMSE), dado pela seguinte expressão

$$\varepsilon_N^2 = \frac{\varepsilon^2}{\sigma_x^2}, \quad (2.15)$$

em que σ_x^2 é a variância amostral da série temporal usada para criar o modelo $\hat{g}(\cdot)$, ou

⁵ Assume-se aqui que o modelo que gera as previsões está correto!

seja,

$$\sigma_x^2 = \frac{\sum_{n=1}^N (x(n) - \bar{x})^2}{N}, \quad (2.16)$$

em que N é o comprimento da série temporal e \bar{x} é a média amostral desta série.

Comparando as Equações (2.14) e (2.16) percebe-se que a diferença entre elas está somente no segundo termo da diferença. Na expressão do MSE, este termo é $\hat{x}(n)$, enquanto na expressão da variância é \bar{x} . Desta forma, pode-se entender a variância como equivalente ao MSE calculado para o caso em que o modelo gera previsões sempre iguais à média amostral. A lógica deste estimador é a seguinte: na dúvida ou na impossibilidade de gerar uma previsão mais exata de uma grandeza, adota-se seu valor médio. Esta estratégia é usada, por exemplo, pelas companhias de energia elétrica ou água quando o funcionário que faz a leitura do equipamento medidor não consegue fazê-lo por algum motivo. Na conta de luz/água correspondente àquele mês de leitura não-feita, vem o valor médio dos últimos 12 meses.

Conclui-se então que ao dividir MSE pela variância da série observada, se está na verdade comparando o erro de previsão de um dado modelo mais confiável com o erro de previsão gerado pelo preditor mais trivial. Quando a previsão é considerada boa, tem-se ε_N^2 próximo de zero. Previsões ruins geram valores de ε_N^2 próximos de 1, o que significa que o modelo $\hat{g}(\cdot)$ é tão ruim quanto o modelo que gera previsões pelo valor médio.

2.5.4 Sensibilidade às Condições Iniciais

A idéia de sensibilidade às condições iniciais foi primeiramente estudada por Lorenz, ao elaborar o problema da imprevisibilidade atmosférica. Muito provavelmente, as equações completas, que descrevem com maior precisão a circulação atmosférica, apresentam uma sensibilidade às condições iniciais, o que torna efetivamente impossível qualquer previsão do tempo a longo prazo. O menor erro nas medidas das condições iniciais climáticas, num dado instante, pode comprometer a validade de qualquer previsão do tempo para os instantes seguintes. A dependência sensível das equações da circulação atmosférica é conhecida como efeito borboleta. Segundo Lorenz, pequenas perturbações causadas pelo bater de asas de uma borboleta no Brasil pode provocar o surgimento de um tornado no Texas⁶ (MONTEIRO, 2006).

Desta forma, sensibilidade às condições iniciais é uma característica essencial de sis-

⁶ *Predictability: Does the Flap of a Butterfly's Wings in Brazil Set off a Tornado in Texas?*. Título de um seminário apresentado por Lorenz, em 1972.

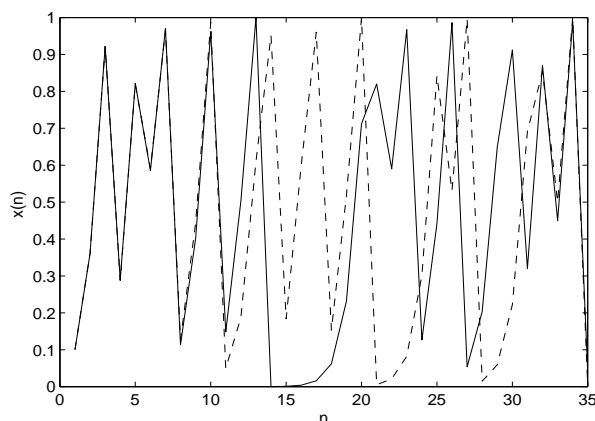


Figura 2.6: séries temporais geradas a partir do mapa logístico para duas condições iniciais diferentes, linha cheia, $x(0) = 0,1$ e linha pontilhada, $x(0) = 0,1001$.

temas caóticos, que significa que dois pontos de um atrator que estão inicialmente próximos se distanciam com o decorrer do tempo. Esta idéia está ilustrada na Figura 2.6, em que um mesmo sistema dinâmico caótico é simulado a partir de duas condições iniciais bem próximas. É importante lembrar que, na prática, uma condição inicial nunca é exatamente igual a outra, visto que erros de medição ou aproximação causados por arredondamento sempre ocorrem quando se usa o computador digital para analisar sistemas caóticos.

Não se deve confundir sensibilidade às condições iniciais com instabilidade, pois as duas séries caóticas mostradas na Figura 2.6 são geradas pelo mesmo sistema dinâmico e, conseqüentemente, são portadoras de informação sobre o mesmo.

Em sistemas dinâmicos determinísticos lineares, uma pequena diferença nas condições iniciais não altera significativamente o curso da série temporal, de modo que as séries resultantes são bem parecidas. Por exemplo, considere um sistema definido pela equação $x(t) = ax(t-1) + 1$, em que $0 < a < 1$. Para $a = 0,5$, duas seqüências geradas a partir de $x(0) = 0,1$ e $x(0) = 0,1001$ são praticamente indistinguíveis pelo tempo que durar a simulação.

Vale a pena contrastar a propriedade da sensibilidade às condições iniciais com a propriedade do determinismo e avaliar suas implicações para o problema de predição de séries temporais caóticas. Foi definido anteriormente que um sistema determinístico é aquele que permite ser predito com exatidão. Contudo, a sensibilidade às condições iniciais de um sistema caótico revela que, embora tal sistema seja determinístico, as trajetórias por eles geradas seguirão cursos históricos diferentes a partir de um certo instante.

Analisando a Figura 2.6 com mais detalhe e observando a propriedade da sensibilidade

às condições iniciais com a propriedade do determinismo, nota-se que até o instante de tempo $n = 11$, as séries temporais são praticamente as mesmas. Deste instante em diante, as trajetórias seguem rumos diferentes. Este exemplo ilustra bem a implicação da atuação destas duas propriedades de sistemas caóticos para o problema de séries temporais, que pode ser colocada da seguinte forma: é possível prever valores da série temporal para horizontes curtos de tempo, mas para horizontes maiores ou mais longos, a predição exata é impossível, uma vez que não se tem certeza da exatidão dos valores das condições iniciais de qualquer sistema real.

É importante destacar que mesmo predições para horizontes longos de tempo são úteis, pois se o modelo $\hat{g}(\cdot)$ for considerado bom, as predições $\hat{x}(n + 1)$ geradas correspondem a uma série temporal pertencente ao sistema dinâmico subjacente e, desta forma, podem ser usadas para reconstruir o espaço de estados. Predições em horizontes pequenos de tempo são chamadas de predições de curto-prazo, enquanto que predições em horizontes mais longos são chamadas de predições de médio ou longo-prazo. Um tipo especial de predição de longo prazo é chamado de predição recursiva ou predição de horizonte infinito. Neste tipo de predição, o modelo $\hat{g}(\cdot)$ é usado como um sistema autônomo, ou seja, ele é realimentado com suas próprias predições anteriores para horizontes longos de tempo.

Um modo de caracterizar a dinâmica de um sistema dinâmico caótico é medir o grau de sensibilidade às condições iniciais, sendo que isto é principalmente feito através do cálculo dos expoentes de Lyapunov do sistema dinâmico (KAPLAN; GLASS, 1995). O conceito de expoentes de Lyapunov já existia bem antes do estabelecimento da moderna teoria de caos e foi desenvolvido para caracterizar a estabilidade de sistemas lineares assim como não-lineares.

Expoentes de Lyapunov, cuja quantidade é igual à dimensão do espaço de estados, são definidos como a média da razão exponencial em que se diverge no tempo trajetórias vizinhas. O conjunto de expoentes de Lyapunov de um sistema dinâmico é chamado de *spectro* deste sistema. Ele resume o nível médio de convergência ou divergência de duas trajetórias vizinhas no espaço de estados, assumindo valores negativos, positivos ou nulos. Detalhes sobre a definição matemática de expoentes de Lyapunov são deixados para o Apêndice B.

Valores negativos significam que duas trajetórias encontram-se próximas uma da outra, tendendo a convergir. Expoentes de Lyapunov positivos, por outro lado, resultam em divergência de trajetórias e aparecem somente em sistemas caóticos. Portanto, um expoente positivo de Lyapunov é um dos mais importantes indicadores de dinâmica

caótica. Um expoente positivo de Lyapunov quantifica a sensibilidade às condições iniciais do sistema dinâmico subjacente, por mostrar como pontos no espaço de fase inicialmente próximos separam-se com o passar do tempo (WILLIAMS, 1997).

A maior dificuldade na análise de séries temporais caóticas é que o espaço de estados é desconhecido, sendo que o espectro é calculado em algum espaço de imersão, que é o espaço de estados reconstruído de acordo com o Teorema de Takens. Assim, o número de expoentes depende da ordem d_E da reconstrução, e pode ser maior do que no espaço de estados original. A menos que a dimensão do espaço seja baixa e que os dados obtidos sejam de elevada qualidade (baixo nível de ruído), não se deve calcular todo o espectro. É interessante tentar computar primeiramente o maior expoente de Lyapunov. Este pode ser determinado até mesmo sem a construção explícita de um modelo para a série temporal (KANTZ; SCHREIBER, 1997).

Pode-se também calcular todo o espectro de Lyapunov, porém requer consideravelmente mais esforço computacional do que apenas o cálculo do maior expoente. Dentre as principais implementações de métodos que calculam o maior expoente de Lyapunov, duas se destacam, Kantz & Schreiber (1997) e Rosenstein et al. (1993). Elas diferem somente na definição das trajetórias vizinhas. O algoritmo de Wolf et al. (1985) é outra implementação bastante conhecida e tem sido amplamente usado, mas devido a sua instabilidade e a impossibilidade de diferenciar a divergência exponencial da divergência devido ao ruído, não é muito recomendado.

Esta dissertação é motivada em grande parte pelo problema de predição de séries temporais caóticas, sendo por esta razão que as ferramentas anteriores de caracterização do Caos em sistemas físicos são discutidas. Na próxima seção, tenta-se estabelecer uma relação com sistemas caóticos e geometria fractal. Em especial, está-se interessado em uma propriedade de geometria fractal conhecida com auto-similaridade, cuja manifestação tem sido verificada em diversos sistemas físicos, tais como tráfego em redes de comunicações.

2.6 Caos e Fractais

Na seção anterior, discutiu-se a propriedade dinâmica de sistemas caóticos que se manifesta na sensível dependência da evolução deste sistema às suas condições iniciais. Este estranho comportamento no tempo de um sistema caótico determinístico é refletido na geometria do conjunto de pontos que formam as trajetórias do sistema no espaço de estados, ou seja, em seu atrator. Atratores de sistemas caóticos têm geralmente uma geometria

muito complicada, o que levou pesquisadores a chamá-los de *atratores estranhos* (KANTZ; SCHREIBER, 1997).

É bom lembrar que existem semelhanças e diferenças entre atratores estranhos e atratores convencionais encontrados em sistemas dinâmicos lineares, tais como pontos de equilíbrio e ciclos-limites. Assim, como os atratores lineares estáveis, trajetórias que começam longe do atrator são atraídas para ele. Contudo, uma vez que tais trajetórias estejam no atrator, elas não se repetem como nos ciclos-limites estáveis, e sim divergem uma das outras, permanecendo porém ainda dentro do atrator.

A estrutura geométrica do atrator estranho desenhada no espaço de estados tem dimensão fracionária, ou seja, não é um número inteiro como nos objetos geométricos clássicos (MONTEIRO, 2006). Daí a razão do termo geometria fractal, usado para designar a geometria dos atratores estranhos. Costuma-se dizer que um certo atrator estranho é um objeto fractal, ou simplesmente fractal. O conceito de fractal pode ser introduzido como uma generalização da noção familiar de dimensão. Assim como um ponto, uma linha, um quadrado e um cubo podem ser ditos ter dimensões de 0, 1, 2, e 3 respectivamente, objetos fractais têm dimensões não-integrais (fracionárias).

Pode-se estudar sistemas caóticos sem conexão direta e até de forma independente da geometria fractal, porém muitos sistemas físicos reais exibem propriedades típicas de fractais, tal como auto-similaridade, evidenciando que a geometria fractal fornece uma base natural para descrever fenômenos irregulares. Estudos recentes revelam, por exemplo, que tráfego de pacotes em redes de comunicações possuem características que podem ser descritas mais eficientemente em termos de processos fractais, em vez de processos estocásticos convencionais (ERRAMILI et al., 2002; LELAND et al., 1994).

Em particular, a auto-similaridade é uma propriedade típica de processos fractais. Um objeto, processo ou fenômeno é auto-similar quando se mantém certas características em diferentes escalas de tempo ou espaço. Cada escala lembra outras escalas, embora elas sejam diferentes. Esta propriedade tem sido observada em dados de tráfego de rede, em que para diversas escalas de tempo, pode-se verificar que o tráfego parece ser o mesmo. Modelos não-lineares, tais como redes neurais artificiais, que sejam capazes de modelar a dinâmica auto-similar, podem ser úteis na previsão de tráfego em redes de comunicações. Maiores detalhes do tráfego com propriedades fractais ou auto-similares são apresentados no próximo capítulo.

2.7 Conclusão

Este capítulo dedicou-se à definição formal de sistemas dinâmicos não-lineares, realçando as semelhanças e diferenças principais entre séries temporais caóticas e estocásticas lineares. As principais técnicas e métodos de caracterização de sistemas caóticos foram descritos. Estes métodos, de forma geral, funcionam como testes para se conhecer se certo sistema é caótico ou possui irregularidades provocada apenas por elementos aleatórios. Também há a necessidade de se medir o quanto uma série é aperiódica, o grau de determinismo e o quanto certos sistemas são sensíveis às condições iniciais.

Desta forma, a caracterização do caos, juntamente com a estimação dos parâmetros de imersão para reconstrução de um sistema são pré-requisitos e ferramentas importantes para a predição e modelagem de séries temporais. Neste contexto, torna-se importante analisar sistemas dinâmicos sem que se conheça detalhes sobre a sua dinâmica, não possuindo portanto um modelo matemático estabelecido. Uma alternativa para isto é a análise de séries temporais que podem ser obtidas diretamente a partir de um experimento, como por exemplo dados extraídos pela medição do número de pacotes por um determinado período de tempo em uma rede de computadores, com o objetivo de modelar e prever o tráfego de pacotes na rede.

3 MODELAGEM DE TRÁFEGO DE REDES

3.1 Introdução

A teoria de sistemas dinâmicos, em especial sistemas caóticos e fractais, fornece ferramentas de análise, modelagem e predição que podem ser usados em uma ampla gama de aplicações. Uma das possíveis aplicações dos modelos de redes neurais estudados nesta dissertação é justamente a predição de tráfego em redes de alta velocidade, Internet e tráfego de vídeo. Do ponto de vista da modelagem, o principal atrativo de modelos não-lineares, baseados em redes neurais, encontra-se na possibilidade de capturar a complexidade do problema de uma maneira concisa e mais eficiente que técnicas convencionais de predição de séries temporais.

Métodos oriundos da teoria do caos e fractais são usados de forma bem sucedida na descrição de fenômenos físicos complexos em vários ramos da ciência, tais como modelos de sistemas cujo comportamento se dá em rajadas (*“bursty”*); ou seja, o sinal observado contém períodos de muita atividade intercalados com períodos de pouca ou nenhuma atividade. Estudos recentes revelam que o tráfego real de pacotes tem características que podem ser mais eficientemente descritas em termos de processos fractais ou processos auto-similares, em vez de processos estocásticos convencionais, tais como processos de Poisson ou ARMA (ERRAMILI et al., 1994).

Desta forma, percebe-se que o comportamento do tráfego de rede é muito diferente dos antigos paradigmas do convencional tráfego telefônico e dos modelos usuais de tráfego; modelos de Poisson, modelo de Poisson modulado por uma cadeia de Markov, modelos de vazão de fluidos e modelos lineares. Assim, a partir deste novo comportamento, o tráfego vem sendo classificado em de duas formas: tráfego com dependência temporal de curta duração e tráfego com dependência temporal de longa duração. Este último é, às vezes, chamado simplesmente de tráfego com memória longa.

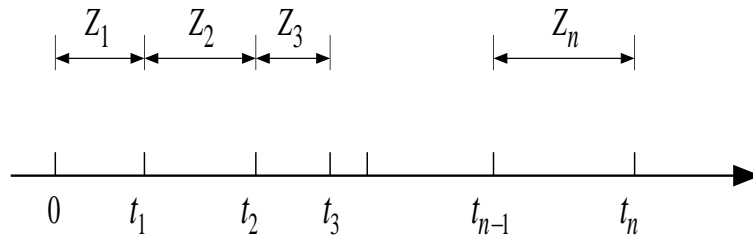


Figura 3.1: esboço dos pontos de ocorrência e tempo entre-chegadas.

Neste capítulo, são apresentados conceitos básicos sobre a teoria de tráfego de redes. Inicialmente são abordadas técnicas tradicionais de modelagem estocástica de tráfego, modelos estocásticos com dependência de curta duração. Em seguida é discutida a recente caracterização de tráfego em redes de comunicação como auto-similar, ou seja, como processos que tem como características principais as propriedades fractais de auto-similaridade e dependência temporal longa. Também discutem-se mapas caóticos que são usados na literatura para modelagem e previsão de tráfego de redes de comunicações.

3.2 Modelos de Tráfego

Um tráfego simples consiste de chegadas únicas de entidades discretas (pacotes, mensagens, células, bytes, etc.). Estas entidades são representadas internamente por estruturas de dados (mensagens ou pacotes) com o formato definido por um protocolo. As observações descrevem, por exemplo, os intervalos entre chegadas sucessivas de comandos de um usuário, mostrando o nível de comportamento do usuário. Podem também descrever os intervalos entre chegadas de pacotes ou os tamanhos dos pacotes de dados, mostrando o nível de comportamento da aplicação ou da rede (FROST; MELAMED, 1994).

Seja t a variável tempo. Suponha um certo experimento que começa em $t = 0$. Eventos (ou seja, resultados do experimento) de um tipo particular ocorrem aleatoriamente, o primeiro no instante T_1 , o segundo em T_2 e assim por diante. Assim, pode-se entender T_i como uma variável aleatória que representa o instante em que o i -ésimo evento ocorre, e os valores t_i que T_i ($i = 1, 2, \dots$) assume são chamados *pontos de ocorrência* (ver Figura 3.1).

Considera-se a seguinte definição para a variável aleatória Z_n

$$Z_n = T_n - T_{n-1}, \quad (3.1)$$

sendo $T_0 = 0$. Assim, Z_n representa o tempo entre o n -ésimo e o $(n - 1)$ -ésimo eventos. A seqüência *ordenada* de variáveis aleatórias $\{Z_n, n \geq 1\}$ é comumente conhecida como

um **Processo Entre-chegadas** (*Interarrival Process*).

Se todas as variáveis aleatórias Z_n são independentes e identicamente distribuídas (i.i.d), então $\{Z_n, n \geq 1\}$ é chamada de **Processo de Renovação** (*Renewal Process*) ou **Processo Recorrente** (*Recurrent Process*). A partir da Equação (3.1) pode-se notar que

$$T_n = Z_1 + Z_2 + \dots + Z_n, \quad (3.2)$$

em que T_n denota o tempo transcorrido do início até a ocorrência do n -ésimo evento. Deste modo, $\{T_n, n \geq 0\}$ é comumente chamado de **Processo de Chegada** (*Arrival Process*) (HSU, 1997).

Definição 1: Um processo aleatório $\{X(t), t \geq 0\}$ é dito ser um **Processo de Contagem** (*Counting Process*) se $X(t)$ representa o número total de “eventos” que ocorreram no intervalo $(0, t)$. A partir desta definição, pode-se ver que para ser um processo de contagem, $X(t)$ deve satisfazer as seguintes condições:

1. $X(t) \geq 0$ e $X(0) = 0$;
2. $X(t)$ é um número inteiro;
3. $X(s) \leq X(t)$, se $s < t$;
4. $X(t) - X(s)$ é o número de eventos ocorridos no intervalo (s, t) .

Uma realização típica de $X(t)$ é mostrada na Figura 3.2. Esta figura pode representar, por exemplo, o número de clientes entrando em um banco. Toda vez que um cliente chega, um contador é incrementado. O instante de chegada do i -ésimo cliente é denotado t_i . Visto que não se pode “adivinhar” ou determinar com precisão absoluta o instante que cada novo cliente chega, então a seqüência $\{t_1, t_2, \dots, t_n\}$, representada simplesmente por $\{t_i\}$, é uma seqüência de números aleatórios. Usando raciocínio semelhante, o número de clientes que chegam no intervalo $(t_0, t]$ é uma variável aleatória.

Definição 2 - Um processo de contagem $X(t)$ possui *incrementos independentes* se o número de eventos que ocorrem em intervalos de tempo disjuntos (i.e. que não se sobrepõem) são independentes.

Definição 3 - Um processo de contagem $X(t)$ possui *incrementos estacionários* se o número de eventos no intervalo $(s + h, t + h)$, ou seja, $X(t + h) - X(s + h)$, tem a mesma distribuição que o número de eventos no intervalo (s, t) , ou seja, $X(t) - X(s)$, para todo $s < t$ e $h > 0$.

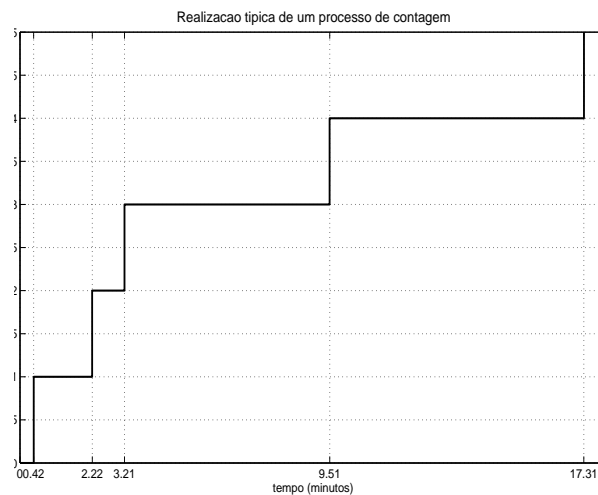


Figura 3.2: realização típica de um processo de contagem.

Processos de renovação tem um longa história de aplicações, devido a sua relativa simplicidade matemática. A independência das variáveis significa que as observações no tempo t não dependem de qualquer observação do passado ou do futuro. Os processos de renovação não capturam a correlação de uma dada seqüência. A importância de se detectar autocorrelações provém do fato de que esta função expressa dependências temporais (JAGERMAN et al., 1997).

A seguir são discutidos dois importantes casos de processos de tráfego de renovação: processos de Poisson e processos de Bernoulli (HSU, 1997).

3.2.1 Processos de Poisson

Um dos processos de contagem mais importantes é chamado de **Processo de Poisson** (ou Processo de Contagem de Poisson). Assim, um processo de contagem $X(t)$ é dito ser um processo de Poisson com taxa de chegada (ou *intensidade*) $\lambda > 0$ se:

1. $X(0) = 0$;
2. $X(t)$ tem incrementos independentes;
3. O número de eventos em qualquer intervalo de comprimento t obedece a uma distribuição de probabilidade de Poisson com média λt , ou seja, para $s, t > 0$,

$$P[X(t+s) - X(s) = n] = e^{-\lambda t} \frac{(\lambda t)^n}{n!}, \quad n = 0, 1, 2, \dots \quad (3.3)$$

Segue da definição anterior que um processo de Poisson tem incrementos estacionários e que

$$E[X(t)] = \lambda t \quad \text{e} \quad \text{Var}[X(t)] = \lambda t. \quad (3.4)$$

Assim, o número esperado de eventos em um intervalo unitário $(0, 1)$, ou qualquer outro intervalo de comprimento unitário, é apenas λ ; daí, o nome *taxa* ou *intensidade* de chegada. A função de autocorrelação $\rho_X(t, s)$ de um processo de Poisson $X(t)$ com taxa λ é dada por

$$\rho_X(t, s) = \lambda \min(t, s) + \lambda^2 ts. \quad (3.5)$$

Resumindo, um processo de Poisson nada mais é do que uma regra matemática que atribui probabilidades ao número de ocorrências de um evento. O único parâmetro que se precisa especificar no modelo de Poisson é o número médio de ocorrências em um intervalo unitário, ou seja, λ . Pode-se mostrar que em um processo de Poisson, os intervalos entre eventos sucessivos são variáveis aleatórias independentes e exponencialmente distribuídas. Desta forma, costuma-se também identificar o processo de Poisson como um processo de renovação com intervalos distribuídos exponencialmente.

Processos de Poisson são usados como modelos probabilísticos em uma ampla gama de aplicações nas mais diversas áreas, tais como número de chamadas telefônicas chegando em uma central em certo intervalo de tempo, número de erros tipográficos em uma página de livro, número de clientes entrando em um banco durante um dado intervalo, número de pacotes que chegam em um servidor Web em certo período, número de acidentes em um cruzamento em uma semana, dentre outros. O modelo de Poisson é um dos mais usuais modelos de tráfego, com origem no advento da telefonia.

Na modelagem de tráfego de pacotes e conexões de chegadas estes são geralmente assumidos como processos de Poisson (FROST; MELAMED, 1994). Contudo, Paxson & Floyd (1995) discutem algumas limitações dos processos de Poisson, e que estes são válidos somente para a modelagem da chegada de sessões do usuário (e.g. conexões TELNET e controle de conexões FTP); mas falham como modelos para outros processos de chegada WAN (*Wide Area Network*). Desta forma, processos de chegada de pacote da rede WAN são melhores modelados usando processos auto-similares.

Processos de Bernoulli são o equivalente discreto de Processos de Poisson. Jagerman et al. (1997) discutem que as ocorrências podem acontecer em algum fatia (*slot*) de tempo. A probabilidade de uma chegada em um *slot* de tempo é p , independente das outras

chegadas. A probabilidade que aconteçam k ocorrências em n slots é dada por

$$P(X_n = k) = \binom{n}{k} p^k (1-p)^{n-k}, \quad (3.6)$$

para $k = 0, \dots, n$, de tal forma que o tempo entre chegadas tem uma distribuição geométrica com parâmetro p

$$P(Z_n = j) = p(1-p)^j, \quad (3.7)$$

para $j = 0, 1, \dots$.

A função de autocorrelação associada a modelos de tráfego, baseados em processos de Poisson ou Bernoulli, são típicas de processos estocásticos independentes, ou seja, tais processos não capturam (modelam) dependências temporais entre amostras sucessivas do sinal de tráfego. Quando tais dependências existem, elas evidenciam que o processo estocástico ou determinístico, em análise, possui memória e, por isso, os modelos de Poisson e Bernoulli não são adequados para sua modelagem. Os modelos de tráfego descritos a seguir incluem em sua formulação mecanismos que modelam dependências temporais na seqüência $\{Z_n\}$.

3.3 Cadeias de Markov

Processos de Markov descrevem um tipo de dependência entre as amostras de um processo estocástico, em que o valor da variável aleatória no instante seguinte depende apenas do valor atual do processo estocástico. Processos estocásticos com esta dependência são muito importantes como ferramentas para avaliação de desempenho de redes, já que simplificam bastante o tratamento analítico (MOURA et al., 1986). Processos de Markov introduzem uma pequena dependência entre os elementos de um seqüência $\{X(t), t \geq 0\}$; conseqüentemente, podem capturar tráfego explosivo (rajada), devido a autocorrelação diferente de zero. Sendo assim, o primeiro passo para descrever dependências entre as ocorrências de tráfego é dado pelos processos de Markov.

Sem entrar em maiores detalhes sobre a teoria de Processos de Markov, considere uma cadeia de Markov de tempo contínuo $\mathcal{X} = \{X(t)_{t=0}^{\infty}\}$, com espaço de estados discreto. Neste caso, \mathcal{X} se comporta como segue: a cadeia permanece no estado i por um tempo distribuído exponencialmente com parâmetro λ_i que depende do estado atual i . A cadeia então muda para o estado j com probabilidade p_{ij} , de acordo com uma matriz de probabilidades de transição $\mathbf{P} = [p_{ij}]$. Uma transição de um estado para outro no instante

seguinte, ou a possível volta para um mesmo estado é interpretada como sinalização de uma chegada de pacotes, de modo que os tempo entre chegadas são exponencialmente distribuídos e os parâmetros de taxa de chegada seriam dependentes dos estados onde a cadeia estava antes da mudança.

Modelos de Markov capturam as dependências entre intervalos de chegada da seguinte forma. Esta modelagem pode ser definida para processos entre-chegadas $\{Z_n\}$ em termos da matriz de $\mathbf{P} = [p_{ij}]$. Neste caso, o estado i corresponde a i slots de tempo sem atividade separando chegadas sucessivas, e p_{ij} é a probabilidade de se ter uma separação de j slots, dado que a separação anterior durou i slots. Chegadas podem acontecer na forma de uma única entidade de tráfego, um lote (*batch*) de entidades, ou uma grandeza contínua.

3.3.1 Modulação via Processos de Markov

Processos estocásticos, que são em si modulados por processos de Markov (*Markov-modulated Processes*), constituem uma classe muito importante de modelos de tráfego. A idéia é introduzir explicitamente a noção de estado na descrição do fluxo (*stream*) de tráfego na forma de um processo de Markov auxiliar que se desenvolve no tempo de tal modo que seu estado atual controla a lei de probabilidade do mecanismo gerador de tráfego.

Seja um processo de Markov de tempo contínuo $\mathcal{X} = \{X(t)_{t=0}^{\infty}\}$, com espaço de estados discreto e finito $E = \{1, 2, \dots, m\}$. Assumindo que, enquanto \mathcal{X} está no estado k , a lei de probabilidade que governa a chegada de tráfego é completamente determinada por k , valendo para cada $1 \leq k \leq m$. Quando \mathcal{X} passa por uma transição, por exemplo, para o estado j , então uma nova lei de probabilidade para as chegadas passa a regular o tráfego enquanto durar o estado j , e assim por diante. Diz-se então que a lei de probabilidade é *modulada* pelo estado de \mathcal{X} . Tais sistemas são chamados de **processos duplamente estocásticos** (*doubly stochastic processes*) ou **processos modulados por Markov** (*Markov modulated processes*). O último termo é mais comum pelo fato de deixar claro que o tráfego está estocasticamente subordinado a \mathcal{X} .

Um importante caso de processo modulado por Markov é chamado de **Processo de Poisson Modulado por Markov** (*Markov-Modulated Poisson Process*), *MMPP*. Neste caso, o mecanismo de modulação simplesmente estipula que no estado k de \mathcal{X} , as chegadas devem ocorrer de acordo com um processo de Poisson a uma taxa λ_k . Assim, à medida que o estado se altera, a taxa do processo de Poisson também o faz. Este modelo tem sido largamente usado para modelar fontes de tráfego de voz (HEFFES; LUCANTONI, 1986).

Os modelos de tráfego baseados em processos (ou cadeias) de Markov são adequados para capturar dependências temporais em horizontes muito curtos de tempo, em geral restritos à relação entre o estado atual e o próximo do sinal de tráfego. Além disso, tais modelos são bons para explicar ou simular tráfego, e não fazer previsões sobre o tráfego futuro. Tais previsões são úteis, por exemplo, para o gerenciamento de recursos da rede (YOUSEFZADEH; JONCKHEERE, 2005), detecção de falhas (GONÇALVES, 2003), detecção de intrusos (HELLERSTEIN et al., 2001), congestionamento, dentre outras aplicações. Na próxima seção, dá-se início à descrição de modelos que tentam capturar dependências que cobrem horizontes mais longos de tempo e utilizar o modelo para fazer previsões de tráfego.

3.4 Modelos Lineares de Box-Jenkins

Processos estocásticos conhecidos genericamente como modelos de Box-Jenkins (BOX et al., 1994) são utilizados para capturar dependências de horizonte mais longos que aquelas capturadas por processos de Markov. Dentre os modelos de Box-Jenkins mais conhecidos destacam-se os modelos autoregressivos (AR), médias móveis (MA) e combinações destes, tais como os modelos ARMA e ARIMA. Todos eles são paramétricos, ou seja, possuem um número finito de parâmetros cujos valores são estimados a partir do sinal ou série temporal de tráfego medido.

Morettin & Tolo (2004) descreve um ciclo iterativo para a construção de um modelo de Box-Jenkins que melhor se ajusta a uma da série temporal. O ciclo é formado por basicamente quatro etapas:

- a primeira etapa do ciclo é a fase de **especificação**, em que uma classe geral de modelos é considerada para análise;
- com base na análise das funções de autocorrelação e autocorrelação parcial define-se a etapa de **identificação** do modelo mais adequado;
- o próximo passo é a etapa de **estimação**, em que os parâmetros do modelo identificado são estimados;
- através de uma série de testes, sendo o principal a análise dos resíduos (erros de previsão), o modelo ajustado chega na fase de **validação** ou **diagnóstico**.

Se o modelo não for satisfatório, o ciclo é repetido, voltando-se à fase de identificação.

A etapa de identificação é a mais crítica, visto que é possível chegar a uma situação em que vários modelos diferentes se adaptam bem a uma determinada série temporal. O princípio da parcimônia, também conhecido como lâmina de Occam (*Occam's razor*), serve como orientação geral nestes casos. Em linhas gerais, este princípio prega que se utilize o modelo mais simples, i.e. com menos parâmetros, caso mais de um modelo explique a série adequadamente.

A utilização de modelos de Box-Jenkins não é tão imediata como nos modelos baseados em processos de Markov, pois requer experiência no trato de ferramentas de processamento de sinais (MORETTIN; TOLOI, 2004). Além disso, os modelos de Box-Jenkins não são destinados a explicar fenômenos não-lineares ou que possuam irregularidades não-estocásticas, tais como sistemas complexos determinísticos e caóticos. Mesmo diante destas dificuldades, sua utilização é bastante difundida e, por isso, são descritos a seguir.

3.4.1 Modelos Autoregressivos

Modelos autoregressivos de ordem p , $AR(p)$, são os modelos de Box-Jenkins mais simples, em que se escreve o valor atual da variável aleatória $x(n)$ como uma soma ponderada de seus valores passados $x(n), x(n-1), x(n-2), \dots$ mais o ruído branco gaussiano $a(n)$

$$x(n) = \phi_0 + \phi_1 x(n-1) + \phi_2 x(n-2) + \dots + \phi_p x(n-p) + a(n) = \phi_0 + \sum_{i=1}^p \phi_i x(n-i) + a(n), \quad (3.8)$$

em que $\phi_i, i = 0, \dots, p$, são os coeficientes do modelo, que juntamente com a ordem da memória p , constituem os parâmetros do modelo. Na Equação (3.8) a seqüência $\{a(n), n \geq 0\}$ de ruído branco gaussiano tem média nula e variância $\sigma_a^2 \neq 0$.

Na forma preditiva, o modelo AR pode ser escrito da seguinte maneira

$$x(n+1) = \phi_0 + \phi_1 x(n) + \phi_2 x(n) + \dots + \phi_p x(n-p+1) + a(n), \quad (3.9)$$

em que valem todas as definições definidas para a Equação (3.8). Independentemente da formulação escolhida, existem várias técnicas para calcular os coeficientes ϕ_i de um modelo AR, sendo a mais comum a dos *Mínimos Quadrados* (MQ) [(AGUIRRE, 2000)], que é equivalente ao método de estimação por máxima verossimilhança (*maximum likelihood*) quando o ruído é gaussiano.

De acordo com a técnica MQ, usando o modelo da Equação (3.9) em uma série temporal com N observações, ou seja, $\{x(n)\}_{n=1}^N$, os coeficientes são calculados por meio da

seguinte expressão

$$\phi = (\mathbf{Y}^T \mathbf{Y})^{-1} \mathbf{Y}^T \mathbf{p}, \quad (3.10)$$

em que $\phi = [\phi_0 \ \phi_1 \ \phi_2 \ \cdots \ \phi_p]^T$ é o vetor de coeficientes, \mathbf{p} é o vetor de predição e \mathbf{Y} é a matriz de regressão. Estes dois vetores e a matriz \mathbf{Y} são dados por

$$\mathbf{p} = \begin{pmatrix} x(p+1) \\ x(p+2) \\ \vdots \\ x(N) \end{pmatrix}, \quad \mathbf{Y} = \begin{pmatrix} 1 & x(p) & \cdots & x(1) \\ 1 & x(p+1) & \cdots & x(2) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & x(N-1) & \cdots & x(N-p-1) \end{pmatrix}. \quad (3.11)$$

Uma vez calculados os coeficientes, estes são utilizados na Equação (3.9) para estimar valores futuros da série temporal. Apesar de sua simplicidade, este método pode apresentar problemas de instabilidade numérica devido à inversão de matrizes, principalmente para valores elevados de p e N pequeno. De qualquer modo, o uso do modelo AR com coeficientes calculados pelo método MQ está amplamente disseminado, não só na Estatística e ciências naturais, como também em Engenharia, Economia e Ciência da Computação, servindo sempre como referência para estudos comparativos.

3.4.2 Modelos de Médias Móveis

Modelos de médias móveis de ordem q , denotados MA(q), são descritos como uma combinação linear finita de q valores passados da seqüência de ruído branco

$$x(n) = a(n) + \theta_1 a(n-1) + \theta_2 a(n-2) + \cdots + \theta_q a(n-q), \quad (3.12)$$

em que θ_i são os coeficientes do modelo, que juntamente com sua ordem q , constituem os parâmetros do modelo. Estes modelos são mais difíceis de aplicar que modelos AR(p) e o cálculo de seus coeficientes, a partir dos dados observados, é geralmente feito através do método de máxima verossimilhança. Em geral, modelos MA(q) são usados em conjunção com modelos AR(p), a fim de reduzir o número de parâmetros deste último.

3.4.3 Modelos Autoregressivos e de Médias Móveis

Para muitas séries encontradas na prática, quando se deseja modelos com um número menor de parâmetros do que os obtidos para um modelo AR(p) ajustado à mesma série, o uso combinado de termos autoregressivos e de médias móveis é a solução adequada (MORETTIN; TOLOI, 2004). Nestes casos, os modelos ARMA(p, q) são a forma mais

simples de combinação

$$x(n) = \phi_1 x(n-1) + \phi_2 x(n-2) + \dots + \phi_p x(n-p) + a(n) + \theta_1 a(n-1) + \theta_2 a(n-2) + \dots + \theta_q a(n-q), \quad (3.13)$$

em que θ_i e ϕ_i são os coeficientes autoregressivos e de médias móveis do modelo, que juntamente com as ordens p e q , constituem os parâmetros do mesmo.

3.4.4 Modelos Auto-Regressivos Integrado de Médias Móveis

Os modelos lineares de Box-Jenkins discutidos até aqui são apropriados somente para descrever séries estacionárias, isto é, séries que se desenvolvem no tempo próximo de uma média constante. Visto que as séries encontradas na prática não são geralmente estacionárias, tais como séries econômicas e financeiras, (MORETTIN; TOLOI, 2004), faz-se necessário discutir um modelo que seja capaz de tratar processos não-estacionários.

Em geral, a estacionariedade de uma série temporal pode ser conseguida através de transformações atuando sobre a série original. Uma forma de tornar séries não-estacionárias em séries estacionárias é através de diferenças entre seus valores consecutivos. Por exemplo, dada uma série $\{x(n)\}_{n=1}^N$ não-estacionária, seja uma nova série $\{w(n)\}_{n=1}^{N-1}$ obtida por meio da seguinte operação

$$w(n) = \Delta x(n) = x(n) - x(n-1). \quad (3.14)$$

Caso esta série ainda não seja estacionária, o mesmo procedimento pode ser novamente aplicado sobre as amostras $w(n)$ até que uma série estocástica seja estacionária o suficiente para permitir que um modelo linear de Box-Jenkins possa ser ajustado a ela.

Uma série temporal $\{x(n)\}_{n=1}^N$ tal que, tomando-se um número finito de diferenças entre amostras sucessivas torna-se estacionária, é chamada *não-estacionária homogênea*. Como o processo é reversível, a série não-estacionária original $\{x(n)\}_{n=1}^N$ pode ser obtida a partir da série estacionária omitida pela soma (ou integração) de amostras sucessivas, daí este modelo ser chamado de **Autoregressivo Integrado de Médias Móveis** de ordens p , d e q , ou simplesmente ARIMA(p,d,q).

Do exposto conclui-se que modelos ARIMA são modelos ARMA em que se lança mão um número d de vezes do expediente de diferenças sucessivas mostrado na Equação (3.14) para produzir uma série estacionária, a partir de uma série-não estacionária homogênea. Na maioria dos casos, raramente se recomenda valores maiores que $d = 1$ ou $d = 2$. Por isso, o modelo ARIMA é adequado para descrever séries cujo o comportamento não-

estacionário é não-explosivo (*non-bursty*), ou seja, séries que apresentam homogeneidade em seu comportamento não-estacionário.

Dois pontos restam ainda ser destacados, quando se considera modelos AR, ARMA e ARIMA para predição ou modelagem de tráfego. Primeiramente, modelos AR, ARMA e ARIMA são processos com função de autocorrelação que decaem geometricamente com o *lag* (k), ou seja, $\rho(k) \sim r^n$ para algum $0 < r < 1$, à medida que $n \rightarrow \infty$. Desta forma, tais modelos são processos indicados para capturar dependência temporal (memória) de curta duração e, portanto, incapazes de capturar os fenômenos observados em modernas redes de alta velocidade. Em segundo lugar, tem sido observado que distribuições empíricas (histograma) das amostras de certas séries de tráfego são tipicamente não-gaussianas, ou seja, elas inclinam-se para a direita, o que pode ser indicativo de presença de não-linearidades no fenômeno observado (HEYMAN et al., 1992; MELAMED; SENGUPTA, 1992).

3.5 Processos Auto-Similares

O termo *auto-similar* foi introduzido por Mandelbrot (1965). Ele e seus colaboradores trouxeram processos auto-similares para atenção da estatística, principalmente através de aplicações em áreas como Hidrologia e Geofísica.

Estudos realizados por Leland et al. (1994, 1995) revelam que o tráfego de pacotes em redes locais (*Local Area Network*, LAN) é auto-similar, com graus diferentes de auto-similaridade que depende da carga na rede. Desta forma então, modelos matemáticos que capturam auto-similaridade são aplicados a uma grande variedade de tráfego de redes. Ledesma & Liu (2000) e Willinger et al. (1996) reúnem uma lista de referências nas quais os modelos auto-similares de tráfego são aplicados com sucesso.

Uma consequência do trabalho de Leland et al. (1994) é a nova forma como o tráfego pode ser classificado: tráfego com dependência de longa duração e tráfego com dependência de curta duração. Na Estatística, processos com dependência de longa duração são também chamados de processos com memória longa. Willinger et al. (1997) procura também fazer uma análise abrangente do tráfego de redes LAN, provendo uma explicação plausível para a ocorrência de auto-similaridade neste tipo de tráfego.

Outra importante bibliografia datada do início da década passada, tal como a de Leland, é o trabalho de Paxson & Floyd (1995) que estuda dados de redes de longa distância (*Wide Area Network*, WAN) e mostra que a modelagem de tráfego, baseada em processos de Poisson, falha em capturar a dependência de longa duração presentes no

tráfego deste tipo de rede. É demonstrado neste trabalho que os processos da chegada de Poisson são completamente limitados para modelar as explosividades/rajadas (*burstiness*) típicas deste tipo de tráfego, especialmente quando muitas fontes são multiplexadas juntas. Como consequência mostra-se que o tráfego WAN é muito mais explosivo, em diferentes escalas de tempo, do que as previsões do modelo de Poisson.

Crovella & Bestavros (1997) observa que o tráfego devido as transferências pela rede mundial de computadores (*World Wide Web*, WWW) apresenta também características que são consistentes com a auto-similaridade. Também é demonstrada neste trabalho a presença da auto-similaridade em superposições de fontes de tráfego ON/OFF; ou seja, superposições de fontes em que são consideradas ON quando pacotes são enviados para rede e OFF quando a rede não é utilizada. Outra importante presença de auto-similaridade em tráfego de redes dá-se através de tráfego de dados de frames gerados por codificadores de vídeo com taxa de bit variável, VBR (BERAN et al., 1995; HEYMAN; LAKSHMAN, 1996).

Do exposto acima, percebe-se que inúmeros trabalhos atestam que diversas modalidades de tráfego de rede apresentam características auto-similares com memória longa, ou seja, seu comportamento é basicamente em rajadas cobrindo uma ampla faixa de escalas de tempo. Portanto, é essencial que os modelos de tráfego capturem as características de natureza auto-similar para que se possa gerar indicadores de desempenho de redes ou fazer previsões confiáveis de seu desempenho. Uma das principais implicações, que esta modelagem mais adequada traz, consiste no desenvolvimento de algoritmos que possam ser utilizados para evitar o congestionamento de tráfego de dados (SILVA et al., 2001). Caso os modelos tradicionais sejam utilizados, o atraso ou perda de pacotes são consideráveis.

3.5.1 Descrição de Processos com Auto-Similaridade

Leland et al. (1994) argumenta que a auto-similaridade e a estimação de parâmetros estatísticos de série temporais na presença de dependência a longo prazo tornam-se cada vez mais comuns em vários campos da ciência. Com esta necessidade iminente, a intenção desta subseção é fornecer algumas informações básicas sobre auto-similaridade, processos auto-similares e estimativas de uma métrica de análise conhecida como **Parâmetro de Hurst**, denotado por H .

3.5.1.1 Definições e Propriedades

Seja $\mathcal{X} = (X_t : t = (0, 1, 2, 3, \dots))$ um processo estocástico fracamente estacionário com média μ , variância σ^2 e função de autocorrelação $r(k), k > 0$. Em particular, assume-se que \mathcal{X} tem uma FAC com a seguinte forma genérica

$$r(k) \sim k^{-\beta} L(k), \quad k \rightarrow \infty, \quad (3.15)$$

em que $0 < \beta < 1$ e L varia lentamente para o infinito. Por simplicidade, assume-se que L é assintoticamente constante. Para cada $m = 1, 2, 3, \dots$ seja $\mathcal{X}^{(m)} = (X_k^{(m)} : k = 1, 2, 3, \dots)$ uma nova série temporal fracamente estacionária (com FAC denotada por $r^{(m)}$), obtida calculando a média da série original \mathcal{X} sobre segmentos não superpostos de comprimento m . Ou seja, para cada $m = 1, 2, 3, \dots, X^{(m)}$ é dado por

$$X_k^{(m)} = \frac{X_{(k-1)m} + \dots + X_{km-1}}{m}, \quad (3.16)$$

para $k \geq 1$. Processos gerados de acordo com a Equação (3.16) são chamados de **processos agregados** (*aggregated processes*), sendo bastante utilizados para avaliar as propriedades estatísticas de uma dada série temporal em diferentes escalas de tempo, de acordo com o valor de m escolhido.

De posse da Equação (3.16) pode-se fazer as seguintes definições:

Definição 1 - Um processo \mathcal{X} é chamado exatamente auto-similar de segunda ordem com parâmetro de auto-similaridade $H = 1 - \beta/2$ se, para todo $m = 1, 2, \dots$, tem-se que

$$\text{var}(X^{(m)}) = \sigma^2 m^{-\beta} \quad \text{e} \quad r^{(m)}(k) = r(k), \quad (3.17)$$

para todo $k \geq 0$.

Definição 2 - X é chamada assintoticamente auto-similar de segunda ordem com parâmetro de auto-similaridade $H = 1 - \beta/2$ se, para todo k bastante grande, tem-se que

$$r^{(m)}(k) \rightarrow r(k), \quad m \rightarrow 0, \quad (3.18)$$

com $r(k)$ dado pela Equação (3.15).

Em outras palavras, o processo \mathcal{X} é exatamente ou assintoticamente auto-similar de segunda ordem se os correspondentes processos agregados $\mathcal{X}^{(m)}$, $m = 1, 2, \dots$, são os mesmos que X ou tornam-se indistinguível deste, pelo menos com respeito a suas funções de autocorrelação.

Matematicamente, as principais propriedades dos processos auto-similares são as seguintes:

Decaimento Lento da Variância - A variância da média aritmética dos processos agregados decai lentamente com o recíproco do tamanho da amostra, isto é, $\text{var}(X^{(m)}) \sim am^{-\beta}$, com $m \rightarrow \infty$ e com $0 < \beta < 1$.

Dependência de Longa Duração - A função de autocorrelação decai hiperbolicamente e não exponencialmente rápida em processos auto-similares, implicando em uma FAC não-somável¹ $\sum_k r(k) = \infty$, isto é, $r(k)$ está de acordo com a Equação (3.15). Em palavras, processos auto-similares apresentam autocorrelações não-nulas mesmo para grandes valores de k . Esta característica juntamente com a anterior é uma das mais enfáticas dos processos auto-similares e sempre são encontradas nas referências sobre o assunto (ERRAMILI et al., 1994; ERRAMILI; WILLINGER, 1993).

Ruído 1/f - A função densidade espectral $f(\cdot)$ obedece uma lei de potência próximo à origem, isto é, $f(\lambda) \sim a\lambda^{-\gamma}$, com $\lambda \rightarrow 0$, com $0 < \gamma < 1$ e $\gamma = 1 - \beta$. Esta é uma manifestação do domínio da frequência da dependência de longa duração.

Distribuição de Caudas Pesadas - Seja X uma variável aleatória com função de densidade f_X e função de distribuição acumulada F_X . X tem distribuição de caudas pesadas ou X segue uma distribuição de caudas pesadas se

$$P(X > x) \sim x^{-a}, \quad x \rightarrow \infty, \quad 0 < a < 2. \quad (3.19)$$

Portanto, uma distribuição deste tipo possui probabilidade diferente de zero para grande valores de x , isto é, valores relevantes (pesos) para uma grande faixa de probabilidade, explicando o uso do termo “peso” neste contexto. Uma característica importante das variáveis aleatórias que têm distribuição de caudas-pesadas é que elas apresentam grande variabilidade nos valores observados (SILVA et al., 2001).

Pode-se medir o grau de auto-similaridade de um processo estocástico e observar então se ele possui dependência de longo alcance ou de curto alcance. Para tal é utilizado o parâmetro de Hurst. Esta explanação e melhores descrições matemáticas são dadas a seguir.

¹Ou não-integrável, no caso de tempo contínuo.

3.5.1.2 Parâmetro de Hurst

Historicamente, a importância dos processos auto-similares encontra-se no fato que eles fornecem uma elegante explicação e interpretação de uma lei empírica que é comumente referenciada como *efeito de Hurst*.

Para um certo conjunto de observações $(X_n : n = 1, 2, \dots, N)$ com média amostral $\bar{X}(n)$ e variância amostral $S^2(n)$, a estatística R/S (R/S statistic) é dado por

$$\frac{R(n)}{S(n)} = \frac{1}{S(n)} [\max(O, W_1, W_2, \dots, W_n) - \min(O, W_1, W_2, \dots, W_n)], \quad (3.20)$$

em que,

$$W_n = (X_1 + X_2 + \dots + X_n) - k\bar{X}(n), \quad (3.21)$$

para $k \geq 1$. Enquanto muitas séries temporais naturais parecem ser bem representadas pela relação $E[R(n)/S(n)] \sim an^H$, para $n \rightarrow \infty$, com parâmetro de Hurst H aproximadamente 0,7, observações X_n de um processo com dependência de curta duração são conhecidos por satisfazer $E[R(n)/S(n)] \sim an^{0.5}$, para $n \rightarrow \infty$. Geralmente esta discrepância é conhecida como efeito de *Hurst* ou *fenômeno de Hurst*.

A título de comparação, processos com dependência temporal de curta duração, i.e. $H = 0,5$, mostram as seguintes propriedades:

- $\text{var}(X^{(m)}) \sim a_1 m^{-1}$;
- $0 < \sum_k r(k) < \infty$;
- $f(\lambda)$ para $\lambda = 0$ é positivo e finito.

3.5.1.3 Estimação do Parâmetro de Hurst

A seguir é feita uma breve descrição de como detectar e estimar o nível de auto-similaridade presente em uma série temporal. Existem diversas técnicas para este propósito, tais como diagrama variância \times tempo, análise das estatísticas R/S , método do Periodograma e estimador de máxima verossimilhança. Estes dois últimos não são discutidos a seguir, porque, pela sua complexidade, fogem do escopo desta dissertação. Assim, são descritas a seguir as estatísticas correspondentes e ferramentas gráficas para o cálculo do parâmetro de Hurst dos métodos restantes. Esta descrição matemática pode ser encontrada em Leland et al. (1994), Silva et al. (2001), Silva (2004) e para maior aprofundamento do parâmetro de Hurst pode ser utilizado a referência Peters (1991).

Análise da estatística R/S : esta análise consiste em determinar o gráfico de $\log(R(n)/S(n))$ versus $\log(n)$, para valores espaçados de n , começando com $n \approx 10$. Quando H é bem definido, uma típica estatística de R/S começa com uma zona de transiente, representando a natureza de dependências de curto alcance dos dados, mas estabiliza-se em seguida e caminha para uma reta com uma certa declividade. Assim, a estimativa \hat{H} de H é dada por uma inclinação assintótica que pode ter valor entre $1/2$ e 1 .

Diagrama variância \times tempo: como observado anteriormente em um processo auto-similar de segunda ordem, a variância do processo agregado $X^{(m)}$, $m \geq 1$, decresce linearmente, para grandes valores elevados de m . Lembre-se que o parâmetro m define as diferentes escalas de tempo associadas ao sinal de tráfego observado. O gráfico conhecido com **diagrama variância \times tempo** (*variance-time plots*) é obtido traçando-se $\log(\text{var}(X^{(m)}))$ versus $\log(m)$ (“tempo”) e por ajustar uma reta através dos pontos resultantes no plano, através do método dos mínimos quadrados, ignorando valores pequenos de m . Valores da inclinação desta reta, correspondendo a estimativa $\hat{\beta}$, entre -1 e 0 sugerem auto-similaridade. Uma estimativa do grau de auto-similaridade é dado então por $\hat{H} = 1 - \hat{\beta}/2$. Este método possui menor complexidade dentre os vários existentes para inferir o parâmetro de Hurst.

Nas seções anteriores, é feito um apanhado geral dos vários modelos matemáticos para análise e predição de tráfego de rede de alta velocidade, que culmina na descrição das principais características estatísticas de processos auto-similares e na apresentação do parâmetro de Hurst como uma grandeza útil na determinação empírica da auto-similaridade. Em muitas situações, a capacidade preditiva de um certo modelo pode ser avaliada num primeiro momento por simulação, sendo que no caso de algoritmos desenvolvidos para predição de tráfego, por exemplo, faz-se necessário que tais algoritmos sejam avaliados usando sinais com características auto-similares. Uma das formas mais simples de se gerar séries temporais auto-similares é através de sistemas dinâmicos caóticos (ERRAMILLI et al., 1994).

3.6 Mapas Caóticos como Fonte de Tráfego

Nesta seção, é discutido o uso de mapas caóticos como geradores de processos fractais, que podem ser usados para modelar fontes de tráfego. São também apresentadas que as características do tráfego que podem ser emuladas usando mapas lineares ou não-lineares

por partes (*piecewise linear/nonlinear maps*); ou seja, mapas cuja formulação é dada por expressões diferentes dependendo da faixa de valores da variável.

Do ponto de vista da avaliação de desempenho de algoritmos de predição é importante que se tenham métodos de geração de séries temporais artificiais que exibam características similares às do tráfego medido. Erramilli et al. (1994) apresenta as principais motivações para se usar mapas caóticos determinísticos como modelos para tráfego de pacotes, destacando as seguintes:

- **Mapas caóticos possuem dinâmicas compatíveis com o comportamento do tráfego de pacotes:** dependendo da tecnologia, da aplicação, e mesmo das escalas de tempo sob consideração, o tráfego de pacotes pode ser constante, periódico ou aleatório, com correlações temporais que variam em muitas escalas de tempo. Esta riqueza de comportamentos pode ser capturada por um simples sistema dinâmico determinístico, caótico ou não. Como uma ilustração, o mapa logístico, discutido no Capítulo 2 possui diferentes comportamentos que são controlados pelo ajuste de um único parâmetro.
- **Determinismo na geração de modelos de pacotes:** existe um grau de determinismo na geração modelos de pacotes, em fontes do tráfego. Recentes estudos de medição de tráfego indicam que em curtas escalas de tempo existem complexas, quase determinísticas, correlações no tráfego de pacotes. Muita destas estruturas complexas e determinísticas podem de fato ser explicadas pela vazão de pacotes deterministicamente espaçados, e por suas superposições. Os modelos caóticos determinísticos permitem uma descrição mais sucinta destas manifestações.
- **Propriedades fractais no tráfego de pacote:** como descrito no Capítulo 2, mapas caóticos são geradores convenientes de propriedades fractais. Entretanto, resta ainda analisar os impactos destas propriedades no projeto e na engenharia de redes de pacotes.
- **Necessidade para a análise do transiente:** redes de pacote são capazes de manifestar interessante comportamento ao longo do tempo, que pode substancialmente ter impacto no desempenho. Desta forma, é importante analisar seu comportamento dinâmico. Dado um mapa caótico que seja um modelo razoável do tráfego, é possível modelar o comportamento dinâmico da rede de pacotes pelo estudo de um sistema de equações não-lineares determinísticas.

- **Experiência bem sucedida em outras áreas:** modelos caóticos foram propostos, como alternativas viáveis aos modelos estocásticos, em outros domínios de aplicação e disciplinas.

Considere um mapa unidimensional em que a variável de estado $x(n)$ evolui sobre o tempo de acordo com o mapa não-linear

$$x(n+1) = \begin{cases} f_1(x(n)), & 0 \leq x(n) < d \\ f_2(x(n)), & d \leq x(n) \leq 1 \end{cases} \quad (3.22)$$

em que $0 < d < 1$ é uma constante. Note que esta formulação é completamente determinística, e uma condição inicial dada define inteiramente um trajetória no espaço de fase. Isto é análogo a uma *realização* de um processo estocástico.

Pode-se agora modelar um processo de geração de pacote supondo que a fonte está em um estado ativo ou inativo no tempo n , dependendo de o valor de $x(n)$ estar abaixo ou acima de um certo limiar d . Cada iteração do mapa no estado ativo corresponde à geração de um pacote (ou um grupo dos pacotes). O processo da chegada do pacote é descrito então pela evolução de um variável indicadora associada, $y(n)$, que pode ser definida como

$$y(n) = \begin{cases} 0, & 0 \leq x(n) < d \\ 1, & d \leq x(n) \leq 1 \end{cases} \quad (3.23)$$

Na prática, os valores de y_n correspondem ao que é observado, enquanto que a dinâmica de $x(n)$ é omitida. O desafio é encontrar $f_1(\cdot)$ e $f_2(\cdot)$ tal que $y(n)$ marque as propriedades do pacote de tráfego real que são relevantes para o desempenho das filas. A seguir são apresentados dois mapas que possuem o formato descrito na Equação (3.22).

3.6.1 Mapa Intermitente Simples

Erramilli et al. (1994) usa o mapa intermitente simples extensivamente para modelar tráfego de pacotes, que conforme é discutido anteriormente, possui características típicas de processos fractais, que processos estocásticos convencionais não possuem. Esta classe de mapas é formulada por partes, consistindo em dinâmicas lineares e não-lineares,

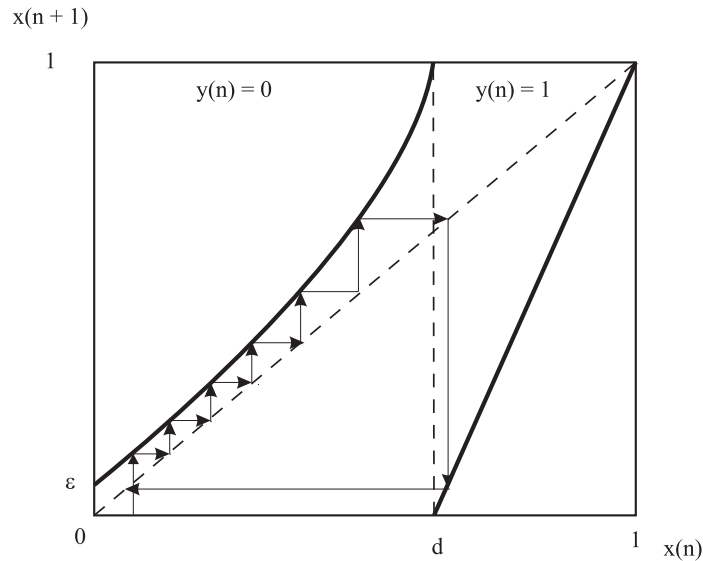


Figura 3.3: dinâmica do mapa intermitente simples.

dependendo do valor de $x(n)$

$$x(n+1) = \begin{cases} \varepsilon + x(n) + cx(n)^m, & 0 \leq x(n) \leq d \\ \frac{x(n) - d}{1 - d}, & d \leq x(n) \leq 1 \end{cases} \quad (3.24)$$

$$\text{tal que, } c = \frac{1 - \varepsilon - d}{d^m},$$

em que $0 < \varepsilon \ll 1$ é uma constante, cujo efeito é limitar a duração máxima dos períodos inativos. Este mapa é conhecido como mapa intermitente, porque ele é relacionado a vários mapas usados para modelar um fenômeno conhecido como **intermitência**, comum no estudo da turbulência (SCHUSTER, 1988). Tais fenômenos são caracterizados por alternância entre períodos com longas fases regulares e períodos relativamente curtos com “explosões” (rajadas) irregulares. A Figura 3.3 ilustra o comportamento qualitativo da dinâmica do mapa intermitente simples. Esta Figura é conhecida como “Teia de aranha” e tem o objetivo de determinar, graficamente, sucessivas interações de um mapa.

O período inativo ($0 \leq x(n) < d$) é representado por um mapa não-linear. Este mapa modela o comportamento de longa duração dos períodos inativos que são relacionados a muitas das propriedades fractais observadas em dados reais. O vasto leque de escalas de tempo, observado em dados reais, pode ser capturada pelo mapa intermitente escolhendo ε , tal que seja, muito menor que d . Se o estado inicial $x(0)$ começa em algum período passivo e se não estiver perto da origem, o período passivo resultante é relativamente curto. Entretanto, $x(0)$ ficando perto da origem, a variável de estado evolue muito lentamente (isto é, $x_{n+1} \approx x_n + \varepsilon$), fazendo-se necessárias muitas iterações do mapa para sair desta

região.

O período ativo ($d < x(n) < 1$) é representado por um mapa linear. Os períodos de tempo no estado inativo são processos de caudas pesadas com variância infinita quando $3/2 < m < 2$. Erramilli et al. (1994) mostra que o sinal de tráfego gerado pelo mapa intermitente simples possui ruído $1/f$ e é assintoticamente auto-similar de segunda ordem. O parâmetro de Hurst é dado por $H = (3m - 4)/2m - 2$ e varia de $(1/2 - 1)$ quando m está limitado em $(3/2 < m < 2)$.

3.6.2 Mapa Intermitente Duplo

O período ativo pode ser projetado de modo a também possuir a propriedade de caudas pesadas, bastando para isto trocar a expressão do segmento linear do mapa intermitente simples por um segmento não-linear apropriado, ou seja,

$$x(n-1) = \begin{cases} \varepsilon_1 + x(n) + c_1 x(n)^m, & 0 \leq x(n) < d \\ \varepsilon_2 + x(n) + c_2 (1-x(n))^m, & d \leq x(n) \leq 1 \end{cases} \quad (3.25)$$

$$\text{tal que, } c_1 = \frac{1 - \varepsilon_1 - d}{d^m}, \quad c_2 = \frac{1 - \varepsilon_2 - d}{(1-d)^m}.$$

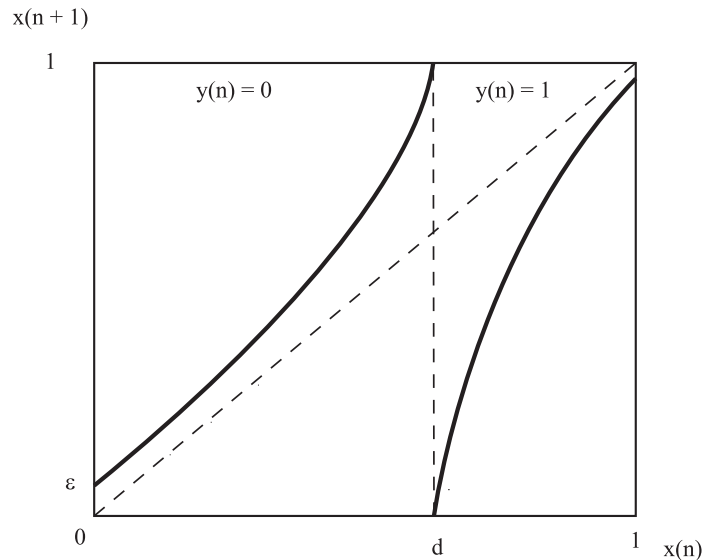


Figura 3.4: dinâmica do mapa intermitente duplo.

A Figura 3.4 ilustra o comportamento qualitativo da dinâmica do mapa intermitente duplo. Neste caso, o parâmetro de Hurst é dado por $H = (3m - 4)/(2m - 2)$ para m no intervalo $(3/2 - 2)$. Este mapa é uma boa representação do do comportamento

ON/OFF verificado nos estudos preliminares de Leland et al. (1994). Conclui-se esta seção resumindo em poucas palavras que tanto o mapa intermitente simples, quanto o intermitente duplo, simulam tráfego que é assintoticamente auto-similar no sentido de que o sinal produzido exhibe ruído $1/f$.

3.7 Conclusão

Neste capítulo foi mostrado que o tráfego de modernas redes de alta velocidade não segue a modelagem tradicionalmente adotada na literatura. O atual tráfego de redes possui uma variância explosiva em várias escalas de tempo, dependências de longo alcance e propriedade fractais. Estas características se devem ao fenômeno de auto-similaridade presentes nos tráfegos VBR e WWW e nas redes WAN e LAN. Vários trabalhos são discutidos e todos são unânimes em mostrar que o comportamento do tráfego agregado em redes de alta velocidade não pode ser modelado de forma convencional.

Foi também apresentada uma importante medida para aferir o grau de auto-similaridade, parâmetro de Hurst, usando estatísticas conhecidas. Esta ferramenta veio se somar com as diversas encontradas no Capítulo 2 para a caracterização de sistemas dinâmicos. Uma técnica importante proposta por Erramilli et al. (1994) para geração de tráfego em rajadas é a que utiliza mapas caóticos. Nesta modelagem é capturada as principais características do tráfego real, com parâmetro de Hurst condizente com sistemas auto-similares e com dependência de longa duração.

No próximo capítulo, as arquiteturas de redes neurais mais utilizadas na análise de séries temporais são apresentadas. O objetivo é utilizar tais arquiteturas em problemas de modelagem e predição de sinais que possuam características complexas (não-linearidades, memórias longas e auto-similaridade) e que são comumente encontradas em séries temporais reais, tais como as séries de tráfego.

4 REDES NEURAI SUPERVISIONADAS DINÂMICAS

4.1 Introdução

Este capítulo tem por objetivo apresentar sucintamente as arquiteturas de redes neurais avaliadas nesta dissertação, a fim de facilitar a compreensão dos métodos de modelagem de predição que são propostos nos capítulos seguintes.

De forma geral, redes neurais artificiais podem ser divididas quanto ao tipo de aprendizado em duas categorias: *(i)* redes com aprendizado supervisionado e *(ii)* redes com aprendizado não-supervisionado. No caso supervisionado, cada entrada apresentada à rede vem acompanhada de uma saída desejada, a fim de permitir uma modificação dos parâmetros ajustáveis em função do erro entre a resposta fornecida pela rede e a saída real desejada. Ao final da etapa de ajuste dos parâmetros, chamada genericamente de treinamento, as respostas da rede para todas as entradas devem ser próximas das saídas desejadas. No caso não-supervisionado, a rede neural detecta padrões e características estatísticas do espaço de entrada, de forma a construir uma representação de dimensionalidade reduzida do mesmo no conjunto de pesos sinápticos de seus neurônios.

As arquiteturas de redes neurais descritas neste capítulo são todas de aprendizado supervisionado, diferenciando umas das outras apenas pelo modo que processam informação temporal, ou seja, se utilizam ou não laços de realimentação (*feedback loops*). Desta forma, arquiteturas supervisionadas que não contenham tais laços, comumente chamadas de *não-recorrentes* ou *feedforward*, são discutidas em primeiro lugar. Em seguida, arquiteturas contendo laços de realimentação, doravante chamadas de *redes neurais recorrentes* são apresentadas. As descrições das arquiteturas apresentadas neste capítulo são baseadas principalmente nos livros de Principe et al. (2000), Hertz et al. (1991) e Haykin (1994). Referências adicionais são citadas quando necessárias.

4.2 Redes Neurais Estáticas

Redes neurais não-recorrentes são as mais populares e de maior uso em aplicações práticas, devido ao seu comprovado desempenho em tarefas de aproximação de funções e classificação de padrões, fruto da combinação de propriedades computacionais importantes, tais como não-linearidade, capacidade de aprendizado e generalização.

Como toda rede neural supervisionada, redes não-recorrentes necessitam de uma fonte externa que forneça informação sobre o problema em questão. Esta informação é fornecida através de um conjunto de N pares de vetores $\{\mathbf{x}(n), \mathbf{d}(n)\}$, $n = 1, 2, \dots, N$, em que $\mathbf{x}(n) \in \mathbb{R}^{(p+1)}$ simboliza o vetor de entrada no instante de tempo discreto n e $\mathbf{d}(n) \in \mathbb{R}^m$ denota o vetor de saídas (respostas) desejadas para aquele vetor de entrada.

Cada vetor de entrada é representado como

$$\mathbf{x}(n) = \begin{pmatrix} x_0(n) \\ x_1(n) \\ \vdots \\ x_p(n) \end{pmatrix} = \begin{pmatrix} -1 \\ x_1(n) \\ \vdots \\ x_p(n) \end{pmatrix}, \quad (4.1)$$

em que $p > 0$ denota o número efetivo de variáveis de entrada usadas no problema. O termo “efetivo” é usado aqui porque a componente de entrada $x_0(n) = -1$ não é propriamente uma variável no sentido usual, sendo mantida fixa com o objetivo de permitir uma formulação única para o ajuste dos limiares (*bias*) de ativação dos neurônios nas redes supervisionadas. De modo semelhante, o vetor de saída no instante n é representado da seguinte forma

$$\mathbf{d}(n) = \begin{pmatrix} d_1(n) \\ \vdots \\ d_m(n) \end{pmatrix}, \quad (4.2)$$

em que $m > 0$ indica o número de variáveis de saída da rede neural. Uma componente qualquer do vetor de entrada é simbolizada por $x_j(n) \in \mathbf{R}$, enquanto uma componente qualquer do vetor de saída é simbolizada como $d_k(n) \in \mathbf{R}$.

Para um dado problema de interesse, considera-se que os vetores, $\mathbf{x}(n)$ e $\mathbf{d}(n)$ estão relacionados segundo alguma relação matemática desconhecida $\mathbf{F}(\cdot)$,

$$\mathbf{d}(n) = \mathbf{F}[\mathbf{x}(n)], \quad (4.3)$$

sendo que o objetivo principal do problema é lançar mão de alguma ferramenta matemática

que possa emular o comportamento de $\mathbf{F}[\cdot]$, com base apenas nos pares de vetores $\{\mathbf{x}(n), \mathbf{d}(n)\}$ disponíveis.

Para isto pode-se utilizar uma rede neural supervisionada e não-recorrente para gerar um modelo matemático que atue como uma aproximação do mapeamento $\mathbf{F}[\cdot]$, denotada por $\hat{\mathbf{F}}[\cdot]$

$$\mathbf{y}(n) = \hat{\mathbf{F}}[\mathbf{x}(n)], \quad (4.4)$$

em que se espera que a saída gerada pela rede neural $\mathbf{y}(n)$ seja muito próxima da saída desejada $\mathbf{d}(n)$.

Redes neurais *feedforward* são aproximadores universais de funções (CYBENKO, 1989; HORNIK, 1991; HORNIK et al., 1989), ou seja, são capazes de aproximar mapeamentos entrada-saída não-lineares, tais como aqueles genericamente descritos pela Equação (4.4), com grau de precisão arbitrário, sejam tais mapeamentos contínuos ou descontínuos. Esta propriedade é uma das responsáveis pela ampla popularização do uso de redes neurais artificiais em tarefas de reconhecimento de padrões e aproximação de funções. Assim, vale destacar que a formulação geral do problema de aprendizado de uma rede neural se aplica a arquiteturas de redes neurais envolvidas tanto em tarefas de aproximação de funções, quanto em tarefas de classificação de padrões.

As arquiteturas de redes neurais a serem descritas neste capítulo são aplicadas unicamente em problemas de predição e modelagem de séries temporais, problema este caracterizado com uma modalidade de problema de aproximação de função. Na próxima seção é apresentada uma das mais utilizadas arquiteturas de redes neurais não-recorrentes, e que também é usada como base para a construção de grande parte das arquiteturas recorrentes mais conhecidas.

4.2.1 Rede Perceptron Multicamadas

Tipicamente, uma rede Perceptron Multicamada (*Multilayer Perceptron*, MLP) é constituída de uma camada de entrada que recebe os sinais, uma ou mais camadas intermediárias, compostas por neurônios somadores com função de ativação não-linear e uma camada de saída, também composta por neurônios somadores, embora estes possam ter funções de ativação lineares.

As camadas intermediárias são comumente chamadas de camadas escondidas, visto que os neurônios nelas localizados não têm acesso direto aos sinais da entrada nem da saída. A existência de camadas escondidas não-lineares confere à rede MLP o poder com-

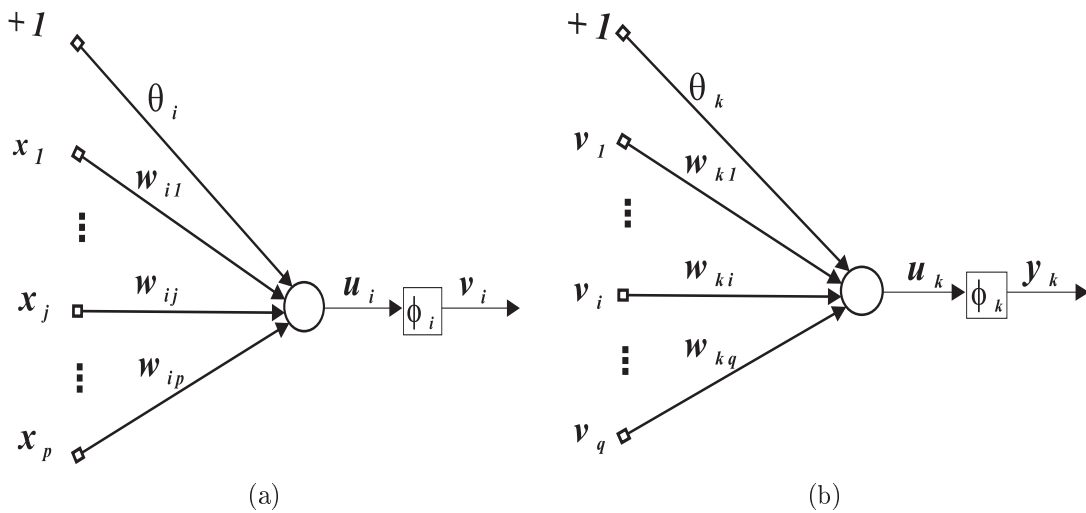


Figura 4.1: (a) neurônios da camada escondida; (b) neurônios de saída.

putacional de resolver problemas complexos, pois tais camadas têm a função de promover sucessivas alterações na representação dos dados originais até que o problema possa ser resolvido pela última camada de neurônios (camada de saída).

Outra característica da rede MLP é seu alto grau de conectividade, determinado pelas sinapses da rede, interligações entre os neurônios de diferentes camadas, em que cada uma delas está associada a um valor numérico chamado de peso sináptico. A Figura 4.1 mostra a arquitetura geral de uma rede MLP de uma única camada escondida. Uma vez especificado o número de camadas e a quantidade de neurônios em cada uma delas, o processo de aprendizado da rede MLP é realizada através do ajuste dos pesos sinápticos e limiares de ativação por meio do algoritmo de retropropagação do erro (*Error Backpropagation*). Este algoritmo de treinamento é detalhado a seguir.

4.2.1.1 Algoritmo de Retropropagação do Erro

O vetor de pesos associado ao i -ésimo neurônio da camada escondida é representado como

$$\mathbf{w}_i(n) = \begin{pmatrix} w_{i0}(n) \\ \vdots \\ w_{ip}(n) \end{pmatrix} = \begin{pmatrix} \theta_i(n) \\ \vdots \\ w_{ip}(n) \end{pmatrix}, \quad (4.5)$$

em que w_{ij} é o peso sináptico conectado a j -ésima entrada ao i -ésimo neurônio da camada escondida e $\theta_i(n)$ é o limiar (*threshold*) associado ao neurônio i . Os neurônios desta camada são chamados de neurônios escondidos por não terem acesso direto à saída da rede MLP, onde são calculados os erros de aproximação.

De modo semelhante, o vetor de pesos associado ao k -ésimo neurônio da camada de saída é representado da seguinte forma

$$\mathbf{m}_k(n) = \begin{pmatrix} m_{k0}(n) \\ \vdots \\ m_{kq}(n) \end{pmatrix} = \begin{pmatrix} \theta_k(n) \\ \vdots \\ m_{kq}(n) \end{pmatrix}, \quad (4.6)$$

na qual m_{ki} é o peso sináptico conectando o i -ésimo neurônio da camada escondida ao k -ésimo neurônio da camada de saída e $\theta_k(n)$ é o limiar associado ao neurônio de saída k . O número de neurônios da camada escondida é denotado por q_1 , $q_1 \geq 2$.

Para cada vetor de entrada apresentado à entrada da rede MLP no instante n , o ajuste dos parâmetros se dá em duas fases: uma direta e outra reversa.

Sentido Direto: esta etapa de funcionamento do algoritmo *backpropagation* envolve o cálculo das ativações e saídas de todos os neurônios da camada escondida e de todos os neurônios da camada de saída. Assim, o fluxo de sinais (informação) se dá dos neurônios de entrada para os neurônios de saída, passando obviamente pelos neurônios da camada escondida. Por isso, diz-se que a informação está se propagando no sentido direto, ou seja,

Entrada \rightarrow Camada Intermediária \rightarrow Camada de Saída.

Assim, após a apresentação de um vetor de entrada \mathbf{x} , na iteração n , o primeiro passo é calcular as ativações dos neurônios da camada escondida

$$u_i(n) = \sum_{j=0}^p w_{ij}(n)x_j(n) = \mathbf{w}_i^T(n)\mathbf{x}(n), \quad i = 1, \dots, q_1, \quad (4.7)$$

em que T denota a operação de transposição dos vetores e q_1 indica o número de neurônios da camada escondida. Em seguida, as saídas correspondentes são calculadas por meio das seguintes equações

$$v_i(n) = \phi[u_i(n)] = \phi \left[\sum_{j=0}^p w_{ij}(n)x_j(n) \right] = \phi [\mathbf{w}_i^T(n)\mathbf{x}(n)], \quad (4.8)$$

em que para este trabalho $\phi(\cdot)$ é definida pela função *tangente hiperbólica*:

$$\phi[u_i(n)] = \frac{1 - \exp[-u_i(n)]}{1 + \exp[-u_i(n)]}. \quad (4.9)$$

O segundo passo consiste em repetir as operações das Equações (4.7) e (4.8) para os neurônios da camada de saída, ou seja

$$u_k(n) = \sum_{i=0}^{q_1} m_{ki}(n)v_i(n), \quad k = 1, \dots, m, \quad (4.10)$$

na qual $m \geq 1$ é o número de neurônios de saída. Em seguida, as saídas dos neurônios da última camada são calculadas pela seguinte equação

$$y_k(n) = \phi[u_k(n)] = \phi \left[\sum_{i=0}^{q_1} m_{ki}(n)v_i(n) \right], \quad k = 1, \dots, m, \quad (4.11)$$

tal que a função de ativação $\phi(\cdot)$ assume a forma definida na Equação (4.9).

Sentido Reverso: Esta etapa de funcionamento do algoritmo *backpropagation* envolve o cálculo de gradientes locais e o ajuste dos pesos de todos os neurônios da camada escondida e da camada de saída. Assim, o fluxo de informação se dá dos neurônios de saída para os neurônios da camada escondida. Por isso, diz-se que a informação está se propagando no sentido reverso, ou seja,

Camada de Saída \rightarrow Camada Escondida.

Assim, após os cálculos das ativações e saídas na fase direta, o primeiro passo da fase reversa consiste em calcular os gradientes locais $\delta_k(n)$ dos neurônios da camada de saída

$$\delta_k(n) = e_k(n)\phi'[u_k(n)]. \quad (4.12)$$

em que $e_k(n)$ é o erro entre a saída desejada $d_k(n)$ para o k -ésimo neurônio da camada de saída e a resposta gerada por ele, $y_k(n)$:

$$e_k(n) = d_k(n) - y_k(n). \quad (4.13)$$

A derivada $\phi'[u_k(n)]$ da função tangente hiperbólica, requerida na Equação (4.12), é dada por

$$\phi'[u_k(n)] = \frac{1}{2} [1 - y_k^2(n)]. \quad (4.14)$$

O segundo passo da fase reversa consiste em calcular os gradientes locais $\delta_i(n)$, dos neurônios da camada escondida

$$\delta_i(n) = \phi'[u_i(n)] \sum_{k=1}^m m_{ki}\delta_k(n), \quad i = 1, \dots, q_1, \quad (4.15)$$

tal que a derivada $\phi'[u_i(n)]$ é calculada através da Equação (4.14).

O terceiro passo da fase reversa corresponde ao processo de atualização ou ajuste dos parâmetros (pesos sinápticos e limiares) da rede MLP com uma camada escondida. Assim, a regra de atualização dos pesos, w_{ij} , que correspondem aos pesos entre a entrada e a camada de saída, é dada por

$$w_{ij}(n+1) = w_{ij}(n) + \eta \delta_i(n) x_j(n), \quad (4.16)$$

em que η é a taxa de aprendizagem. E para os pesos que ligam a camada escondida com a de saída, tem-se que a regra de atualização é dada por

$$m_{ki}(n+1) = m_{ki}(n) + \eta \delta_k(n) y_i(n). \quad (4.17)$$

O algoritmo de retropropagação do erro é descrito acima para uma rede MLP com uma única camada de neurônios escondidos, mas o mesmo pode ser generalizado para redes com duas ou mais camadas escondidas sem muito esforço. Redes de uma camada escondida são capazes de aproximar com precisão arbitrária funções contínuas, enquanto redes MLP de duas ou mais camadas podem aproximar até funções descontínuas. Na prática, redes MLP com mais de duas camadas de neurônios escondidos são difíceis de encontrar.

O término do treinamento da rede MLP é, em geral, avaliado com base no valor da média do erro quadrático calculado ao final de cada época de treinamento

$$\epsilon_{med} = \frac{1}{N} \sum_{t=1}^N \epsilon(n) = \frac{1}{2N} \sum_{t=1}^N \sum_{k=1}^m e_k^2(n), \quad (4.18)$$

em que uma *época* de treinamento corresponde à apresentação de todos os pares entrada-saída disponíveis. Neste trabalho, a rede MLP é treinada por várias épocas até que um número máximo de épocas permitido seja alcançado. O gráfico de ϵ_{med} pelo número de épocas é chamado de curva de aprendizagem da rede neural.

Para avaliar o desempenho da rede treinada é importante avaliar a sua resposta a dados de entrada diferentes daqueles utilizados durante o treinamento, calculando-se o valor de ϵ_{med} para estes vetores. Na fase de teste, os pesos da rede não são ajustados. Para este fim, o procedimento mais adotado consiste em treinar a rede apenas com uma parte dos dados selecionados aleatoriamente, guardando a parte restante para ser usada para testar o desempenho da rede. Assim, ter-se-á dois conjuntos de dados, um para treinamento, de tamanho $N_1 < N$, e outro de tamanho $N_2 = N - N_1$, para o teste.

Em geral, escolhe-se N_1 tal que a razão N_1/N esteja na faixa de 0,75 a 0,90, ou seja, se $N_1/N \approx 0,75$ tem-se que 75% dos vetores de dados devem ser selecionados aleatoriamente, sem reposição, para serem utilizados durante o treinamento. Os 25% restantes são usados para testar a rede.

O valor de ϵ_{med} calculado para os dados de teste é chamado de erro médio de generalização da rede, pois testa a capacidade da mesma em extrapolar o conhecimento aprendido durante o treinamento para novos casos. É importante ressaltar que, geralmente, o erro de generalização é maior do que o erro de treinamento, pois trata-se de um novo conjunto de dados, mas seu valor deve ser suficiente baixo de modo a garantir o bom desempenho da rede MLP.

Os procedimentos de treinamento e teste são repetidos por um número K ($K \gg 1$) de vezes, a fim de se ter uma noção da variabilidade estatística das taxas de erro. Para cada bateria de treinamento e teste, os elementos que compõem os conjuntos de treinamento e teste são selecionados aleatoriamente. O valor final da taxas de acerto é dado então pela média das taxas obtidas para as K baterias. O intervalo de confiança da taxa de acerto também pode ser estimado a partir da amostra obtida para as K baterias de treinamento e teste. O mesmo procedimento é levado a cabo caso se queira ter uma noção do valor médio do erro de generalização.

Por fim, certos itens importantes para o bom funcionamento da rede MLP em tarefas de predição de séries temporais são listadas a seguir.

Dimensão do vetor de Entrada (p) - Em predição de séries temporais esta dimensão se confunde com a ordem da memória ou regressão de entrada, visto que o vetor de entrada da rede MLP é construído a partir da amostra atual da série $x(n)$ e $p - 1$ amostras passadas¹

$$\begin{aligned} \mathbf{x}(n) &= [x_0(n) \ x_1(n) \ \cdots \ x_p(n)]^T, \\ &= [-1 \ x(n) \ \cdots \ x(n-p+1)]^T, \end{aligned} \quad (4.19)$$

em que se nota que o valor mínimo permitido de p é 1. O limite superior para p está associado à ordem do sistema que gerou a série temporal. É importante ter em mente que um valor alto para p não indica necessariamente um melhor desempenho para a rede neural, pois pode haver redundância na informação provida. Em predição não-linear de séries temporais, a dimensão p está associada ao conceito de dimensão

¹Além do termo constante $x_0 = -1$, é claro!

de imersão visto no Capítulo 2 e cujo o valor pode ser determinado por diferentes métodos.

Número de camadas escondidas - Em geral, escolhe-se redes com uma ou duas camadas de neurônios escondidos. Conforme já mencionado, redes de uma camada escondida são capazes de aproximar com precisão arbitrária funções contínuas, enquanto redes MLP de duas ou mais camadas podem aproximar até funções descontínuas. O princípio da Navalha de Occam (*Occam's Razor*) sugere que se comece os testes com uma rede MLP de uma camada escondida. Caso esta não tenha produzido bons resultados, parte-se para a inclusão de uma outra camada escondida. Este é o procedimento adotado nesta pesquisa, chegando-se à conclusão de que as redes a serem utilizadas deveriam ter duas camadas escondidas.

Número de neurônios em cada camada escondida - Este ítem juntamente com o anterior definem o poder computacional da rede MLP. Um valor subótimo para o número de neurônios em cada camada escondida é geralmente encontrado por experimentação, em função da capacidade de generalização da rede. Grosso modo, esta grandeza mede o desempenho da rede neural ante situações não-previstas, ou seja, que valor de erro médio quadrático ela produz quando novos dados de entrada são apresentados. Se muitos neurônios existirem na camada escondida, a generalização é muito bom para os dados de treinamento, mas tende a ser ruim para os novos dados. Se existirem poucos neurônios, o desempenho é ruim também para os dados de treinamento. O valor ideal é aquele que permite atingir as especificações de desempenho adequadas tanto para os dados de treinamento, quanto para os novos dados.

O número de neurônios da primeira camada escondida (q_1) é fixado após alguma experimentação no valor $q_1 = 20$, que se mostra plenamente satisfatório para o propósito desta dissertação. Já para determinar o número de neurônios da segunda camada escondida (q_2) é utilizada a seguinte fórmula

$$q_2 = \sqrt{q_1}, \quad (4.20)$$

em que o valor resultante é arredondado para o maior valor inteiro mais próximo. Para o exemplo anterior, ter-se-ia $q_2 = \sqrt{20} = 4,472$, resultando em $q_2 = 5$.

Taxa de aprendizagem variável - Nas expressões de ajuste de pesos sinápticos, Equações (4.16) e (4.17), é usada uma taxa de aprendizagem variável no tempo, $\eta(n)$, que decai

linearmente a zero com o passar das iterações de treinamento

$$\eta(n) = \eta_0 \left(1 - \frac{n}{n_{max}} \right), \quad (4.21)$$

em que η_0 é o valor inicial da taxa de aprendizagem e n_{max} é o número máximo de iterações, dado por

$$n_{max} = N \times \text{Número máximo de épocas.} \quad (4.22)$$

A idéia representada na Equação 4.21 está em começar o treinamento da rede MLP com um valor alto para η (e.g. $\eta_0 \approx 0,5$), para então ir decaindo o valor de $\eta(n)$ a fim de estabilizar o processo de aprendizado (HAYKIN, 1994).

A partir da próxima seção começam a ser apresentadas arquiteturas neurais desenvolvidas a partir da rede MLP especialmente para lidar com problemas de predição não-linear de séries temporais. Estas redes são chamadas genericamente de redes neurais dinâmicas. Primeiramente são apresentadas redes dinâmicas não-recorrentes e, em seguida, descrevem-se as redes dinâmicas recorrentes.

4.3 Redes Neurais Dinâmicas Não-Recorrentes

O problema de predição não-linear de séries temporais é geralmente colocado na forma de um problema de aproximação de funções, conforme mostrado na Equação (4.4). Assim, o interesse em usar a rede MLP para predição está fundamentado justamente na capacidade de aproximação universal desta arquitetura neural. Além disso, para ter bom desempenho em tarefas de predição de séries temporais, a rede MLP deve também ser capaz de representar a dinâmica temporal (relações de causa-e-efeito) do processo não-linear que gerou a série temporal e que está implicitamente representada nesta.

Uma rede neural capaz de modelar a dinâmica de um processo, a partir de uma série temporal, é chamada de rede neural dinâmica, caso contrário, a rede é dita estática. Uma rede neural pode já ser concebida como dinâmica ou ser tornada dinâmica a partir de uma rede estática. As arquiteturas de redes neurais dinâmicas são, em sua grande maioria, extensões da rede MLP, que por sua vez é originalmente uma rede estática, ou seja, voltada para processamento de dados estáticos provenientes de sistemas sem memória.

A rede MLP e suas variantes dinâmicas são tornadas sensíveis à estrutura temporal dos sinais portadores de informação através da inclusão de mecanismos de memória de

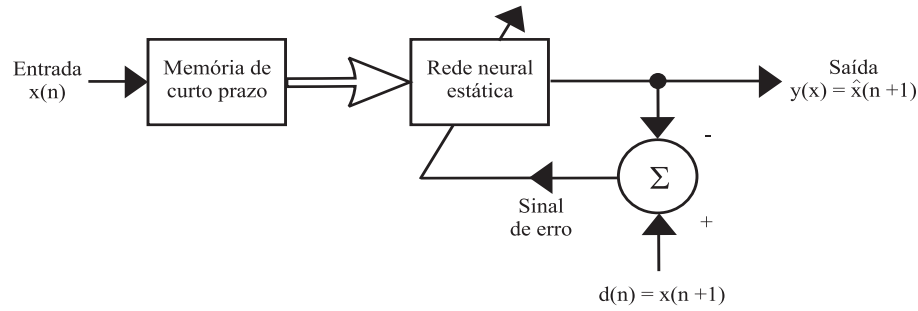


Figura 4.2: arquitetura genérica de uma rede neural dinâmica construída a partir de uma rede neural estática por meio de mecanismos externos de memória de curta duração.

curta duração (*short-term memory*, STM). Estes mecanismos são responsáveis por manter a informação temporal disponível por vários instantes de tempo, a fim de que a rede MLP possa manipulá-la e armazená-la adequadamente em seus pesos sinápticos. Uma forma simples de inserir memória de curta duração dá-se através de atrasadores (*time delays*) ou laços de realimentação (*feedback loops*), que podem ser inseridos tanto interna quanto externamente à rede.

A Figura 4.2 ilustra a arquitetura geral de uma rede neural estática com memória de curta duração externa. O uso de atrasadores como mecanismos de memória de curta duração da rede MLP dá origem às chamadas redes dinâmicas não-recorrentes, enquanto que o uso de laços de realimentação dá origem às redes dinâmicas recorrentes. A seguir são descritas algumas redes dinâmicas não-recorrentes, para em seguida redes dinâmicas recorrentes serem apresentadas.

4.3.1 Rede MLP com Atrasadores na Entrada

A Figura 4.3 ilustra como um número finito de atrasadores podem ser colocados na entrada de uma rede neural a fim de torná-la dinâmica.

A janela de tempo formada pelos atrasadores percorre toda a extensão da série temporal, a fim de converter o sinal unidimensional $\{x(n)\}_{t=1}^N$ em $N - p - 1$ vetores de dimensão $p + 1$. Estes vetores é que são apresentados na entrada da rede neural. Uma pergunta importante é como selecionar o comprimento p da janela de modo a capturar adequadamente as propriedades de uma série temporal. Neste trabalho, a janela de tempo é construída de acordo com a Equação (2.5), representando o teorema de Takens (1981), repetida abaixo para facilitar o entendimento

$$\mathbf{x}(n) = [x(n) \ x(n - \tau) \ \cdots \ x(n - (d_E - 1)\tau)]^T, \quad (4.23)$$

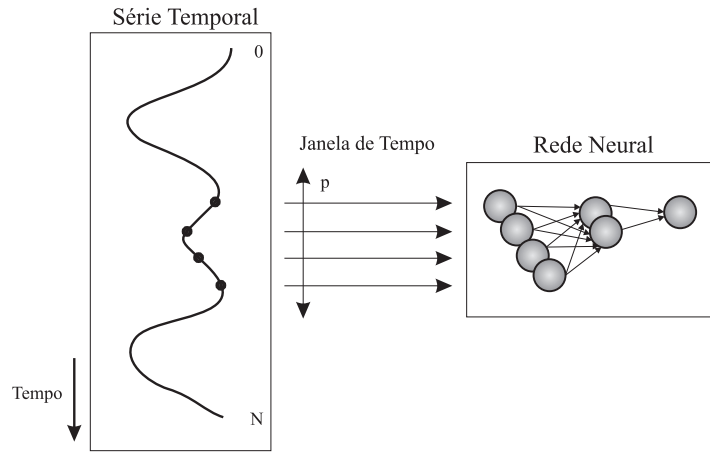


Figura 4.3: exemplo de atrasadores formando uma janela de tempo de comprimento na entrada de uma rede neural.

em que $\mathbf{x}(n)$ é um vetor que contém d_E elementos da série, contados a partir do elemento atual $x(n)$, espaçados um do outro de τ unidades de tempo. No âmbito da análise de séries temporais caóticas, o parâmetro d_E é chamado de dimensão de imersão e o parâmetro τ é chamado de atraso de imersão.

A rede neural dinâmica formada pela introdução de atrasadores na entrada de uma rede MLP estática é conhecida pela sigla FTDNN (*Focused Time Delay Neural Network*) (PRINCIPE et al., 2000), sendo aqui chamada simplesmente de Rede MLP com Atrasadores na Entrada. Assim, como a rede MLP, que lhe dá origem, a FTDNN é uma rede *feedforward* multicamadas cujos pesos sinápticos e limiares são ajustados de acordo com o algoritmo *backpropagation*.

A arquitetura de uma rede FTDNN com uma camada escondida está mostrada na Figura 4.4. Nesta figura, o vetor de entrada é definido de acordo com a Equação (4.23), sendo os parâmetros da rede FTDNN modificados a fim de minimizar o erro quadrático médio entre a saída da rede, $y(n) = \hat{x}(n + 1)$ e a resposta desejada $x(n + 1)$.

Para o tipo de problema de predição de séries temporais que se está interessado nesta dissertação, utiliza-se apenas um neurônio na saída da rede. Matematicamente, isto equivale a fazer $m = 1$ na Equação (4.11). Assim, uma vez treinada a rede FTDNN, sua saída no instante n é calculada pela seguinte expressão:

$$\begin{aligned}
 y_1(n) &= \hat{x}(n + 1) = \phi \left[\sum_{i=1}^{q_1} m_{1i} v_i(n) \right], \\
 &= \phi \left[\sum_{i=1}^{q_1} m_{1i} \phi \left(\sum_{j=0}^{d_E-1} w_{ij} x(n - j\tau) - \theta_i \right) - \theta_1 \right]. \quad (4.24)
 \end{aligned}$$

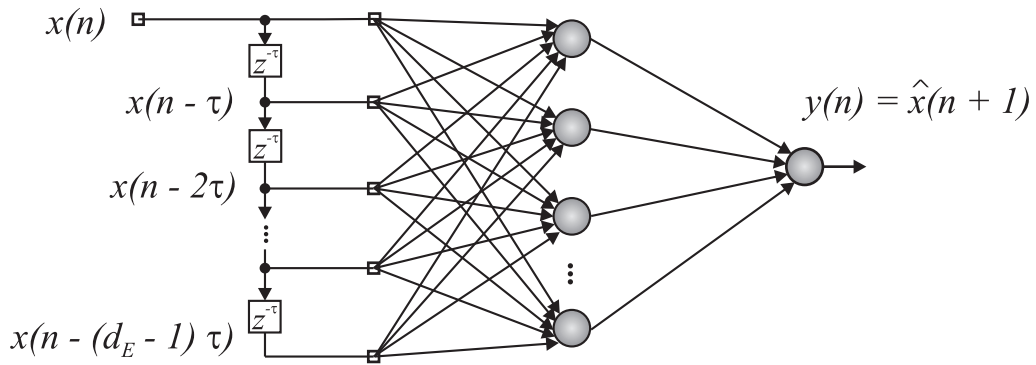


Figura 4.4: arquitetura genérica de uma rede FTDNN de uma camada escondida.

É comum encontrar na literatura, variantes dinâmicas da rede MLP que possuem não só atrasadores externos, como na rede FTDNN, mas também atrasadores internos, ou seja, inseridos dentro da arquitetura da rede. Tais atrasadores internos são colocados nas saídas dos neurônios das camadas escondidas. Este tipo de rede é denotada genericamente pela sigla TDNN (PRINCIPE et al., 2000; HAYKIN, 1994), sendo proposta inicialmente em Waibel et al. (1989). Deve-se frisar, portanto, que a rede FTDNN é um caso particular da rede TDNN em que não há atrasadores internos, apenas atrasadores externos. Daí a razão do termo *focused* na sigla FTDNN, para indicar que a memória de curta duração está “concentrada” na entrada.

4.3.2 Rede MLP com Neurônios do Tipo FIR

Uma arquitetura de rede MLP dinâmica não-recorrente com atrasadores internos é conhecida pela sigla FIR-MLP (do inglês *Finite Impulse Response Multilayer Perceptron*), proposta por Wan (1994, 1990). A rede FIR-MLP utilizada considera que cada sinapse de um neurônio é, em si, um filtro de resposta ao impulso de duração finita (FIR), conforme mostrado na Figura 4.5. Assim, cada componente $x_j(n)$ do vetor de entrada é tratada como um sinal em si, tal que em cada sinapse há disponível no instante n um número L de valores passados de $x_j(n)$. No instante n , a saída da j -ésima sinapse do neurônio i , denotada por $s_{ij}(n)$, é dada por

$$s_{ij} = \sum_{l=0}^L w_{ij}(l)x_j(n-l). \quad (4.25)$$

A Figura 4.6 ilustra como passa a ser representado um neurônio artificial cujas sinapses são modeladas como filtros FIR. Neste caso, a saída total do i -ésimo neurônio da camada

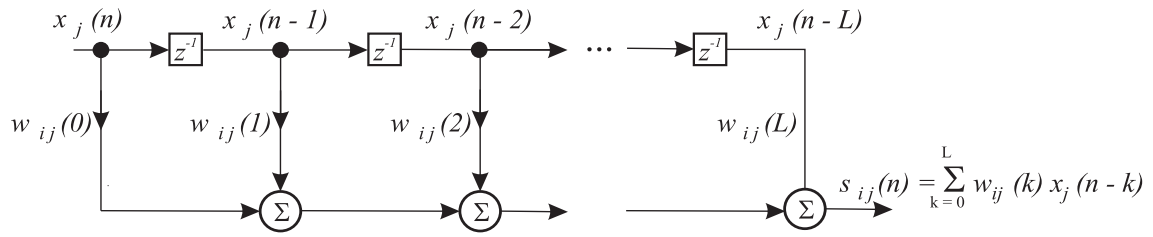


Figura 4.5: sinapse representada como um filtro FIR.

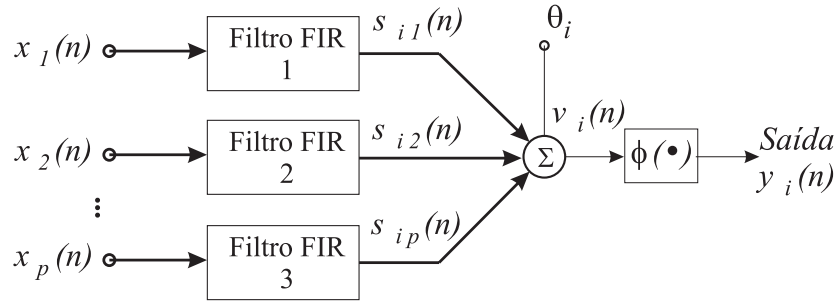


Figura 4.6: neurônio formado por sinapses do tipo FIR.

escondida passa então a ser calculada pela seguinte expressão

$$v_i(n) = \phi[u_i(n)] = \phi \left[\sum_{j=1}^p s_{ij}(n) - \theta_i \right] = \phi \left[\sum_{j=1}^p \sum_{l=0}^L w_{ij}(l) x_j(n-l) - \theta_i \right], \quad (4.26)$$

em que L é a ordem do filtro FIR, considerada a mesma para todas as sinapses. Note que se $L = 0$ na Equação (4.26), a saída do neurônio reduz-se à Equação (4.8), usada pela redes MLP e FTDNN. A saída do k -ésimo neurônio da camada de saída passa a ser expressa da seguinte forma

$$y_k(n) = \phi[u_k(n)] = \phi \left[\sum_{i=1}^{q_1} s_{ki}(n) - \theta_k \right] = \phi \left[\sum_{j=1}^{q_1} \sum_{l=0}^L m_{ki}(l) v_i(n-l) - \theta_k \right], \quad (4.27)$$

que, no caso em que $L = 0$, reduz-se à Equação (4.11).

Haykin (1994) mostra que esta rede é adequada para modelagem de sinais não-estacionários, diferentemente da rede FTDNN, mais indicada para modelagem de sistemas invariantes no tempo. A rede FIR-MLP é o algoritmo vencedor em uma importante competição sobre predição de séries temporais caóticas, cujo o resultado foi publicado originalmente em Wan (1994), sendo posteriormente republicado em Haykin (1994).

Para treinar a rede FIR-MLP é necessário uma variante temporal do algoritmo de retropropagação descrita na Seção 4.2.1.1, conhecido como algoritmo de retropropagação temporal. Este algoritmo apesar de ser mais poderoso do ponto de vista da capacidade de extrair informação dinâmica da série temporal, possui elevado custo computacional

quando comparado ao algoritmo de retropropagação padrão.

Outra dificuldade da rede FIR-MLP é a necessidade de ajustar muito mais parâmetros do que a rede FTDNN. Por exemplo, se todos os filtros existentes numa rede FIR-MLP qualquer tiverem ordem L , então o número total de parâmetros ajustáveis M desta rede é dado por

$$M = ((p \times L) + 1) \times q_1 + ((q_1 \times L) + 1) \times m, \quad (4.28)$$

enquanto que para uma rede FTDNN o valor de M é dado por

$$M = (p + 1) \times q_1 + (q_1 + 1) \times m. \quad (4.29)$$

Como exemplo hipotético, considere as seguintes constantes $p = 4$, $q_1 = 10$ e $m = 1$. Assim, uma rede FIR-MLP com $L = 5$, terá $M = 210 + 51 = 261$ parâmetros ajustáveis, enquanto uma rede FTDNN terá apenas $M = 50 + 11 = 61$ parâmetros.

A partir da próxima seção serão apresentadas arquiteturas de redes neurais dinâmicas recorrentes. Tais redes diferem das redes dinâmicas apresentadas até o presente momento por apresentar laços de realimentação internos ou externos, também chamados laços locais ou globais, respectivamente.

4.4 Redes Neurais Dinâmicas Recorrentes

Uma rede neural dinâmica recorrente, ou simplesmente rede recorrente, é aquela que contém conexões sinápticas realimentadas (ou laços de realimentação), permitindo o fluxo de sinais de ativação e saída neurais entre neurônios de camadas distintas, entre neurônios de uma mesma camada, ou ainda de um neurônio para ele mesmo.

Assim como os atrasadores, a recorrência é um tipo de mecanismo de memória de curta duração que permite a rede lembrar informações de um passado recente. A diferença básica entre estes tipos de memória é que, enquanto os atrasadores disponibilizam no instante atual os valores exatos da informação passada, os laços de realimentação realizam algum tipo de processamento (filtragem) sobre informação passada.

Redes neurais recorrentes constituem uma das mais importantes famílias de arquiteturas de redes neurais e, conseqüentemente, um grande número de algoritmos são desenvolvidos com o passar dos anos (NARENDRA; PARTHASARATHY, 1990; HERTZ et al., 1991; HAYKIN, 1994; HORNE; GILES, 1995; TSOI; BACK, 1997; PRINCIPE et al., 2000). Tsoi & Back (1997) discutem e listam como diversas arquiteturas neurais recorrentes podem

ser geradas pelas mais variadas combinações de realimentações entre neurônios de uma mesma camada e entre neurônios de diferentes camadas, de tal forma que pode-se facilmente entender a razão da grande diversidade de arquiteturas recorrentes encontrada na literatura.

Pode-se apresentar o modelo de redes neurais dinâmicas recorrentes sob a forma de equações de variáveis de estado. Assim, considerando um caso especial de uma rede neural em que um vetor $\mathbf{x}(n)$ p -por-1 represente o vetor de entrada e o vetor $\mathbf{v}(n)$ q -por-1 represente a saída da camada oculta no tempo n , pode-se então descrever o comportamento dinâmico do modelo de redes dinâmicas recorrentes pelo par acoplado:

$$\mathbf{v}(n+1) = \phi(\mathbf{v}(n), \mathbf{x}(n)), \quad (4.30)$$

$$\mathbf{y}(n) = \phi(\mathbf{v}(n)), \quad (4.31)$$

onde $\phi(\cdot)$ é uma função não-linear que caracteriza a camada oculta e a camada de saída (HAYKIN, 1994). Embora essa seja uma representação de redes neurais dinâmicas recorrentes, no decorrer desta dissertação, a representação escolhida será a do mapeamento entrada-saída, isto é, a que já vinha sendo utilizada nas redes dinâmicas.

De um ponto de vista prático, é importante procurar uma resposta à seguinte pergunta: que rede neural dinâmica deve ser usada para tratar com o complexo problema de predição e modelagem não-linear de séries temporais? Recorrente ou não-recorrente? Infelizmente², não há resposta fácil a esta pergunta. A arquitetura apropriada é dependente do problema e já que várias co-existem na literatura, resta ao usuário experimentar um bom número delas, antes de encontrar a(s) arquitetura(s) apropriada(s). Esta postura é adotada nesta dissertação, em que o desempenho de diversas redes neurais dinâmicas, recorrentes ou não, são testadas no problema supracitado.

4.4.1 Tipos de Conexão de Realimentação

Uma conexão sináptica é definida como uma ligação entre dois neurônios quaisquer. Existem dois tipos maiores de conexões, a saber: conexão de alimentação direta (*feed-forward*) e conexão de realimentação (*feedback*). A conexão de alimentação direta ocorre quando um sinal tem orientação da entrada para a saída. Em contraste, a conexão de realimentação tem orientação da saída para a entrada. Desta forma, as conexões reali-

²Ou felizmente, para quem gosta de diversidade!

mentam para um dada camada (ou parte dela) sinais de ativação/saída produzidos por neurônios de outras camadas.

Um modo de classificar redes recorrentes consiste em verificar a extensão espacial das conexões de realimentação existentes, ou seja, se ela envolve apenas os neurônios de uma única camada ou se envolve neurônios de outras camadas. Pode-se enquadrar esta definição em três grupos.

- **Conexão Recorrente Local:** este tipo de conexão envolve apenas um neurônio. Neste caso, o termo local refere-se ao fato de a saída do neurônio ser realimentada para a entrada deste mesmo neurônio. É importante salientar que não é possível ter uma conexão local de alimentação direta. As conexões de alimentação direta devem necessariamente envolver dois neurônios diferentes.
- **Conexão Recorrente Global:** este tipo de conexão acontece entre um neurônio de uma camada para um neurônio de uma camada anterior, ou seja, um sinal de saída de um neurônio é realimentado para a entrada de um outro neurônio localizado em uma camada anterior.
- **Conexão Recorrente Não-Local:** Está é um tipo especial de conexão global, visto que envolve neurônios distintos, porém a conexão é estabelecida entre neurônios de uma mesma camada. Assim, a saída de um neurônio de uma certa camada é realimentada para a entrada de um outro neurônio da mesma camada.

Tendo em vista todas as possíveis conexões que podem ocorrer em redes recorrentes, é fácil perceber a grande variedade de arquiteturas que podem ser formadas pela combinação de tipos diferentes de conexões e com o número de camada de neurônios. Neste trabalho, o problema de predição e modelagem não-linear de séries temporais será também analisado lançando-se mão de redes dinâmicas recorrentes. A seguir são descritas as arquiteturas com recorrências globais e locais estudadas nesta dissertação.

4.4.2 Redes Recorrentes Simples

Assim como as redes FTDNN e FIR-MLP, uma grande parcela das redes recorrentes de maior utilização são extensões da rede MLP convencional. Não fugindo a esta regra, as duas arquiteturas com recorrência global a serem descritas a seguir são bastante utilizadas na prática, sendo obtidas facilmente a partir da rede MLP. Vale ressaltar que, como os pesos das conexões de realimentação não são ajustáveis, pode-se usar o algoritmo

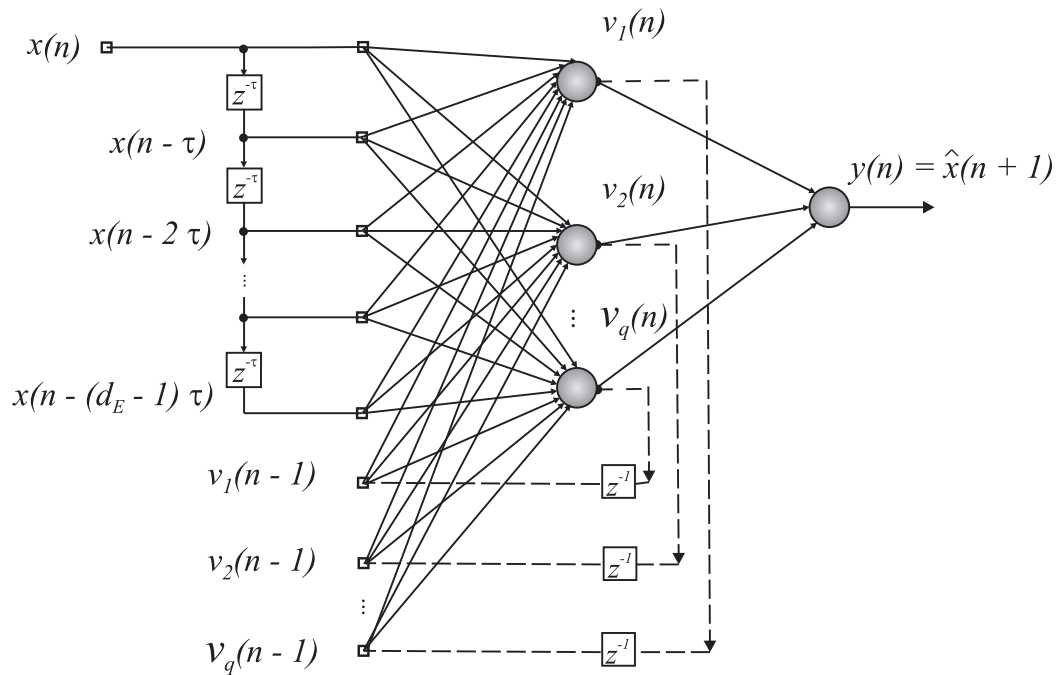


Figura 4.7: arquitetura da rede de Elman aplicada ao problema de previsão não-linear de séries temporais.

backpropagation padrão para treinar tais redes. A este tipo de rede recorrente dá-se o nome de redes recorrentes simples (PRINCIPE et al., 2000; HAYKIN, 1994; HERTZ et al., 1991).

4.4.2.1 Rede Recorrente de Elman

Esta arquitetura recorrente é proposta por Elman (1990), sendo obtida a partir da rede MLP através da redefinição da camada de entrada da rede, que passa a ser dividida em duas partes. A primeira parte corresponde ao vetor de entrada propriamente dito, conforme definido na Equação (4.23). A segunda parte, chamada de unidades de contexto, consiste na cópia das saídas dos neurônios da camada escondida no instante $n - 1$. O termo “cópia”, na verdade, é implementado computacionalmente através de um conjunto de conexões de realimentação com pesos fixos e iguais a 1. Desta forma, os valores exatos das ativações dos neurônios da camada escondida no instante $n - 1$ são utilizados pelas unidades de contexto para compor o vetor de entrada no instante n . A Figura 4.7 ilustra uma rede recorrente de Elman com uma camada escondida. A entrada e a saída da rede estão definidas de acordo com o problema de previsão de séries temporais.

Se uma rede MLP tem duas camadas escondidas pode-se escolher de qual camada escondida realimentar as ativações neurais a fim de gerar uma rede recorrente de Elman. Três opções são possíveis: (i) realimentar as ativações da primeira camada escondida para

as unidades de contexto, (ii) realimentar as ativações da segunda camada escondida para as unidades de contexto, ou (iii) realimentar as ativações de ambas as camadas escondidas para as unidades de contexto. Após experimentação, observou-se melhores resultados para a segunda opção. Dessa forma, utiliza-se nessa dissertação apenas as ativações da segunda camada escondida que são realimentadas para as unidades de contexto.

Com relação à quantidade de unidades de contexto, esta depende diretamente de qual das opções listadas no parágrafo anterior for escolhida. No primeiro caso, a quantidade de unidades de contexto é igual ao número de neurônios da primeira camada escondida (q_1). No segundo caso, a quantidade de unidades de contexto é igual ao número de neurônios da segunda camada escondida (q_2). No terceiro e último caso, a quantidade de unidades de contexto é igual a $q_1 + q_2$. Todas as outras conexões da rede de Elman são ajustáveis e do tipo *feedforward*, de tal forma que esta arquitetura pode ser treinada pelo algoritmo *backpropagation*. Do ponto de vista das conexões, a rede de Elman possui apenas recorrências globais, sendo a maior parte de suas conexões sinápticas do tipo *feedforward*.

As ativações dos neurônios da primeira camada escondida da rede de Elman são calculadas por

$$u_i(n) = \sum_{j=0}^p w_{ij}(n)x_j(n) + \sum_{l=1}^{q_1} w_{il}(n)v_l(n-1), \quad i = 1, \dots, q_1, \quad (4.32)$$

tal que a saída dos mesmos é dada por $v_i(n) = \phi[u_i(n)]$. As ativações e as saídas dos neurônios da última camada são calculadas como nas Equações (4.10) e (4.11). Durante o treinamento os pesos w_{ij} e w_{il} são ajustados segundo as regras do algoritmo *backpropagation*.

Para o cálculo do número de parâmetros da rede recorrente de Elman com realimentação das ativações da primeira camada escondida para as unidades de contexto temos:

$$M = ((p_1 + q_1 + 1) \times q_1) + ((q_1 + 1) \times m). \quad (4.33)$$

Como exemplo hipotético, considere as seguintes constantes $p = 4$, $q_1 = 10$ e $m = 1$. Assim, uma rede de Elman, terá $M = 150 + 11 = 161$ parâmetros ajustáveis.

4.4.2.2 Rede Recorrente de Jordan

A rede de Jordan (1986) é outra arquitetura recorrente clássica, sendo inicialmente usada para reconhecimento de seqüências temporais. Assim como a rede de Elman, a rede

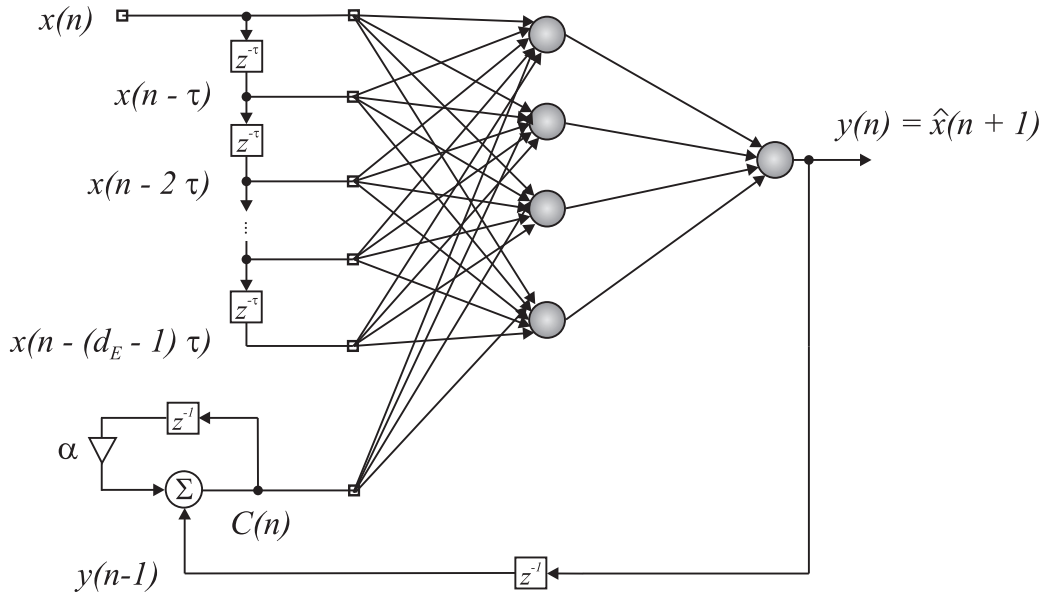


Figura 4.8: arquitetura da rede de Jordan aplicada ao problema de previsão não-linear de séries temporais.

de Jordan também não possui recorrência entre neurônios da mesma camada, sendo por isso enquadrada entre as redes globalmente recorrentes.

Em vez de realimentar as ativações dos neurônios da camada escondida, a rede de Jordan envolve conexões de realimentação dos neurônios da camada de saída para as unidades de contexto. Além disso, este tipo de rede recorrente possui auto-conexões ou auto-realimentações, em que a saída de uma unidade de contexto é realimentada para sua entrada. A Figura 4.8 ilustra uma rede recorrente de Jordan com uma camada escondida. A entrada e a saída da rede estão definidas de acordo com o problema de previsão de séries temporais.

A saída da k -ésima unidade de contexto no instante n , denotada por $C_k(n)$, é dada pela seguinte expressão

$$C_k(n) = \alpha C_k(n-1) + y_k(n-1), \quad (4.34)$$

em que $y_k(n)$ é a resposta do k -ésimo neurônio de saída, calculada como na Equação (4.11), e $0 < \alpha < 1$ é chamado de coeficiente de auto-realimentação. Se a saída $y_k(n)$ for fixada, então C_k decai exponencialmente para $y_k(n)/(1-\alpha)$, esquecendo assim gradualmente valores passados de C_k .

Desta forma, a ativação do i -ésimo neurônio da camada escondida é dada por

$$u_i(n) = \sum_{j=0}^p w_{ij}(n)x_j(n) + \sum_{k=1}^m w_{ik}(n)C_k(n), \quad i = 1, \dots, q_1, \quad (4.35)$$

tal que a saída dos mesmos é dada por $v_i(n) = \phi[u_i(n)]$. As ativações e as saídas dos neurônios da última camada são calculadas como nas Equações (4.10) e (4.11). Durante o treinamento os pesos w_{ij} e w_{il} são ajustados segundo as regras do algoritmo *backpropagation*.

Para o cálculo do número de parâmetros da rede recorrente de Jordan temos:

$$M = ((p_1 + 1 + m) \times q_1) + ((q_1 + 1) \times m). \quad (4.36)$$

Como exemplo hipotético, considere as seguintes constantes $p = 4$, $q_1 = 10$ e $m = 1$. Assim, uma rede de Jordan, terá $M = 60 + 11 = 71$ parâmetros ajustáveis.

As redes dinâmicas descritas até agora, recorrentes ou não, são relativamente fáceis de aplicar ao problema de predição e modelagem não-linear de séries temporais. Para este fim, basta definir o vetor de entradas como na Equação (4.23) e definir uma rede com um único neurônio de saída ($m = 1$) cuja saída desejada durante o treinamento é dado pelo próximo valor da série, ou seja, $d(n) = x(n + 1)$. Durante o teste, a saída da rede fornece uma estimativa do próximo valor da série, ou seja, $y(n) = \hat{x}(n + 1)$. A seguir é descrita uma rede dinâmica, que pode funcionar de modo recorrente ou não, que foi originalmente proposta para lidar não com predição/modelagem não-linear de séries temporais, mas sim com problemas um pouco mais gerais, genericamente chamados de identificação de sistemas não-lineares.

4.5 Rede Dinâmica NARX

Uma importante e útil classe de sistemas não-lineares de tempo discreto é matematicamente representada pelo modelo NARX (*Nonlinear AutoRegressive model with exogenous inputs*) (LEONTARITIS; BILLINGS, 1985; NORGAARD et al., 2000)

$$y(n) = f[y(n - 1), \dots, y(n - d_y); u(n), u(n - 1), \dots, u(n - d_u + 1)], \quad (4.37)$$

em que $u(n) \in \mathbb{R}$ e $y(n) \in \mathbb{R}$ representam, respectivamente, a entrada e a saída do modelo no instante n , enquanto $d_u > 0$ e $d_y > 0$, $d_u \leq d_y$, são as ordens de memória de entrada e memória de saída.

A função $f(\cdot)$ é uma função não-linear, geralmente desconhecida. Quando esta função é aproximada por uma rede MLP, a topologia resultante é chamada de *rede recorrente NARX* (CHEN et al., 1990; NARENDRA; PARTHASARATHY, 1990), constituindo uma importante classe de arquiteturas neurais dinâmicas computacionalmente equivalentes à

máquina de Turing (SIEGELMANN et al., 1997). A Figura 4.9 mostra uma rede NARX com uma camada escondida. É importante notar que a rede NARX possui atrasadores em sua entrada e um laço de realimentação global.

A rede NARX é treinada e utilizada em um dos seguintes modos de operação (NARENDRA; PARTHASARATHY, 1990):

- **Modo de Identificação Paralelo** - Neste caso, também chamado de modo recorrente, a saída estimada é realimentada e incluída na saída do regressor, ou seja

$$\hat{y}(n) = \hat{f}[\hat{y}(n-1), \dots, \hat{y}(n-d_y); u(n), u(n-1), \dots, u(n-d_u+1)]. \quad (4.38)$$

- **Modo de Identificação Série-Paralelo** - Neste caso, também chamado de modo não-recorrente, a saída do regressor é formada somente por valores atuais da saída do sistema, ou seja

$$\hat{y}(n) = \hat{f}[y(n-1), \dots, y(n-d_y); u(n), u(n-1), \dots, u(n-d_u+1)]. \quad (4.39)$$

É interessante notar que o caminho de realimentação mostrado na Figura 4.9 é apresentado somente no Modo de Identificação Paralelo. Como uma ferramenta para identificação de sistemas não-lineares, a rede NARX vem sendo aplicada com sucesso em uma ampla gama de problemas de modelagem entrada-saída, tal como trocadores de calor, plantas de tratamento da água servidas, sistemas de transformação catalítica em uma refinaria do petróleo e em predição de série temporais não-lineares (ver referências em (LIN et al., 1998)).

A rede NARX não é comumente utilizada na tarefa de predição e modelagem de séries temporais, que é a principal motivação desta dissertação. Na revisão bibliográfica desenvolvida nesta dissertação é encontrado o emprego das redes NARX nesta tarefa apenas no trabalho de Lin et al. (1997). Contudo, quando a rede NARX é aplicada a este tipo de problema, a ordem da memória da saída é feita $d_y = 0$, reduzindo assim a rede NARX a uma rede FTDNN convencional:

$$y(n) = f[u(n), u(n-1), \dots, u(n-d_u+1)]. \quad (4.40)$$

A formulação da rede NARX como uma rede FTDNN elimina uma porção considerável das capacidades representativas da rede NARX; isto é, toda a informação dinâmica

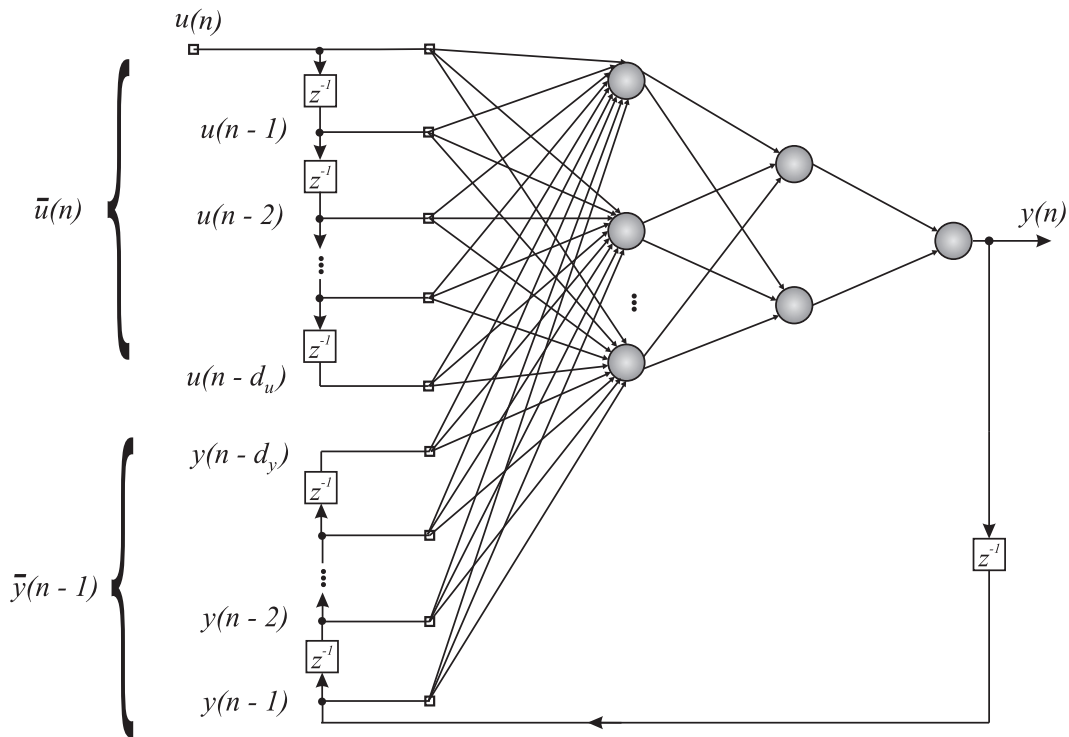


Figura 4.9: rede NARX com d_u entradas e d_y atrasos da saída.

que poderia ser aprendida das memórias passadas da saída é descartada. Para muitas aplicações práticas, tal como modelagem de tráfego de rede de computadores (GROSS-GLAUSER; BOLOT, 1998), a rede neural deve ser capaz de armazenar informação durante um período longo de tempo na presença de ruído. Bengio et al. (1994) explicaram analiticamente porque tal classe de problema é difícil de lidar com redes neurais dinâmicas que utilizam algoritmos de aprendizagem baseados no método do gradiente-descendente, tal como o algoritmo *backpropagation*.

A formulação original da rede NARX não resolve totalmente o problema de dependências temporais de longa duração, mas é demonstrado que ela tem freqüentemente um desempenho muito melhor que as RNAs recorrentes padrões nesta classe de problemas, alcançando uma convergência mais rápida e um melhor desempenho de generalização (LIN et al., 1996). No entanto, se a memória de saída é plenamente descartada, como na Equação (4.40), não há garantia de que estas propriedades sejam observadas.

Considerando este uso limitado das potencialidades da rede NARX em tarefas de predição de série temporais não-lineares, propõe-se nesta dissertação uma nova estratégia para permitir que as capacidades computacionais da rede NARX possam ser plenamente exploradas em tarefas de predição e modelagem não-linear de série temporais.

4.5.1 Rede NARX para Predição de Séries Temporais

Apesar da possibilidade concreta de se usar a rede FTDNN em predição de séries temporais, é importante lembrar que esta rede pode ser entendida como uma versão simplificada da rede NARX, obtida pela eliminação da memória de saída. Para usar toda o poder computacional da rede NARX como uma arquitetura dinâmica para predição de série temporais, uma nova definição para a entrada e saída dos regressores é proposta a seguir.

Dada uma série temporal $\{x(n)\}_{n=1}^N$, define-se o regressor do sinal de entrada da rede NARX, denotado por $\mathbf{U}(n)$, como na Equação (4.23). Assim, tem-se que

$$\begin{aligned}\mathbf{U}(n) &= [u(n), u(n-1), \dots, u(n-d_u+1)], \\ &= [x(n), x(n-\tau), \dots, x(n-(d_E-1)\tau)],\end{aligned}\quad (4.41)$$

em que se percebe que o regressor $\mathbf{U}(n)$ é composto de d_E valores observados da série temporal, amostrados a cada τ unidades de tempo.

Para levar em conta a informação dinâmica fornecida pelo laço de realimentação, a rede NARX pode ser treinada e usada nos modos paralelo e série-paralelo. No modo paralelo (ou recorrente), o regressor do sinal de saída, representado por $\mathbf{Y}(n)$, é definido como segue

$$\mathbf{Y}(n-1) = [y(n-1), y(n-2), \dots, y(n-d_y)], \quad (4.42)$$

em que se percebe que o regressor de saída $\mathbf{Y}(n)$ compreende d_y saídas prévias da rede neural. Vale lembrar que, para uma rede treinada adequadamente, a saída da rede no instante n é uma estimativa do valor futuro da série, ou seja, $y(n) = \hat{x}(n+1)$. Assim, a Equação (4.42) pode ser escrita também da seguinte forma:

$$\mathbf{Y}(n-1) = [\hat{x}(n), \hat{x}(n-1), \dots, \hat{x}(n-d_y+1)]. \quad (4.43)$$

Já, no modo série-paralelo (ou não-recorrente), o regressor de saída é definido como segue

$$\mathbf{Y}(n-1) = [x(n), x(n-1), \dots, x(n-d_y+1)], \quad (4.44)$$

sendo, portanto, construído com amostras reais da série temporal de interesse. Qualquer que seja o modo de uso da rede NARX, ambas as arquiteturas implementam o seguinte mapeamento entrada-saída

$$y(n) = \hat{x}(n+1) = \hat{f}[\mathbf{Y}(n-1), \mathbf{U}(n)], \quad (4.45)$$

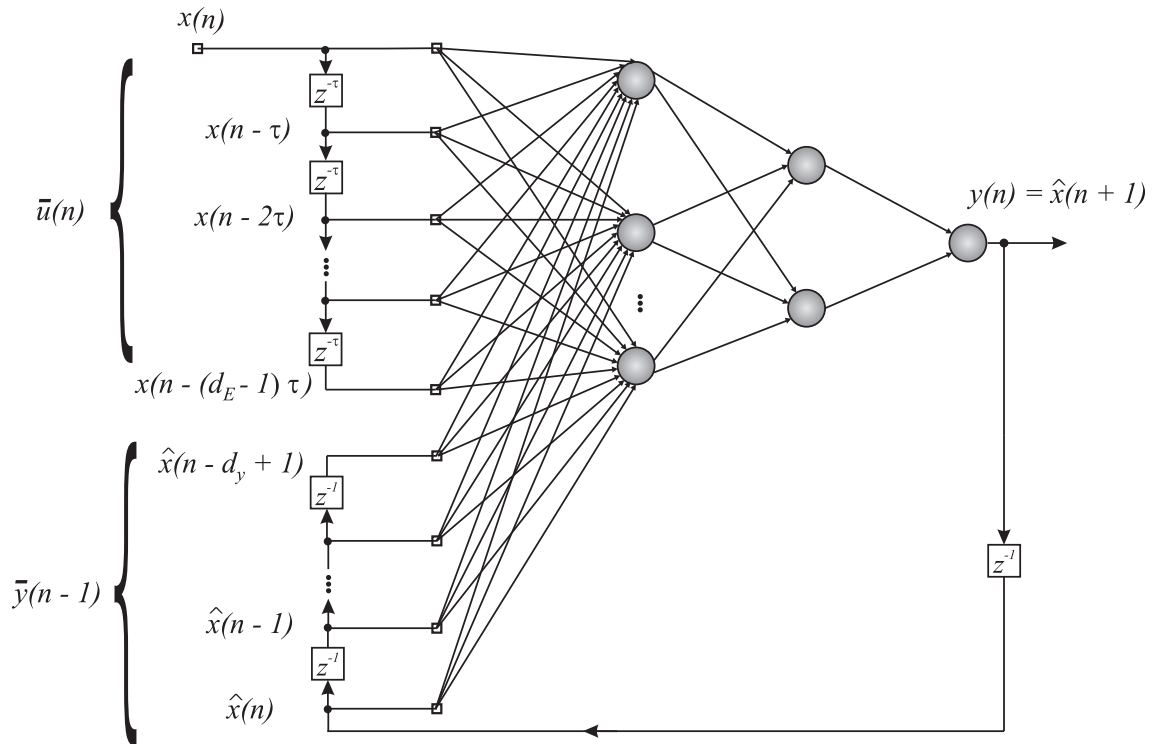


Figura 4.10: rede NARX com modo paralelo.

em que a função não-linear $\hat{f}(\cdot)$ pode ser realizada prontamente pela rede MLP padrão e treinada pelo algoritmo *backpropagation* simples.

A aproximação proposta é resumida como segue. Uma rede recorrente NARX é definida de modo que o seu regressor de entrada $\mathbf{U}(n)$ contenha d_u amostras da variável observada $x(n)$, espaçadas de $\tau > 0$ unidades de tempo, enquanto que o regressor de saída $\mathbf{Y}(n-1)$ contém valores reais ou *estimativas* da mesma variável, porém amostradas em instantes consecutivos.

Particularmente interessante é o caso do treinamento da rede NARX no modo paralelo. À medida que o treinamento da rede NARX avança, as estimativas $y(n) = \hat{x}(n+1)$ tornam-se cada vez mais próximas dos valores desejados $d(n) = x(n+1)$, indicando convergência do processo de treinamento. Assim, as saídas da rede que estão sendo realimentadas para o regressor de saída $\mathbf{Y}(n-1)$ tendem a replicar o comportamento de curto-prazo da série temporal real, pois o atraso entre as estimativas é unitário. Já o regressor de entrada fornece informação de médio/longo prazo sobre a dinâmica da série temporal, visto que o atraso τ é sempre muito maior que a unidade.

Desta forma, independente de seu modo de treinamento/uso, a rede NARX codificará em seus pesos sinápticos informação sobre a dinâmica da série temporal tanto em horizontes de curto prazo, quanto de médio ou longo prazos. Conforme será visto no próximo

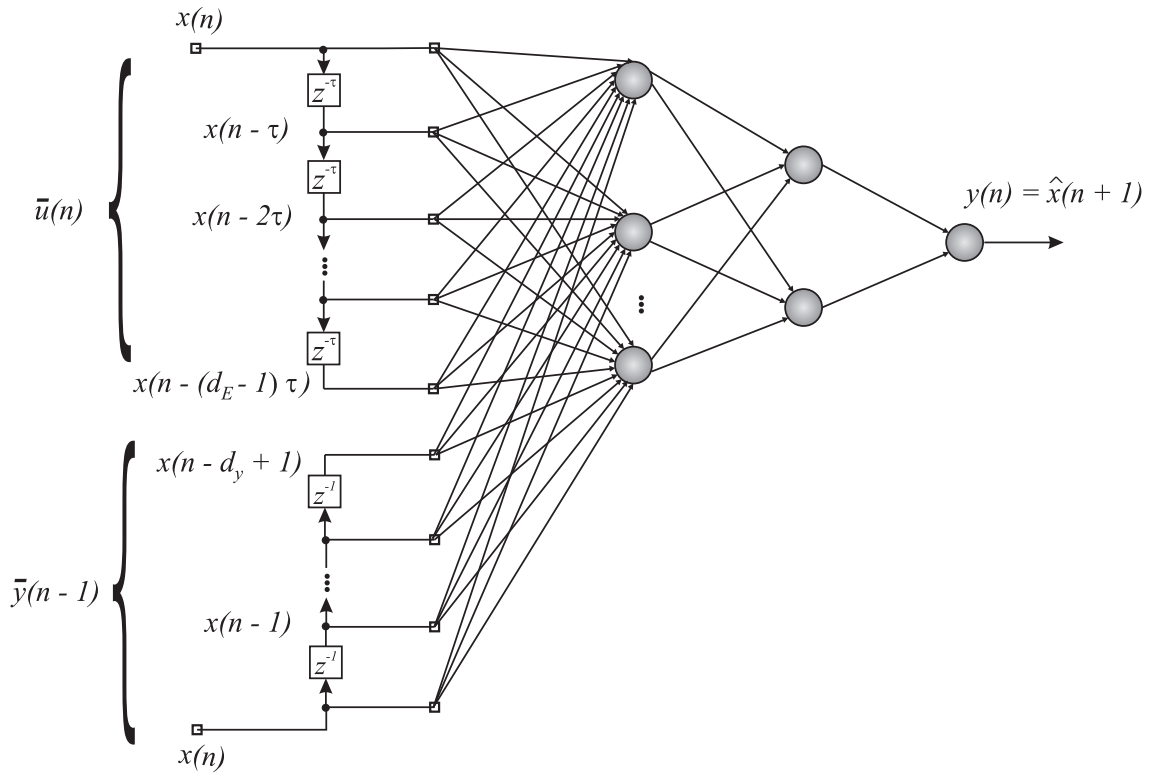


Figura 4.11: rede NARX com modo série-paralelo.

capítulo, a abordagem proposta permite que a rede NARX tenha desempenho superior ao das outras redes dinâmicas descritas neste capítulo.

Outro ponto importante a ser enfatizado é que além dos parâmetros d_E e τ da Equação (4.41), a abordagem proposta requer a especificação do parâmetro d_y . A determinação deste parâmetro é tornada automática se for lembrado que a Equação (4.23) tem uma forma alternativa sugerida por Haykin & Principe (1998)

$$\mathbf{x}(n) = [x(n), x(n-1), \dots, x(n-\tau \cdot d_E + 1)], \quad (4.46)$$

tal que ao se comparar diretamente esta equação com as Equações (4.42) e (4.43), chega-se a conclusão de que pode-se adotar sempre $d_y = \tau \cdot d_E$. Contudo, este valor é um limite superior para d_y pois o modelo NARX, expresso na Equação (4.37), exige apenas que $d_y > d_u$. Como $d_u = d_E$ no regressor de entrada $\mathbf{U}(n)$ definido na Equação (4.41), pode-se combinar estas duas restrições e estabelecer a seguinte faixa de valores para d_y

$$d_E < d_y \leq \tau \cdot d_E, \quad (4.47)$$

tal que o valor ótimo de d_y dentro desta faixa é dependente da série temporal sendo modelada.

Para o cálculo do número de parâmetros da rede NARX adotando-se $d_y = \tau \cdot d_E$ temos:

$$M = ((d_E + 1 + d_y) \times q_1) + ((q_1 + 1) \times m). \quad (4.48)$$

enquanto que para uma rede FTDNN o valor de M é dado por

$$M = (p + 1) \times q_1 + (q_1 + 1) \times m. \quad (4.49)$$

Como exemplo hipotético, considere as seguintes constantes $p = d_E = 4$, $q_1 = 10$, $m = 1$ e $\tau = 5$. Assim, uma rede NARX terá $M = 250 + 11 = 261$ parâmetros ajustáveis, enquanto uma rede FTDNN terá $M = 50 + 11 = 61$ parâmetros.

Do exposto no parágrafo anterior, a rede NARX usando abordagem proposta tem $\tau \cdot d_E$ mais pesos que a rede FTDNN, para um número fixo dos neurônios na primeira camada escondida. Entretanto, o impacto deste aumento da dimensionalidade é mínimo em termos de aumento no custo computacional. Principalmente, se for observado que outras arquiteturas dinâmicas podem ter bem mais parâmetros ajustáveis, como é o caso da rede FIR-MLP.

Além disso, no que diz respeito ao impacto do aumento da dimensão no desempenho preditivo da rede neural, pode-se argumentar que este aumento pode em princípio deteriorar o desempenho preditivo da rede neural. Este problema é comumente conhecido como maldição da dimensionalidade (*curse of dimensionality*) (HAYKIN, 1994). É comum encontrar na literatura especializada exemplos de sistemas com desempenho muito bom em problemas de pequena escala, ou seja, problemas em que a dimensão do vetor de entrada é pequena, mas que não são capazes de manter o mesmo desempenho em problemas de maior escala.

Uma análise importante, levada a cabo por Barron (1993), demonstra que os erros quadráticos médios gerados pela rede MLP em problemas de diferentes escalas é independente da dimensão do espaço de entrada, para conjuntos de treinamento com muitos dados. Neste caso, o erro decresce com o inverso do número de neurônios da camada escondida (i.e. $O(\frac{1}{q_1})$). Para efeito de comparação, este resultado é bem melhor do que o apresentado por aproximadores polinomiais, para os quais o erro decresce exponencialmente com a dimensão do espaço de entrada (i.e. $O(\frac{1}{\sqrt{N^2}})$) (PRINCIPE et al., 2000). Desta forma, conclui-se que o ganho no desempenho preditivo da abordagem obtido com o aumento da dimensão da entrada proposta justifica o esforço computacional adicional.

Outro importante destaque a respeito da aproximação proposta é relacionada com

a estabilidade da aprendizagem da rede NARX. Em Narendra & Parthasarathy (1990) é mostrado que o modo de Identificação Paralelo é muito mais instável do que o Modo Série-Paralelo. Isto se deve basicamente à incerteza propagada pelas estimativas realimentadas para o regressor de saída. Isto é um assunto importante a ser considerado somente quando se trabalha com a identificação de sistemas de entrada-saída, em que a entrada e a saída dos regressores da rede NARX são compostos de variáveis diferentes. Na abordagem proposta aqui, a instabilidade do modo de identificação paralelo é de certa forma atenuada pelo fato de o regressor de entrada e o de saída usarem a mesma variável $x(n)$ (ou estimativas dela). Desta forma, a estabilidade da aprendizagem é mantida pelo regressor de entrada $\mathbf{U}(n)$, levando a rede a convergir sempre.

4.6 Conclusão

Este capítulo apresentou sucintamente, porém de forma auto-contida, as arquiteturas de redes neurais dinâmicas avaliadas nesta dissertação. Grosso modo, estas redes podem ser classificadas em redes dinâmicas recorrentes e não-recorrentes de acordo com a presença ou não de conexões de realimentação:

Redes Dinâmicas Não-Recorrentes - Redes FTDNN, FIR-MLP e NARX (no modo de operação série-paralelo);

Redes Dinâmicas Recorrentes - Redes de Elman, Jordan e NARX (no modo de operação paralelo).

Todas estas arquiteturas, derivadas da rede MLP a partir de introdução de mecanismos de memória de curta-duração, cobrem uma parcela razoável das técnicas neurais utilizadas em problemas de modelagem de sistemas dinâmicos não-lineares.

O próximo Capítulo dedica-se à avaliação das arquiteturas aqui apresentadas em tarefas de modelagem e predição não-linear de séries temporais, usando tanto séries temporais geradas artificialmente a partir da equação dinâmica de um determinado sistema não-linear, quanto séries obtidas de sistemas reais.

5 RESULTADOS

5.1 Introdução

Após a introdução teórica feita em capítulos anteriores sobre comportamento caótico e sobre redes neurais para predição de séries temporais, este capítulo se detém na apresentação dos resultados obtidos através da aplicação destes métodos a dados reais e simulados. O desempenho de várias redes neurais dinâmicas, discutidas no capítulo anterior, é avaliado pela capacidade de predição um-passo-adiante, de vários passos-adiante e pelo desempenho na modelagem dinâmica, isto é, na tarefa de reconstrução autônoma, preservando características e invariâncias da série original. Além disso, os algoritmos de redes neurais em questão são comparados quanto à velocidade de convergência, à sensibilidade a variações de determinados parâmetros de treinamento e à capacidade de generalização.

Para a extração de tais resultados discutidos acima são utilizadas séries temporais caóticas artificiais, séries temporais caóticas reais e séries de tráfego de rede de computadores. É importante enfatizar que não são realizadas simulações exaustivas para cada uma das séries escolhidas. A intenção é demonstrar o potencial das técnicas de reconstrução do atrator e desempenho das redes NARX-P e NARX-SP. Ao longo da apresentação dos resultados, diversos pontos de discussão são abordados com respeito aos modelos propostos, para então se chegar às conclusões finais sobre o emprego dos mesmos.

O restante do Capítulo está organizado da seguinte maneira. Na Seção 5.3 são descritas as séries temporais usadas para avaliar as redes NARX-P e NARX-SP. A Seção 5.4 tem como objetivo apresentar a metodologia empregada nos testes das redes neurais e entender como os resultados são extraídos e comparados entre os diversos algoritmos. Por fim, na Seção 5.5 são apresentados os diversos testes feitos, abordando os principais pontos e aspectos dos resultados.

5.2 Estudo de Caso I - Sistemas Caóticos

Nesta seção são apresentadas as séries temporais caóticas usadas ao longo deste Capítulo. Para cada série é feita uma breve descrição do conjunto de dados que a origina e a motivação em adotá-la como exemplo. O conjunto de séries temporais utilizadas compreende séries temporais caóticas, simuladas a partir de equações diferenciais, e séries temporais caóticas reais.

5.2.1 Série Caótica de Mackey-Glass

Observações da série temporal de Mackey-Glass são produzidas por uma equação diferencial com atrasos de tempo (Δ) (MACKEY; GLASS, 1977)

$$\frac{dy(t)}{dt} = \beta y(t) + \frac{\alpha y(t - \Delta)}{1 + y^{10}(t - \Delta)}, \quad (5.1)$$

em que $y(t)$ é o valor da série temporal no instante t . Para $y(0) \in [0, \Delta]$, o sistema converge para um ponto de equilíbrio estável se $\Delta < 4,53$, para um ciclo-limite se $\Delta \in [4,53 - 13,3]$, tornando-se caótico para $\Delta > 16,8$, após uma série de duplicações de períodos para $\Delta \in [13,3 - 16,8]$. Para as simulações deste Capítulo os seguintes valores para os parâmetros são utilizados: $\alpha = 0,2$, $\beta = -0,1$ e $\Delta = 17$.

A série de Mackey-Glass é gerada a partir da discretização da Equação (5.1), usando o método de Euler (Seção 2.2). É adotado $\Delta t = 1$ e as primeiras amostras geradas são descartadas para eliminar o efeito transitório devido às condições iniciais. Um trecho contendo 500 amostras da série é apresentado na Figura 5.2(a).

A Equação (5.1) modela a dinâmica de produção de células brancas (neutrófilos) no corpo humano. Pelo fato de as taxas de proliferação destas células envolverem um atraso de tempo, dinâmicas periódicas e caos podem ser verificadas. Mackey e Glass sugeriram que flutuações de longo prazo no número de células, observadas em certas formas de leucemia, apresentam uma dinâmica semelhante à observada na Equação (5.1) (KAPLAN; GLASS, 1995).

Na Figura 5.1(b) observa-se o resultado do cálculo da dimensão de imersão estimada pelo método de Cao e o valor recomendado encontra-se no joelho da curva; isto é, quando a curva passa a ter um menor crescimento. Neste caso, o valor escolhido para uma boa reconstrução do atrator é 5. Este valor está de acordo com o teorema de imersão ($d_E \geq 2[d] + 1$), pois sendo o valor da dimensão intrínseca do atrator de Mackey-Glass

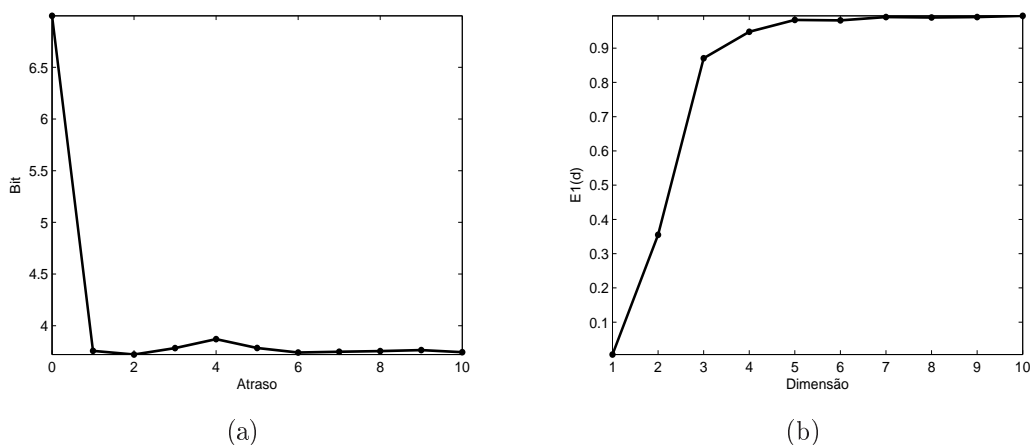


Figura 5.1: série caótica de Mackey-Glass: (a) informação mútua para o cálculo do atraso de imersão; (b) método de Cao para o cálculo da dimensão de imersão.

encontrado em Farmer (1982) igual a 1,95 para Δ igual a 17, uma condição suficiente para o valor da dimensão de imersão seria $d_E = 5$ também. Na Figura 5.1(a) encontra-se o valor igual a 2 para o atraso de imersão calculado pelo método da informação mútua. Como descrito na Seção 2.3.2, uma boa estimativa para o atraso de imersão é quando a função da informação mútua atinge o primeiro mínimo.

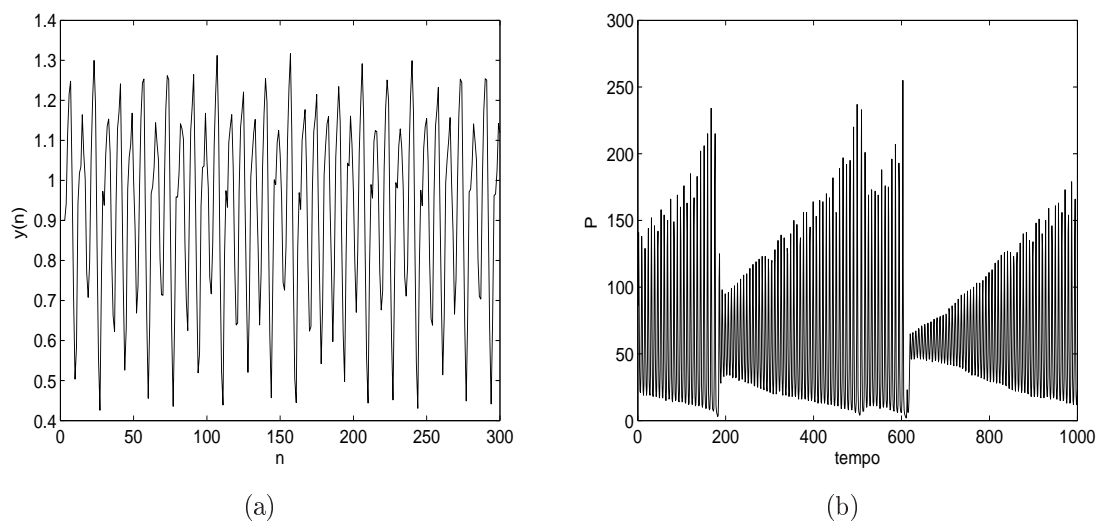


Figura 5.2: (a) série caótica de Mackey-Glass; (c) série caótica do Laser.

5.2.2 Série do Laser Caótico

Um outro exemplo de sinal caótico provém de uma seqüência de medidas da intensidade de pulsação de um laser de NH_3 infravermelho, obtida de um experimento realizado por Hübner et al. (1989). Esta série temporal foi disponibilizada inicialmente como parte

de uma competição de predição de séries temporais promovida pelo Instituto Santa Fé, ocorrida nos Estados Unidos em 1992. A Figura 5.2(b) contém um trecho contendo 2000 amostras da série caótica do Laser. Na Figura 5.3(a) determina-se o atraso de imersão igual a 2, utilizando o método da informação mútua, e na Figura 5.3(b) encontra-se a dimensão de imersão da série real caótica do Laser como sendo igual a 7.

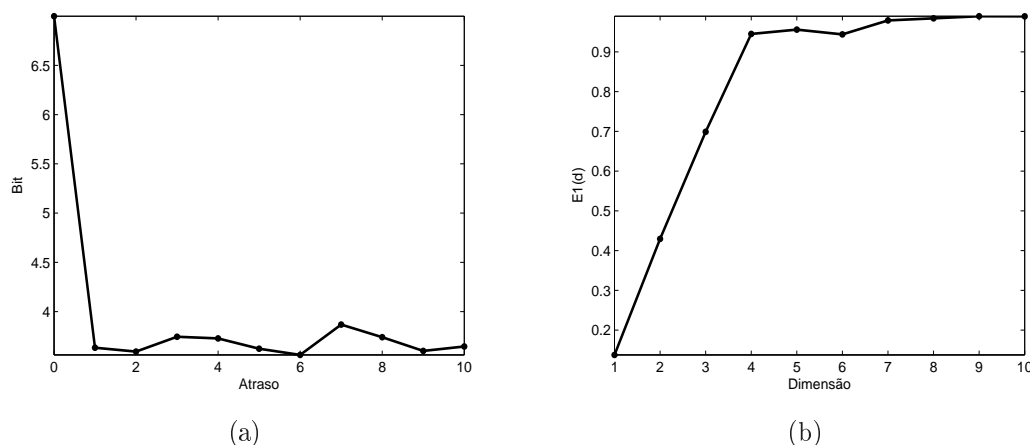


Figura 5.3: série caótica do Laser: (a) informação mútua para o cálculo do atraso de imersão; (b) método de Cao para o cálculo da dimensão de imersão.

Observando a série do Laser caótico, percebe-se que a potência de saída exibe trechos em que há oscilações regulares de amplitude crescente. Quando um valor crítico é atingido, uma instabilidade ocorre e a oscilação recomeça com uma amplitude de valor mais baixo. Devido a instabilidade, perde-se informação sobre a fase de oscilação e a amplitude se comporta de forma aparentemente imprevisível. A taxa de amostragem dos dados é tal que, para cada oscilação, oito medidas são feitas. Estas medidas são digitalizadas por um conversor A/D, tal que o erro de discretização tem amplitude igual a $\frac{1}{512}$ da faixa de variação do sinal.

5.3 Estudo de Caso II - Séries Temporais de Tráfego de Redes

Nesta seção são apresentadas as séries temporais de tráfego de redes usadas ao longo deste Capítulo. Para cada série é feita uma breve descrição de como este tráfego foi adquirido e a motivação em adotá-lo como exemplo. O conjunto de séries de tráfego utilizados compreende séries de tráfego real de uma rede local e tráfego de vídeo MPEG VBR.

5.3.1 Tráfego de Internet - Série Bellcore

A série original é constituída de dados de tráfego LAN coletados por Leland e Wilson (LELAND; WILSON, 1991). Foram gravados centenas de milhões de pacotes IP em um ambiente Ethernet sem perdas (sem levar em conta a sobrecarga do tráfego) e com *timestamps* (marcas de tempo) de gravação com exatos $100\mu s$. Os dados foram coletados entre agosto de 1989 e fevereiro 1992 na rede Ethernet do Bellcore Morris Research and Engineering Center.

São escolhidas 1.000.000 de amostras, das centenas de milhões de amostras da série original, que representa o monitoramento por 122.797,83 segundos (≈ 35 horas) do número de pacotes Ethernet externos que chegam na rede local do Bellcore Morris Research and Engineering Center, contados a partir das 23h46min de 3 de outubro 1989, em Morristown, USA. Destas 1.000.000 de amostras, optou-se em usar uma série agregada, em que os 1000 pontos desta série correspondem à média das 1000 amostras de cada um dos 1000 intervalos da série de 1.000.000 de amostras. Esta nova série é um processo agregado da série original, que possui uma escala de tempo diferente (CASTRO, 2001).

A representação gráfica da série Bellcore agregada é mostrada na Figura 5.4(a). Os valores estimados para a dimensão e atraso de imersão são, respectivamente, 10 e 8.

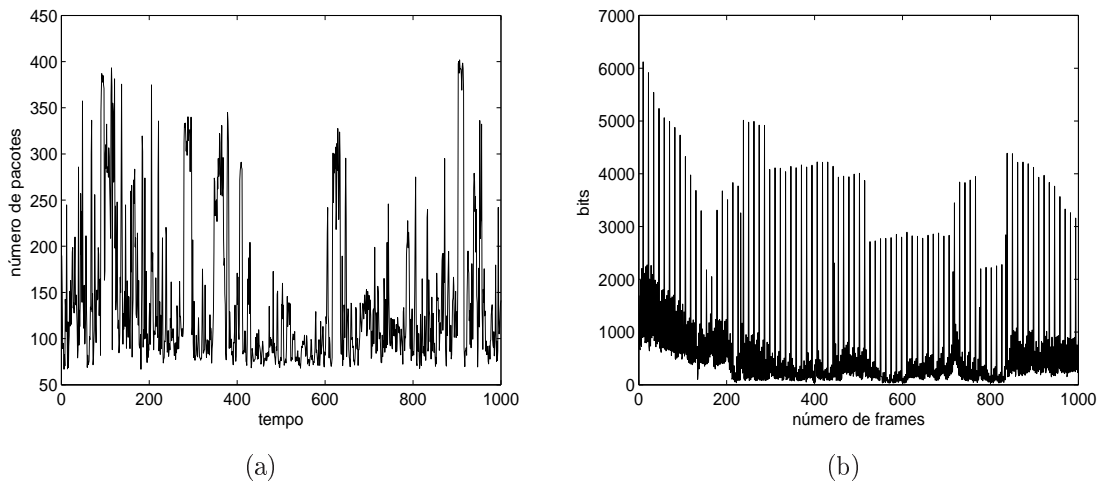


Figura 5.4: (a) série Bellcore; (b) série Tráfego de Vídeo VBR.

5.3.2 Tráfego de Vídeo MPEG VBR

Um arquivo MPEG é um arquivo digital contendo vídeo e áudio digitais codificados seguindo determinados padrões de compressão e armazenados em um dado formato especí-

fico. O MPEG tem grande relevância nas aplicações multimídia (TV digital, aplicações gráficas interativa e multimídia interativa) da indústria. Com isso, a transmissão de vídeo em redes digitais de telecomunicações está presente nas diversas classes das chamadas aplicações em banda larga.

Pesquisas em predição do tráfego de vídeo MPEG VBR visam desenvolver estratégias de gerência de rede satisfazendo requisitos de QoS. Outra motivação para estudos da predição do tráfego de rede está na importante descoberta de auto-similaridade e dependência de longo prazo em tráfego de redes de banda larga (LELAND et al., 1994). Estudos posteriores observaram que o tráfego de vídeo MPEG VBR tipicamente ocorre em rajadas em múltiplas escalas de tempo (BERAN et al., 1995; HEYMAN; LAKSHMAN, 1996), fenômeno este característico de sistemas com dependência de longo prazo e auto-similares.

Nesta dissertação, os modelos NARX-P e NARX-SP são avaliados usando série temporais de tráfego de vídeo MPEG VBR (traços), extraído do filme “Jurassic Park” (ROSE, 1995). Estes traços do tráfego de vídeo foram codificados na Universidade de Würzburg com MPEG-1. O algoritmo MPEG usa três tipos de frames diferentes: Intraframe (I), Preditivo (P) Bidirecionalmente-Preditivo (B). Estes três tipos de frames são organizados como um grupo (*Group of Pictures*, GoP) definido por uma distância L entre frames I e uma distância M entre frames P. Se o padrão cíclico de frames é {IBBPBBPBBPBBBI}, então L = 12 e M = 3. Estes valores para L e M são usados neste trabalho. Na Figura 5.4(b) são apresentados 1000 frames extraídos do filme Jurassic Park.

5.4 Metodologia de Avaliação

Como o foco desta dissertação são os problemas de predição e modelagem de séries temporais, discutem-se aqui as principais abordagens, métodos e técnicas de validação de modelos. Maiores detalhes sobre esta área de aplicação podem ser encontrados em: Box et al. (1994), Kantz & Schreiber (1997) e Aguirre (2000). No decorrer deste Capítulo procura-se mostrar a eficiência dos modelos NARX-P e NARX-SP na modelagem e predição de séries temporais de tráfego.

A capacidade de prever o comportamento futuro de uma série particular de eventos, com conhecimento apenas do seu presente e do seu passado, é uma das formas de verificar se um modelo matemático definitivamente “entendeu” os dados observados. Neste contexto, “entender” significa remover as possíveis redundâncias nos dados e, conseqüentemente, descobrir regularidades estatísticas ou dinâmicas na série apresentada. Desta

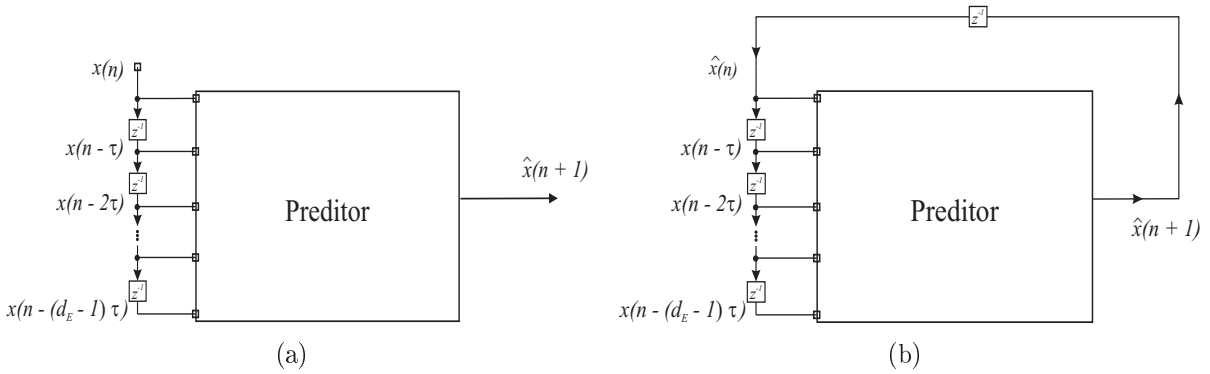


Figura 5.5: (a) preditor sem realimentação; (b) preditor recursivo, com realimentação.

forma, predição de séries temporais é um problema de processamento de sinais em que se tem uma seqüência de N amostras de uma determinada variável escalar, $\{x(n), x(n-1), \dots, x(n-N+1)\}$, uniformemente espaçadas no tempo, e cujo objetivo é obter uma estimativa $\hat{x}(n+1)$, para o próximo elemento da série. Este procedimento é conhecido como predição um-passo-adiante (UPA), em que se estima somente o próximo valor da série temporal, sem realimentação do valor predito para a entrada do regressor, conforme ilustrado na Figura 5.5(a). Em outras palavras, o regressor de entrada contém somente observações exatas da série temporal.

Quando se está interessado num horizonte de predição mais amplo, um outro método para construção de preditores é comumente utilizado, sendo conhecido como predição k-passos-adiante (KPA) ou “predição com realimentação dos valores preditos”, Figura 5.5(b). A saída do modelo deve ser realimentada para o regressor de entrada. Neste caso, os componentes do regressor de entrada, previamente compostos apenas de valores exatos da série temporal, são gradativamente trocados por valores preditos. Um exemplo hipotético para predição KPA é mostrada na Tabela 5.1.

Se o horizonte de predição tende ao infinito, em algum momento no tempo, a entrada do regressor começa a ser composto somente de valores previamente estimados da série temporal. Neste caso, a tarefa de predição KPA torna-se uma tarefa de modelagem dinâmica, em que o modelo com Redes Neurais atua como um sistema autônomo (HAYKIN; PRINCIPE, 1998).

Predição KPA e a modelagem dinâmica são mais complexas de se trabalhar do que a predição UPA, e acredita-se que estas são tarefas em que redes neurais desempenham uma importante função, em particular arquiteturas neurais recorrentes (PRINCIPE et al., 2000).

Como métrica de avaliação do desempenho em tarefa de predição e modelagem de

Tabela 5.1: Predição k-passos-adiante

Instante	Regressor	Saída
n	$x(n), y(n-1), y(n-2), \dots, x(n-(d_E-1))$	$\widehat{x}(n+1)$
n + 1	$\widehat{x}(n+1), x(n), x(n-1), \dots, x(n-(d_E-2))$	$\widehat{x}(n+2)$
n + 2	$\widehat{x}(n+2), \widehat{x}(n+1), x(n), \dots, x(n-(d_E-3))$	$\widehat{x}(n+3)$
⋮	⋮	⋮

séries temporais, define-se o erro de predição (ou resíduo) como a diferença entre o valor realmente observado para a próxima amostra da série e a estimativa $\widehat{x}(n+1)$ produzida pelo preditor, ou seja,

$$e(n) = x(n) - \widehat{x}(n). \quad (5.2)$$

A seqüência de resíduos, $\{e(n)\}$, $n = 1, \dots, K$, é utilizada para avaliar a precisão do modelo por meio do Erro Quadrático Médio Normalizado (*Normalized Mean-Squared Error*, NMSE), dado pela seguinte expressão, já mostrada na seção 2.5.3, como forma de verificar o determinismo de séries temporais

$$NMSE(K) = \frac{1}{K \cdot \sigma_x^2} \sum_{n=1}^K e^2(n) = \frac{\widehat{\sigma}_e^2}{\sigma_x^2}, \quad (5.3)$$

em que σ_x^2 é a variância da série a ser predita, $\widehat{\sigma}_e^2$ é a variância dos resíduos e K é o tamanho da seqüência de resíduos.

Aguirre (2000) sugere outros métodos para validação de modelos com dinâmica caótica, como por exemplo a análise das invariâncias de sistemas, pois, a simples predição UPA não é suficiente. Abarbanel et al. (1993) sugere o uso da predição KPA como validação para sistemas não-lineares caóticos, pois, embora erros de predições UPA sejam frequentemente elevados, estes modelos podem reproduzir melhor o comportamento de um sistema. Desta forma, nem sempre uma boa predição UPA leva a uma boa predição KPA.

5.5 Simulações e Resultados

Com a caracterização de algumas séries caóticas no início do Capítulo e o respectivo parâmetros da janela de imersão para cada uma delas, pode-se agora testar o desempenho das principais redes neurais dinâmicas descritas nesta dissertação. Estas séries têm suas amplitudes normalizadas entre $[-1, 1]$ e todas as redes avaliadas nesta dissertação possuem duas camadas escondidas. A primeira camada escondida possui 20 neurônios, a segunda 5 e a camada de saída possui 1 neurônio. Todos neurônios das camadas escondidas e de

saída usam a função de ativação tangente hiperbólica.

Para uma primeira avaliação do desempenho das redes neurais dinâmicas na tarefa de predição e modelagem de séries temporais caóticas as redes NARX-P e NARX-SP são treinadas com a série Mackey-Glass. Vale lembrar que as redes NARX descritas nesta dissertação se diferenciam entre si apenas pelo modo de identificação. A primeira é a rede neural NARX utilizando o modo paralelo, em que existe realimentação no treino e teste. A segunda rede usa o modo série-paralelo, em que não existe realimentação no treino e o contexto da entrada é formado pelos próprios elementos da série.

Ambas as redes neurais usam as mesmas configurações: treinamento pelo algoritmo *backpropagation* com 500 épocas, taxa de aprendizagem igual a 0,01, janela de entrada da rede formada por uma dimensão (d_E) igual a 5 e um atraso (τ) igual a 3. A série Mackey-Glass possui 2000 amostras, sendo que as 1700 primeiras são destinadas para o treino, 200 para validação e 100 últimas para teste.

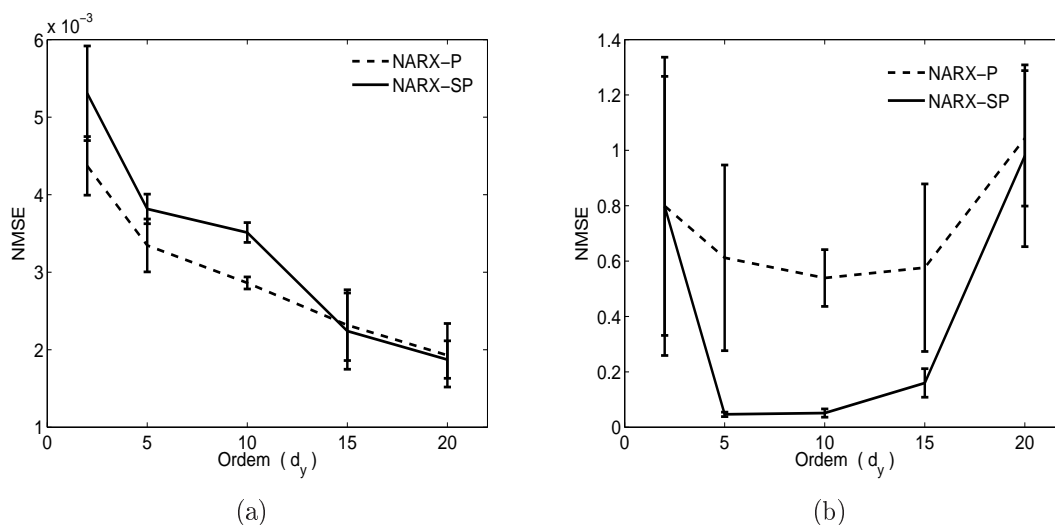


Figura 5.6: NMSE versus ordem do regressor de saída (d_y) dos modelos NARX: (a) predição UPA; (b) predição KPA.

A ordem do regressor de saída (d_y) dos modelos NARX são testadas com 5 casos: 2, 5, 10, 15 e 20. Para conhecer como estas redes neurais se comportam quando a ordem do regressor de saída varia, duas curvas para NMSE são mostradas. Na Figura 5.6(a) encontra-se o erro gerado no teste de predição UPA, calculado para um horizonte de 100 amostras. Por sua vez, a Figura 5.6(b) mostra o resultado da predição KPA para um horizonte de 50 passos adiante. Os resultados dessas simulações são construídos após 10 repetições e apresentados nas Figuras 5.6(a) e (b) as médias do NMSE das predições UPA e KPA, sendo que as barras verticais mostrando as variâncias do NMSE.

A primeira observação a ser feita sobre estes resultados é que, quando o d_y se aproxima de zero, estas duas redes tendem a uma rede FTDNN, isto é, sem realimentação. Pelos resultados apresentados nas Figuras 5.6(a) e (b), o erro é alto quando d_y é baixo, mostrando a necessidade de uma memória extra para melhor capturar a dinâmica da série. Outro importante resultado desta simulação é o desempenho superior da rede NARX-SP na predição KPA.

Também deve ser discutido a deteriorização dos resultados das redes com o aumento de d_y na predição KPA. Este fato pode ser explicado pela saturação da rede com o aumento de informações na entrada desta. Por outro lado, na predição UPA não há perda de desempenho das redes com o aumento de d_y ; pelo contrário, há uma melhora dos resultados.

No próximo teste, avaliam-se as redes NARX-P e NARX-SP com a série do laser caótico, largamente usada em estudos de *benchmark*. Esta série temporal possui 1500 amostras, sendo que as 1000 primeiras são destinadas para o treino e 500 últimas para o teste. Nesta comparação, todas as redes possuem a mesma janela de imersão com dimensão de imersão igual a 7 e atraso de imersão igual a 2. A ordem do regressor de saída das redes NARX-P e NARX-SP são fixados em $n_y = \tau d_E = 2 \times 7 = 14$. O algoritmo *backpropagation* é utilizado para treinar a rede por 3000 épocas, com taxa de aprendizagem igual a 0,01.

Os resultados são apresentados nas Figuras 5.7, 5.8(a) e 5.8(b) para as redes NARX-SP, Elman e FTDNN, respectivamente. A linha sólida representa os 500 valores da amplitude estimados recursivamente e a linha tracejada são os valores exatos da série. Uma inspeção visual demonstra claramente que o modelo NARX-SP tem um desempenho melhor do que as outras duas arquiteturas neurais. É importante salientar que a situação crítica do laser caótico ocorre por volta do instante de tempo 60, quando ocorrem colapsos da intensidade do laser repentinamente (passam de um valor alto para um valor baixo), para então começar uma recuperação gradual da intensidade do laser.

O modelo NARX-SP é capaz de reproduzir as dinâmicas do laser caótico muito mais fielmente que as redes Elman e FTDNN. A rede Elman faz esta tarefa bem até o ponto crítico. Deste ponto em adiante, é incapaz de reproduzir as dinâmicas do laser caótico fielmente, ou seja, as intensidades preditas de laser têm valores muito mais baixos que as amplitudes exatas. Já a rede FTDNN tem um desempenho muito inferior. Do ponto de vista dinâmico, os resultados sugerem que a saída da rede FTDNN é capturada por de um ciclo limite, pois oscila intermitentemente.

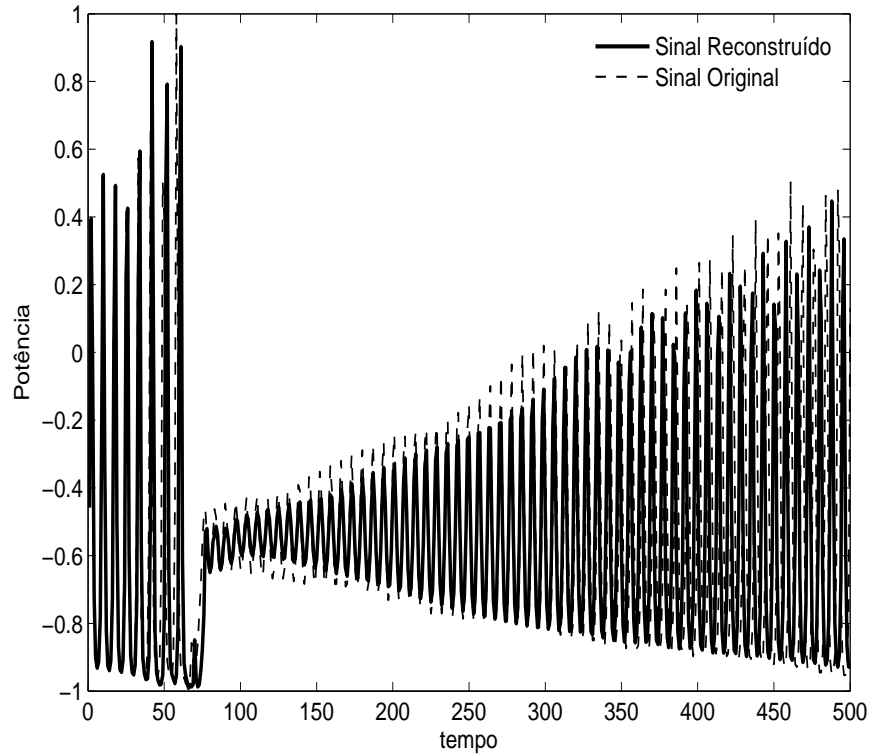


Figura 5.7: predição KPA para a série do laser caótico utilizando a rede NARX-SP.

Vale mencionar que os resultados anteriores não significam que as rede FTDNN e Elman não possam aprender as dinâmicas do laser caótico. De fato, isto é mostrado ser possível em Haykin & Principe (1998). Os resultados só expressam que para um mesmo pequeno número de neurônios $N_{h,1} + N_{h,2} + 1 = 20$, uma curta série temporal de treinamento e um número fixo de épocas de treinamento, a rede NARX-SP desempenha melhor esta tarefa do que as redes FTDNN e Elman. Em resumo, a arquitetura NARX-SP é computacionalmente mais poderosa do que as redes Elman e FTDNN em capturar a dinâmica não-linear do laser caótico.

Na Figura 5.9 encontra-se a média de 10 resultados do desempenho da predição KPA para as redes FTDNN, Elman, NARX-P e NARX-SP. Em outras palavras, esta Figura mostra a evolução de NMSE em função do horizonte de predição K . Vale destacar dois tipos de comportamentos nesta Figura. Abaixo do ponto crítico $K = 60$, as curvas NMSE são aproximadamente as mesmas, com uma pequena vantagem para a rede Elman. Isto significa que, enquanto o ponto crítico não é alcançado, todas as redes têm um bom desempenho de predição. A partir do ponto $K = 60$, os modelos NARX-P e NARX-SP revelam seu desempenho superior, apresentando erros menores do que os das redes Elman e FTDNN.

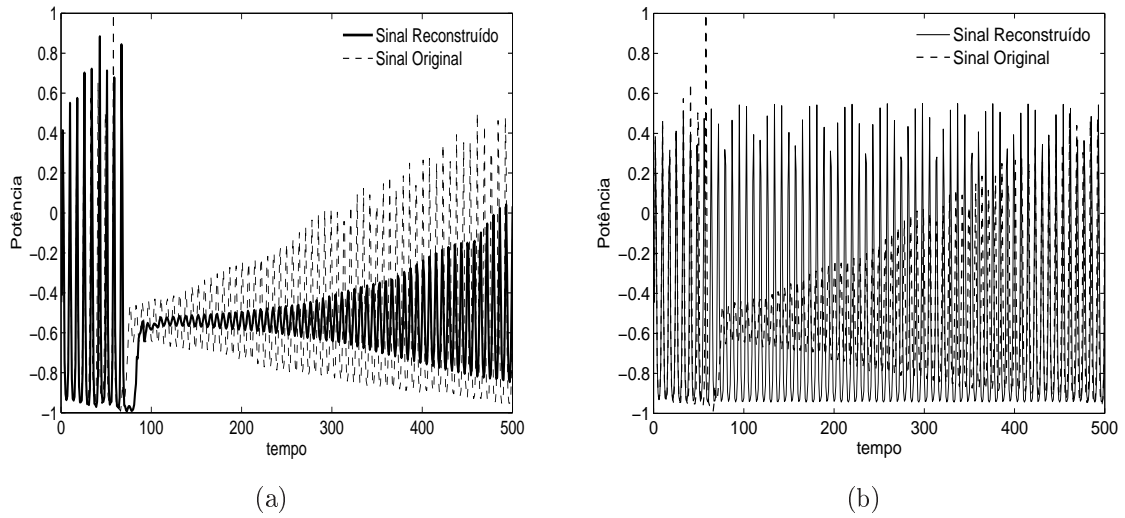


Figura 5.8: predição KPA para a série caótica do Laser. (a) Elman; (b) FTDNN.

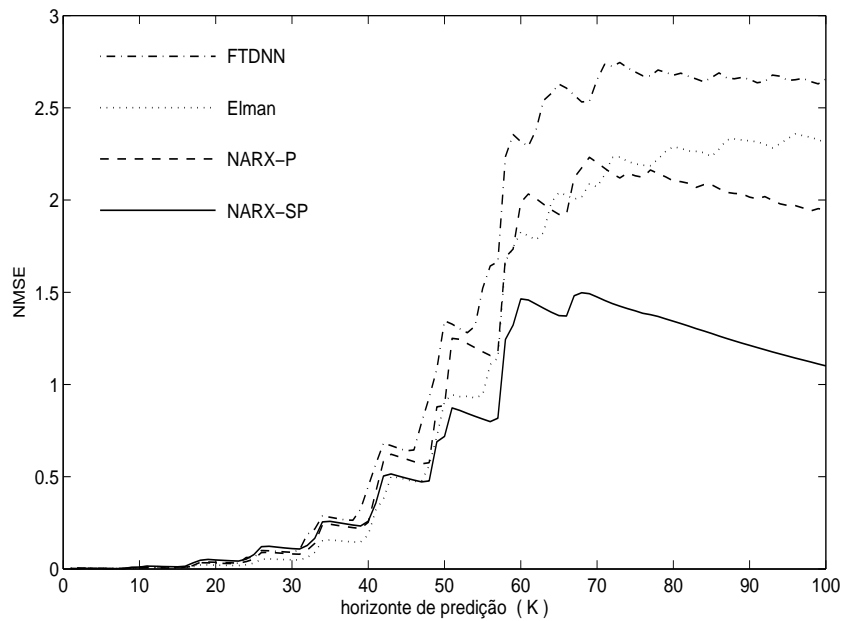


Figura 5.9: NMSE versus horizonte de predição para a rede FTDNN, Elman, NARX-P e NARX-SP.

Como mais uma forma de avaliar o desempenho da rede NARX-SP utilizando a predição da série do laser caótico, é interessante comparar os diagramas de recorrências (Seção 2.5.2) das séries preditas pelas redes neurais NARX-SP, FTDNN e Elman com o diagrama de recorrência da série original. O valor de referência para a distância entre dois pontos no espaço de imersão é fixado em todos os diagramas como $r = 0,4$. O resultado é apresentado na Figura 5.10 e as séries preditas correspondem a $K = 200$ passos adiante. As Figuras 5.10(a) e 5.10(b) são as que apresentam padrões mais semelhantes entre si,

revelando que a rede NARX descreve uma trajetória reconstruída mais próxima da série do laser caótico original.

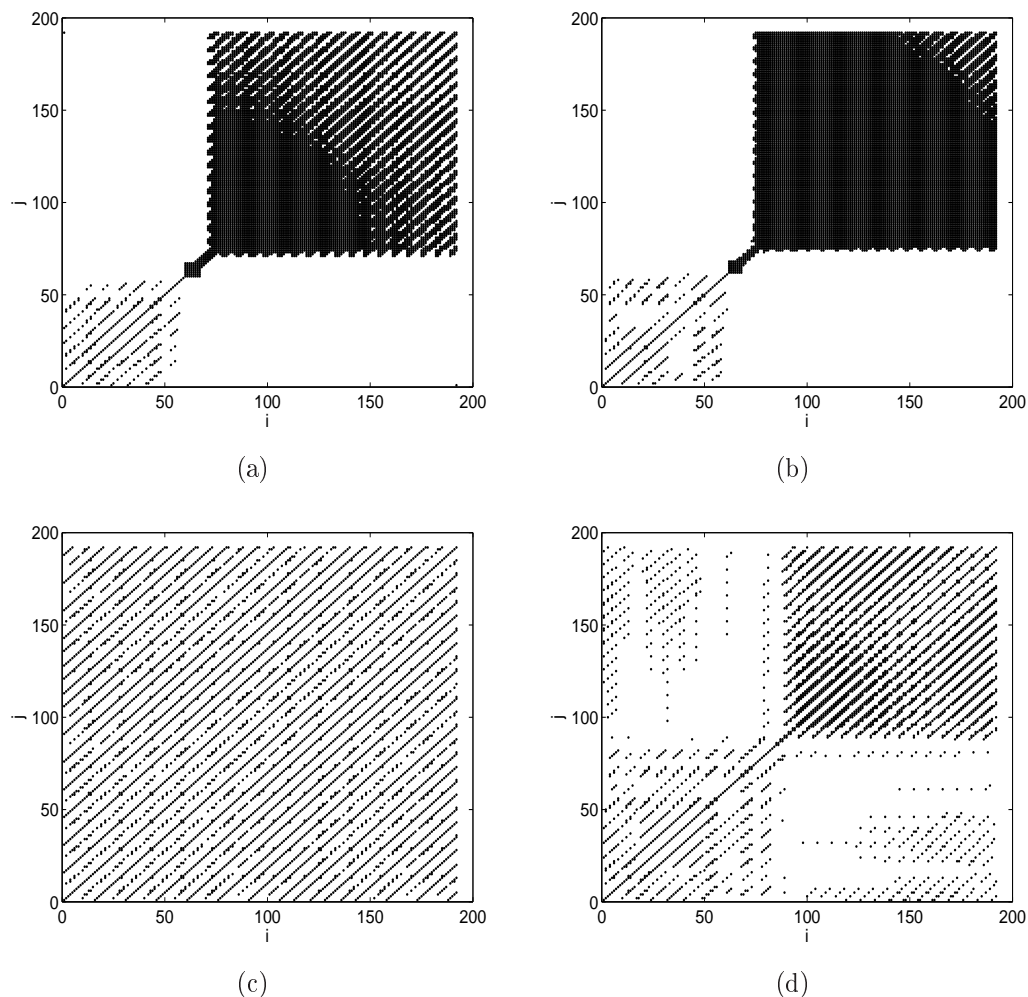


Figura 5.10: diagrama de recorrência: (a) série original; (b) NARX-SP; (c) FTDNN; (d) Elman.

Após a avaliação dos modelos NARX na tarefa de predição e modelagem de séries temporais caóticas, o objetivo agora é aplicá-las na predição de séries temporais de tráfego de redes. Conforme mostrado no Capítulo 3, o tráfego atual de redes de alta velocidade possui características fractais, irregulares e não-estacionárias, sendo de difícil tratamento por meio de modelos lineares ou modelos que não levem em conta as dependências de longo prazo.

A série Bellcore é usada para uma primeira avaliação do desempenho das redes neurais dinâmicas na tarefa de predição e modelagem de séries temporais de tráfego de redes. O algoritmo *backpropagation* é usado para treinar as redes com taxa de aprendizagem igual a 0,001. Esta série possui 1000 pontos, destes 800 são destinados para o treino, 125 para validação e 75 restante para teste. Esta série possui um nível alto de dificuldade,

principalmente devido ao fato de se tratar de um série real, possuir ruído e ter poucas amostras disponíveis.

Percebeu-se nos testes preliminares que a curva de generalização indicava perda de desempenho na predição UPA numa faixa de 300 a 400 épocas, dependendo do algoritmo empregado. No primeiro teste apresentado, o número de épocas é fixado em 300 e varia-se a ordem do regressor de saída das duas redes neurais: NARX-P e NARX-SP. O resultado desta simulação pode ser visto na Figura 5.11(a), em que se observa um melhor desempenho da rede neural NARX-SP na tarefa de predição UPA, quando d_y é inferior a 30. A partir deste valor, as redes Elman, FTDNN e NARX-P geram um melhor desempenho. Por sua vez, a Figura 5.11(b) mostra o resultado da variação da ordem do regressor de saída na tarefa de predição KPA, com $K = 12$, sendo o valor $d_y = 30$ aquele que gera o melhor resultado para a rede neural NARX-SP.

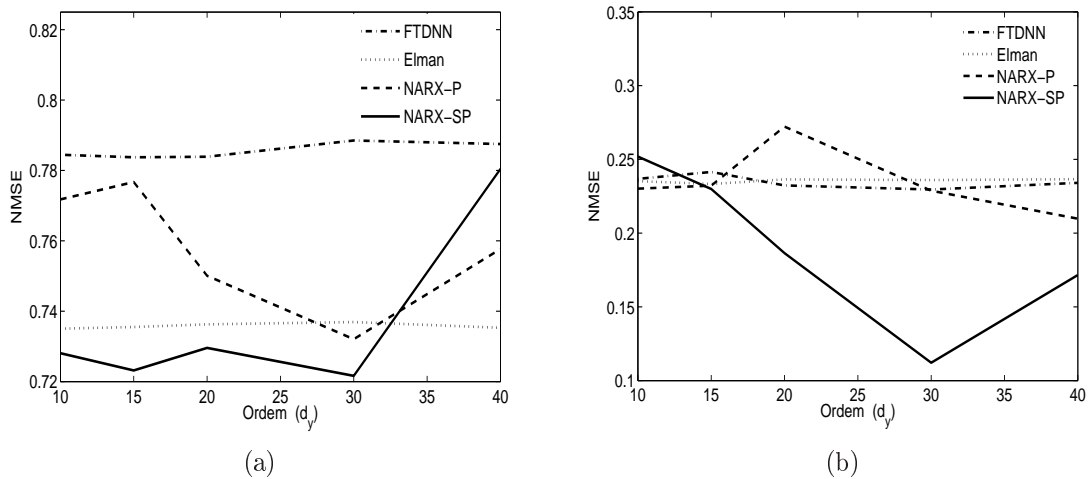


Figura 5.11: NMSE versus d_y : (a) UPA; (b) KPA, $K = 12$.

Como mais um forma de avaliar o desempenho das redes neurais utilizando a série Bellcore, são utilizadas novamente as redes FTDNN, Elman, NARX-P e NARX-SP, mas agora variando o número de épocas de treinamento. Nesta simulação é fixado $d_y = 30$, de acordo com o apresentado anteriormente, por ser o melhor resultado para o problema com a série de Bellcore. O resultado apresentado na Figura 5.12(a) é a resposta das redes na tarefa de predição UPA. A rede NARX-SP obteve um desempenho superior e mais rápido nesta tarefa, quando comparado com as outras redes, demonstrando a boa capacidade de generalização das redes NARX. Isto pode ser verificado na Figura 5.12(a), em que a rede NARX-SP obtém o melhor desempenho com 100 épocas. O resultado apresentado na Figura 5.12(b) é a resposta das redes na tarefa de predição KPA, com $K = 12$. A rede NARX-SP obteve um menor erro a partir de 150 épocas, comprovando-se, mais um vez,

a superioridade da rede NARX-SP.

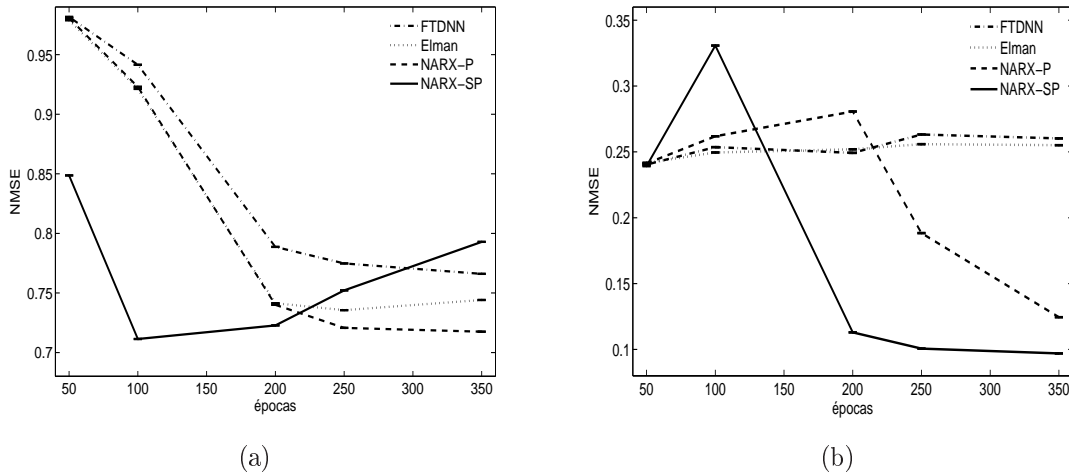


Figura 5.12: NMSE versus números de épocas de treinamento : (a) UPA; (b) KPA, $K = 12$.

Após as simulações apresentadas com a série Bellcore, mostram-se agora os gráficos com valores exatos e preditos para UPA e KPA utilizando apenas a rede neural NARX-SP. A ordem do regressor de saída é fixada em $d_y = 30$, e 400 como o número de épocas no treinamento. Na Figura 5.13 o desempenho desta rede com a série Bellcore é apresentado. No teste de predição UPA houve um resultado satisfatório, mas já esperado pela complexidade da série. Já na predição KPA, apesar de se não conseguir uma predição para um horizonte mais longo, há uma boa predição para $K = 10$ passos adiante.

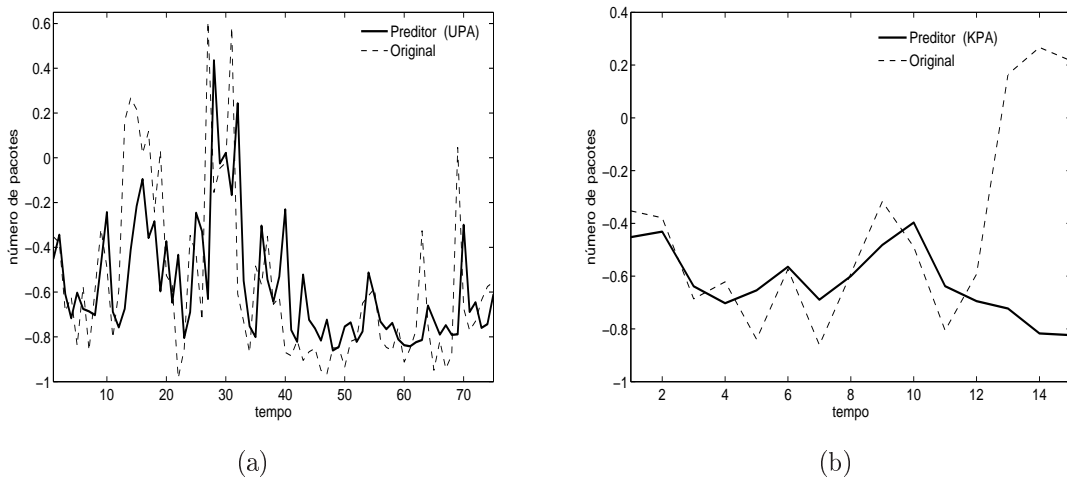


Figura 5.13: predição recursiva da série Bellcore, valores estimados (linha sólida) e valores exatos (linha tracejada): (a) resposta para predição um-passo-adiante; (b) resposta para predição recursiva.

Como o última simulação, avalia-se a predição da série de tráfego de vídeo com taxa de bit variável (VBR). Esta série temporal possui 2500 pontos que são normalizados entre

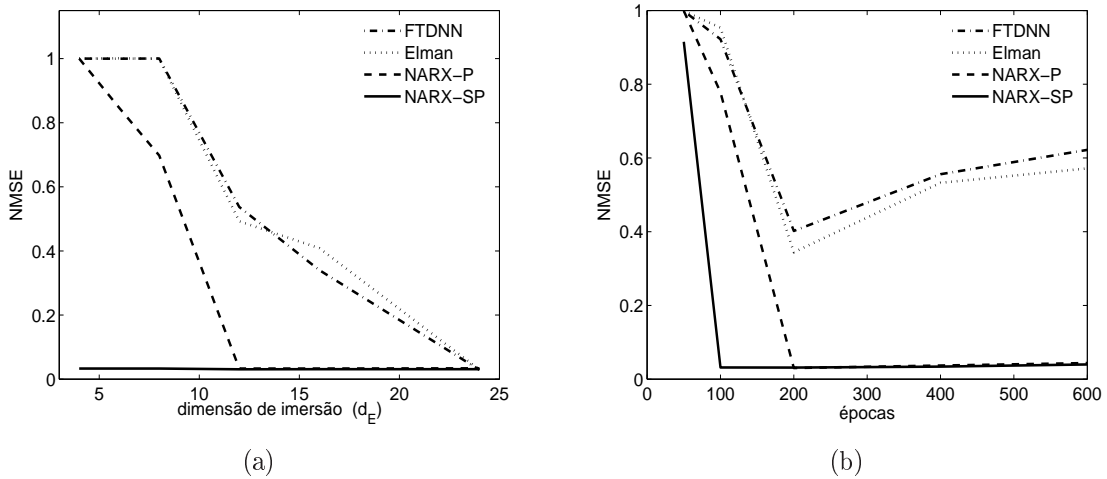


Figura 5.14: avaliação da sensibilidade da rede neural: (a) dimensão de imersão e (b) número de épocas de treinamento.

[-1, 1]. A série temporal normalizada é dividida então em dois grupos: 2000 amostras para treino e 500 amostras para teste.

Compara-se o desempenho das arquiteturas NARX-P e NARX-SP com as redes recorrentes FTDNN e Elman. O algoritmo *backpropagation* é usado para treinar todas as redes com taxa de aprendizagem igual a 0,001.

A avaliação do desempenho da predição KPA em todas as redes podem também ajudar a avaliação da sensibilidade dos modelos neurais de importantes parâmetros de treinamento, tais como, o número de épocas de treinamento e a dimensão de imersão (d_E), como mostra a Figura 5.14.

A Figura 5.14(a) mostra as curvas do NMSE para todas as redes neurais simuladas versus o valor da dimensão de imersão d_E , que varia de 3 a 24. Para esta simulação, todas as redes são treinadas por 300 épocas, $\tau = 1$ (Figura 5.5) e $d_y = 24$. Pode-se notar que os modelos NARX-P e NARX-SP apresentam melhores resultados do que as redes FTDNN e Elman. Em particular, o desempenho do NARX-SP é bastante superior aos demais. De $d_E \geq 12$ em adiante, o desempenho das redes NARX-P e NARX-SP são praticamente os mesmos. Vale notar que o desempenho das redes FTDNN e Elman aproximam-se das redes NARX-P e NARX-SP quando d_E é da mesma ordem de magnitude de d_y . Isto sugere que, para as redes NARX-SP (ou NARX-P), pode-se selecionar um valor pequeno para d_E e ainda assim se tem um desempenho muito bom quando comparado com as redes FTDNN e Elman.

A Figura 5.14(b) mostra as curvas do NMSE obtido nas redes neurais simuladas versus

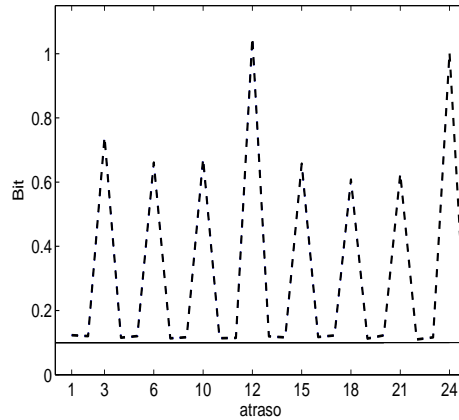


Figura 5.15: informação mútua da série de tráfego de vídeo VBR.

o número de épocas de treinamento, variado de 90 a 600. Para esta simulação todas as redes são treinadas com $\tau = 1$, $d_E = 12$ e $d_y = 2\tau d_E = 24$. Novamente, melhores resultados são alcançados pelo NARX-P e NARX-SP. O desempenho do NARX-SP é praticamente o mesmo depois de 100 épocas. O mesmo comportamento é observado para a rede NARX-P, a partir de 200 épocas. Isto pode ser explicado lembrando-se que o NARX-P usa valores estimados para compor o regressor de saída $\mathbf{y}_p(n)$ e, por causa disso, aprende mais devagar que a rede NARX-SP.

Outro importante comportamento pode ser observado para as redes FTDNN e Elman. Depois de 200 épocas, estas redes aumentam seus valores do NMSE em vez de diminuí-los. Isto pode ser uma evidência de *overfitting*, um fenômeno observado quando modelos não-lineares, com muitos graus de liberdade (pesos sinápticos), são treinados durante um período longo para um conjunto finito de dados. Neste sentido, os resultados da Figura 5.14(b) sugerem fortemente que as redes NARX-SP e NARX-P são mais robustas, ou seja, menos sensíveis a variações paramétricas do que as redes FTDNN e Elman.

Finalmente, é mostrado na Figura 5.16(a), 5.16(b) e 5.17 a estimação típica de traços de tráfego de vídeo VBR gerado pelas redes FTDNN, Elman e NARX-SP, respectivamente. Para esta simulação, todas as redes neurais devem estimar recursivamente os valores das amostras dos traços do tráfego de vídeo VBR por $K = 300$ passos adiante no tempo. Para todas as redes, usa-se $d_E = 12$, $\tau = 1$ e $d_y = 24$. Os modelos neurais são treinados por 300 épocas. Para estes parâmetros de treinamento, as redes FTDNN e Elman possuem resultados muito equivalentes, não conseguindo prever mais do que $k = 200$ passos adiante. Já a rede NARX-SP prediz o traço de tráfego de vídeo muito melhor do que as redes FTDNN e Elman, como apresentado na Figura 5.17, que possui o tráfego predito por $k = 400$ passos adiante.

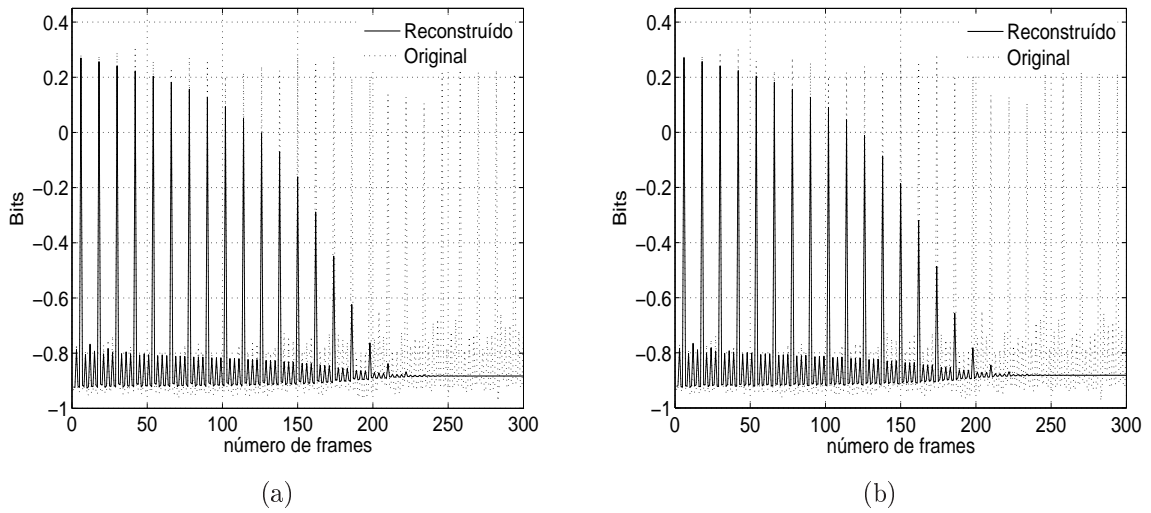


Figura 5.16: predição recursiva obtidas pelas redes (a) FTDNN e (b) Elman.

Vale enfatizar que os resultados mostrados na Figura 5.16 não significam que as redes FTDNN e Elman não possam eventualmente prever os traços do tráfego de vídeo tão bem quanto a rede NARX-SP. Eles só querem dizer que, para a mesma configuração de treinamento e de parâmetros, a rede NARX-SP tem um maior poder computacional. Lembrando que a rede MLP é um aproximador universal de função, então, qualquer modelo neural baseado nesta rede, tal como as redes FTDNN e Elman, são, em princípio, capazes de aproximar funções complexas com exatidão arbitrária, uma vez que sejam fornecidos dados suficientes e épocas de treinamento adequados.

5.5.1 Conclusão

Neste Capítulo foram avaliados os desempenhos das redes neurais dinâmicas descritas no Capítulo 4, em especial a rede neural NARX, na tarefa de predição de séries caóticas e séries de tráfego de redes. Tais algoritmos foram avaliados segundo sua capacidade de predição UPA e predição KPA. Em particular, buscou-se comparar o desempenho das redes dinâmicas mais comuns, como as redes FTDNN e Elman. Dando ênfase especial à nova proposta introduzida nesta dissertação, a rede neural NARX.

Como conclusão geral deste Capítulo, pode-se afirmar que as redes neurais NARX-P e NARX-SP têm um melhor desempenho na predição UPA, predição KPA e na tarefa de modelagem do que as redes FTDNN e Elman. Em particular, o desempenho da rede NARX-SP é superior as demais redes neurais aqui discutidas. Esta rede obteve desempenho superior e mais rápido na tarefa de predição UPA e KPA, demonstrando a sua boa capacidade de generalização.

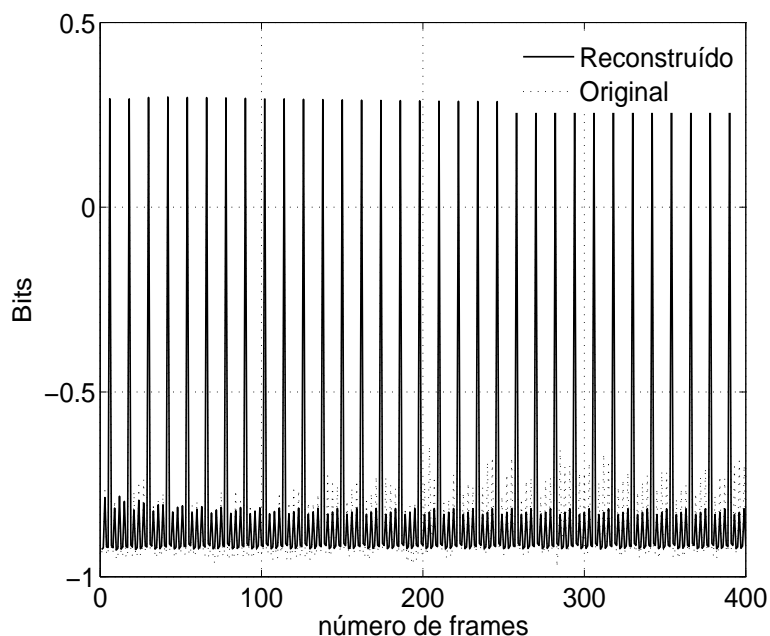


Figura 5.17: predição recursiva obtidas pela rede NARX-SP.

Este melhor desempenho das redes NARX pode ser explicado pela sua capacidade de extrair memória de curto e longo prazo. Em especial, o caso do modelo de treinamento série paralelo da rede NARX obter um desempenho melhor do que todas as redes testadas, pode ser explicado por dois motivos. Primeiro, o regressor de saída durante o treino é composto de amostras exatas da série temporal de interesse e não de valores estimados, deixando este modelo mais preciso. Em segundo, a rede NARX-SP tem puramente uma arquitetura *feedforward*, e pode ser treinada pelo algoritmo *backpropagation*.

O próximo Capítulo traz um resumo dos tópicos abordados e dos principais resultados e conclusões obtidas, além de discutir alguns possíveis temas para futuros trabalhos na área.

6 CONCLUSÕES E PERSPECTIVAS

Este estudo é focado na predição e modelagem de séries temporais complexas, isto é, séries com irregularidades, típicas de sistemas não-lineares que apresentam características complexas. Para esta tarefa são usadas as redes neurais artificiais, motivada pela sua capacidade de aproximar funções, extrair informações temporais importantes e possuir melhor desempenho do que os procedimentos estatísticos convencionais.

Muitos dos problemas reais da engenharia, informática e medicina são dinâmicos, necessitando de ferramentas capazes de modelá-los. No Capítulo 2 fez-se um estudo destes sistemas dinâmicos e caóticos, definindo os principais termos, mostrando as possíveis trajetórias de sistemas dinâmicos, descrevendo e apresentando as principais diferenças dos sistemas caóticos e dinâmicos dos sistemas lineares e estocásticos. Este Capítulo é de extrema importância, pois ele serve de embasamento teórico das séries temporais usadas como estudo de caso e também por servir como base para o Capítulo 3 de modelagem de tráfego de redes, em especial para a nova abordagem dos modelos de tráfego, conhecida como modelagem auto-similar ou fractal.

No Capítulo 3 é destacado que o modelo mais recente de tráfego possui características, tais como dependências de longo alcance, densidade de caudas pesadas, dimensões fractais e ruído $1/f$, diferente dos modelos tradicionais de tráfego, em que predomina a modelagem por processos estocásticos. Desta forma, deve-se possuir a habilidade de modelar e, portanto, a capacidade de prever tráfego de redes, pois isto é de fundamental importância para que sistemas de gerenciamento possa evitar perdas e atrasos de pacotes.

Redes neurais aplicadas para predição e modelagem no Capítulo 4 são assim o objetivo principal deste trabalho, pois realizam tarefas difíceis, como a predição e reconstrução de sistemas caóticos. São usadas para tal redes neurais supervisionadas dinâmicas não-recorrentes, que possuem atrasadores na entrada da rede para prover representação dinâmica temporal. O foco ainda maior desta dissertação é para com as redes neurais recorrentes, em que existem laços de realimentação, permitindo o fluxo de sinais de ati-

vação e saída neurais entre neurônios de camadas distintas, entre neurônios de uma mesma camada, ou ainda de um neurônio para ele mesmo. Este tipo de configuração permite a rede relembrar informações de um passado recente pelo processamento de informação passada. Assim, juntamente com a capacidade das redes dinâmicas de extrair informações temporais, redes neurais recorrentes são uma excelente ferramenta nas tarefas propostas neste estudo.

A rede neural NARX, descrita na Seção 4.5, é um modelo que estima o valor da próxima saída em função de valores prévios dos sinais de saída e de entrada. Tem sido demonstrado que a formulação original da rede NARX, aplicada nos problemas que motivaram esta dissertação, a ordem da memória da saída é reduzida, e assim não resolve totalmente o problema de dependências temporais de longa duração, mas é demonstrado que ela tem freqüentemente um desempenho muito melhor que as RNAs recorrentes padrões, alcançando uma convergência mais rápida e um melhor desempenho de generalização. Assim, esta dissertação apresenta um novo método, embora simples, é bastante eficiente, por permitir que as capacidades computacionais da rede NARX possam ser plenamente exploradas em tarefas de predição e modelagem não-linear de série temporais.

No Capítulo 5 é testado e comprovado o bom desempenho das redes dinâmicas na predição e modelagem de sistemas dinâmicos, observando-se assim a habilidade destas redes neste tipo de problema. Pode destacar ainda neste estudo que as redes neurais recorrentes conseguiram superar as redes neurais dinâmicas sem recorrência, principalmente na velocidade de convergência do algoritmo *backpropagation* e desempenho no aprendizado de informações temporais. E por fim, como principal destaque deste capítulo, tem-se as redes neurais NARX, que conseguiram um melhor desempenho no teste recursivo, logo melhores respostas na tarefa de predição de longo prazo e modelagem dinâmica. Desta forma, a rede neural NARX, quando treinada de modo paralelo ou série-paralelo, obtém-se melhores resultados que as redes neurais mais comuns, tal como a rede Elman e a rede FTDNN.

Uma perspectiva deste trabalho de mestrado é entender a utilização da metodologia proposta para construir um mecanismo de predição de possíveis congestionamentos e tendências do tráfego. Na mesma área de interesse é a aplicação das redes neurais recorrentes em algoritmos preditivos de anormalidades na rede, *Intrusion Detection System* (IDS).

Uma outra aplicação de interesse é a análise temporal de séries financeiras, como exemplo a variação dos preços das ações de uma determinada empresa. Estas séries

temporais possuem uma grande complexidade, mas que aparentemente são deterministas. Desta forma, o uso de redes neurais recorrentes, em especial a rede neural NARX, e juntamente com técnicas poderosas de pré-processamento poderiam vir a trazer bons resultados neste campo.

APÊNDICE A – Estabilidade de Estados de Equilíbrio

Considere um sistema dinâmico autônomo descrito pela equação do espaço de estados (2.3) (HAYKIN, 1994). Diz-se que um vetor constante $\mathbf{x} \in m$ é um estado de equilíbrio (estacionário) do sistema se a seguinte condição for satisfeita

$$\mathbf{F}(\bar{\mathbf{x}}) = \mathbf{0}, \quad (\text{A.1})$$

onde $\mathbf{0}$ é o vetor nulo. O vetor velocidade $d\mathbf{x}/dt$ desaparece no estado de equilíbrio $\bar{\mathbf{x}}$, e portanto a função constante $\mathbf{x}(t) = \bar{\mathbf{x}}$ é uma solução da Equação 2.3. Além disto, devido à propriedade de unicidade de soluções, nenhuma outra curva de solução pode passar através do estado de equilíbrio $\bar{\mathbf{x}}$. O estado de equilíbrio é também referido como um ponto singular, significando o fato de que no caso de um ponto de equilíbrio a trajetória degenerará para o próprio ponto.

Para se desenvolver um entendimento mais profundo da condição de equilíbrio, suponha-se que a função não-linear $\mathbf{f}(\mathbf{x})$ seja suave o suficiente para que a equação do espaço de estados (2.3) seja linearizada na vizinhança de $\bar{\mathbf{x}}$. Especificamente, considere

$$\mathbf{x}(t) = \bar{\mathbf{x}} + \Delta\mathbf{x}(t), \quad (\text{A.2})$$

onde $\Delta\mathbf{x}(t)$ é um pequeno desvio de $\bar{\mathbf{x}}$. Então, retendo os primeiros dois termos na expansão em série de Taylor de $\mathbf{f}(\mathbf{x})$, pode-se aproximá-la como segue,

$$\mathbf{f}(\mathbf{x}) \simeq \bar{\mathbf{x}} + J\Delta\mathbf{x}(t). \quad (\text{A.3})$$

A matriz J é a Jacobiana da função não-linear $\mathbf{f}(\mathbf{x})$, calculada no ponto $\mathbf{x} = \bar{\mathbf{x}}$, como mostrado por

$$J = \left. \frac{\partial}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) \right|_{\mathbf{x}=\bar{\mathbf{x}}}. \quad (\text{A.4})$$

Substituindo as Equações A.2 e A.3 em 2.3 e então usando a definição de um estado de

Tabela A.1: Classificação do Estado de Equilíbrio de um Sistema de Segunda Ordem

Tipo de Estado de Equilíbrio $\bar{\mathbf{x}}$	Autovalores da Matriz Jacobiana J
Nó estável	Reais e negativos
Foco estável	Complexos conjugados com partes reais negativas
Nó instável	Reais e positivos
Foco instável	Complexos conjugados com partes reais positivas
Ponto de sela	Reais com sinais opostos
Centro	Conjugados puramente imaginários

equilíbrio, obtém-se

$$\frac{d}{dt}\Delta\mathbf{x}(t) \simeq J\Delta\mathbf{x}(t). \quad (\text{A.5})$$

Desde que a matriz Jacobiana J seja não-singular, isto é, que exista a matriz inversa J^{-1} , a aproximação descrita na Equação A.5 é suficiente para determinar o comportamento local das trajetórias do sistema na vizinhança do estado de equilíbrio $\bar{\mathbf{x}}$. Se J for não singular, a natureza do estado de equilíbrio é essencialmente determinada pelo seus autovalores, e portanto pode ser classificada de uma forma correspondente.

Para o caso especial de um sistema de segunda ordem, pode-se classificar o estado de equilíbrio como resumido na Tabela A.1 e ilustrado na Figura A.1. Sem perda de generalidade, o estado de equilíbrio é assumido como estando na origem do espaço de estados, isto é, $\mathbf{x} = \mathbf{0}$. Note também que no caso de um ponto de sela, mostrado na Figura A.1(e), as trajetórias indo para o ponto de sela são estáveis, enquanto que as trajetórias saindo do ponto de sela são instáveis.

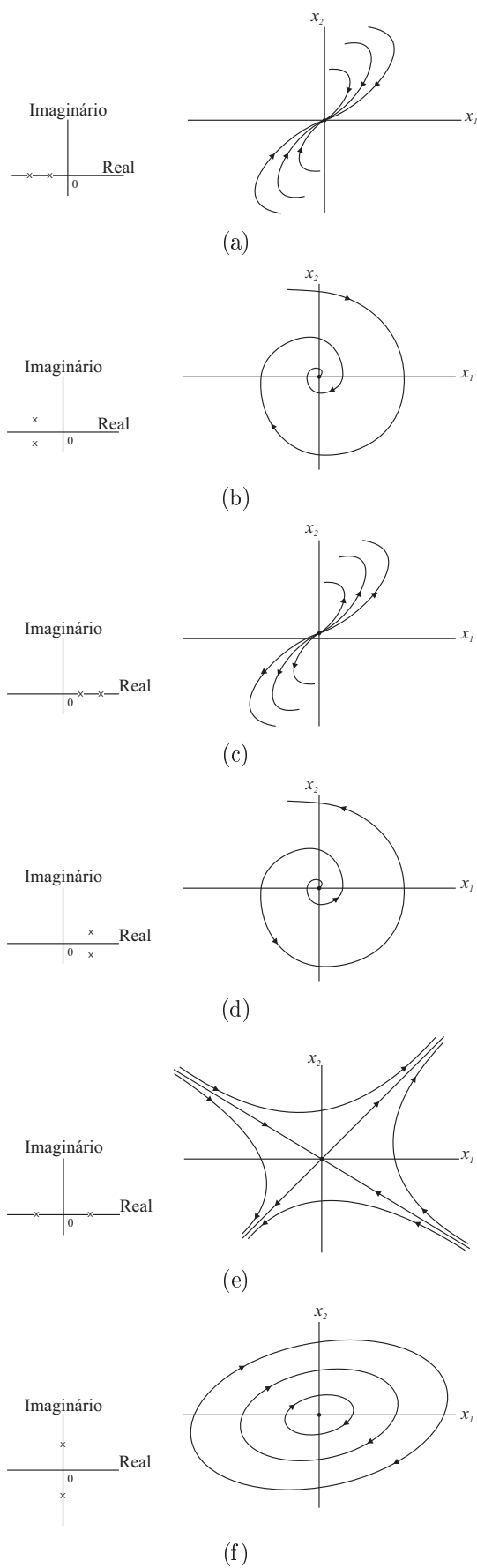


Figura A.1: (a) nó estável, (b) foco estável, (c) nó instável, (d) foco instável, (e) ponto de sela e (f) centro.

APÊNDICE B – Expoentes de Lyapunov

Expoentes de Lyapunov, cuja quantidade é igual à dimensão do espaço de estados, são definidos como a média da razão exponencial em que trajetórias vizinhas se divergem no tempo. Seja um sistema de m equações diferenciais ordinárias. Considere uma hipersfera, espaço fechado de três ou mais dimensões, de condições iniciais centrada num ponto $\mathbf{x}(t_0)$. Conforme o tempo passa, este volume se deforma. Assume-se que, ao longo de cada uma das m dimensões, o raio inicial $d_j(t_0)$ varia exponencialmente no tempo, de maneira que a relação entre $d_j(t_0)$ e o valor correspondente no instante t , dado por $d_j(t)$, é definida como

$$d_j(t) = d_j(t_0)^{\lambda_j(t-t_0)}, \quad j = 1, 2, \dots, n. \quad (\text{B.1})$$

Essa relação pode ser reescrita como

$$\lambda_j = \frac{\ln[d_j(t)/d_j(t_0)]}{t - t_0}. \quad (\text{B.2})$$

Os números λ_j são desta forma os expoentes de Lyapunov.

Em sistemas de tempo discreto, por exemplo, para um mapa unidimensional $x(n+1) = F(x(n))$, seu expoente de Lyapunov (λ) pode ser calculado da seguinte maneira (MONTEIRO, 2006). Sejam $x(0)$ e $x(0) + \delta_0$ duas condições iniciais vizinhas, separadas por uma “pequena” distância δ_0 . Considere que após N interações do mapa, com $N \rightarrow \infty$, a distância entre estes dois pontos seja δ_N . Se δ_N relaciona-se com δ_0 por

$$|\delta_N| \simeq |\delta_0| e^{\lambda N}, \quad (\text{B.3})$$

então λ é o expoente de Lyapunov procurado. A expressão anterior pode ser reescrita como

$$\lambda = \frac{1}{N} \ln \left(\left| \frac{\delta_N}{\delta_0} \right| \right). \quad (\text{B.4})$$

A distância δ_N é a diferença entre a N -ésima interação a partir do ponto $x(0) + \delta_0$ e

a N-ésima interação a partir do ponto $x(0)$. Portanto

$$\delta_N = F^{(N)}(x(0) + \delta_0) - F^{(N)}(x(0)). \quad (\text{B.5})$$

Assim, λ é dado por

$$\lambda = \frac{1}{N}(\ln) \left| \frac{F^{(N)}(x(0) + \delta_0) - F^{(N)}(x(0))}{\delta_0} \right|. \quad (\text{B.6})$$

Admitindo que os dois pontos iniciais $x(0) + \delta_0$ e $x(0)$ estavam infinitesimalmente separados, ou seja, que $\delta_0 \rightarrow 0$, então o argumento do logaritmo na expressão anterior é a derivada de $F^{(N)}$ calculada por $x(0)$. Ou seja,

$$\lambda = \frac{1}{N}(\ln) \left| \frac{dF^{(N)}(x)}{dx} \right|_{x=x(0)}. \quad (\text{B.7})$$

Aplicando a regra da cadeia para o cálculo desta derivada, obtém-se que

$$\frac{dF^{(N)}(x)}{dx} \Big|_{x(0)} = \frac{dF(x)}{dx} \Big|_{x(N-1)} \frac{dF(x)}{dx} \Big|_{x(N-2)} \cdots \frac{dF(x)}{dx} \Big|_{x(0)}, \quad (\text{B.8})$$

sendo $x(n) = F^{(n)}(x(0))$. Assim, λ pode ser calculado pela expressão

$$\lambda = \lim_{n \rightarrow \infty} \frac{1}{n} \ln \prod_{n=0}^{N-1} \left| \frac{dF(x)}{dx} \right|_{x=x(n)} = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{n=0}^{N-1} \ln \left| \frac{dF(x)}{dx} \right|_{x=x(n)}, \quad (\text{B.9})$$

em que um ingrediente essencial é a estimativa do Jacobiano local, $dF(x)/dx$, isto é, da dinâmica linearizada que governa o crescimento das perturbações infinitesimais (KANTZ; SCHREIBER, 1997).

APÊNDICE C – Método de Cao

C.1 Definições Preliminares

Seja uma série temporal composta por N amostras, $x(t)$, $t = 1, 2, \dots, N$:

$$\mathbf{X}_N = \{x(1), x(2), \dots, x(N)\}. \quad (\text{C.1})$$

A trajetória do do sinal temporal $x(t)$ no espaço de fases d -dimensional é reconstruída a partir de vetores d -dimensionais, $\mathbf{y}_t(d)$, definidos como:

$$\mathbf{y}_t(d) = [x(t), x(t + \tau), \dots, x(t + (d - 1)\tau)], \quad t = 1, 2, \dots, N - (d - 1)\tau, \quad (\text{C.2})$$

onde d é a dimensão de imersão (*embedding dimension*) e τ é o passo ou atraso de reconstrução (*time delay*). Este método de reconstrução de atratores, chamado “método dos atrasos temporais”, foi proposto por (TAKENS, 1981). Este autor demonstrou que a trajetória (ou atrator) reconstruída não é idêntica à trajetória real geradora da série temporal observada, mas as características topológicas do atrator reconstruído permanecem preservadas.

A utilização dos vetores $\mathbf{y}_t(d)$ na reconstrução de atratores no espaço de fase só é possível se forem determinados valores adequados para o passo de reconstrução e para a dimensão de imersão.

Embora, em princípio, a dimensão de imersão é independente do atraso τ , a dimensão de imersão *mínima* o é. Assim, diferentes valores de τ resultam em diferentes valores para a dimensão de imersão mínima. A seguir, descreve-se um método que tem sido bastante utilizado, graças a sua simplicidade, para determinar a dimensão de imersão mínima.

C.2 Cálculo da Dimensão de Imersão pelo Método de Cao

Usando como base o método dos falsos vizinhos, descrito brevemente na Seção 2.3.1, Cao (1997) fez a seguinte definição:

$$a(t, d) = \frac{\|\mathbf{y}_t(d+1) - \mathbf{y}_{n(t,d)}(d+1)\|}{\|\mathbf{y}_t(d) - \mathbf{y}_{n(t,d)}(d)\|}, \quad t = 1, 2, \dots, N - d\tau. \quad (\text{C.3})$$

Onde:

- o vetor $\mathbf{y}_t(d+1)$ é o t -ésimo vetor de reconstrução com dimensão $d+1$,

$$\mathbf{y}_t(d) = [x(t), x(t+\tau), \dots, x(t+d\tau)]. \quad (\text{C.4})$$

- O número inteiro $n(t, d)$, $1 \leq n(t, d) \leq N - d\tau$, é tal que $\mathbf{y}_{n(t,d)}(d)$ é o vizinho mais próximo de $\mathbf{y}_t(d)$ no espaço de fase d -dimensional reconstruído, no sentido determinado pela função distância $\|\cdot\|$.
- A função distância $\|\cdot\|$ é definida como a norma máxima de seu argumento, ou seja,

$$\|\mathbf{y}_k(m) - \mathbf{y}_l(m)\| = \max_{0 \leq j \leq m-1} |x(k+j\tau) - x(l+j\tau)|. \quad (\text{C.5})$$

É importante fazer algumas observações sobre os componentes da Equação C.3 antes de continuar a apresentação do método de Cao:

1. O número inteiro $n(t, d)$ que aparece no numerador desta equação é o mesmo que o do denominador.
2. Se $\mathbf{y}_{n(t,d)}(d)$ é igual a $\mathbf{y}_t(d)$, toma-se o segundo vizinho mais próximo em seu lugar.
3. Se d é qualificada como uma dimensão de imersão pelos teoremas de (TAKENS, 1981), então dois pontos que estão próximos no espaço de fases d -dimensional reconstruído, permanecerão próximos no espaço de fases $d+1$ -dimensional. Tal par de pontos são chamados de “vizinhos verdadeiros”, caso contrário são “vizinhos falsos”. Esta é a idéia subjacente ao método dos “vizinhos falsos”, proposto por (ABARBANEL et al., 1993).

O método de Cao se baseia na definição do valor médio de todos os $a(t, d)$'s, ou seja,

$$E(d) = \frac{1}{N - d\tau} \sum_{t=1}^{N-d\tau} a(t, d), \quad (\text{C.6})$$

onde $E(d)$ depende apenas da dimensão d e do passo τ . Para investigar a variação de $E(d)$ quando a dimensão aumenta de d para $d + 1$, define-se a seguinte quantidade:

$$E_1(d) = \frac{E(d+1)}{E(d)}. \quad (\text{C.7})$$

Cao verificou que $E_1(d)$ para de variar quando d é maior que um certo valor d_0 se a série provém de um atrator. Assim, o valor d_0 é tomado como a mínima dimensão de imersão.

Referências

- ABARBANEL, H. D. I. et al. The analysis of observed chaotic data in physical systems. *Reviews of Modern Physics*, v. 65, n. 4, p. 1331–1392, 1993.
- AGUIRRE, L. A. *Introdução à Identificação de Sistemas*. Belo Horizonte, MG: Editora UFMG, 2000.
- ATIYA, A. F.; ALY, M. A.; PARLOS, A. G. Sparse basis selection: New results and application to adaptive prediction of video source traffic. *IEEE Transactions on Neural Networks*, v. 16, n. 5, p. 1136–1146, 2005.
- ATIYA, A. F. et al. A comparison between neural-network forecasting techniques-case study: River flow forecasting. *IEEE Transactions on Neural Networks*, v. 10, n. 2, p. 402–409, 1999.
- BARRON, A. R. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, v. 39, n. 3, p. 930–945, 1993.
- BENGIO, Y.; SIMARD, P.; FRASCONI, P. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, v. 5, n. 2, p. 157–166, 1994.
- BERAN, J. et al. Long-range dependence in variable-bit-rate video traffic. *IEEE Transactions on Communications*, v. 43, n. 234, p. 1566–1579, 1995.
- BHATTACHARYA, A.; ATIYA, A. F.; PARLOS, A. G. Prediction of mpeg-coded video source traffic using recurrent neural networks. *IEEE Transactions on Signal Processing*, v. 51, n. 8, p. 2177–2190, 2003.
- BOX, G.; JENKINS, G. M.; REINSEL, G. *Time Series Analysis: Forecasting & Control*. 3rd. ed. [S.l.]: Prentice Hall, 1994.
- CAO, L. Practical method for determining the minimum embedding dimension of a scalar time series. *Physica D*, v. 110, n. 1–2, p. 43–50, 1997.
- CASTRO, M. C. F. *Predição Não-Linear de Séries Temporais Usando Redes Neurais RBF por Decomposição em Componentes Principais*. Tese (Doutorado) — Universidade Estadual de Campinas, UNICAMP, 2001.
- CHEN, S.; BILLINGS, S. A.; GRANT, P. M. Nonlinear system identification using neural networks. *International Journal of Control*, v. 11, n. 6, p. 1191–1214, 1990.
- COYLE, D.; PRASAD, G.; MCGINNITY, T. M. A time-series prediction approach for feature extraction in a brain-computer interface. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, v. 13, n. 4, p. 461–467, 2005.

- CROVELLA, M. E.; BESTAVROS, A. Self-similarity in world wide web traffic: Evidence and possible causes. *IEEE/ACM Transactions on Networking*, v. 5, n. 6, p. 835–846, 1997.
- CYBENKO, G. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, v. 2, p. 303–314, 1989.
- DABLEMONT, S. et al. Time series forecasting with SOM and local non-linear models - Application to the DAX30 index prediction. In: *Proceedings of the 4th Workshop on Self-Organizing Maps, (WSOM)'03*. [S.l.: s.n.], 2003. p. 340–345.
- DOULAMIS, A. D.; DOULAMIS, N. D.; KOLLIAS, S. D. An adaptable neural network model for recursive nonlinear traffic prediction and modelling of MPEG video sources. *IEEE Transactions on Neural Networks*, v. 14, n. 1, p. 150–166, 2003.
- ELMAN, J. Finding structure in time. *Cognitive Science*, v. 14, p. 179–211, 1990.
- ERRAMILI, A. et al. Self-similar traffic and network dynamics. *Proceedings of the IEEE*, v. 90, n. 5, p. 800–819, 2002.
- ERRAMILI, A.; SINGH, R.; PRUTHI, P. Modeling packet traffic with chaotic maps. In: *Proceedings of the 14th International Teletraffic Congress (ITC'94)*. [S.l.: s.n.], 1994. p. 329–338.
- ERRAMILI, A.; WILLINGER, W. Fractal properties in packet traffic measurements. In: *In Proceedings of the St. Petersburg Regional ITC Seminar*. St. Petersburg, Russia: [s.n.], 1993. p. 144–158.
- FARMER, J. D. Chaotic attractors of an infinite-dimensional dynamical system. *Physica D*, v. 4, p. 66–393, 1982.
- FRASER, A. M.; SWINNEY, H. L. Independent coordinates for strange attractors from mutual information. *Physical Review A*, v. 33, p. 1134–40, 1986.
- FROST, V. S.; MELAMED, B. Traffic modelling for telecommunications networks. *IEEE Communications Magazine*, v. 32, n. 3, p. 70–79, 1994.
- GLASS, L.; MACKEY, M. C. *Dos Relógios ao Caos*. São Paulo, SP: Edusp, 1997.
- GONÇALVES, C. H. R. *Utilizando Redes Neurais Artificiais para Predição de Falhas em Links de Redes óticas*. Dissertação (Mestrado) — Mestrado em Ciência da Computação, Universidade Federal do Ceará, Fortaleza, Ceará, 2003.
- GROSSGLAUSER, M.; BOLOT, J. C. On the relevance of long-range dependence in network traffic. *IEEE/ACM Transactions on Networking*, v. 7, n. 4, p. 329–640, 1998.
- HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. Englewood Cliffs, NJ: Macmillan Publishing Company, 1994.
- HAYKIN, S.; PRINCIPE, J. C. Making sense of a complex world. *IEEE Signal Processing Magazine*, v. 15, n. 3, p. 66–81, 1998.

- HEFFES, H.; LUCANTONI, D. M. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE Journal on Selected Areas in Communications*, v. 4, n. 6, p. 856–867, 1986.
- HELLERSTEIN, J. L.; ZHANG, F.; SHAHABUDDIN, P. A statistical approach to predictive detection. *Computer Network*, v. 35, p. 77–95, 2001.
- HERTZ, J.; KROGH, A.; PALMER, R. G. *Introduction to the theory of neural computation*. Redwood City, CA: Addison-Wesley, 1991.
- HEYMAN, D.; LAKSHMAN, T. What are the implications of long-range dependence for VBR video traffic engineering? *IEEE/ACM Transactions on Networking*, v. 4, n. 3, p. 301–317, 1996.
- HEYMAN, D.; TABATABAI, A.; LAKSHMAN, T. Statistical analysis and simulation study of video teleconference traffic in ATM networks. *IEEE Transactions on Circuits and Systems for Video Technology*, v. 2, n. 1, p. 49–59, 1992.
- HORNE, B. G.; GILES, C. L. An experimental comparison of recurrent neural networks. In: TESAURO, G.; TOURETZKY, D.; LEEN, T. (Ed.). *Advances in Neural Information Processing Systems*. [S.l.]: MIT Press, 1995. v. 7, p. 697–704.
- HORNIK, K. Approximation capabilities of multilayer feedforward networks. *Neural Networks*, v. 4, p. 251–257, 1991.
- HORNIK, K.; STINCHCOMBE, M.; WHITE, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, v. 2, p. 359–366, 1989.
- HSU, H. P. *Probability, Random Variables and Random Processes*. [S.l.]: Schaum Outline, 1997.
- HÜBNER, U.; ABRAHAM, N. B.; WEISS, C. O. Dimensions and entropies of chaotic intensity pulsations in a single-mode far-infrared NH₃ laser. *Physical Review*, A 40, p. 6354–6365, 1989.
- JAGERMAN, D.; MELAMED, B.; WILLINGER, W. Stochastic modeling of traffic processes. In: DSHALALOW, J. (Ed.). *Frontiers in Queueing: Models, Methods and Problems*. [S.l.]: CRC Press, 1997. p. 271–370.
- JORDAN, M. I. Attractor dynamics and parallelism in a connectionist sequential machine. In: *Proceedings of the 8th Annual Conference of the Cognitive Science Society*. Amherst, MA: [s.n.], 1986. p. 531–546.
- KANTZ, H.; SCHREIBER, T. *Nonlinear time series analysis*. Cambridge: Cambridge University Press, 1997.
- KAPLAN, D.; GLASS, L. *Understanding Nonlinear Dynamics*. New York: Springer, 1995.
- KOLEN, J. F.; KREMER, S. C. *A Field Guide to Dynamical Recurrent Networks*. [S.l.]: Wiley-IEEE Press, 2001.

- KUGIUMTZIS, D.; LILLEKJENDLIE, B.; CHRISTOPHERSEN, N. Chaotic time series - part I: Estimation of some invariant properties in state space. *Modeling, Identification and Control*, v. 15, n. 4, p. 205–224, 1994.
- LEDESMA, S.; LIU, D. Synthesis of fractional gaussian noise using linear approximation for generating self-similar network traffic. *SIGCOMM, Computer Communication Review*, ACM Press, New York, NY, v. 30, n. 2, p. 4–17, 2000.
- LELAND, W. E. et al. On the self-similar nature of ethernet traffic. *IEEE/ACM Transactions on Networking*, v. 2, n. 1, p. 1–15, 1994.
- LELAND, W. E. et al. Self-similarity in high speed packet traffic: Analysis and modelling of network traffic measurements. *Statistical Science*, v. 10, p. 67–85, 1995.
- LELAND, W. E.; WILSON, D. V. High time-resolution measurement and analysis of LAN traffic: Implications for LAN interconnection. In: *INFOCOM (3)*. [S.l.: s.n.], 1991. p. 1360–1366.
- LENDASSE, A. et al. Time series prediction competition the CATS benchmark. In: *International Joint Conference on Neural Networks (IJCNN'04)*. [S.l.: s.n.], 2004. p. 1615–1620.
- LEONTARITIS, I. J.; BILLINGS, S. A. Input-output parametric models for nonlinear systems - Part I: deterministic nonlinear systems. *International Journal of Control*, v. 41, n. 2, p. 303–328, 1985.
- LIANG, Y. Real-time VBR video traffic prediction for dynamic bandwidth allocation. *IEEE Transactions on Systems, Man and Cybernetics*, C-34, n. 1, p. 32–47, 2004.
- LILLEKJENDLIE, B.; KUGIUMTZIS, D.; CHRISTOPHERSEN, N. Chaotic time series - part II: System identification and prediction. *Modeling, Identification and Control*, v. 15, n. 4, p. 225–243, 1994.
- LIN, T.; HORNE, B. G.; GILES, C. L. How embedded memory in recurrent neural network architectures helps learning long-term temporal dependencies. *Neural Networks*, v. 11, n. 5, p. 861–868, 1998.
- LIN, T. et al. Learning long-term dependencies in NARX recurrent neural networks. *IEEE Transactions on Neural Networks*, v. 7, n. 6, p. 1424–1438, 1996.
- LIN, T. et al. A delay damage model selection algorithm for NARX neural networks. *IEEE Transactions on Signal Processing*, v. 45, n. 11, p. 2719–2730, 1997.
- LORENZ, E. Deterministic nonperiodic flow. *Science*, v. 20, p. 130–141, 1963.
- LUZ, A. C. C. *Modelo de Previsão Aplicado à Gerência de Redes de Computadores*. Dissertação (Mestrado) — Mestrado em Engenharia Elétrica, Universidade Federal do Ceará, 2003.
- MACKEY, M. C.; GLASS, L. Oscillations and chaos in physiological control systems. *Science*, v. 197, p. 287–289, 1977.

- MANDELBROT, B. Self-similar error clusters in communication systems and the concept of conditional stationarity. *IEEE Transactions on Communication Technology*, p. 71–90, 1965.
- MELAMED, B.; SENGUPTA, B. Tes modeling of video traffic. *IEICE Trans. on Communications*, E75-B, n. 12, p. 1292–1300, 1992.
- MONTEIRO, L. H. A. *Sistemas Dinâmicos*. 2nd. ed. São Paulo, SP: Livraria da Física, 2006.
- MORETTIN, P. A.; TOLOI, C. M. C. *Análise de Séries Temporais*. [S.l.]: Editora Edgard Blücher, 2004.
- MOURA, J. A. B. et al. *Redes Locais de Computadores - Protocolos de Alto Nível e Avaliação de Desempenho*. [S.l.]: McGraw-Hill - EMBRATEL, 1986.
- NARENDRA, K. S.; PARTHASARATHY, K. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, v. 1, n. 1, p. 4–27, 1990.
- NORGAARD, M. et al. *Neural Networks for Modelling and Control of Dynamic Systems*. [S.l.]: Springer, 2000.
- PAXSON, V.; FLOYD, S. Wide area trafic: The failure of poisson modeling. *IEE/ACM Transactions on Networking*, v. 3, n. 3, p. 226–244, 1995.
- PETERS, E. E. *Chaos and Order in the Capital Markets*. [S.l.]: Wiley Finance Editions, 1991.
- PRINCIPE, J. C.; EULIANO, N. R.; LEFEBVRE, W. C. *Neural Adaptive Systems: Fundamentals Through Simulations*. [S.l.]: John Willey and Sons, 2000.
- ROSE, O. Statistical properties of mpeg video traffic and their impact on traffic modeling in atm systems. In: *LCN '95: Proceedings of the 20th Annual IEEE Conference on Local Computer Networks*. Washington, DC, USA: IEEE Computer Society, 1995. p. 397.
- ROSENSTEIN, M. T.; COLLINS, J. J.; LUCA, C. J. D. A practical method for calculating the largest lyapunov exponents from small data sets. *Physica D*, v. 65, p. 117–134, 1993.
- SAVI, M. A. *Dinâmica Não Linear e Caos*. Universidade Federal do Rio de Janeiro - COPPE, Engenharia Mecânica: [s.n.], 2004.
- SCHREIBER, T. Interdisciplinary application of nonlinear time series methods. *Physics Reports*, v. 308, n. 1, p. 1–64, 1999.
- SCHUSTER, H. G. *Deterministic Chaos: An Introduction*. 2nd revised. ed. [S.l.]: VCH, 1988.
- SIEGELMANN, H. T.; HORNE, B. G.; GILES, C. L. Computational capabilities of recurrent NARX neural networks. *IEEE Transactions On Systems, Man, and Cybernetics*, B-27, n. 2, p. 208–215, 1997.

- SILVA, J. L. de Castro e. *ProCon - Prognóstico de Congestionamento de Tráfego de Redes usando Wavelets*. Tese (Doutorado) — Centro de Informática, Universidade Federal de Pernambuco (UFPE), 2004.
- SILVA, J. L. de Castro e; CAMPOS, M. A.; CUNHA, P. R. F. Modelagem estocástica em redes de comunicação. In: *XXI Simpósio Brasileiro de Telecomunicações, (SBrT2001)*. Fortaleza, CE: [s.n.], 2001.
- TAKENS, F. Detecting strange attractors in turbulence. In: RAND, D. A.; YOUNG, L.-S. (Ed.). *Dynamical Systems and Turbulence*. [S.l.]: Springer, 1981. (Lecture Notes in Mathematics, v. 898), p. 366–381.
- TSOI, A. C.; BACK, A. D. Discrete-time recurrent neural network architectures: a unifying review. *Neurocomputing*, v. 15, n. 3, p. 183–223, 1997.
- WAIBEL, A. et al. Phoneme recognition using time-delay neural networks. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, v. 37, n. 3, p. 328–339, 1989.
- WAN, E. A. Temporal backpropagation for FIR neural networks. In: *Proceedings of the IEEE International Joint Conference on Neural Networks*. [S.l.: s.n.], 1990. v. 1, p. 575–580.
- WAN, E. A. Times series prediction using a connectionist network with internal delay lines. In: WEIGEND, A. S.; GERSHENFELD, N. A. (Ed.). *Time Series Prediction: Forecasting the Future and Understanding the Past*. Reading, MA: Addison-Wesley, 1994. p. 195–217.
- WHITNEY, H. Differentiable manifolds. *Annals of Mathematics*, v. 37, n. 3, p. 645–680, 1936.
- WILLIAMS, G. P. *Chaos Theory Tamed*. Washington, DC: National Academies Press, 1997.
- WILLINGER, W. et al. Self-similarity through high-variability: Statistical analysis of ethernet LAN traffic at the source level. *IEEE/ACM Transactions on Networking*, v. 5, n. 1, p. 71–86, 1997.
- WILLINGER, W.; TAQQU, M. S.; ERRAMILI, A. A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks. In: KELLY, F. P.; ZACHARY, S.; ZIEDINS, I. (Ed.). *Stochastic Networks*. [S.l.]: Oxford University Press, 1996. p. 339–366.
- WOLF, A. J. et al. Determining lyapunov exponents from a time series. *Physica D*, v. 16, p. 285–317, 1985.
- YOUNG, E. F.-Y.; CHAN, L. W. Using recurrent network in time series prediction. In: . [S.l.: s.n.], 1993. v. 4, p. 332–336.
- YOUSEFI'ZADEH, H. *Performance Modeling of a Class of Queuing with Self-Similar Characteristics*. Tese (Doutorado) — University of Southern California, 1997.

YOUSEFI'ZADEH, H. Neural network modeling of self-similar teletraffic patterns. In: *Workshop on Fractals and Self-similarity in Data Mining: Issues and Approaches. In conjunction with 8th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. [S.l.: s.n.], 2002.

YOUSEFI'ZADEH, H.; JONCKHEERE, E. A. Dynamic neural-based buffer management for queueing systems with self-similar characteristics. *IEEE Transactions on Neural Networks*, v. 16, n. 5, p. 1163–1173, 2005.