

Universidade Federal de Minas Gerais
Programa de Pós-Graduação em Engenharia Elétrica
Centro de Pesquisa e Desenvolvimento em
Engenharia Elétrica

Uso de Redes Neurais e Conhecimento a Priori na Identificação de Sistemas Dinâmicos Não-Lineares

Gleison Fransoares Vasconcelos Amaral

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

Orientador: Luis Antonio Aguirre

Um amigo fiel é uma poderosa proteção,
quem o encontrou descobriu um tesouro.
Nada se pode comparar com um amigo fiel,
e nada se iguala ao seu valor.
Um amigo fiel é um remédio de vida;
aquele que teme o Senhor achará tal amigo.
O que teme o Senhor terá também uma excelente
amizade, porque o seu amigo será semelhante a Ele.

Eclesiástico 6, 14-17.

Resumo

Este trabalho investiga o uso de redes neurais artificiais do tipo *perceptron* de múltiplas camadas MLP na identificação de sistemas dinâmicos não-lineares. As principais questões investigadas foram a possibilidade e a viabilidade de se utilizar informação a priori tanto para restringir a topologia da rede quanto os pesos da mesma durante o treinamento.

A fim de testar algumas hipóteses, trabalhou-se com dois estudos de caso, que têm se tornado “testes de bancada” na identificação de sistemas não-lineares. Os sistemas usados foram o oscilador de Duffing-Ueda, representado por equação diferencial ordinária não-linear e um aquecedor elétrico, que está disponível em laboratório. No caso do oscilador de Duffing-Ueda, o objetivo principal foi o de treinar uma rede afim de reproduzir a bifurcação do tipo forquilha. Para esse fim, utilizou-se informação sobre a simetria dos pontos fixos do sistema autônomo para restringir a topologia da rede. Para o caso do aquecedor elétrico, o objetivo básico foi treinar uma rede de tal maneira a garantir que a mesma tivesse o comportamento em estado estacionário conhecido previamente do sistema.

Os principais resultados mostraram que foi possível treinar uma rede que reproduz muito bem o diagrama de bifurcação do sistema de Duffing-Ueda, dentro da faixa de parâmetros investigada, inclusive apresentando a bifurcação do tipo forquilha. Até onde vai o conhecimento dos autores, redes MLP com tais características ainda não foram divulgadas na literatura internacional. O uso da informação de simetria de pontos fixos parece ter sido importante na obtenção desse resultado. Foi obtida uma rede que representa tanto a dinâmica quanto o estado estacionário de um aquecedor elétrico para isso utilizando restrições sobre os pesos da rede durante seu treinamento. A comparação das técnicas de poda investigadas mostrou que em geral, elas conseguem reduzir o tamanho da rede sem significativa perda de desempenho, mas não foi possível melhorar o desempenho em estado estacionário usando-se tais ferramentas. Assim, acredita-se que os procedimentos desenvolvidos e testados neste trabalho, têm um importante potencial no uso de redes neurais em problemas de identificação de sistemas não-lineares.

Abstract

This work investigates the use of multilayer *perceptron* (MLP), a kind of artificial neural networks, in nonlinear system identification problems. The main issue investigated in this work is the possibility and the viability of using a priori information in order to restrict the topology and the weights of the network during the training.

Two benchmark systems for nonlinear system identification were studied to test some hypotheses, namely Duffing-Ueda's oscillator and an electrical heater, which is available in laboratory. In the case of Duffing-Ueda's oscillator, the main objective was to train a network in order to reproduce a pitchfork bifurcation, observed in the system. To this end, information about the symmetry of the fixed points of the autonomous system was used to restrict the topology of the network. In the case of the electrical heater, the basic objective was to train a network to guarantee a particular steady-state behaviour, which was available a priori.

The main results showed that it was possible to train a network that reproduces the bifurcation diagram of the Duffing-Ueda's oscillator, within the range of parameters investigated, also presenting the pitchfork bifurcation. To the best of the author's knowledge, this is the first time MLP networks with such features are reported. The use of the information of symmetry of fixed points was important to obtain this result. Network model was also obtained to represent the dynamics and steady-state behaviour of the electrical heater, using restrictions on the weights of the network during training. Some pruning techniques are, in general, able to reduce the size of the network without significant loss of performance. However, it was not possible to improve the performance in steady-state using such tools. Thus, it is believed that the procedures developed and tested in this work have an important potential in the training of neural networks in problems of nonlinear system identification.

Agradecimentos

Agradeço primeiramente a Deus por sempre me dar condições e força para desenvolver o trabalho.

Ao meu orientador Luis Antonio Aguirre que possui o rigor de um pai severo e a ternura de uma mãe amorosa. Por sua dedicação e paciência, obrigado!

Aos professores Marcelo Vieira Corrêa, Antônio de Pádua Braga, Waldir Matos Caminhas e Luis Antonio Aguirre pela disposição de avaliar este trabalho em tão pouco tempo e pelas valiosas sugestões que contribuíram para melhorar seu conteúdo e apresentação.

Aos amigos e funcionários do CPDEE, em especial a Erivelton Geraldo Nepomuceno, Mara Cristina da Silveira Coelho, Ubiratan Santos Freitas e de maneira particular a Márcio Falcão Santos Barroso, por entre outras coisas me ajudar na estruturação do texto.

Aos amigos do Movimento Shalom da Arquidiocese de Mariana, que sempre estiveram perto em momentos difíceis. Com os quais também pude dividir momentos de muita alegria e luz!

Aos meus Tios Waldete, Afonso e Primos Soraia e Daniel, que me acolheram em sua casa durante o tempo do mestrado, recebendo-me como filho e irmão.

Às três mulheres de minha vida. Minha mãe Marlene, que me ensina o rigor, minha irmã Miriã que me ensina a alegria e minha namorada Rita de Cássia de Paula que me ensina o Amor. Obrigado!!!

A meu Pai José Soares do Amaral, que mesmo depois de sua partida sempre esteve presente em minha vida. E que me deu uma grande herança,

x

o gosto de aprender.

Ao Prof. José do Carmo Toledo que contribuiu de maneira muito intensa em minha formação acadêmica e cujas lições e posturas guardo sempre comigo.

Ao prof. e amigo Eduardo M.A.M. Mendes que me proporcionou condições de caminhar pelos meandros do mundo da ciência.

Aos amigos da memorável república Mitologia.

Ao amigo Marcelo Elias de Oliveira que me introduziu na área de redes neurais. Vale Marcelo!

A todos aqueles que de forma direta ou indireta contribuíram.

Ao CNPq que possibilitou a realização deste trabalho.

Conteúdo

Resumo	v
Abstract	vii
Agradecimentos	ix
Conteúdo	xi
Lista de Tabelas	xv
Lista de Figuras	xvii
Lista de Símbolos	xix
Nomenclatura	xxi
1 Introdução	1
1.1 Motivação	1
1.2 Objetivo	2
1.3 Apresentação do trabalho	3
2 Sistemas dinâmicos e Redes Neurais	5
2.1 Sistemas dinâmicos	5
2.1.1 Atrator	6
2.1.2 Pontos Fixos	6
2.1.3 Ciclos limite	7
2.1.4 Caos	7
2.1.5 Estabilidade Estrutural	7
2.1.6 Bifurcação	8
2.2 Identificação de Sistemas Dinâmicos	9
2.2.1 O processo de identificação	9
2.2.2 Validação de modelos com dinâmica caótica	10

2.2.3	Modelos NARX e NARMAX	12
2.3	Redes neurais em identificação de sistemas	13
2.4	Considerações finais	14
3	Uso de informação a priori em redes MLP	15
3.1	Introdução	15
3.2	Objetivo	16
3.3	Pontos fixos em redes MLP	16
3.3.1	Obtenção de pontos fixos (caso autônomo)	17
3.4	Imposição de Simetria em Redes MLP	19
3.5	Metodologia para imposição de restrições dada a curva estática	21
3.5.1	O treinamento como um problema de otimização	22
3.5.2	Condições de otimalidade de Kuhn-Tucker	23
3.6	Considerações finais	24
4	Identificação do Sistema de Duffing-Ueda	25
4.1	Introdução	25
4.2	Sistema dinâmico de Duffing-Ueda	26
4.2.1	Dedução da equação	27
4.2.2	Pontos Fixos	28
4.2.3	A dinâmica do sistema	28
4.3	O processo de identificação	30
4.3.1	Excitação do sistema	30
4.3.2	Implementação das redes	32
4.3.3	Cancelamento do termo de polarização	37
4.4	Sobreparametrização da rede	37
4.5	Identificação utilizando excitação periódica	40
4.5.1	Identificação utilizando dados com ruído	45
4.6	Conclusão	46
5	Identificação de um aquecedor	49
5.1	Introdução	49
5.2	Aquecedor com Dissipação Variável	49
5.2.1	Descrição do Sistema	50
5.2.2	Teste Dinâmico	50
5.2.3	Teste Estático	52
5.3	Identificação do aquecedor elétrico	53
5.3.1	Identificação caixa preta	54
5.3.2	Identificação caixa cinza	65
5.4	Análise dos resultados	71
5.5	Conclusão	72

6	Conclusão	73
6.1	Discussão	73
6.2	Conclusão	73
6.3	Sugestões de trabalhos futuros	74
	Bibliografia	80
A	Redes Neurais	81
A.1	Introdução	81
A.2	Neurônio	82
A.3	Função de ativação	83
A.4	Polarização (Bias)	83
A.5	Arquitetura	85
A.6	Aprendizagem	86
	A.6.1 Processos de aprendizagem	86
A.7	Perceptron	87
A.8	Redes MLP (perceptron de múltiplas camadas)	87
A.9	O algoritmo de retropropagação (<i>backpropagation</i>)	88
A.10	O método Levenberg-Marquardt	88
B	Técnicas de poda (<i>pruning</i>)	91
B.1	Introdução	91
B.2	Método de decaimento de pesos	91
B.3	Poda estatística	92
B.4	Cirurgião cerebral ótimo	93

Lista de Tabelas

4.1	Valores percentuais das redes que conseguiram reconstruir a forquilha	34
4.2	Valores percentuais das redes sobreparametrizadas que conseguiram reconstruir a forquilha.	40
4.3	Valores percentuais das redes que conseguiram reconstruir a forquilha, no caso de entrada cossenoidal.	45
4.4	Valores percentuais das redes que conseguiram reconstruir a forquilha, para o caso de dados com ruído.	45
5.1	Valores RMSE das 5 estratégias utilizadas na identificação do aquecedor elétrico.	71

Lista de Figuras

2.1	Diagrama de bifurcação da equação normal (2.4).	9
2.2	Espaço de estados do sistema de Duffing-Ueda em regime caótico e uma de suas seções de Poincaré.	11
2.3	Arquiteturas sugeridas em (Narendra and Parthasarathy, 1990)	13
2.4	Rede neural NARX com 3 atrasos da entrada e 3 atrasos da saída.	14
4.1	O oscilador de Duffing-Ueda	26
4.2	Regimes dinâmicos apresentados pela equação de Duffing-Ueda	29
4.3	Dados de identificação	31
4.4	Arquitetura da rede HH.	33
4.5	Arquitetura da rede HH com <i>bias</i>	33
4.6	Arquitetura da rede HL.	33
4.7	Arquitetura da rede HL com <i>bias</i>	34
4.8	Seções de Poincaré dos modelos neurais.	35
4.9	Diagramas de bifurcação típicos dos modelos neurais.	36
4.10	Seções de Poincaré dos modelos neurais sobreparametrizados.	38
4.11	Diagramas de bifurcação dos modelos neurais sobreparametrizados.	39
4.12	Dados de identificação (entrada cossenoidal).	41
4.13	Seções de Poincaré dos modelos neurais obtidos usando dados da Figura 4.12.	43
4.14	Diagramas de bifurcação dos modelos neurais obtidos usando dados da Figura 4.12.	44
5.1	Diagrama em blocos funcionais do sistema (Aquecedor).	50
5.2	Dados de identificação - Aquecedor.	51
5.3	Dados de validação - Aquecedor.	52
5.4	Curva estática do sistema	53
5.5	Resposta dinâmica da rede HHH	55
5.6	Característica estática da rede HHH	56
5.7	Resposta dinâmica da rede HHHP com alta penalização	57

5.8	Característica estática da rede HHHDP com alta penalização .	57
5.9	Resposta dinâmica da rede HHHDP	58
5.10	Característica estática da rede HHHDP	59
5.11	Arquitetura da rede podada pelo ERR	61
5.12	Resposta dinâmica da rede HHHERR	61
5.13	Característica estática da rede HHHERR	62
5.14	Critérios utilizados após aplicação do OBS para se determinar a melhor rede.	63
5.15	Arquitetura da rede podada pelo OBS	63
5.16	Resposta dinâmica da rede HHHOBS	64
5.17	Característica estática da rede HHHOBS	64
5.18	Arquitetura da Rede proposta em (Narendra and Parthasarathy, 1990).	65
5.19	Tangente aproximando uma quadrática.	66
5.20	Gaussiana aproximando uma quadrática.	67
5.21	Arquitetura da Rede GGGL.	69
5.22	Resposta dinâmica da rede GGGL.	70
5.23	Característica estática da rede GGGL.	70
A.1	Modelo de um neurônio artificial.	82
A.2	Funções de ativação.	83
A.3	Efeito do bias nas funções de ativação.	84
A.4	Aprendizagem supervisionada.	86
A.5	Aprendizagem não-supervisionada.	87

Lista de Símbolos

\dot{y}	derivada primeira da função y
\ddot{y}	derivada segunda da função y
k	tempo discreto
t	tempo contínuo
\mathbb{R}	Conjunto dos Números Reais
\mathbb{Z}	Conjunto dos Números Inteiros
∞	Infinito Numérico
$\bar{\cdot}$	Estado Estacionário
$ \cdot $	Módulo
D_f	Matriz Jacobiana
ρ	Autovalores da matriz Jacobiana
Δ_i	Derivada Parcial
T_p	Atraso de Tempo
τ	Ponto do Diagrama de Bifurcação
ω	Frequência em rad/s
σ^2	Variância
A	Parâmetro de Bifurcação
F	Mapeamento Não-Linear
$y(k - i)$	Regressor de Saída
$u(k - i)$	Regressor de Entrada
$v(k - i)$	Regressor de Ruído
$e(k)$	Qualquer Incerteza
$v(k)$	Ruído Branco
n_y	Máximo Atraso do Sinal de Saída
n_u	Máximo Atraso do Sinal de Entrada
n_v	Máximo Atraso do Sinal de Ruído

\mathbf{Y}	Vetor Contendo os Regressores da Saída
\mathbf{U}	Vetor Contendo os Regressores da Entrada
\mathbf{W}_L	Matriz de Pesos
w_{Lij}	Elemento (i, j) da Matriz de Pesos \mathbf{W}_L
H	Função de Ativação Tangente Hiperbólica
L	Função de Ativação Linear
G	Função de Ativação Gaussiana
β	Parâmetro de regularização
E	Erro Médio Quadrático
$J(\cdot)$	Função Custo Genérica
S	Espaço Factível
h_i	Restrições de Igualdade
∇	Gradiente
$L(\cdot, \cdot)$	Função Lagrangeana
λ	Multiplicador de Lagrange
ϵ	Tolerância do Algoritmo

Nomenclatura

ARX	Auto-Regressivo com entradas Exógenas
DB	Diagrama de Bifurcação
ERR	Taxa de Redução de Erro
GAUSS	Função de Ativação Gaussiana
LIN	Função de Ativação Linear
MLP	Perceptron de Múltiplas Camadas
NARMAX	Não-linear Auto-Regressivo de Média Móvel com Entrada Exógena
NARX	Não-linear Auto-Regressivo com Entrada Exógena
NB	Neurônio sem <i>bias</i>
OBS	Cirurgião Cerebral Ótimo
p.u.	Por Unidade
RLC	Resistivo, Indutivo e Capacitivo
RMSE	Raíz Quadrada do Erro Quadrático Médio
tanh ou TANH	Função de Ativação Tangente Hiperbólica

Capítulo 1

Introdução

As pesquisas em redes neurais são motivadas desde o seu início pelo reconhecimento de que o cérebro humano processa informação de uma forma totalmente diferente de um computador digital convencional. Diariamente tarefas são realizadas pelo cérebro humano em alguns milissegundos, enquanto tarefas bem mais simples podem levar dias para serem executadas em computadores convencionais (Haykin, 2001; Braga et al., 2000).

Além desta alta capacidade do sistema de processamento do cérebro, há características particulares deste sistema de processamento, como capacidade de aprender e generalizar (Kröse and Smagt, 1996; Hornik et al., 1988). As redes empregadas neste trabalho são redes perceptrons de múltiplas camadas (Rumelhart et al., 1986a,b) treinadas com o algoritmo Levenberg-Marquardt (Hagan and Menhaj, 1994).

Em (Narendra and Parthasarathy, 1990) foi sugerido que identificação e controle usando redes neurais podem ser melhor entendidos dentro do contexto da teoria de sistemas. Desde então muitos trabalhos abordando o uso de redes em identificação de sistemas foram produzidos, entre eles (Isermann et al., 1997; Narendra and Mukhopadhyay, 1997; Ljung and Sjöberg, 1992; Corrêa et al., 1999; Tsoi and Back, 1997; Norgaard, 1997; Sjöberg, 1995; Vega et al., 2000).

1.1 Motivação

Apesar da grande capacidade de modelagem não-linear e por serem modelos altamente não-lineares as redes geralmente estão sujeitas, como todo modelo não-linear em geral, a sobreparametrização (Aguirre and Billings, 1995a), gerando assim dinâmicas espúrias que ficam especialmente evidentes em diagramas de bifurcação.

Além disso existe o problema da rede memorizar, isto é, ela não aprende um comportamento realmente, pois incorpora também os ruídos presentes nos dados de treinamento. Redes muito flexíveis, isto é, com um grande número de parâmetros (pesos e *bias*) modelam muito bem os dados de treinamento, porém, demonstram pouca capacidade de generalização. Este problema é conhecido como dilema de polarização e variância (Gemam et al., 1992).

Algumas técnicas conhecidas como técnicas de poda, são citadas na literatura no sentido de tentar contornar este problema (Reed, 1993; Mozer and Smolensky, 1989; Le Cun et al., 1990; Hassibi et al., 1992; Hagiwara, 1990; Hinton, 1989; Bhat and McAvoy, 1992; Henrique et al., 2000).

A sobreparametrização mostra-se tão ou mais nociva ao aprendizado das redes que a presença de certos níveis de ruído. Muitas vezes apenas estratégias caixa-preta não são suficientes para obter-se modelos globais, exatamente devido a sobreparametrização.

Recentes trabalhos têm sido publicados usando redes neurais na área de identificação de sistemas dinâmicos não-lineares, em que diagramas de bifurcação são usados como critério de validação de redes treinadas. Em (Corrêa et al., 1999) é mostrado via diagrama de bifurcação, que a dinâmica global não foi aprendida pela rede. Novamente em (Billings and Coca, 1999) as redes não puderam aprender a dinâmica do sistema.

Ainda uma outra abordagem é combinar modelos semi-físicos com redes neurais (Forsell and Lindskop, 1997). Tem-se ainda ao lado destas técnicas a identificação caixa-cinza que é aquela em que, outras informações, além dos dados de entrada e saída, são levadas em consideração o que facilita o processo de seleção de estrutura do modelo.

Este trabalho consiste em investigar a possibilidade de se reproduzir a bifurcação do tipo forquilha no sistema de Duffing-Ueda. Para este fim foi utilizada informação sobre a simetria de pontos fixos, uma vez que conjectura-se que o sistema apresenta esta propriedade para o caso autônomo. Um segundo estudo de caso verifica-se a possibilidade de restringir os pesos da rede durante o treinamento de modo a impor que a rede tenha um determinado comportamento estático.

1.2 Objetivo

O objetivo principal deste trabalho é examinar, em dois estudos de caso específicos, a possibilidade e a viabilidade de restringir a arquitetura e os parâmetros de uma rede neural perceptron de múltiplas camadas na identificação de sistemas dinâmicos não-lineares usando outras informações do

sistema e não apenas dados de entrada e saída.

1.3 Apresentação do trabalho

A dissertação está organizada da seguinte forma:

Capítulo 2 - Neste capítulo são revistos alguns conceitos da teoria de sistemas dinâmicos não-lineares que são fundamentais para o entendimento do restante da dissertação. Um breve comentário é feito sobre redes neurais no contexto de identificação de sistemas dinâmicos.

Capítulo 3 - O capítulo 3 descreve os procedimentos de incorporação de informação a priori em redes MLP. É enunciado um teorema que define condições suficientes para que redes MLP apresentem simetria de pontos fixos e ponto fixo trivial. Dada a característica estática do sistema, um treinamento com restrições de igualdade é proposto no sentido da rede aprender o comportamento dinâmico assegurando o estático.

Capítulo 4 - Este capítulo apresenta um estudo de caso, para o sistema de Duffing-Ueda. A partir da informação de que o sistema apresenta bifurcação forquilha, infere-se simetria, logo as condições do teorema do capítulo 3 são impostas às redes para que estas sejam mais adequadas à identificação.

Capítulo 5 - O segundo estudo de caso é abordado. Trata-se de um aquecedor elétrico. O objetivo do estudo é, obter um modelo dinâmico e preservar aproximadamente o comportamento do sistema em estado estacionário. Para tal aplica-se o treinamento com restrições de igualdade enunciado no capítulo 3.

Capítulo 6 - Apresenta as conclusões finais da dissertação e as sugestões de trabalho futuros que foram levantadas ao longo da dissertação.

Capítulo 2

Sistemas dinâmicos e Redes Neurais

2.1 Sistemas dinâmicos

O objetivo deste capítulo é apresentar alguns conceitos necessários ao entendimento do trabalho realizado. Basicamente, pode-se dividir este capítulo em duas partes. A primeira trata de conceitos referentes à teoria de sistemas e identificação, sendo que a segunda parte trata de redes neurais no contexto de identificação de sistemas dinâmicos não-lineares.

Sistema é uma combinação de componentes que atuam conjuntamente e realizam um determinado objetivo (Ogata, 1993). Porém, o conceito pode ser entendido de maneira mais ampla, pois não se limita a algo físico. Qualquer relacionamento de variáveis físicas ou abstratas, variáveis das ciências econômicas, por exemplo, pode ser entendido como *sistema*. Um sistema pode ser classificado como estático se dada uma entrada $u(t)$, a saída $y(t)$ depende unicamente da entrada atual, e como dinâmico se a saída depende não só da entrada atual, mas também dos valores passados da entrada e da saída, por esse motivo sistemas dinâmicos são chamados de sistemas com memória. Sistemas dinâmicos são representados matematicamente por equações diferenciais em tempo contínuo (Fiedler-Ferrara and Prado, 1994) e equações de diferença em tempo discreto (Scheinerman, 1996; Spiegel, 1994). Exemplos podem ser vistos nas equações (2.1) e (2.2).

$$\ddot{y} + k\dot{y} + y = u(t) \quad t \in \mathbb{R}, \quad (2.1)$$

$$y(k) = ay(k-1) + by(k-2) + cu(k-1) + du(k-2) \quad k \in \mathbb{Z}, \quad (2.2)$$

Sistemas são autônomos se não dependem explicitamente do tempo, ou em outras palavras, o sistema não possui entradas. E em caso contrário, isto é,

de depender explicitamente do tempo o sistema é chamado de não-autônomo.

2.1.1 Atrator

Atrator é uma região do espaço de estados para o qual a trajetória de um sistema dissipativo converge em regime permanente, ou seja ($t \rightarrow \infty$). A dimensão do atrator será necessariamente menor do que o número de graus de liberdade do sistema. O único atrator possível para a dinâmica de sistemas lineares autônomos estáveis é a origem do seu espaço de estados. Para sistemas não-lineares destacam-se os pontos fixos (dimensão 0), os ciclos limite (dimensão 1), os toróides (dimensão 2) e caos (dimensão maior que 2 e normalmente fracionária). É chamado de bacia de atração o lugar geométrico das condições iniciais de um sistema dinâmico dissipativo que em regime permanente convergem ao mesmo atrator.

2.1.2 Pontos Fixos

A análise de sistemas dinâmicos, em geral, começa pelos seus pontos fixos (Guckenheimer and Holmes, 1983). Ponto fixo é um estado do sistema no qual este permanece estático. Em termos matemáticos é quando no caso contínuo todas as derivadas da saída são nulas ou no caso discreto, todos os atrasos forem iguais $\bar{y} = y(k-1) = \dots = y(k-n_y)$ em que \bar{y} é o valor do ponto fixo, o mesmo para os atrasos da entrada $\bar{u} = u(k-1) = \dots = u(k-n_u)$.

Estabilidade de pontos fixos

Pontos fixos, no caso de sistemas lineares apresentam três cenários de estabilidade (Scheinerman, 1996):

- **Estável.** Os autovalores da matriz jacobiana são $\rho < 0$ para o caso contínuo ou $|\rho| < 1$ para o caso discreto.
- **Marginalmente estável.** Difere do caso anterior, apenas no fato de que pelo menos um dos autovalores está sobre o eixo $j\omega$, no caso contínuo e sobre o círculo de raio unitário, no caso discreto, mas não necessariamente todos.
- **Instável.** Se pelo menos um dos autovalores da matriz jacobiana são $\rho > 0$, para o caso contínuo, ou $|\rho| > 1$ para o caso discreto.

No caso de sistemas não-lineares uma alternativa de análise é linearizar o sistema em torno do ponto fixo, tal tarefa é feita usando a matriz jacobiana. A matriz jacobiana dos modelos tratados neste trabalho, está representada na equação (2.3).

$$Df = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & 0 \\ 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & 1 \\ \Delta_{n_y} & \Delta_{n_y-1} & \Delta_{n_y-2} & \dots & \Delta_1 \end{bmatrix} \quad (2.3)$$

Em que:

$$\begin{aligned} \Delta_1 &= \left. \frac{\partial y(k)}{\partial y(k-1)} \right|_{\bar{y}, \bar{u}} \\ &\vdots \\ \Delta_{n_y} &= \left. \frac{\partial y(k)}{\partial y(k-n_y)} \right|_{\bar{y}, \bar{u}} \end{aligned}$$

2.1.3 Ciclos limite

Ciclos limite são comportamentos periódicos, em estado estacionário do sistema. Aparecem em sistemas que possuem componentes conservativas e componentes dissipativas. Um exemplo cotidiano é um pêndulo de um relógio de parede (Fiedler-Ferrara and Prado, 1994).

2.1.4 Caos

Caos é um regime dinâmico caracterizado por grande sensibilidade a condições iniciais. Um sistema caótico possui direções de instabilidade local no seu espaço de estados. Por esse motivo, minúsculas variações nas condições iniciais do sistema serão amplificadas exponencialmente ao longo de tais direções de instabilidade.

2.1.5 Estabilidade Estrutural

Além da estabilidade de uma solução estacionária, existe um outro conceito de estabilidade muito importante trata-se do conceito de estabilidade estrutural. Enquanto no estudo da estabilidade assintótica e de Lyapunov

a estabilidade é investigada perturbando-se o sistema, no estudo da estabilidade estrutural o que é verificado é se o retrato de fase é alterado qualitativamente devido à variação de um parâmetro do sistema (Scheinerman, 1996).

2.1.6 Bifurcação

Quando o sistema perde a estabilidade estrutural diz-se que houve uma bifurcação. Em relação aos pontos fixos do sistema pode-se observar variação no número e na sua estabilidade.

Bifurcação forquilha (pitchfork)

Considere a equação normal:

$$\dot{x} = f(x, \mu) = \mu x - x^3, \quad x \in \mathbb{R}, \mu \in \mathbb{R}, \quad (2.4)$$

Os pontos fixos deste sistema são (Wiggins, 1990):

$$x = 0, \quad (2.5)$$

$$x = \pm\sqrt{\mu}, \quad (2.6)$$

Como pode ser deduzido pela equação (2.6), para valores $\mu < 0$ só se tem o ponto fixo trivial e para valores de $\mu > 0$ dois novos pontos fixos surgem. A essa variação do número de pontos fixos devido a variação do parâmetro μ é que dá-se o nome de bifurcação forquilha. É importante observar que neste tipo de bifurcação a estabilidade dos pontos fixos também é alterada com a variação do parâmetro μ . No entanto, para este trabalho apenas a análise do número de pontos fixos é relevante. Duas observações importantes a serem feitas:

1. Existe um ponto fixo trivial,
2. Os pontos fixos não triviais são simétricos em relação ao ponto fixo trivial.

A análise foi feita considerando apenas pontos fixos reais (Wiggins, 1990).

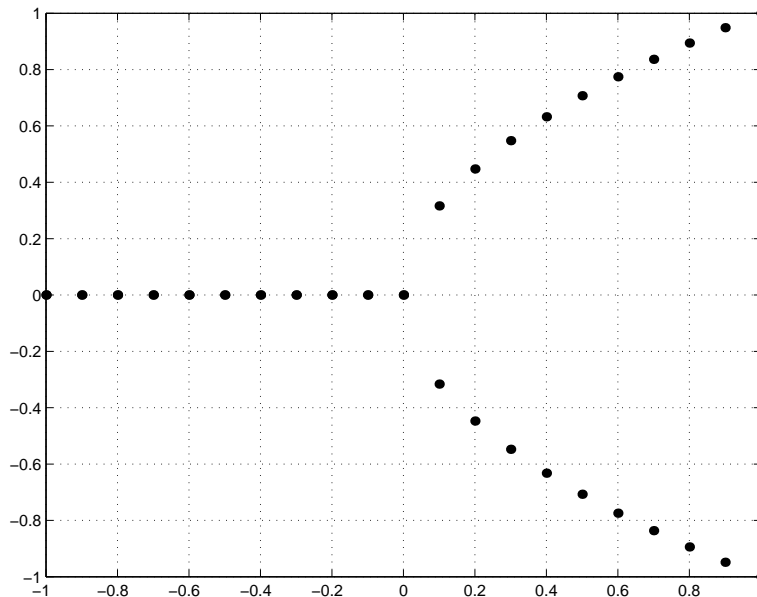


Figura 2.1: Diagrama de bifurcação da equação 2.4. O eixo horizontal é referente ao parâmetro de bifurcação e o eixo vertical os valores de pontos fixos.

2.2 Identificação de Sistemas Dinâmicos

2.2.1 O processo de identificação

O processo de identificação consiste basicamente de 4 etapas:

- **Aquisição dos dados.** Os dados são às vezes gravados durante um projeto de experimento, onde o pesquisador deve determinar quais sinais medir, quando medi-los e também escolher os sinais de entrada para o sistema. O projeto de experimento é feito de acordo com estas escolhas, entre outras, de modo que os dados possam fornecer o máximo de informações sobre o sistema. Em outros casos o pesquisador não tem a possibilidade de realizar um experimento, devendo então usar os dados de operação do sistema (Ljung, 1987).
- **Escolha da representação e da estrutura.** Em relação a escolha de representação o que na prática ocorre é que cada pesquisador utiliza a representação que lhe é mais “familiar”, ou seja, aquela que lhe é mais conhecida. Uma discussão mais aprofundada do assunto pode ser encontrado em (Aguirre, 2000b). Quanto a escolha de estrutura cada

representação possui os seus métodos específicos de escolha de estrutura, em redes neurais, por exemplo, existem as redes construtivas, que são aquelas que aumentam o número de neurônios durante o processo de treinamento, técnicas de poda (pruning), que tentam eliminar os pesos desnecessários da rede após o treinamento e uso de conhecimento a priori que será tratado neste trabalho.

- **Estimação de parâmetros.** Esta etapa consta da busca dos parâmetros tais que o modelo possa reproduzir o comportamento do sistema. Em relação às redes neurais esta etapa corresponde ao treinamento da rede.
- **Validação.** Um modelo nunca é capaz de representar perfeitamente um sistema real, assim testa-se o modelo para verificar se este reproduz bem uma determinada característica do sistema, segundo algum critério.

Um breve histórico sobre a área pode ser encontrado em (Ljung, 1985).

2.2.2 Validação de modelos com dinâmica caótica

Como mencionado anteriormente, sistemas caóticos apresentam grande sensibilidade a condições iniciais, essa propriedade faz com que seja muito difícil comparar a resposta simulada à resposta do sistema, pois pequenas variações nas condições iniciais, ou nos parâmetros do modelo, resultarão em divergência. Conseqüentemente, apenas a comparação no domínio do tempo, da simulação do modelo com dados medidos, não serve como critério para validar o respectivo modelo. Devido a sensibilidade a condições iniciais, a comparação ponto a ponto é dificultada. Um forma de contornar esta dificuldade é trabalhar num espaço onde o tempo não seja explicitamente representado. Esse problema é conhecido como reconstrução do espaço de estados (Aguirre, 1996a,b).

Reconstrução do espaço de estados. O procedimento consiste basicamente em representar $y(t)$ em relação a $y(t - T_p)$. A motivação para isto é que de alguma forma $y(t - T_p)$ esta relacionado a $\dot{y}(t)$ e conseqüentemente as trajetórias representadas no pseudo-plano de fase tem propriedades similares ao atrator original do sistema (Moon, 1987).

Seção de Poincaré. A seção de Poincaré é outra ferramenta de análise, esta permite verificar um corte do atrator. A seção de Poincaré pode ser obtida definindo um plano que deverá ser transversal ao fluxo no espaço de estados. De maneira mais formal tem-se que, a seção de Poincaré, é uma hipersuperfície de dimensão $(n - 1)$ que é transversal às trajetórias no espaço de estados Figura 2.2.

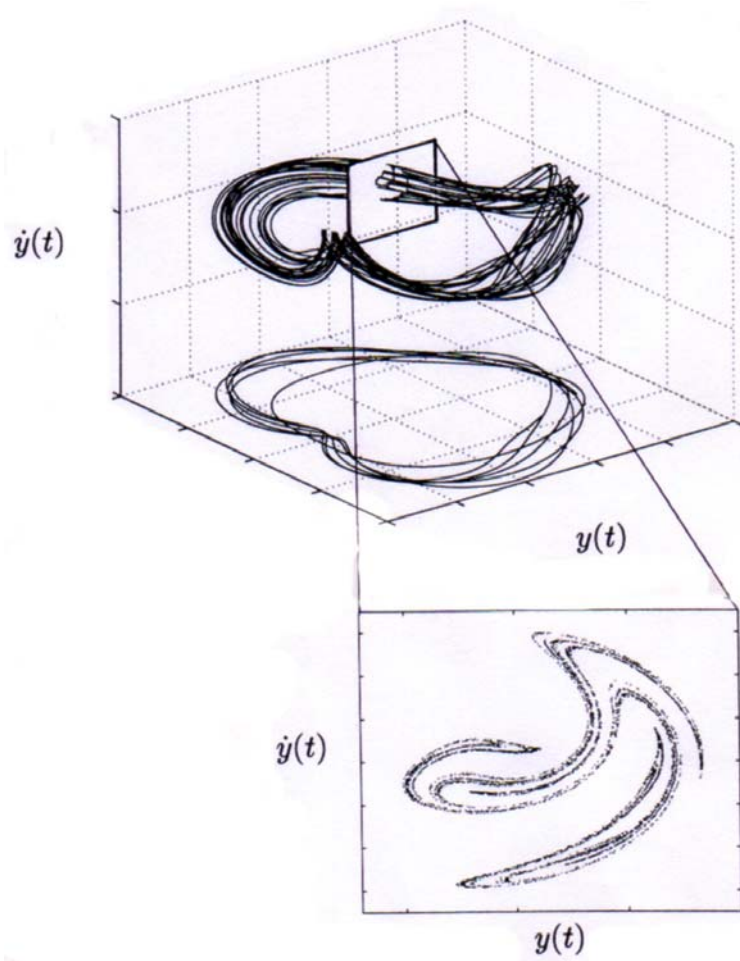


Figura 2.2: Espaço de estados do sistema de Duffing-Ueda em regime caótico e uma de suas seções de Poincaré.

Diagrama de bifurcação. O diagrama de bifurcação mostra os diversos tipos de regimes dinâmicos apresentados por um sistema quando varia-se o parâmetro de bifurcação. Um ponto r do diagrama de bifurcação de um sistema não-autônomo excitado por $A\cos(\omega t)$ com A como parâmetro de bifurcação é definido como:

$$\tau = \{(y, A) \in \mathbb{R} \times I \mid y = y(t_i), A = A_0; \\ t_i = t_0 + K_{ss} \times 2\pi/\omega\}, \quad (2.7)$$

Sendo I é o intervalo $I = [A_i \ A_f] \subset \mathbb{R}$, $0 \leq t_0 \leq 2\pi/\omega$.

O ponto τ é obtido por simulação do sistema por um tempo suficientemente longo para atingir o regime permanente, então é feito o gráfico de $y(K_{ss} \times 2\pi)$ por A_0 . Para cada valor de A tem-se n_b pontos. Esta forma de determinar o diagrama de bifurcação usa força bruta e portanto só mostra características estáveis do sistema. Há outras formas, por exemplo o método de continuação, que não sofre destas dificuldades (Parker and Chua, 1989).

2.2.3 Modelos NARX e NARMAX

A estrutura de um modelo NARX com período de amostragem normalizado é (Leontaritis and Billings, 1985a,b):

$$y(k) = F[y(k-1), y(k-2), \dots, y(k-n_y), u(k-d-1), \dots \\ \dots, u(k-d-n_u+1)] + e(k), \quad (2.8)$$

em que $k = 1, \dots, N$, F é uma função não-linear qualquer, $y(k)$ e $u(k)$ são respectivamente a saída e a entrada do sistema, que têm seus atrasos representados por n_y e n_u , d representa o tempo de retardo do sistema e $e(k)$ representa incertezas de forma geral. A estrutura de um modelo NARMAX com período de amostragem normalizada é dada pela equação (2.9) a seguir.

$$y(k) = F[y(k-1), y(k-2), \dots, y(k-n_y), u(k-d-1), \dots \\ \dots, u(k-d-n_u+1), \dots, v(k-1), v(k-2), \dots \\ \dots, v(k-d-n_v)] + v(k), \quad (2.9)$$

A diferença dos modelos NARX e NARMAX é a possibilidade de incorporar regressores de ruído, o que evita a polarização do modelo. Nesse caso a incerteza $e(k)$ passa a ser ruído branco $v(k)$. O termo n_v é o máximo atraso de v .

2.3 Redes neurais em identificação de sistemas

Redes neurais foram e são amplamente utilizadas em identificação de sistemas estáticos (reconhecimento de padrões). Em (Narendra and Parthasarathy, 1990) as redes foram sugeridas para proposições de identificação de sistemas dinâmicos, principalmente devido a capacidade de generalização e pelo fato de serem aproximadores universais (Hornik et al., 1988). Ainda uma outra razão que justifica o uso de redes em problemas práticos de identificação é o fato de uma grande classe de problemas que ocorrem na prática são problema que apresentam algum tipo de saturação, e as redes neurais, por construção, já apresentam na estrutura essa característica (Ljung and Sjöberg, 1992).

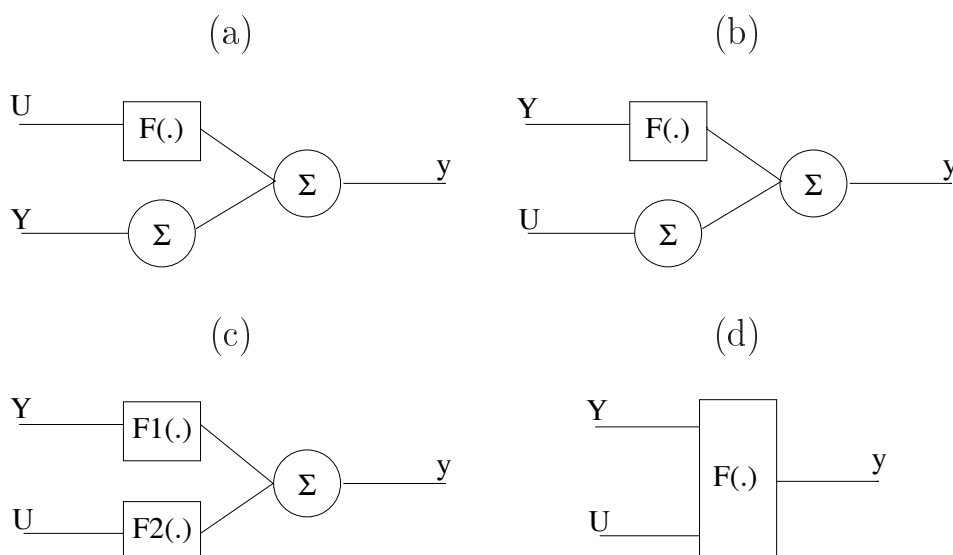


Figura 2.3: Arquiteturas sugeridas em (Narendra and Parthasarathy, 1990). (a) Os regressores da entrada passam por um bloco não-linear (conjunto de neurônios com função de ativação não-linear) e o regressores da saída por um bloco linear (b) Os regressores da saída passam por um bloco não-linear e o regressores da entrada por um bloco linear (c) Tanto os regressores da entrada quanto os regressores da saída passam por blocos não-lineares, porém os blocos são distintos (d) Os regressores da entrada e da saída passam pelo mesmo bloco não-linear

Várias topologias foram sugeridas em (Narendra and Parthasarathy, 1990), como pode ser verificado na Figura 2.3, sendo, Y e U representam respectivamente o regressores referentes as saídas e as entradas, y é o valor da saída no instante t e as funções representadas por F , $F1$ e $F2$ são funções não-lineares.

Em (Isermann et al., 1997), arquiteturas de redes dinâmicas são classificadas em redes com dinâmicas internas e externas. E a capacidade de modelar sistemas dinâmicos não-lineares complexos destas redes é estudada. As redes usadas neste trabalho serão redes com recorrência externa, isto é, redes perceptron de múltiplas camadas estáticas em que na entrada são aplicadas os valores passados da entrada e da saída do sistema (regressores). Estas redes são modelos NARX Figura 2.4. Um outro estudo que trata de arquiteturas de redes dinâmicas pode ser encontrado em (Tsoi and Back, 1997), principalmente as redes com recorrência interna.

Em (Ljung and Sjöberg, 1992; Sjöberg, 1995) redes neurais são vistas dentro da teoria de identificação de sistemas.

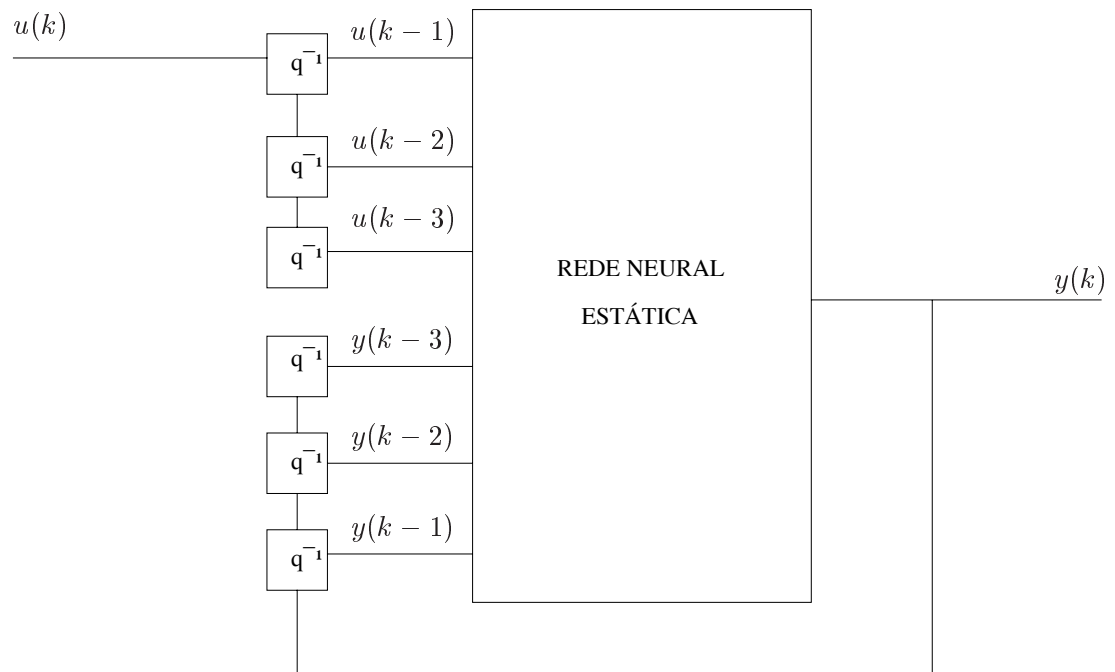


Figura 2.4: Rede neural NARX com 3 atrasos da entrada e 3 atrasos da saída.

2.4 Considerações finais

Este capítulo reviu conceitos da teoria de sistemas dinâmicos não-lineares, que são necessários ao bom entendimento do trabalho desenvolvido. Além destes conceitos foram também citados alguns trabalhos que ajudam a situar o uso de redes neurais no contexto de identificação de sistemas dinâmicos.

Capítulo 3

Uso de informação a priori em redes MLP

3.1 Introdução

O uso de informação sobre o relacionamento funcional das variáveis de um sistema, em redes, ainda é algo que deve ser melhor entendido, para que as potencialidades das redes possam ser melhor exploradas principalmente no âmbito de sistemas dinâmicos não-lineares.

Tendo em vista a necessidade de melhor entender este relacionamento, e não apenas seu uso como modelo caixa preta, este capítulo sugere metodologias que objetivam contribuir nesse sentido. Tais metodologias permitem a incorporação de informação sobre simetria de pontos fixos do sistema autônomo e característica estática a priori em redes neurais do tipo perceptron de múltiplas camadas alimentadas adiante.

A primeira das metodologias parte de um conhecimento qualitativo dos pontos fixos de alguns sistemas dinâmicos não-lineares, a simetria de pontos fixos no caso autônomo. Tal conhecimento pode ser obtido via diagrama de bifurcação, sendo que, uma vez detectada bifurcação do tipo forquilha, pode-se inferir a presença de pontos fixos simétricos e ponto fixo trivial (Guckenheimer and Holmes, 1983). Em geral, a obtenção do diagrama de bifurcação em situações práticas nem sempre é possível, pois é necessária a aplicação de vários sinais de entrada na planta que se deseja identificar. Apesar disso, já existem alguns trabalhos em que tal diagrama foi obtido através de séries temporais (Aguirre et al., 2001; Bagarinao et al., 1999a,b,c), o que futuramente pode ser um caminho no sentido de contornar a dificuldade de obter-se o diagrama. Uma outra maneira de verificar a simetria de pontos fixos, é obtê-los diretamente a partir dos dados (Aguirre and Souza, 1998; Coelho,

2002).

A segunda informação utilizada é a característica estática do sistema. Esta informação pode ser obtida, por exemplo, aplicando vários patamares de sinal contínuo na planta e coletando o sinal de saída após o transitório.

Este capítulo está organizado da seguinte maneira; a seção 3.2 apresenta o objetivo do capítulo, na seção 3.3 é discutido brevemente sobre o cálculo de pontos fixos em redes MLP, a seção 3.4 apresenta um teorema em que, condições de suficiência são enunciadas para que redes MLP apresentem pontos fixos triviais e quando apresentar pontos fixos não-trivias, estes serem simétricos, a seção 3.5 apresenta uma metodologia que apresenta um treinamento com restrição quando a informação sobre a curva estática é disponível, a seção 3.5 comenta sucintamente os algoritmos utilizados na dissertação, por fim a seção 3.6 fornece considerações finais do capítulo.

3.2 Objetivo

A problemática que se levantou neste trabalho é: supondo que no processo de identificação de sistemas se tenha informação sobre o comportamento em estado estacionário, como incorporar as características conhecidas do sistema na arquitetura e no treinamento do modelo neural que se deseja obter?

A importância do desenvolvimento deste procedimento está relacionada a dois aspectos, o primeiro deles é que tal metodologia pode ajudar no processo de aprendizado (estimação de parâmetros), uma vez que determinadas características, conhecidas a priori, não precisarão ser aprendidas. E o segundo é que, modelos que incorporam informações a priori tendem a ser mais semelhantes ao sistema físico, e por isso mesmo mais parsimoniosos.

3.3 Pontos fixos em redes MLP

O cálculo de pontos fixos em redes neurais MLP é feito de maneira numérica para a maioria das arquiteturas usadas, pois, as funções de ativação geralmente são funções sigmoidais.

A seguir serão comentadas algumas maneiras de se obter pontos fixos em redes neurais.

3.3.1 Obtenção de pontos fixos (caso autônomo)

Os modelos neurais usados neste trabalho foram do tipo NARX (*Não-linear AutoRegressivo com entradas eXógenas*). Estes modelos explicam a saída do sistema como uma função dos termos passados da entrada e dos termos passados da saída do sistema,

$$y(k) = F(y(k-1), \dots, y(k-n_y), u(k-1), \dots, u(k-n_u)), \quad (3.1)$$

sendo n_y e n_u , respectivamente, os máximos atrasos da saída e da entrada do sistema, e $F(\cdot)$ uma função não-linear qualquer.

O ponto fixo para o caso autônomo é o ponto que satisfaz a relação $y(k) = y(k-1) = \dots = y(k-n_y) = \bar{y}$, sendo $u(k) = u(k-1) = \dots = u(k-n_u) = 0$. Para o caso não autônomo, a cada patamar de sinal de entrada \bar{u} calculam-se os pontos fixos. Para calcular tais pontos, fixam-se os termos atrasados da entrada $u(k) = u(k-1) = \dots = u(k-n_u) = \bar{u}$ na equação (3.1) e calculam-se os valores de \bar{y} correspondentes. Assim no ponto fixo a equação (3.1) se torna,

$$\bar{y} = F(\bar{u}, \bar{y}). \quad (3.2)$$

É importante observar que o número de pontos fixos depende da função $F(\cdot)$, portanto, pode ser maior que um.

Pode-se obter os pontos fixos de diversas maneiras, por exemplo:

- **Método de Newton**

O método de Newton pode ser utilizado para encontrar as raízes da equação $\bar{y} - F(\bar{u}, \bar{y}) = 0$, que são os pontos fixos.

- **Simulação numérica**

A simulação numérica consiste em fixar um patamar de entrada \bar{u} e atribuir valores iniciais aos termos referentes às saídas do sistema e simular a rede até que a variação da saída seja menor do que uma tolerância $0 < \epsilon \ll 1$.

- **Analiticamente**

O cálculo analítico, geralmente feito apenas para um tipo específico de arquitetura proposta em (Narendra and Parthasarathy, 1990). Esta arquitetura possui a particularidade de possuir apenas não-linearidade nos termos referentes à entrada, como pode ser melhor descrito na equação (3.3),

$$y(k) = F(u(k-1), \dots, u(k-n_u)) + a_1 y(k-1), \dots, a_{n_y} y(k-n_y), \quad (3.3)$$

sendo $F(\cdot)$ uma função não-linear. No ponto fixo tem-se

$$\bar{y} = a_t \bar{y} + F(\bar{u}) \quad (3.4)$$

em que a_t é a soma dos coeficientes dos termos lineares. O ponto fixo pode então ser calculado analiticamente pela equação 3.5,

$$\bar{y} = \frac{1}{(1 - a_t)} F(\bar{u}). \quad (3.5)$$

- **Método da “biseção”**. Este método consiste nos seguintes passos:
 1. Determinação do intervalo de busca. Este intervalo é igual a $|x| \leq \sum_{i=1}^n w_{2_i}$, sendo $\sum_{i=1}^n w_{2_i}$, a soma dos pesos da camada de saída e x um valor qualquer dentro do intervalo.
 2. Avaliar a função $F(\bar{u}, \bar{y})$ no intervalo de busca. Para isso, fixa-se \bar{u} no patamar desejado e atribuem-se valores para \bar{y} que cobram todo o intervalo de busca.
 3. Calcular a diferença entre cada ponto de $F(\bar{u}, \bar{y})$ e da função $y = \bar{y}$ (função identidade)
 4. Verificar onde estão as regiões de crescimento e decrescimento da função diferença, estes subintervalos são regiões em que cada ponto fixo será procurado individualmente.
 5. Depois de determinar os subintervalos, repetir os passos de 2 a 4, até que o ponto fixo seja encontrado com a precisão desejada.

A simulação possui a desvantagem de possibilitar apenas a obtenção de pontos fixos estáveis, o cálculo analítico está restrito a uma arquitetura muito específica, o método de Newton é o mais utilizado, porém é necessário avaliar a rede e sua derivada. O método da “biseção” apresenta-se como uma alternativa, pois não necessita da avaliação da derivada.

3.4 Metodologia para imposição de restrições dada simetria de pontos fixos

A seguir será enunciado um teorema que descreve condições de suficiência para que redes MLP apresentem ponto fixo trivial e quando houver pontos fixos não-triviais estes serem simétricos. Tal análise foi feita para o caso autônomo.

Redes simétricas

Teorema: Um modelo NARX, tendo como função não-linear uma rede MLP alimentada adiante (*feedforward*), em que nenhum dos neurônios possua termo de polarização (*bias*) e todas as funções de ativação são contínuas e ímpares, apresenta ponto fixo trivial e simetria de pontos fixos (se houver pontos fixos não-triviais), no caso autônomo.

Prova:

Para garantir simetria de pontos fixos e ponto fixo trivial basta mostrar que toda rede (o mapa) é uma função contínua e ímpar. Neste tipo de rede, os pontos fixos apresentam as características desejadas,

- $f(0) = 0$,
- $-\bar{y} = f(-\bar{y})$.

Continuidade - Se cada neurônio é uma função contínua a rede também será, pois a composição de funções contínuas também é uma função contínua.

Ímpar - A rede neural NARX, enunciada acima pode ser escrita matematicamente como

$$y(k) = f_{\text{saida}} \left(\sum_{i=1}^{Nn} w_{2_{1i}} f_i \left(\sum_{j=1}^{n_y} w_{1_{ij}} y(k-j) \right) \right). \quad (3.6)$$

Para o caso autônomo $y = y(k-1) = \dots = y(k-n_y)$, o que faz com que a equação (3.6) se torne:

$$y = f_{\text{saida}} \left(\sum_{i=1}^{Nn} w_{2_{1i}} f_i (w_{\text{soma}_i} y) \right) \quad (3.7)$$

Sendo

$$w_{soma_i} = \sum_{j=1}^{ny} w_{1_{ij}}. \quad (3.8)$$

Para $y = -a$ tem-se,

$$-a = f_{saida} \left(\sum_{i=1}^{Nn} w_{2_{1i}} f_i (w_{soma_i}(-a)) \right), \quad (3.9)$$

$$-a = f_{saida} \left(\sum_{i=1}^{Nn} w_{2_{1i}} f_i (-(w_{soma_i}a)) \right), \quad (3.10)$$

Como a função f_i é uma função ímpar a equação 3.10 se torna:

$$-a = f_{saida} \left(\sum_{i=1}^{Nn} w_{2_{1i}} (-f_i (w_{soma_i}a)) \right), \quad (3.11)$$

$$-a = f_{saida} \left(\sum_{i=1}^{Nn} -(w_{2_{1i}} f_i (w_{soma_i}a)) \right). \quad (3.12)$$

Como a função do neurônio da saída também é ímpar, tem-se:

$$-a = -f_{saida} \left(\sum_{i=1}^{Nn} w_{2_{1i}} f_i (w_{soma_i}a) \right). \quad (3.13)$$

Logo igualando-se (3.9) e (3.13) tem-se que, $f(-a) = -f(a)$, conclui-se então, que a função descrita pela rede é ímpar. c.q.d.

É importante ressaltar que esta propriedade se verifica independentemente do número de camadas, neurônios e saídas da rede e de qual ou quais funções ímpares foram usadas. Algo importante a se ressaltar é que uma mesma rede pode ter neurônios com funções de ativação diferentes na camada oculta, e ainda assim preservar as características do teorema.

As informações dos pontos fixos podem ajudar na escolha da arquitetura. Dado que um determinado sistema apresente simetria de pontos fixos para o caso autônomo e ponto fixo trivial, pode-se então ser usada as condições definidas no teorema acima, isto é, que os neurônios não possuam o termo de polarização e as funções de ativação sejam funções ímpares. Isso será ilustrado no capítulo 4.

3.5 Metodologia para imposição de restrições dada a curva estática

A metodologia descrita a seguir destina-se a sistemas que apresentam apenas um ponto fixo para cada patamar do sinal de entrada.

Inicialmente verifica-se os dados, pois estes podem ajudar na escolha da função de ativação dos neurônios da camada oculta. Os regressores referentes à entrada do sistema entrarão nos neurônios não-lineares da camada oculta. Se a função estática puder ser representada por uma função par ou ímpar então a função de ativação dos neurônios não-lineares devem ser respectivamente pares ou ímpares, sem o termo de polarização. Os regressores referentes a saída do sistema entram no neurônio linear da camada oculta. Observação: se a função estática não puder ser representada nem por uma função par, nem por uma ímpar, então se usa qualquer função de ativação com o termo de polarização, pois se sabe que redes com uma camada oculta podem aproximar qualquer função contínua. A seguir será apresentada a metodologia para incorporação de conhecimento nas redes dada característica estática.

PASSO 1 - Determinação da arquitetura. Exemplo: Os dados da função estática representam uma função ímpar. Pode-se então utilizar a rede descrita na equação 3.14,

$$y(k) = w_{221} w_{111} y(k-1) + w_{222} w_{112} y(k-2) + w_{211} \tanh(w_{113} u(k-1) + w_{114} u(k-2)), \quad (3.14)$$

sendo \tanh a função tangente hiperbólica.

PASSO 2 - Reescrever a rede em sua forma estática, isto é, substituindo todos os regressores da saída e entrada por \bar{y} e \bar{u} , respectivamente. Em seguida agrupar os termos.

A rede do exemplo torna-se,

$$y = \left(\frac{w_{211}}{1 - w_{221} w_{111} - w_{222} w_{112}} \right) \tanh((w_{113} + w_{114}) \bar{u}). \quad (3.15)$$

Fazendo as seguintes substituições $\left(\frac{w_{211}}{1 - w_{221} w_{111} - w_{222} w_{112}} \right) = a_1$ e $(w_{113} + w_{114}) = a_2$, a equação (3.14) torna-se,

$$y = a_1 \tanh(a_2 u), \quad (3.16)$$

que é a equação da rede neural estática relacionada.

PASSO 3 - Utilizar os dados da curva estática para treinar a rede obtida no passo anterior. No caso do exemplo, a rede neural da equação (3.16). Tal treinamento pode ser feito, por exemplo, via algoritmo de retropropagação (*backpropagation*).

PASSO 4 - Os pesos e os termos de polarização obtidos no passo anterior serão usados para se escrever as restrições. Observação: o número de neurônios na camada oculta determina quantas restrições serão necessárias.

PASSO 5 - treinar a rede dinâmica resolvendo o problema de otimização sujeito a restrições de igualdade.

$$\begin{aligned} \min \quad & \sum_{i=1}^N \frac{1}{N} (y - \hat{y})^2 \\ \text{Sujeito a} \quad & \left(\frac{w_{211}}{1 - w_{221}w_{111} - w_{222}w_{112}} \right) = a_1 \\ & (w_{113} + w_{114}) = a_2 \end{aligned} \tag{3.17}$$

PASSO 6 - Validar tanto estaticamente quanto dinamicamente a rede.

Este procedimento será ilustrado no capítulo 5.

3.5.1 O treinamento como um problema de otimização

Os problemas tratados na dissertação apresentam apenas restrições de igualdade e podem ser enunciados como na equação abaixo:

$$\begin{aligned} \text{Minimize} \quad & J(x) \\ & x \in S \\ \text{Sujeito a} \quad & h_i(x) = 0 \quad i = 1, \dots, m \end{aligned} \tag{3.18}$$

sendo $J(x)$ a função objetivo, $h_i(x)$ as restrições do problema, m é o número de restrições e $S \subseteq \mathbb{R}^n$, isto é, o conjunto de soluções viáveis.

A função objetivo

A função objetivo consiste no erro quadrático médio. Como pode ser observado, o valor da função objetivo depende dos dados disponíveis e da arquitetura escolhida pela rede. É importante ressaltar que a escolha de uma arquitetura inadequada a um determinado problema gera uma superfície de erro com muitos mínimos locais (Dodd, 1992) o que pode dificultar o aprendizado da rede.

As restrições

As restrições são obtidas pelo ajuste da rede em sua forma estática aos dados em estado estacionário. Os pesos e/ou termos de polarização que foram agrupados quando a rede dinâmica foi escrita em sua forma estática devem manter o mesmo valor de parâmetro que a rede estática obteve, isto é, a rede dinâmica deve aprender o comportamento dinâmico, preservando as relações que garantam a característica estática.

3.5.2 Condições de otimalidade de Kuhn-Tucker

As condições de otimalidade de Kuhn-Tucker (Takahashi, 2001; Mateus and Luna, 1986) determinam as condições necessárias para existência de mínimo no problema (3.18).

$$\begin{aligned} \nabla f(\bar{x}) + \sum_{i=1}^m \bar{\lambda}_i \nabla h_i(\bar{x}) &= 0 \\ \bar{\lambda}_i &\geq 0 \quad i = 1, \dots, m \end{aligned} \quad (3.19)$$

em que ∇ , representa o gradiente das funções.

Os algoritmos

No caso de problemas restritos o que se faz, em geral, é transformar um problema restrito em um problema irrestrito que passa a ser base de um processo iterativo. Tal transformação é feita utilizando a função Lagrangeana que consiste na soma da função objetivo e ponderações das restrições, como visto a seguir

$$L(x, \lambda) = f(x) + \sum_{i=1}^m \lambda_i h_i(x), \quad (3.20)$$

sendo, λ_i os multiplicadores de Lagrange e m o número de restrições. Um algoritmo para este problema pode ser enunciado abaixo:

1. Escolher valores de x em S e valores de $\lambda > 0$.
2. Determinar a solução ótima x^* do problema:

$$\min \quad J(x) + \sum_{i=1}^m \lambda_i h_i(x) \quad (3.21)$$

utilizando algum algoritmo de minimização irrestrita, por exemplo, o algoritmo de Levenberg-Marquardt (Hagan and Menhaj, 1994).

3. Maximizar a Lagrangeana em relação aos multiplicadores.
Observação: $\max [L(x, \lambda)] = -\min [-L(x, \lambda)]$.
4. Repetir os passos 2 e 3 até que as condições de Kuhn-Tucker sejam satisfeitas.

3.6 Considerações finais

Foram apresentados procedimentos que objetivam permitir usar conhecimento prévio tanto na definição da arquitetura quanto no treinamento de redes MLP.

Dois tipos de conhecimento prévio foram tratados. Em primeiro lugar, considera-se o caso em que há informação sobre a simetria de pontos fixos em estado estacionário. Apresentou-se um teorema com sua respectiva prova que define condições suficientes para que redes MLP apresentem ponto fixo trivial e simetria de pontos fixos (quando houver pontos não-triviais). Tais características são observadas em sistemas que apresentam bifurcações do tipo forquilha.

Em segundo lugar, o conhecimento da característica estática do sistema permitiu a definição de um procedimento para o treinamento de redes com restrições. Isto é, a rede é treinada para aprender o comportamento dinâmico levando em consideração restrições sobre o desempenho em estado estacionário. Isso permite obter redes com desempenho estático previamente determinado.

Capítulo 4

Identificação do Sistema de Duffing-Ueda

4.1 Introdução

Principalmente com o trabalho de (Narendra and Parthasarathy, 1990), redes neurais vêm sendo bastante utilizadas em identificação de sistemas de processos em geral, devido à sua alta capacidade de ajuste a funções não-lineares. Porém, esta qualidade, se não usada com o devido cuidado pode conduzir a dinâmicas espúrias como consequência de sobreparametrização (Aguirre and Billings, 1995a), que é caracterizada por modelos com número maior de parâmetros do que o necessário para um dado problema. No contexto das redes neurais, tais parâmetros são os pesos (conexões) e o termo de polarização (*bias*). O número de neurônios e a função de ativação também são determinantes.

Ter a capacidade de representar um conhecimento e aprendê-lo, são questões diferentes, em geral redes sobreparametrizadas podem representar um comportamento, mas aprender pode ser muito difícil (Dodd, 1992). Em (Billings and Coca, 1999; Corrêa et al., 1999) pode-se exemplificar tal dificuldade, redes neurais MLP (Multilayer Perceptron) foram treinadas para aprender o comportamento do sistema de Duffing-Ueda, no entanto, nenhum dos modelos conseguiram reproduzir o comportamento global do sistema, conforme constatado via diagrama de bifurcação.

As técnicas de poda, ocasionalmente podem ajudar a contornar problemas como este (Reed, 1993; Mozer and Smolensky, 1989; Le Cun et al., 1990; Hassibi et al., 1992; Hagiwara, 1990; Hinton, 1989; Bhat and McAvoy, 1992; Henrique et al., 2000).

Diante desse cenário surge a necessidade de entender como obter topolo-

gias que sejam mais adequadas, uma vez que redes completas, isto é, redes que possuem todas as conexões, em geral têm maiores dificuldades de aprendizado. Uma das alternativas, é a identificação caixa-cinza, que é um procedimento de identificação de sistemas que utiliza além dos dados de entrada e saída, outras informações disponíveis sobre o sistema.

O objetivo principal deste capítulo é utilizar conhecimento a priori para auxiliar na escolha da arquitetura de uma rede neural tipo perceptron de múltiplas camadas que seja mais adequada à identificação do comportamento dinâmico do sistema de Duffing-Ueda. Além disso é verificado como a rede restrita se comporta em relação a variação do número de neurônios da camada oculta, dados de identificação em que a função de excitação seja “pobre” em relação a informação dinâmica e dados com ruído.

4.2 Sistema dinâmico de Duffing-Ueda

O sistema descrito pela equação (4.1) é um circuito elétrico RLC mostrado na Figura 4.1. O termo cúbico se deve à saturação magnética do núcleo do indutor (Ueda, 1993).

$$\ddot{y} + k\dot{y} + y^3 = u(t) \quad (4.1)$$

sendo que, $k = 0, 1$ e a entrada $u(t) = A\cos(\omega t)$. A freqüência ω usada neste trabalho é $\omega = 1\text{rad/s}$ e o parâmetro A (amplitude do sinal de entrada) é variado na faixa $0 \leq A \leq 12$.

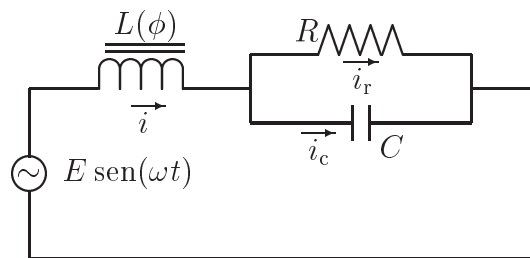


Figura 4.1: O oscilador de Duffing-Ueda

4.2.1 Dedução da equação

Através da lei de Kirchoff das malhas pode-se escrever a equação do circuito como (Ueda, 1993):

$$n \frac{d\phi}{dt} + R_{iR} = E \text{sen}(\omega t), \quad (4.2)$$

$$R_{iR} = \frac{1}{C} \int i_C dt, \quad i = i_R + i_C, \quad (4.3)$$

sendo n o número de espiras do indutor, e ϕ é o fluxo magnético no núcleo. Então desprezando a histerese, pode-se assumir a curva de saturação magnética da seguinte forma:

$$i = a_1 \phi + a_3 \phi^3, \quad (4.4)$$

os termos de potência maior que 3 são desprezados. A corrente i e o fluxo ϕ são colocados numa base apropriada I e Φ .

$$i = I.u, \quad \phi = \Phi.v. \quad (4.5)$$

Com as novas bases a equação (4.4) se torna:

$$u = \frac{a_1 \Phi}{I} v + \frac{a_3 \Phi^3}{I} v^3 = c_1 v + c_3 v^3, \quad (4.6)$$

embora as bases I e Φ possam ser escolhidas arbitrariamente, é preferível, por razões de simplificação da expressão, fixá-las pelas relações

$$n\omega^2 C \Phi = I, \quad c_1 + c_3 = 1. \quad (4.7)$$

Então, depois da eliminação de i_R e i_C das equações (4.2) e (4.3) e usando as equações (4.5), (4.6) e (4.7) o resultado em termos de v é

$$\frac{d^2 v}{d\tau^2} + k \frac{dv}{d\tau} + c_1 v + c_3 v^3 = B \cos(\tau), \quad (4.8)$$

sendo

$$\tau = \omega t - \tan^{-1} k, \quad k = \frac{1}{\omega C R}, \quad B = \frac{E}{n\omega\Phi} \sqrt{1 + k^2}. \quad (4.9)$$

A equação (4.8) é bem conhecida na teoria de oscilações como equação de Duffing. Como pode ser observado a equação Duffing-Ueda é um caso particular desta equação, quando o valor de $c_1 = 0$.

4.2.2 Pontos Fixos

Os pontos fixos para o caso autônomo (entrada nula) podem ser obtidos resolvendo a seguinte equação:

$$c_1 v + c_3 v^3 = 0. \quad (4.10)$$

Esta equação pode ser reescrita como:

$$v(c_1 + c_3 v^2) = 0. \quad (4.11)$$

Os pontos fixos são:

$$\begin{aligned} v_1 &= 0, \\ v_2 &= +\sqrt{-\frac{c_1}{c_3}}, \\ v_3 &= -\sqrt{-\frac{c_1}{c_3}}. \end{aligned} \quad (4.12)$$

Pode-se observar que este sistema apresenta um ponto fixo trivial e os outros pontos fixos não triviais são simétricos em relação ao trivial. No caso de Duffing-Ueda ($c_1 = 0$), o sistema apresenta três pontos fixos triviais, a característica de simetria se mantém pois a característica estática do sistema é raiz cúbica da entrada.

4.2.3 A dinâmica do sistema

No diagrama de bifurcação (DB) do sistema, mostrado na Figura 4.2, podem ser observados vários comportamentos dinâmicos complexos, como caos. Este diagrama fornece uma precisa indicação de como a dinâmica do sistema se altera quando o parâmetro de bifurcação é variado entre 4, $5 \leq A \leq 12$. Em $A \approx 4,8$ o sistema passa por uma bifurcação de duplicação de período. Isto acontece novamente em $A \approx 5,4$ e caracteriza a bem conhecida rota de duplicação de período para o caos (Feigenbaum, 1983). Outra cascata similar começa em $A \approx 9,67$ precedendo um regime caótico. Duas janelas caóticas podem ser distinguidas em aproximadamente $5,5 \leq A \leq 5,8$ e $9,9 \leq A \leq 11,6$. Em $A \approx 6,6$ e $A \approx 8$ o sistema passa por uma bifurcação forquilha (*pitchfork*) seção 2.1.6. O diagrama começa e termina com regimes dinâmicos de período 1. No intervalo $5,8 \leq A \leq 9,6$ o sistema apresenta dinâmica de período 3 conforme visto na Figura 4.2. O diagrama de bifurcação é útil, pois permite conhecer a dinâmica global do sistema em

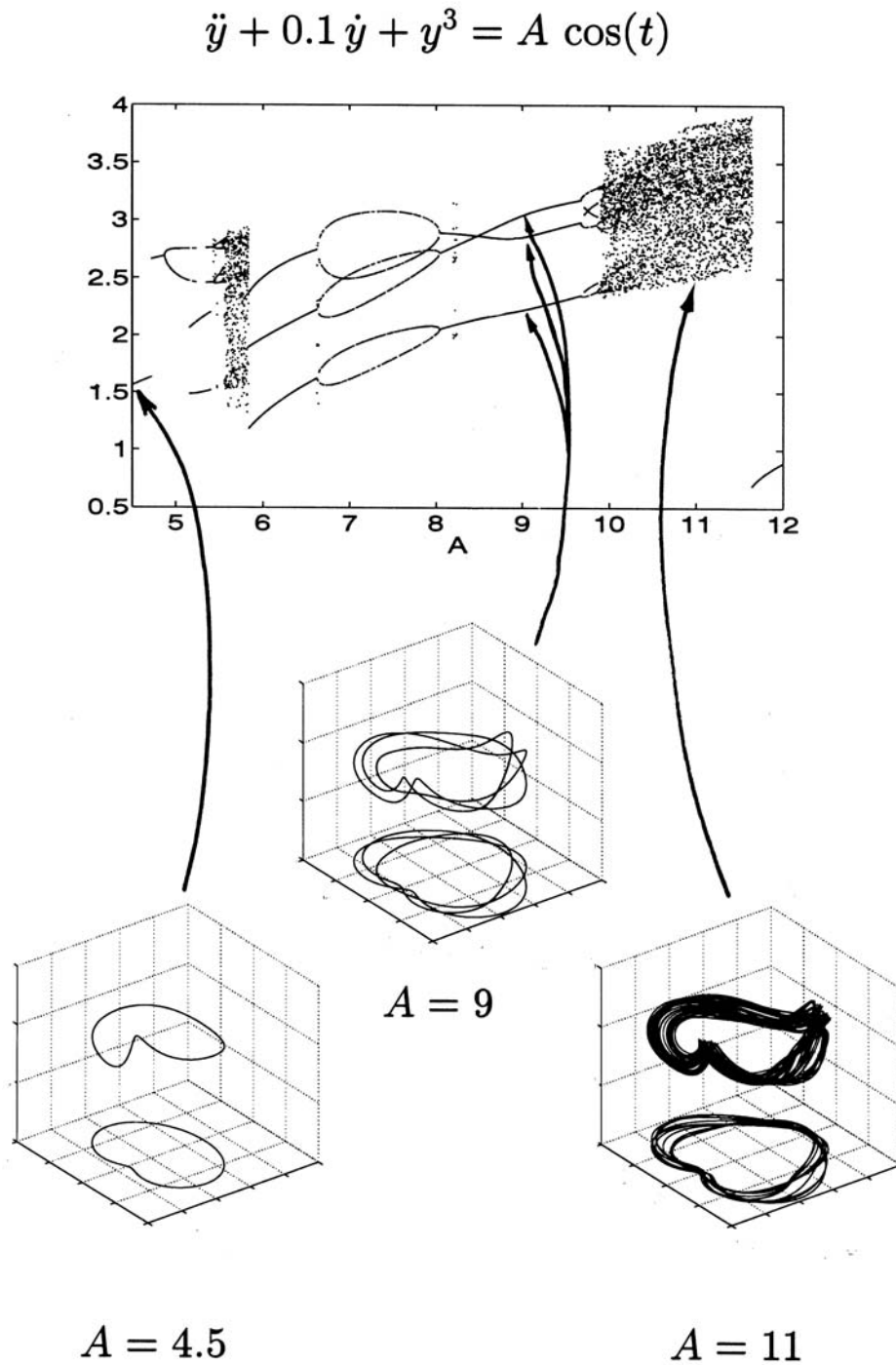


Figura 4.2: Regimes dinâmicos apresentados pela equação de Duffing-Ueda na faixa de operação de $4, 5 \leq A \leq 12$. Para $A = 4, 5$ o sistema apresenta um comportamento periódico (período 1), para $A = 9$ o comportamento também é periódico (período 3), mas, a diferença é que para completar um ciclo da saída são necessários três ciclos da entrada. Para $A = 11$ o sistema apresenta caos.

sua faixa de operação. Contudo, nem sempre é possível construir um diagrama de bifurcação na prática, porém, já existem trabalhos em que o DB foi obtido através dos dados de entrada e saída do sistema (Aguirre et al., 2001; Bagarinao et al., 1999a,b,c).

4.3 O processo de identificação

A partir da informação dos regimes dinâmicos do sistema, pode-se verificar duas importantes características sobre este:

- 1) O oscilador de Duffing-Ueda apresenta bifurcação forquilha na seção de Poincaré para excitação senoidal.
- 2) Da equação normal descrita na seção 2.1.6, que é diferente da equação de Duffing-Ueda, observa-se o aparecimento de simetria sobre a seção de Poincaré.

Motivados por 1 e 2 e tendo em vista o teorema do capítulo 3, decidiu-se impor condições de simetria de pontos fixos a rede para verificar como essa mudança na topologia afetaria o seu treinamento e desempenho dinâmico.

O processo de identificação é descrito a seguir.

4.3.1 Excitação do sistema

Os dados utilizados na estimação de parâmetros ou etapa de treinamento das redes neurais são fundamentais para a obtenção de uma rede eficiente (Vega et al., 2000). O sinal de entrada deve excitar tanto a dinâmica quanto a não-linearidade estática do sistema de modo que se possa obter o máximo de informação do sistema. O sinal de entrada utilizado foi uma onda quadrada em que a amplitude variava na faixa de $\pm 0,5$ a ± 13 e além disso, foi adicionada uma variável aleatória com distribuição gaussiana de média zero e variância $\sigma^2 = 2,0$. A explicação para o uso desse tipo de entrada deve-se ao fato de que para sistemas não-lineares não apenas o conteúdo espectral é importante, como no caso linear, mas também a forma de onda do sinal de excitação é relevante (Aguirre and Billings, 1995a).

Como pode ser observado na Figura 4.3 a amplitude é variada na faixa de operação em que deseja-se obter um modelo. Os dados de saída foram obtidos a partir de simulação digital, através de um algoritmo Runge-Kutta de quarta ordem com intervalo de integração de $\pi/3000$. Tanto a entrada quanto a saída foram uniformemente amostradas com período de $\pi/60$ o que resulta 120 pontos por período da entrada quando esta é senoidal.

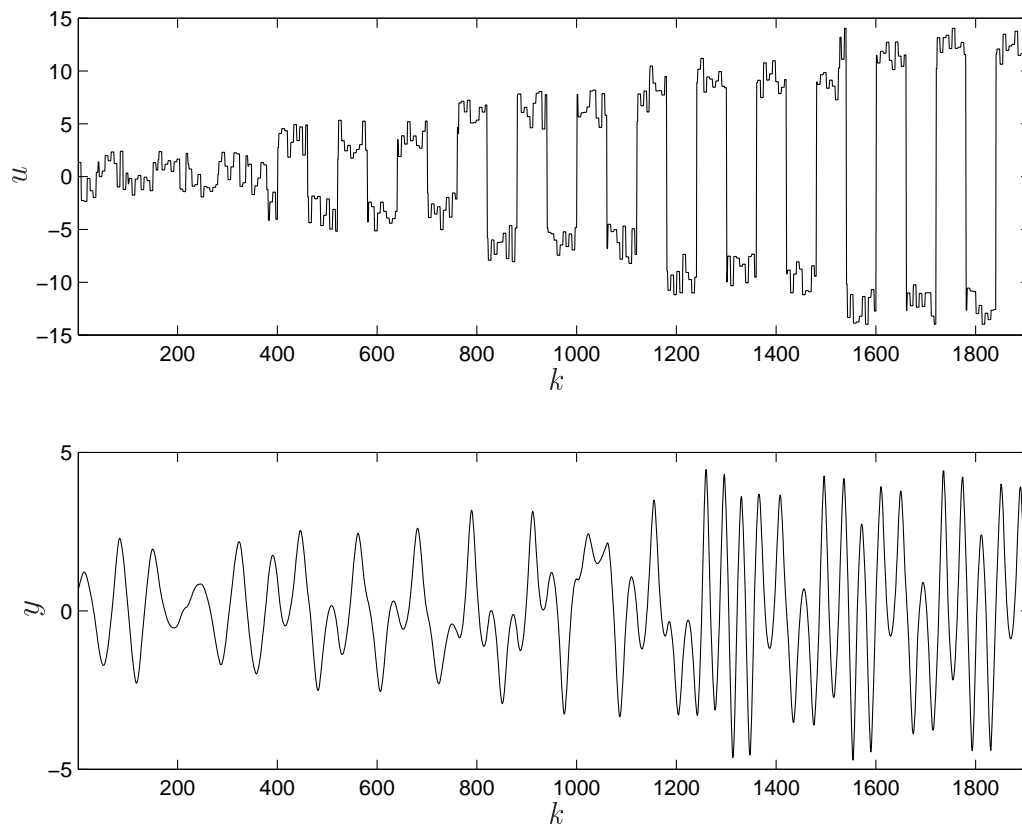


Figura 4.3: Dados de entrada e saída do sistema (Aguirre and Billings, 1995a). A entrada é uma onda quadrada com amplitude variando de $\pm 0,5$ a ± 13 a qual foi adicionada uma variável aleatória de média zero e variância $\sigma^2 = 2,0$, gráfico de cima. O gráfico de baixo é a saída do sistema.

4.3.2 Implementação das redes

Quatro arquiteturas foram escolhidas para obtenção de um modelo, sendo duas com o *bias* e duas sem. As arquiteturas podem ser visualizadas nas Figuras 4.4, 4.5, 4.6 e 4.7. A razão da escolha de arquiteturas de redes que possuem um neurônio linear na camada oculta deve-se a observação do artigo (Aguirre and Billings, 1994), em que o sistema de Duffing-Ueda é modelado por modelos NARMAX polinomiais. Tal trabalho mostra que não foi necessário não-linearidade nos termos referentes aos regressores da entrada rede. O uso de um neurônio linear na camada oculta ajuda no ajuste da parte linear das redes. Porém, a metodologia de começar com modelos lineares, que é o caso em que só tem-se neurônios lineares, e então a partir deste modelo linear ir acrescentando termos não-lineares é sugerido na literatura de redes (Sjöberg, 1997). As redes foram implementadas usando o toolbox de identificação de sistemas baseada em redes neurais (Norgaard, 1997). O treinamento foi feito através da rotina `nnarmax2.m` (utilizando o algoritmo de Levenberg e Marquardt) e a simulação pela rotina `nnsimul.m`.

As redes foram nomeadas de acordo com a notação do toolbox, a letra H representa a função de ativação tangente hiperbólica e a letra L a função de ativação linear. Por exemplo: a rede HL é uma rede que possui dois neurônios na camada oculta sendo um linear e o outro tangente hiperbólica. Quanto a obter arquiteturas que não possuam os termos de polarização (*bias*), basta inicializar com valores nulos porque a rotina `nnarmax2.m` não altera o valor de um parâmetro se este for inicializado com valor zero, em outras palavras, é como se este parâmetro não existisse. O número de regressores usado foi 3,2,10 respectivamente o número máximo de atrasos da saída, entrada e termos de ruído, pois, se trata de um modelo NARMAX neural.

Quatro problemas foram propostos, o primeiro foi identificar o sistema utilizando os dados da Figura 4.3, no segundo foram utilizados os mesmos dados, porém, com redes sobredimensionadas, em seguida dados de uma excitação periódica Figura 4.12, foi utilizada e por fim dados com ruído. Em cada um destas quatro situações, foram treinadas 100 redes de cada uma das arquiteturas propostas, o objetivo foi obter resultados com maior validade estatística.

Os critérios de parada foram escolhidos heurísticamente. Os valores de erro eram variados empiricamente até que uma rede de cada grupo de arquiteturas conseguisse reconstruir a forquilha. Então cada grupo de redes era treinado com o valor de erro da respectiva rede que aprendeu a forquilha.

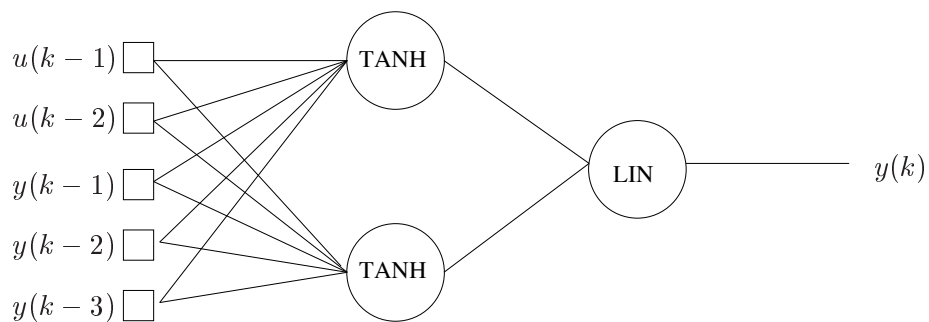


Figura 4.4: Arquitetura da rede HH.

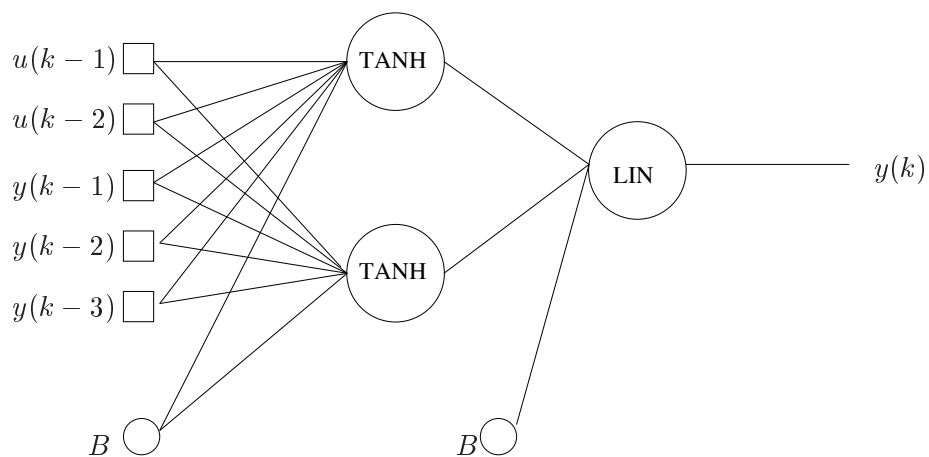
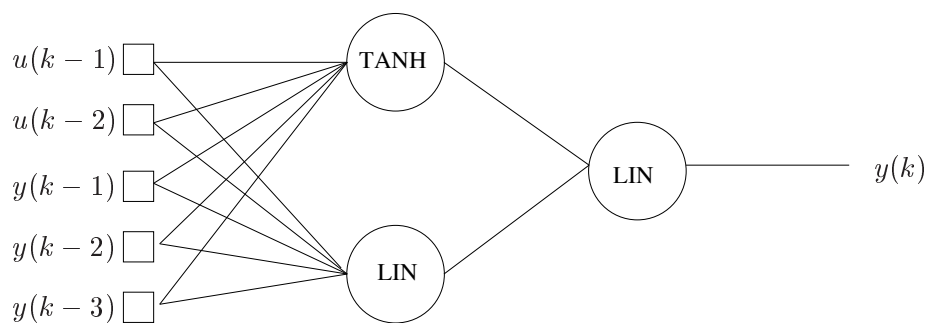
Figura 4.5: Arquitetura da rede HH com *bias*.

Figura 4.6: Arquitetura da rede HL.

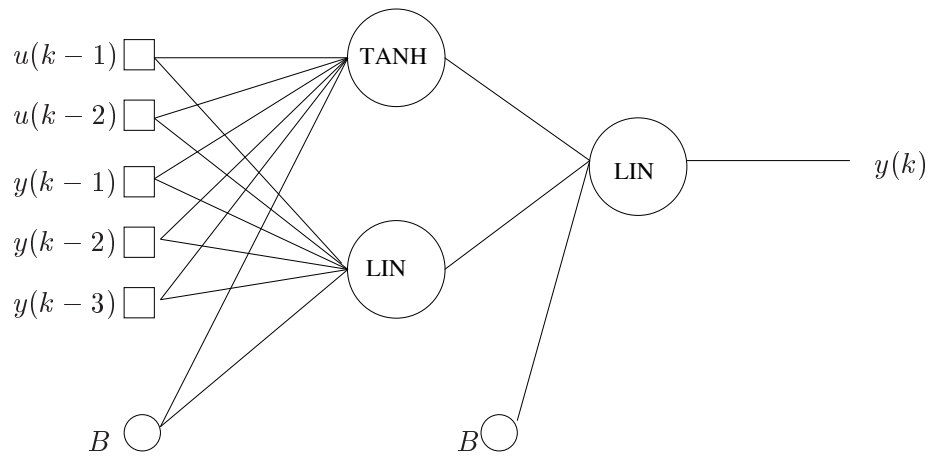


Figura 4.7: Arquitetura da rede HL com *bias*.

As seções de Poincaré de algumas redes foram obtidas para uma entrada cossenoidal de amplitude 11 Figura 4.8. A rede HH com (*bias*) apresenta seção de Poincaré diferente da seção de Poincaré do sistema, pode-se observar que o valor mínimo no eixo $y(t - Tp)$ para o sistema, está próximo de 2,5, enquanto na rede em questão, esse valor é aproximadamente 2,6. Além de poder ser observada diferenças significativas na forma da seção de Poincaré. As seções de poincaré das demais redes, são mais próximas à seção do sistema.

Alguns diagramas de bifurcação são mostrados na Figura 4.9. Para estes resultados a exceção da rede HH com *bias* (d), as redes conseguiram modelar a dinâmica global do sistema. O resultado visto em (d) tem relação com problemas de mínimos locais, o que conduz a soluções inadequadas, pois a rede não conseguiu realizar o cancelamento de (*bias*).

Os resultados mostrados nas Figuras 4.8 e 4.9 são resultados típicos.

Tabela 4.1: Valores percentuais das redes que conseguiram reconstruir a forquilha .

Rede	Forquilha
HH	15%
HHNB	15%
HL	59%
HLNB	61%

Como pode ser observado pela Tabela 4.1, a presença ou não dos termos de polarização não foi tão crítica, neste caso, pois as redes com os termos

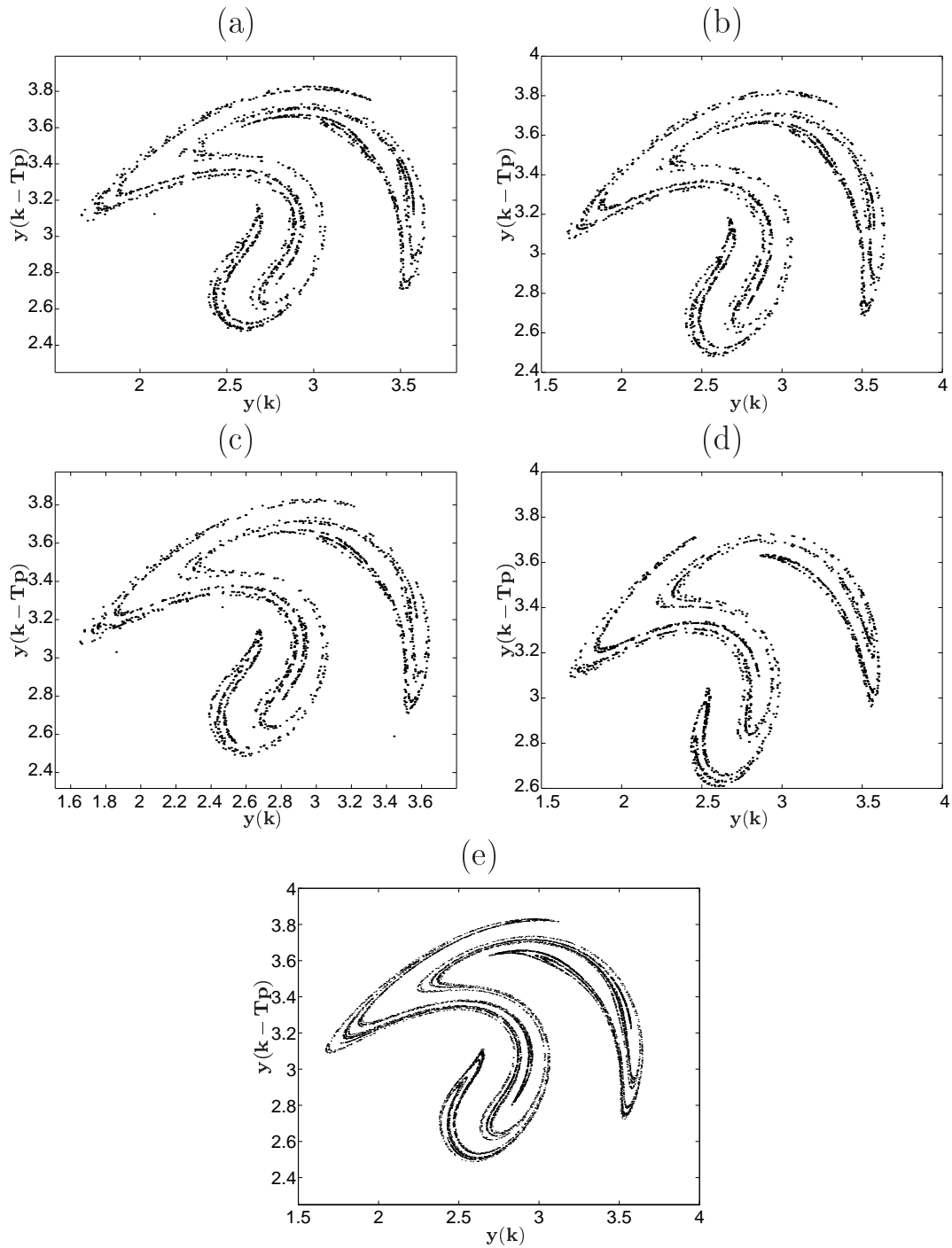


Figura 4.8: Seções de Poincaré dos modelos neurais obtidas a partir de uma entrada cossenoidal com amplitude $A = 11$: (a) Rede HL (b) Rede HL com bias (c) Rede HH (d) Rede HH com *bias* (e) Seção de Poincaré do sistema.

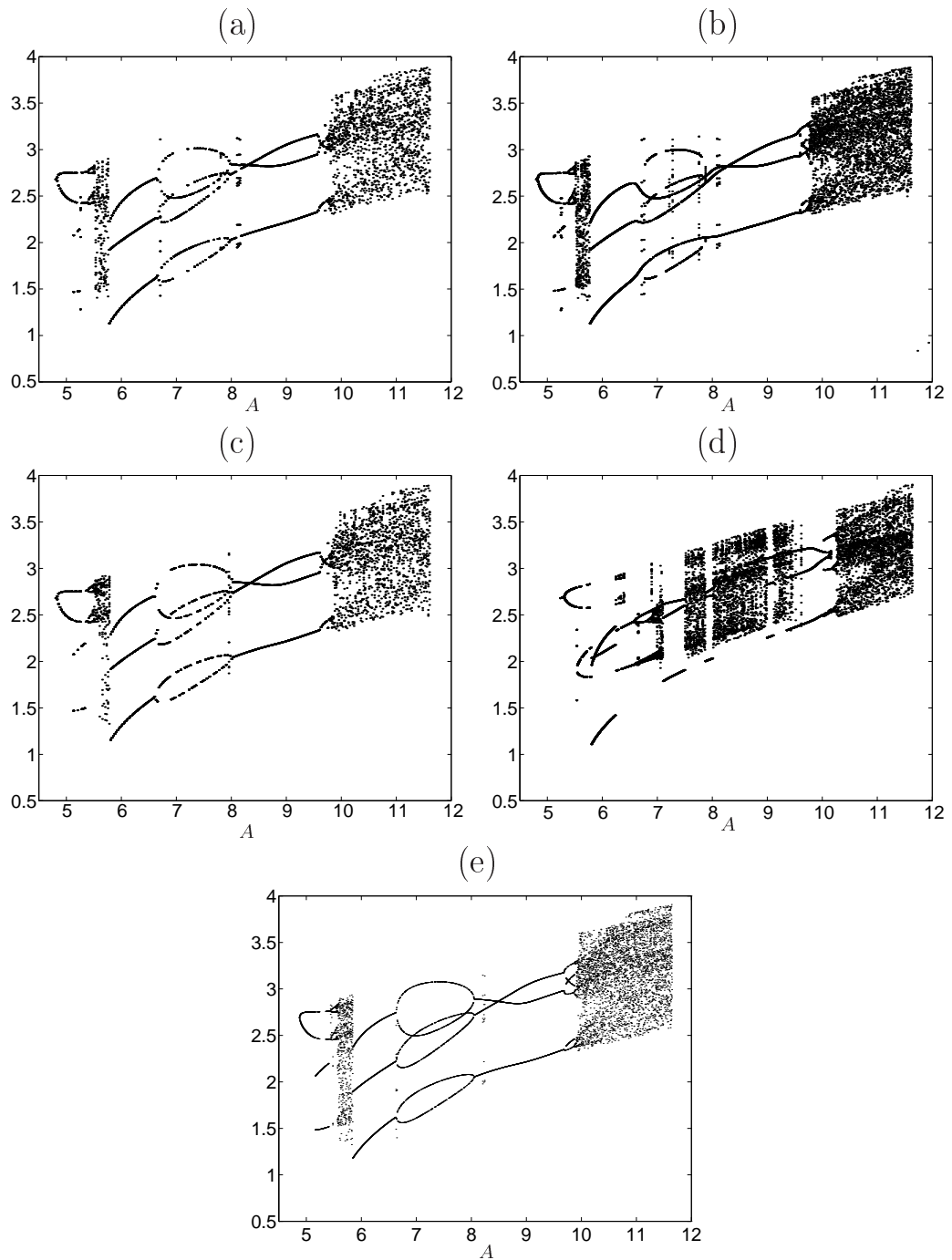


Figura 4.9: Diagramas de bifurcação típicos dos modelos neurais para entrada cossenoidal com amplitude no intervalo $4, 5 \leq A \leq 12$: (a) Rede HL (b) Rede HL com *bias* (c) Rede HH (d) Rede HH com *bias* (e) Diagrama de bifurcação do sistema.

de polarização conseguiram aprender o comportamento esperado, isto é, forquilha. O que parece realmente relevante, neste caso, é a presença de um neurônio linear na camada oculta, pois o grupo de redes que possuía este neurônio, mostrou que o número de redes que reconstruiu o comportamento de forquilha foi quatro vezes maior do que o número daquelas redes que não possuíam este neurônio.

4.3.3 Cancelamento do termo de polarização

As redes que possuíam o termo de polarização e que conseguiram reconstruir a forquilha demonstraram que, como esperado, a presença do termo de polarização é indesejável, pois, este foi eliminado durante o treinamento. Ocorreu um tipo de cancelamento deste termo durante o processo de treinamento. As redes HH após o treinamento tiveram valores de $(bias) \leq 10^{-4}$. Enquanto que nas redes HL o mesmo aconteceu para os termos de polarização do neurônio não-linear e os termos dos neurônios lineares da camada oculta e da camada de saída anularam-se. Este cancelamento dá-se da seguinte maneira, o termo de polarização da camada oculta multiplicado pelo peso que conecta o neurônio da camada oculta a camada de saída, é aproximadamente igual ao negativo do termo de polarização da camada de saída.

4.4 Sobreparametrização da rede

O objetivo desta seção é verificar os efeitos da variação do número de neurônios na camada oculta, no processo de identificação do sistema. A cada arquitetura utilizada na seção anterior foram adicionados dois neurônios. Foi usada a massa de dados da Figura 4.3.

As seções de Poincaré não apresentam grandes mudanças em relação ao atrator do sistema Figura 4.10 como seria de se esperar (Aguirre and Billings, 1994), a exceção da rede HH com $(bias)$, que mostra-se um pouco girada no sentido horário.

Nos diagramas de bifurcação mostrados na Figura 4.11 percebem-se diferenças qualitativas entre os modelos. A primeira delas é que as redes com os bias (b) e (d) não conseguem reproduzir a forquilha. Entre as redes (a) e (c), a rede HL sem $(bias)$ é a rede que possui o diagrama mais semelhante ao diagrama do sistema. A rede (b) apresenta diferenças na região de período 3.

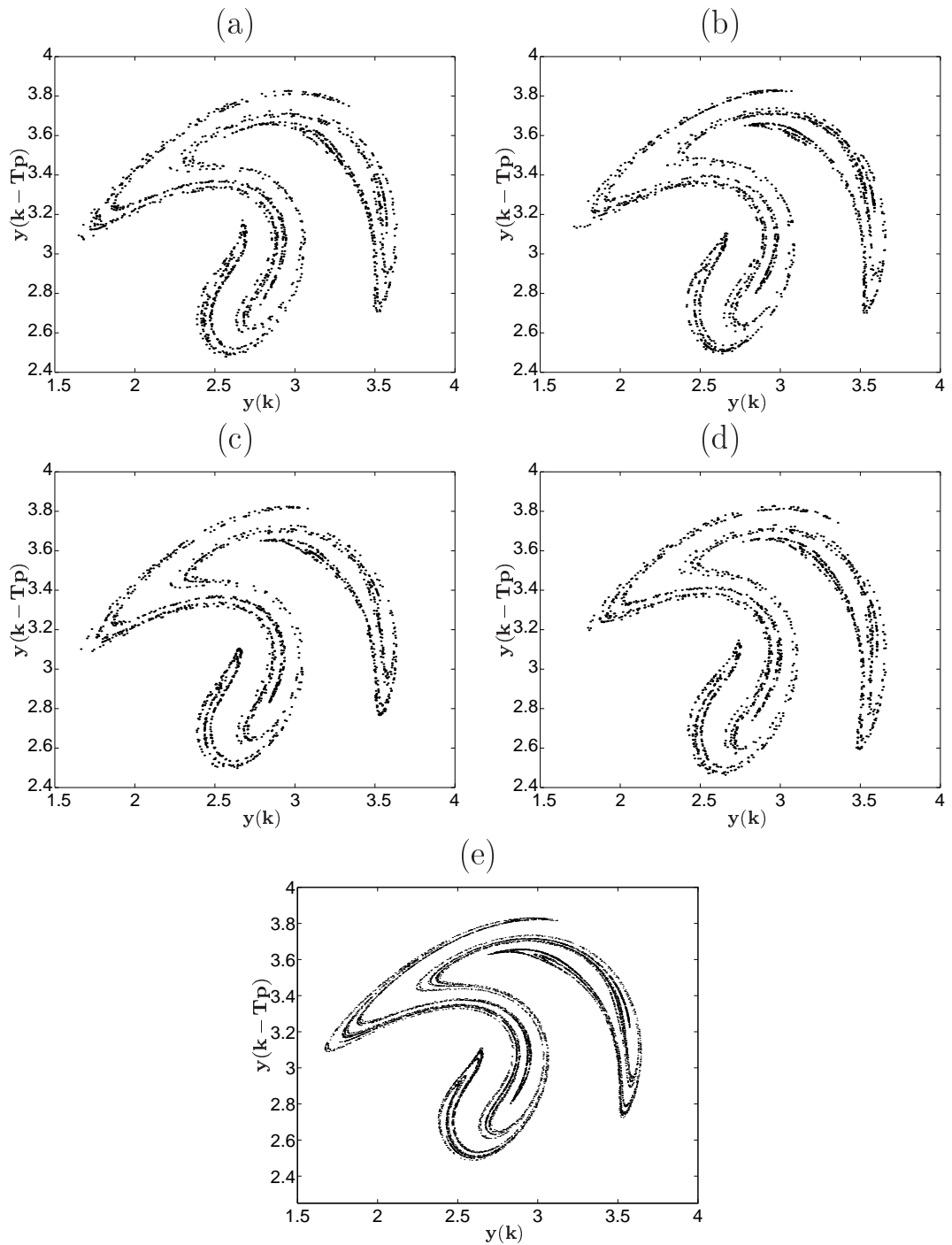


Figura 4.10: Seções de Poincaré dos modelos neurais obtidas a partir de uma entrada cossenoidal com amplitude $A = 11$: (a) Rede HHHL (b) Rede HHHL com bias (c) Rede HHHH (d) Rede HHH com *bias* (e) Seção de Poincaré do sistema.

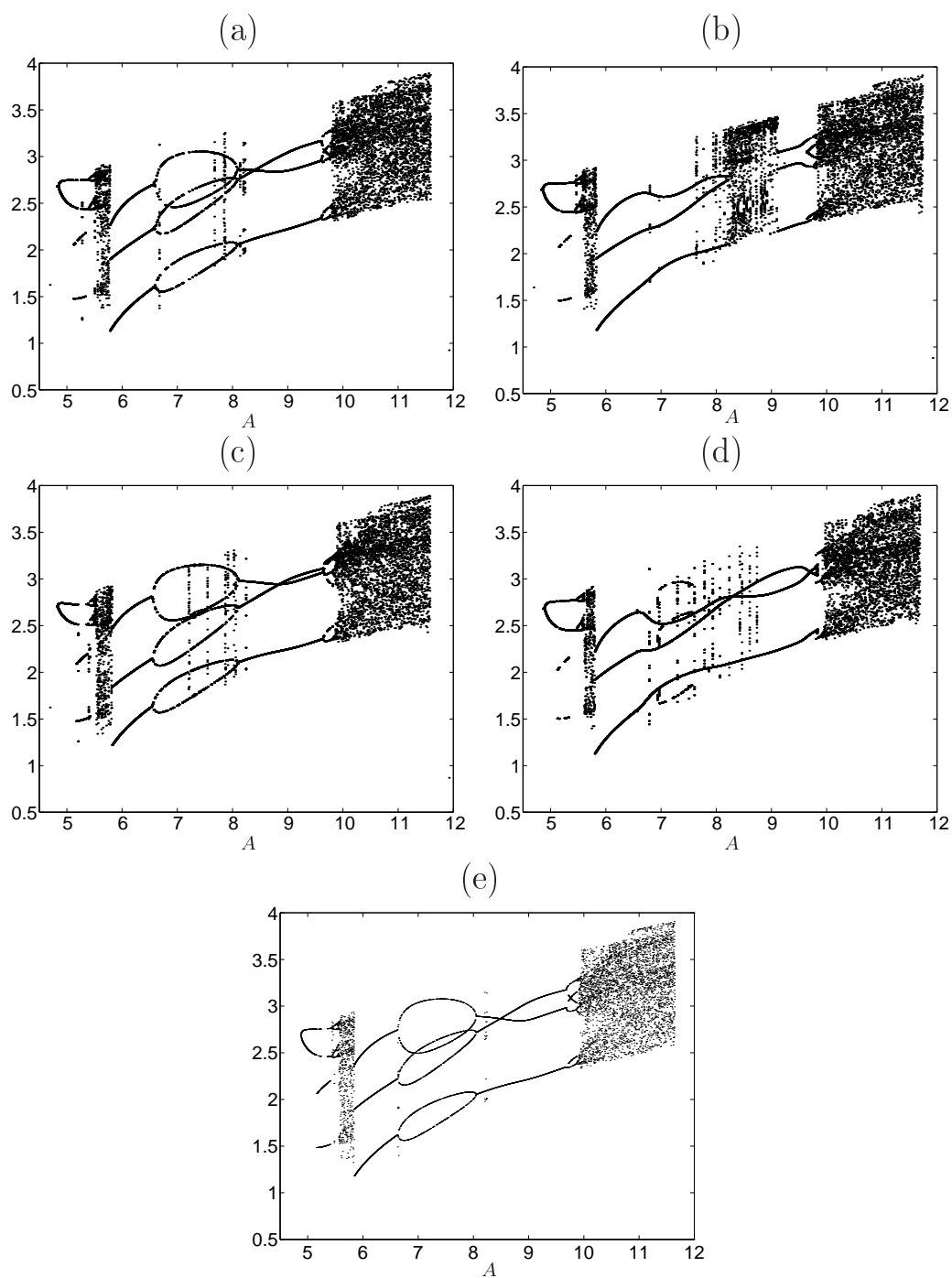


Figura 4.11: Diagramas de bifurcação dos modelos neurais para entrada cossenoidal com amplitude no intervalo $4, 5 \leq A \leq 12$: (a) Rede HHHL (b) Rede HHHL com *bias* (c) Rede HHHH (d) Rede HHHH com *bias* (e) Diagrama de bifurcação do sistema.

Tabela 4.2: Valores percentuais das redes sobreparametrizadas que conseguiram reconstruir a forquilha.

Rede	Forquilha
HHHH	34%
HHHHNB	2%
HHHL	47%
HHHLNB	77%

Na Tabela 4.2 pode-se verificar o efeito da presença do (*bias*) e do neurônio linear. Para melhor avaliar o benefício de zerar o termo de polarização na mão, o mais adequado é comparar as redes HHHLNB e HHHL, uma vez que as redes HHHHNB e HHHH proporcionam um relacionamento não-linear de $u(k)$ o que é inadequado para este caso. Conjectura-se que o termo (*bias*) ajuda no processo de treinamento deste grupo de redes, porém, nas redes onde foram obtidos bons resultados este termo sempre era cancelado. Em relação ao neurônio linear, compara-se as redes HHHH e HHHHNB com HHHL e HHHLNB, respectivamente, e é verificado que a presença do neurônio linear sempre é benéfica para este caso.

4.5 Identificação utilizando excitação periódica

Estes dados também foram gerados por simulação digital através da rotina `ode45.m` do programa Matlab (Mathworks, 1999), as opções `MaxStep` e `InitialStep` foram iguais a $\pi/3000$, a entrada e a saída foram amostradas em $\pi/54, 54$. Para realizar a construção do diagrama de bifurcação e seção de Poincaré a amostragem foi de $\pi/55$ de modo que os dados apresentassem 110 pontos por período. A entrada utilizada foi uma cossenóide de amplitude 11, os dados podem ser vistos na Figura 4.12. Apesar dessa entrada ser pobre no sentido de conter apenas uma frequência, a saída para esta entrada apresenta um amplo espectro de frequência, o que é uma característica de sistemas não-lineares. O objetivo em usar esta entrada é verificar se as redes neurais são robustas em relação aos dados, isto é, dada informação dinâmica de apenas um atrator, as redes pode-se obter bons modelos globais? (Henrique et al., 1998).

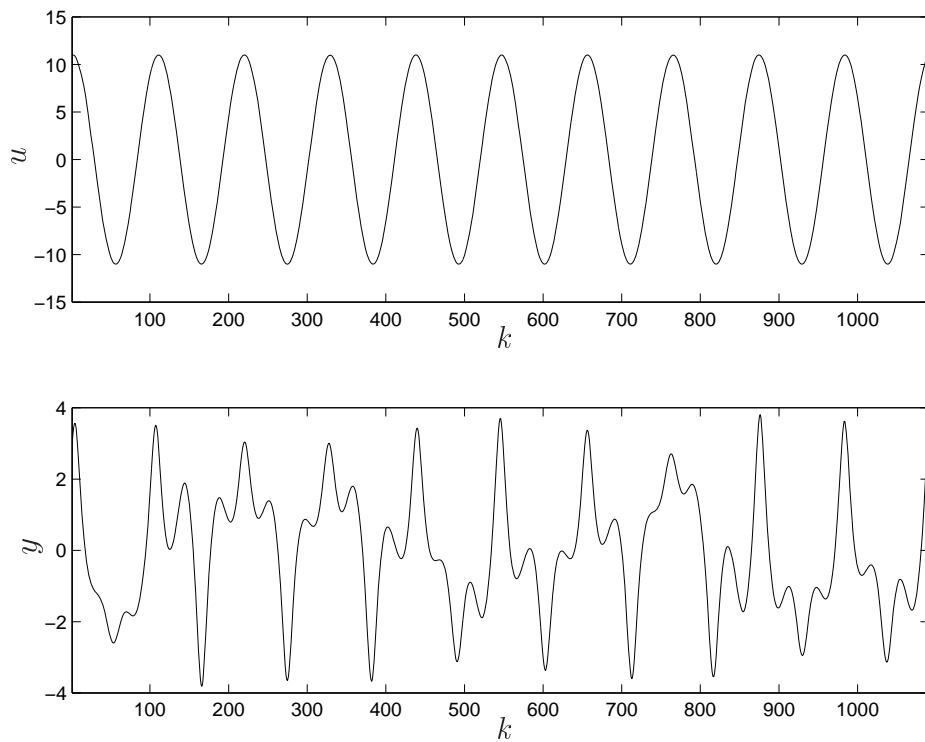


Figura 4.12: Dados de entrada e saída do sistema. A entrada é uma onda cossenoidal de amplitude $A = 11$, em cima. A saída do sistema em regime caótico, embaixo.

É importante salientar que este caso é semelhante a muitas situações práticas, em que muitas vezes não é possível aplicar uma entrada que possa excitar o sistema numa ampla faixa de operação, e além disso o número de dados usados são escassos.

As redes foram treinadas como nas etapas anteriores e utilizam apenas dois neurônios na camada oculta. Na Figura 4.13 estão mostradas algumas seções de Poincaré obtidas por cada grupo de arquitetura utilizada, pode-se observar que as seções estão visivelmente giradas no sentido horário, porém a forma do atrator é bem semelhante, a exceção a rede HHNB que indica um atrator de período 3.

Alguns diagramas de bifurcação podem ser vistos na Figura 4.14, pode-se constatar que as redes que possuíam o termo de polarização apresentaram dinâmicas espúrias. No caso da rede HL é verificado um comportamento de duplicação de período começando em $A \approx 7$ e a rede HH apresenta uma região não periódica aproximadamente em $6, 7 \leq A \leq 7, 8$. As redes HHNB e HLNB conseguiram reproduzir a mesma seqüência de bifurcações do sistema original, no entanto, as faixas onde começa e termina cada comportamento dinâmico está um pouco diferente do sistema original, o que explica o fato da seção de Poincaré da rede HLNB apresentar forma tão diferente. A segunda janela caótica deveria terminar em $A \approx 11, 6$ e o resultado mostra que a transição acontece em $A \approx 11$.

Como pode ser visualizado na Figura 4.13 a seção de Poincaré da rede HH sem o *bias* não conseguiu reproduzir o atrator caótico. As redes restantes apresentam seções de Poincaré com características semelhantes. Existe uma distorção dos atratores, como pode ser visto, parece que as seções estão inclinadas para a direita. Os diagramas de bifurcação das redes HL sem *bias* e HH sem o *bias*, demonstram a mesma seqüência de bifurcações o que variou foram os valores de A para os quais os diversos regimes dinâmicos são observados. Os diagramas das redes com *bias* apresentam dinâmicas espúrias, a rede HL (b) apresenta uma forquilha dentro da forquilha e a rede HH demonstra regime caótico na região em que deveria responder com um período 3.

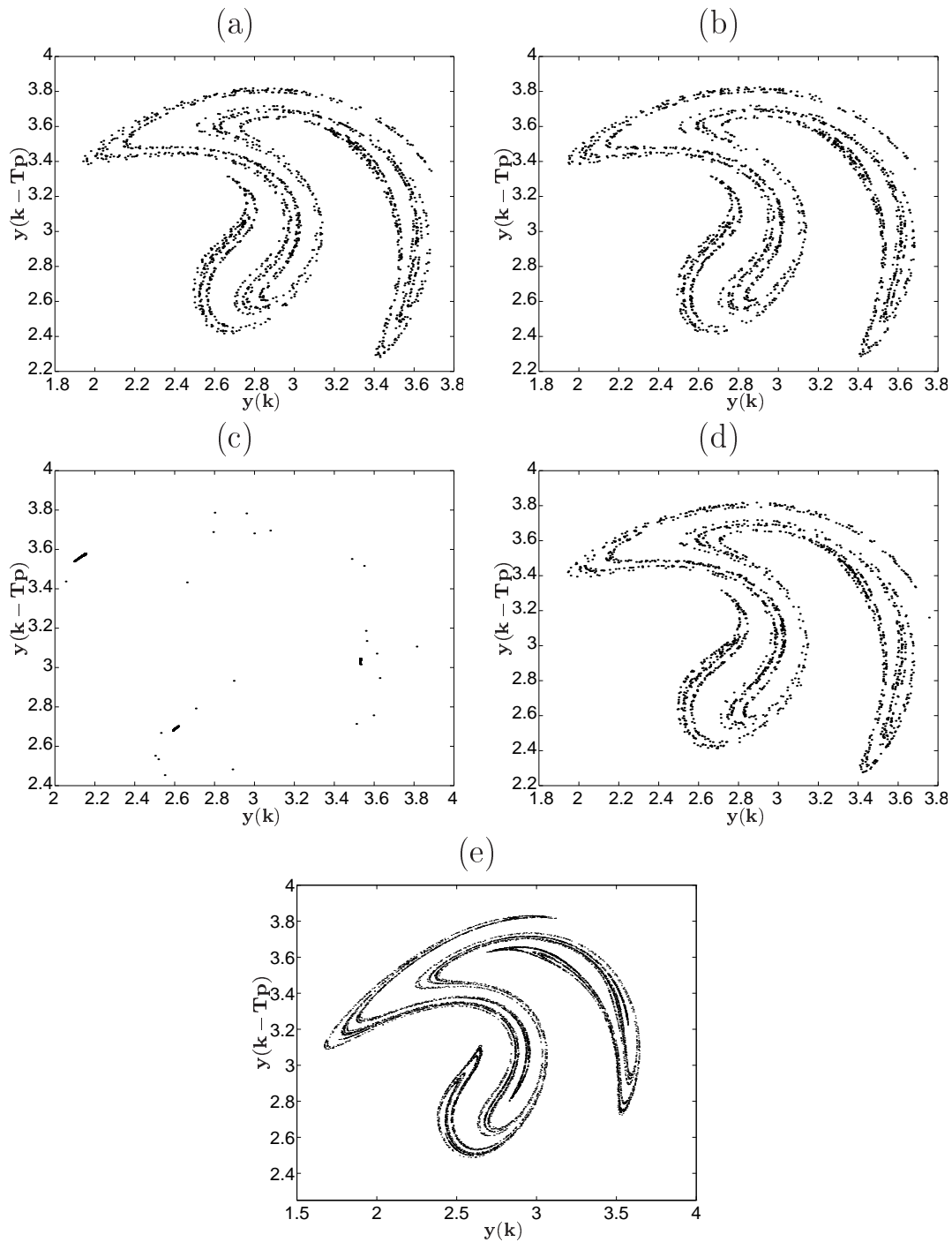


Figura 4.13: Seções de Poincaré dos modelos neurais (redes treinadas com a massa de dados mostrada na Figura 4.12) obtidas a partir de uma entrada cossenoidal com amplitude $A = 11$: (a) Rede HL (b) Rede HL com bias (c) Rede HH (d) Rede HH com *bias* (e) Seção de Poincaré do sistema.

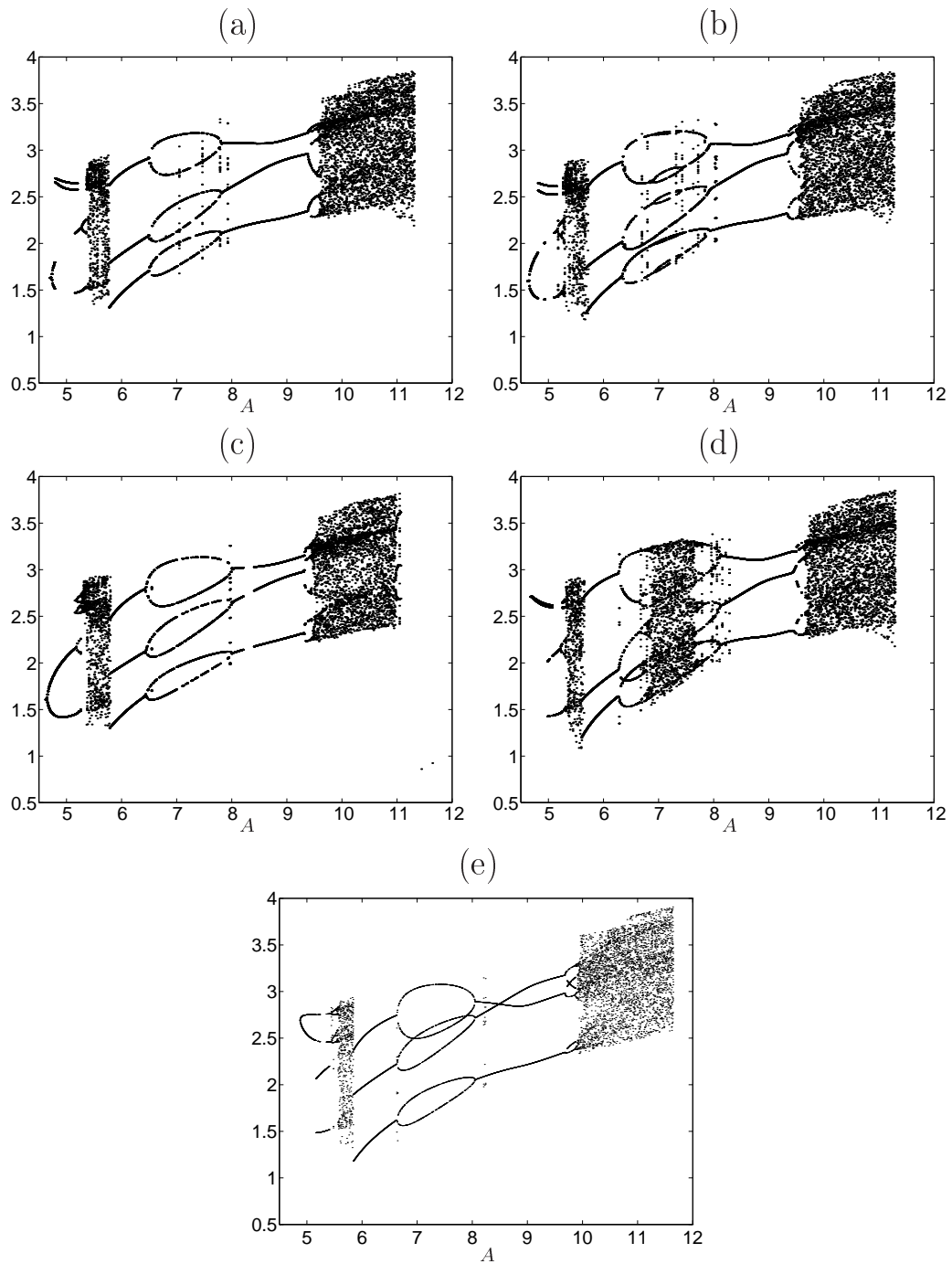


Figura 4.14: Diagramas de bifurcação dos modelos neurais (obtidos pela massa de dados 4.12) para entrada cossenoidal com amplitude no intervalo $4, 5 \leq A \leq 12$: (a) Rede HL (b) Rede HL com *bias* (c) Rede HH (d) Rede HH com *bias* (e) Diagrama de bifurcação do sistema.

Tabela 4.3: Valores percentuais das redes que conseguiram reconstruir a forquilha, no caso de entrada cossenoidal.

Rede	Forquilha
HH	5%
HHNB	38%
HL	51%
HLNB	69%

Pela Tabela 4.3 pode-se concluir que tanto a ausência do termo de polarização, quanto a presença do neurônio linear na camada oculta foram importantes para obtenção de bons resultados. No caso do (*bias*) verifica-se que as redes que não possuíam o neurônio linear foram auxiliadas com a retirada deste termo, o número de redes que conseguiram obter forquilha cresceu de 5% para 38%. As redes que possuíam o neurônio linear também foram auxiliadas pela ausência do (*bias*), o grupo de redes HL que não possuía este termo obteve um aumento de 18% no número de redes que conseguiram reproduzir a forquilha. Em relação a presença do neurônio linear é observado que o número de redes que reproduziram a forquilha é significativamente maior.

4.5.1 Identificação utilizando dados com ruído

Nesta seção dados com ruído foram utilizados como dados de treinamento das redes. O dados são os mesmos da Figura 4.3, porém foi adicionado aos dados de saída do sistema um ruído gaussiano com média zero e desvio padrão 40 vezes menor que o desvio padrão da saída do sistema. A relação sinal ruído é de $\approx 64\text{dB}$, nível geralmente encontrado em situações reais. Neste caso o erro mínimo de parada do treinamento foi de $1,5 \times 10^{-3}$.

Tabela 4.4: Valores percentuais das redes que conseguiram reconstruir a forquilha, para o caso de dados com ruído.

Rede	Forquilha
HH	0%
HHNB	6%
HL	6%
HLNB	15%

Como pode ser verificado na Tabela 4.4, é importante ressaltar o quão inadequadas mostraram-se as redes HH neste caso, pois, nenhuma destas redes conseguiram reproduzir o comportamento de forquilha. No entanto,

com esta mesma arquitetura, porém, sem o termo de polarização 6% destas redes puderam aprender. Isto é um indicativo de que a restrição a este termo ajuda no aprendizado das redes. Como vem sendo observado nos casos discutidos, as redes HL obtiveram a melhor performance.

4.6 Conclusão

Com o objetivo de obter modelos neurais que fossem mais adequados a identificação do sistema de Duffing-Ueda as restrições deduzidas no capítulo 3 foram aplicadas as redes, uma vez que sabe-se que o sistema em questão apresenta bifurcação forquilha o que implica simetria. Estas redes foram testadas em quatro diferentes situações. Inicialmente com uma massa de dados em que a entrada possui grande conteúdo espectral e varre a faixa de valores de interesse. No segundo caso a mesma massa de dados foi utilizada, porém, as redes estavam sobredimensionadas. Um outro caso testado foi identificar o sistema utilizando uma massa de dados em que a entrada é "pobre", tanto em relação ao espectro quanto ao fato de não excitar o sistema em toda faixa de operação. Na última situação estudada, fez-se uso da massa de dados do primeiro caso, porém com um ruído adicionado aos dados de saída do sistema.

Os resultados para o primeiro caso Tabela 4.1, mostram que a ausência do (*bias*) foi pouco relevante, pois, as redes HH e HL realizaram o cancelamento deste nos casos em que se obteve forquilha. Demonstrando que o termo *bias* é indesejável no caso deste sistema. As redes HL e HLNB mostraram-se mais adequadas a este sistema. No segundo caso, em que as redes estavam sobredimensionadas a ausência do termo de polarização mostrou-se auxiliar para as redes com o neurônio linear na camada oculta. As redes que não o neurônio linear na camada oculta necessitaram do (*bias*) como é verificado na Tabela 4.2. Foi observado que mesmo assim o cancelamento do termo de polarização aconteceu, o que leva a supor que o (*bias*) neste caso auxiliou no treinamento. No caso dos dados de treinamento com entrada cossenoidal, os resultados da Tabela 4.3 mostram que o termo de polarização é desnecessário em relação a redes com ou sem o neurônio linear na camada oculta. Redes neurais conseguiram reproduzir a dinâmica do sistema mesmo com uma excitação pobre, o que mostra robustez das redes uma vez incorporado conhecimento a priori. No último caso, em que os dados estavam corrompidos com ruído verifica-se pela Tabela 4.4, que a rede que foi considerada inadequada não conseguiu obter a forquilha em nenhuma das tentativas. A arquitetura de rede que obteve maior número de modelos que apresentavam forquilha foi a rede HLNB.

De modo geral, observou-se que a ausência do (*bias*) e a presença do neurônio linear na camada oculta, auxiliou a obtenção de redes que apresentavam o comportamento dinâmico do sistema de Duffing-Ueda. Principalmente considerando os dois últimos casos, da entrada cossenoidal e com ruído, que são mais próximos de situações reais.

Capítulo 5

Identificação de um aquecedor: Um estudo de caso usando técnicas caixa preta e caixa cinza

5.1 Introdução

No capítulo anterior conhecimento a priori foi utilizada para determinar algumas características da arquitetura da rede MLP. Neste capítulo, um estudo de caso é feito a partir de abordagens caixa preta e caixa cinza (Cassini, 1999; Barroso, 2001; Barroso et al., 2001). Entre as técnicas caixa preta estão, decaimento de pesos, cirurgia cerebral ótimo e a taxa de redução de erro. A abordagem caixa cinza neste capítulo, mostra um exemplo que conduz a um procedimento geral, para os casos em que o sistema apresenta apenas um estado estacionário na saída. Aqui foi possível não só impor restrições à arquitetura, mas também ao treinamento da rede.

5.2 Aquecedor com Dissipação Variável

Para melhor compreensão do leitor a Figura 5.1 apresenta um diagrama em blocos funcionais de todo o sistema que será identificado no presente capítulo (Cassini, 1999).

A entrada e saída do sistema são: a tensão elétrica no divisor de tensão e a tensão de saída do circuito amplificador, respectivamente. O divisor de tensão é construído de maneira que uma variação na faixa (0 a 127 V) corresponda a uma variação de (0 a 5 V) na entrada do sistema. Os dados de entrada e saída foram coletados através de uma placa de aquisição de dados (PCL 711s).

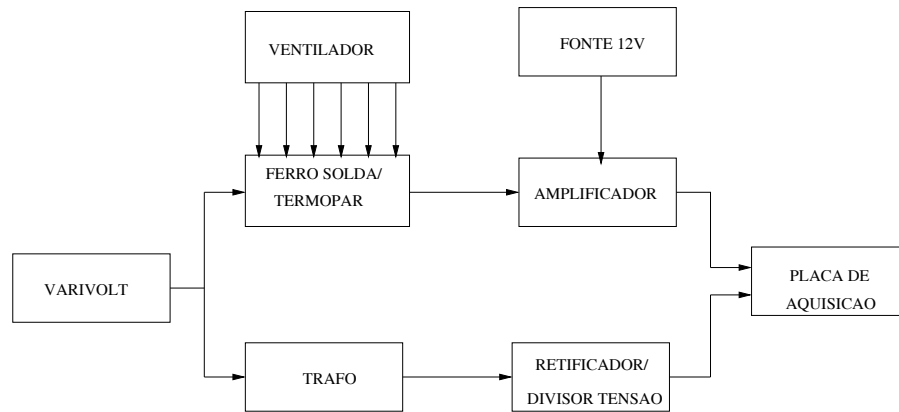


Figura 5.1: Diagrama em blocos funcionais do sistema (Aquecedor).

5.2.1 Descrição do Sistema

A placa de aquisição de dados utilizada tem como limites de tensão, para coleta, as faixas (0 a 5V) e (0 a 10V). O conjunto funcional transformador-retificador-divisor de tensão tem como função, garantir a manutenção dos limites de (0 a 5V).

O varivolt é utilizado para tornar possível a variação da tensão de entrada na faixa de (0 a 136V) da planta (ferro de solda). Esse sinal passa pelo transformador, passando para a faixa de (0 a 18V), e o sinal de saída é retificado.

As variações na temperatura do ferro de solda provocadas pela variação no varivolt são medidas por meio de um termopar acoplado ao mesmo. A tensão nos terminais do termopar são ampliadas por um amplificador de instrumentação para uma faixa de (0 a 4V) e coletadas pela placa de aquisição de dados.

É utilizado ainda um ventilador para variar o ganho estático do sistema.

5.2.2 Teste Dinâmico

Foram realizados dois testes dinâmicos, um com o ventilador desligado e o outro com o ventilador ligado em uma fonte de 127V.

Esses testes foram realizados em uma temperatura ambiente de 24°C por um período de 4h. Os dados foram coletados com uma taxa de amostragem de 6s, resultando num total de 2510 pontos para o teste sem o ventilador

estar ligado e de 2520 pontos para o teste com ventilador ligado. Cada patamar do sinal de excitação do sistema foi mantido por 1 min, gerando aproximadamente 10 amostras por patamar.

Os sinais utilizados nesse trabalho estavam sob a influência do ventilador, e foram dizimados de um fator de 10 seguindo o mesmo procedimento descrito em (Aguirre, 2000a) para determinação da amostragem. A massa de dados foi dividida em duas, sendo a primeira metade utilizada para a identificação do sistema e a segunda metade para a validação do mesmo. Os dados utilizados na identificação podem ser vistos na Figura 5.2 e os dados utilizados na validação na Figura 5.3.

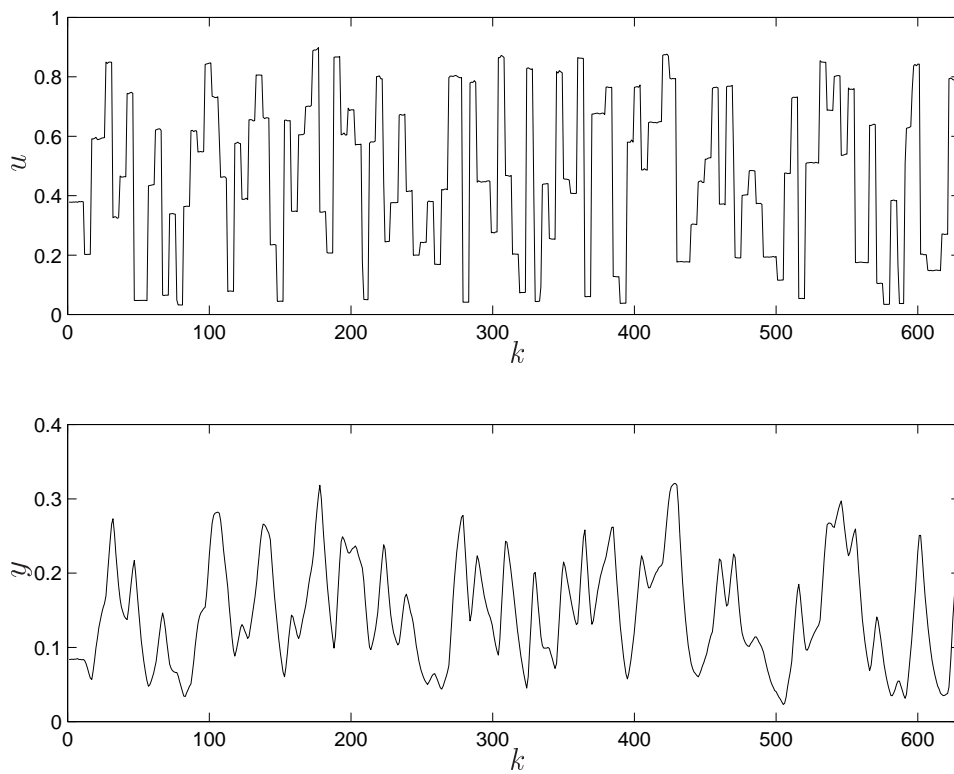


Figura 5.2: Dados de identificação do sistema (`din3n.mat`); (em cima) sinal de entrada do sistema, sendo o eixo dos x o número de amostras k e o eixo dos y o valor em p.u. da tensão de alimentação do sistema; (em baixo) sinal de saída do sistema, sendo o eixo dos x o número de amostras k e o eixo dos y o valor em p.u. da temperatura no aquecedor.

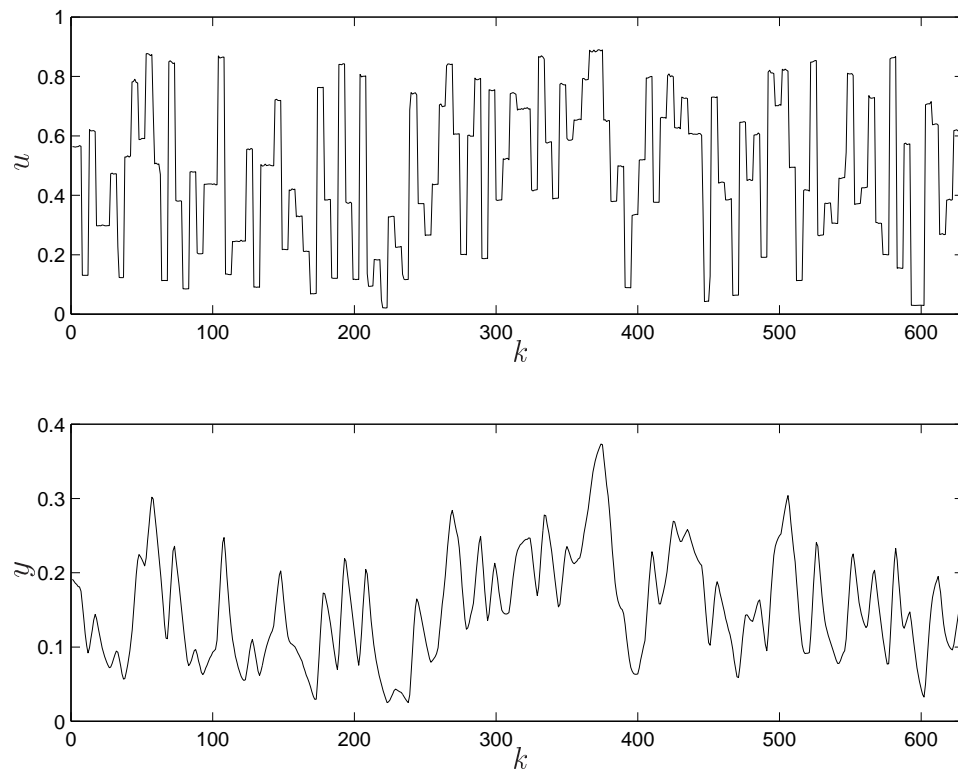


Figura 5.3: Dados de validação do sistema (`din3n.mat`); (em cima) sinal de entrada do sistema, sendo o eixo dos x o número de amostras k e o eixo dos y o valor em p.u. da tensão de alimentação do sistema; (em baixo) sinal de saída do sistema, sendo o eixo dos x o número de amostras k e o eixo dos y o valor em p.u. da temperatura no aquecedor.

5.2.3 Teste Estático

Os testes estáticos têm como finalidade o levantamento da curva estática do sistema. O teste cujos dados são utilizados neste trabalho foi efetuado com o ventilador ligado e teve a duração de 3h (Cassini, 1999).

O teste consistiu em:

1. aplicar um patamar fixo na entrada do sistema;
2. esperar o sistema estabilizar, ou seja, entrar em regime permanente;
3. medir a saída do sistema;
4. colocar os pares ordenados (saída, entrada) em um gráfico;

- o conjunto de todos os pontos no gráfico, descrevem a curva estática do sistema.

A Figura 5.4 mostra a curva estática do modelo. As grandezas estão indicadas em p.u., sendo 1 p.u. correspondente ao referencial AC de 5V do sinal de entrada, visto pelo ferro de solda e a 998,51°C na temperatura do ferro de solda.

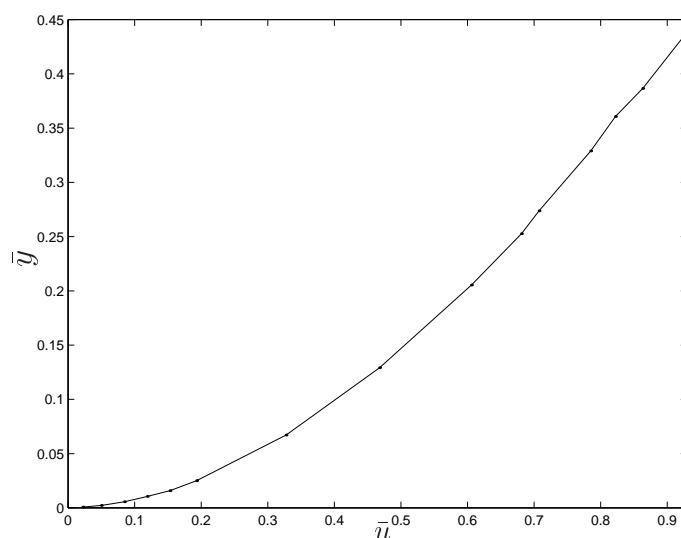


Figura 5.4: Curva estática do sistema (dados `estat3.mat`), sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos y os respectivos dados de saída estáticos (\bar{y}) em p.u.

Outro teste utilizado foi a resposta ao degrau unitário positivo e negativo para determinação da constante de tempo predominante do sistema.

Para o degrau positivo a constante de tempo ficou em torno de 1,81 min e para o degrau negativo em torno de 3,33 min. Por esse motivo os testes realizados foram feitos para uma variação positiva do degrau.

5.3 Identificação do aquecedor elétrico

Um procedimento padrão em identificação de sistemas e em várias outras disciplinas consiste em inicialmente tentar metodologias mais simples primeiro. Partindo desse princípio, a primeira estrutura a tentar modelar a dinâmica do sistema é um modelo linear. Tal tarefa já foi feita em (Cassini, 1999). Os resultados revelam que o modelo tem dificuldades de acompanhar o sistema nos valores extremos. Tal característica é um indicador de “falta

de não-linearidade no modelo”, pois revela que o ganho do modelo embora adequado em uma faixa de operação, é inadequado nos valores extremos. A Figura 5.7(b) mostra a simulação dinâmica do modelo linear e a Figura 5.8(b) a curva estática do modelo.

5.3.1 Identificação caixa preta

Como já foi verificado, um modelo linear não consegue representar bem toda faixa de operação, em vista disso o próximo passo é partir para um modelo não-linear. Várias redes neurais contendo dois neurônios com função de ativação tangente hiperbólica foram identificadas. Para esse número de neurônios na camada oculta os resultados mostraram-se inadequados para representar a estática do sistema, pois para valores de entrada próximos de 1 p.u. a rede apresentava ganho inferior. Aumentando o número de neurônios da camada oculta para três, os resultados não apresentaram diferença aparente na dinâmica, mas a característica estática melhorou consideravelmente. Para número de neurônios na camada oculta maiores que três, melhoras significativas não foram observadas. A função de ativação usada no neurônio da saída foi uma função linear, esta rede foi chamada rede HHH.

Quanto aos regressores, várias redes foram treinadas, variando o número de atrasos da entrada e da saída o atraso resíduo foi fixado em 6, os resultados mostraram que os melhores resultados dinâmicos e estáticos foram obtidos com redes que possuíam 2, 2 e 6, respectivamente, os máximos atrasos da saída, entrada e termos de resíduo.

As redes neurais foram treinadas usando a rotina `nnarmax2.m` (Norgaard, 2000), sendo os critérios de parada o erro médio quadrático ou o número máximo de épocas, respectivamente, 10^{-4} e 300. Estes valores foram escolhidos heurísticamente, os outros critérios de parada do algoritmo ficaram com os valores padrão.

O modelo que se deseja obter, deve aprender não só a dinâmica, mas também a característica estática do aquecedor elétrico. Os resultados foram avaliados pela resposta dinâmica do modelo através da função `nnsimul.m` e a resposta estática também utilizou a mesma função, porém o processo para obtenção da característica estática constou dos seguintes passos:

1. cada valor de entrada do aquecedor u_i é aplicado na rede até que a saída da rede estabilize em torno de um ponto;
2. verificar a estabilização da saída através das diferenças das 4 últimas

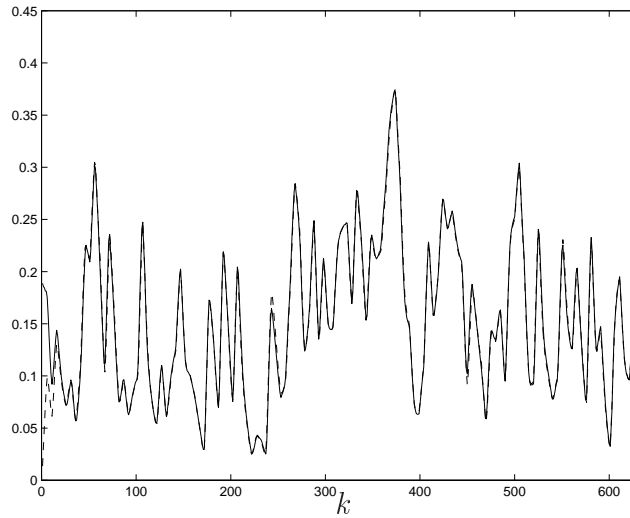


Figura 5.5: Resposta dinâmica da rede HHH para os dados de validação, linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-). Sendo o eixo dos x o número de amostras k e o eixo dos y o valor de saída da rede. Observação: esta rede não utilizou nenhuma técnica de poda.

saídas passadas. Para isso três deltas são calculados:

$$\begin{aligned}\Delta_1 &= y(k) - y(k-1), \\ \Delta_2 &= y(k-1) - y(k-2), \\ \Delta_3 &= y(k-2) - y(k-3).\end{aligned}$$

A condição que determina a estabilização é: quando todos os Δ_i s forem menores ou iguais a 10^{-6} (tolerância) considera-se que a saída está estável.

3. guardar os valores da saída estática da rede \hat{y} em um vetor;
4. colocar os pares ordenados (u, \hat{y}) em um gráfico que descreve a curva estática produzida pela rede neural.

Dentro da abordagem caixa preta três técnicas de poda serão utilizadas objetivando melhorar as respostas dinâmica e estática obtidas Figuras 5.5 e 5.6.

Identificação caixa preta utilizando decaimento de pesos (*Weight Decay*)

Nesta estratégia a mesma arquitetura foi utilizada faltando apenas definir o parâmetro β de regularização. Não há um critério bem definido para a

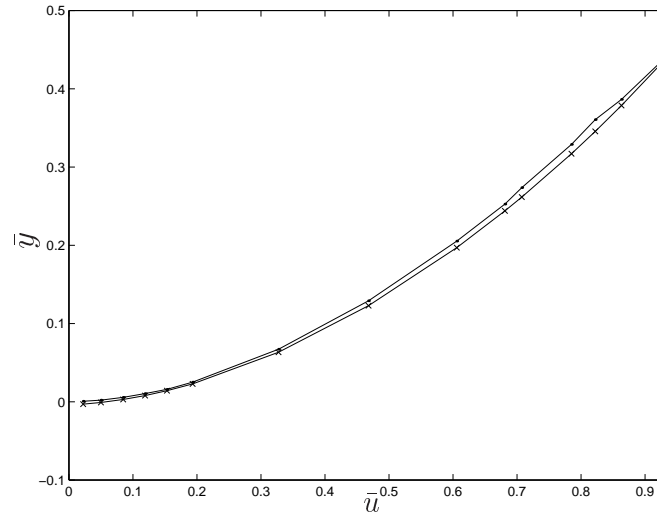


Figura 5.6: Em ponto traço (.-) esta os valores da característica estática coletada do sistema, e em xis traço (x-) a característica estática da rede HHH. Sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos \bar{y} os respectivos valores de saída da rede HHH.

escolha desse parâmetro. Algumas redes foram obtidas variando o parâmetro β . É interessante observar que para valores altos de β (maior que 0,05), a rede apresenta um comportamento semelhante ao de um modelo linear como pode ser visto na Figura 5.7(a). Em um sistema linear a característica estática é uma reta que passa pela origem, como é deduzido abaixo. Seja o sistema dinâmico linear descrito pela equação (5.1):

$$y(k) = a_1 y(k-1) + a_2 y(k-2) + \dots + a_{ny} y(k-ny) + b_1 u(k-1) + b_2 u(k-2) + \dots + b_{nu} u(k-nu). \quad (5.1)$$

Em estado estacionário para entrada constante, $y(k) = y(k-1) = y(k-2) = \dots = y(k-ny) = y$, o mesmo acontece para a entrada $u(k) = u(k-1) = u(k-2) = \dots = u(k-nu) = u$, assim a equação (5.1) se torna:

$$\bar{y} = \left(\frac{b_1 + b_2 + \dots + b_{nu}}{1 - (a_1 + a_2 + \dots + a_{ny})} \right) \bar{u}. \quad (5.2)$$

Exatamente por ser semelhante a um sistema linear, o modelo não responde bem para altos picos dos dados de saída, como pode ser observado na Figura 5.7(a). A característica estática está na Figura 5.8

Nas Figuras 5.9 e 5.10 pode ser visualizado o melhor resultado dinâmico e estático que foi obtido utilizando o método de decaimento de pesos com

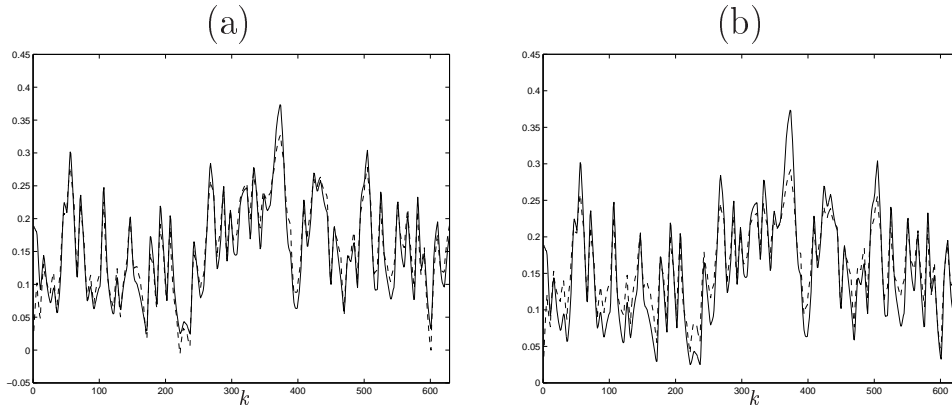


Figura 5.7: (a) Resposta dinâmica da rede HHHDP com parâmetro regularização igual a 0,07, para os dados de validação, linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-), (b) Resposta dinâmica do modelo ARX, para os dados de validação, linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-). Sendo o eixo dos x o número de amostras k e o eixo dos y o valor de saída da rede.

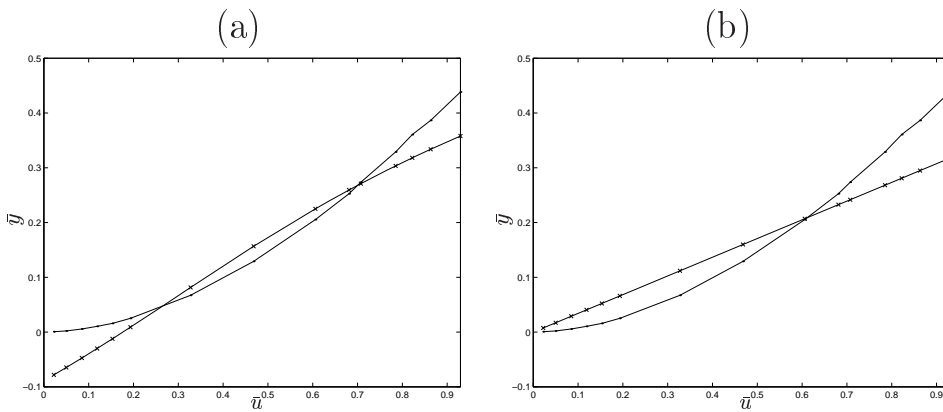


Figura 5.8: (a) Característica estática da rede HHHDP com parâmetro regularização igual a 0,07, para os dados de validação, xis traço (x-). característica estática do sistema em linha contínua (-), (b) Característica estática do modelo ARX, para os dados de validação, linha xis traço (x-). Característica estática do sistema em linha contínua (-). Sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos \bar{y} .

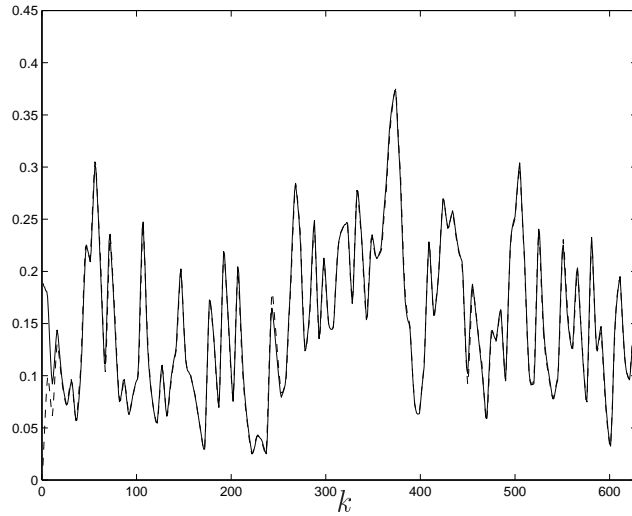


Figura 5.9: Resposta dinâmica da rede HHDPP com parâmetro regularização igual a 0,01, para os dados de validação, linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-). Sendo o eixo dos x o número de amostras k e o eixo dos y o valor de saída da rede.

parâmetro de regularização igual a 0,01. A melhor rede utilizando este método foi chamada rede HHDPP.

Identificação caixa preta utilizando poda estatística

1. Emprego do ERR na poda de redes neurais

Considere uma rede MLP com uma camada oculta, as equações que descrevem a rede são mostradas abaixo:

$$\begin{bmatrix} v'_1(k) \\ v'_2(k) \\ \vdots \\ v'_q(k) \end{bmatrix} = \begin{bmatrix} w_{111} & w_{112} & \cdots & w_{11m} \\ w_{121} & w_{122} & \cdots & w_{12m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{1q1} & w_{1q2} & \cdots & w_{1qm} \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_m(k) \end{bmatrix} + \begin{bmatrix} b_{11} \\ b_{12} \\ \vdots \\ b_{1q} \end{bmatrix}, \quad (5.3)$$

Para que a equação (5.3) fique na forma $y = wu$, será utilizada uma forma estendida da matriz de pesos, em que os termos de polarização são incluídos, como pode ser visto na equação (5.4).

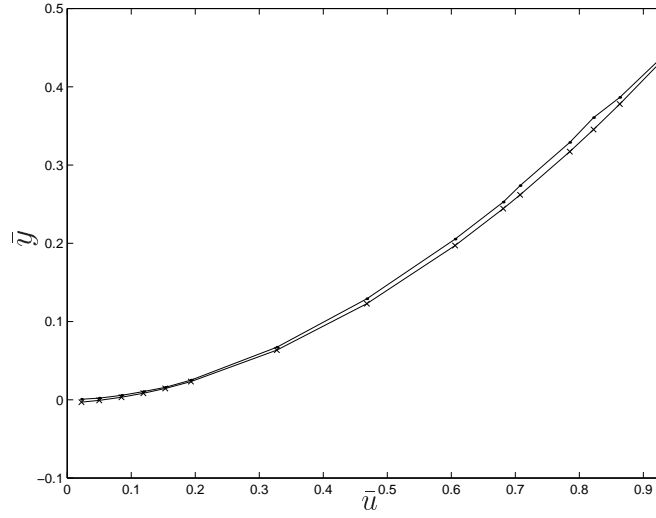


Figura 5.10: Característica estática da rede HHDHP com parâmetro regularização igual a 0,01, para os dados de validação, xis traço (x-). característica estática do sistema em linha contínua (-). Sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos y os respectivos dados de saída estáticos (\bar{y}) em p.u.

$$\begin{bmatrix} v'_1(k) \\ v'_2(k) \\ \vdots \\ v'_q(k) \end{bmatrix} = \begin{bmatrix} w_{111} & w_{112} & \dots & w_{11m} & b_{11} \\ w_{1121} & w_{1122} & \dots & w_{112m} & b_{12} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ w_{1q1} & w_{1q2} & \dots & w_{1qm} & b_{1q} \end{bmatrix} \begin{bmatrix} u_1(k) \\ u_2(k) \\ \vdots \\ u_m(k) \\ 1 \end{bmatrix}, \quad (5.4)$$

$$\begin{bmatrix} v_1(k) \\ v_2(k) \\ \vdots \\ v_q(k) \end{bmatrix} = \begin{bmatrix} f_1(v'_1(k)) \\ f_2(v'_2(k)) \\ \vdots \\ f_3(v'_q(k)) \end{bmatrix}, \quad (5.5)$$

$$\begin{bmatrix} y'_1(k) \\ y'_2(k) \\ \vdots \\ y'_q(k) \end{bmatrix} = \begin{bmatrix} w_{211} & w_{212} & \dots & w_{21m} & b_{21} \\ w_{2121} & w_{2122} & \dots & w_{212m} & b_{22} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ w_{2q1} & w_{2q2} & \dots & w_{2qm} & b_{2q} \end{bmatrix} \begin{bmatrix} v_1(k) \\ v_2(k) \\ \vdots \\ v_m(k) \\ 1 \end{bmatrix}, \quad (5.6)$$

$$\begin{bmatrix} y_1(k) \\ y_2(k) \\ \vdots \\ y_q(k) \end{bmatrix} = \begin{bmatrix} g_1(y'_1(k)) \\ g_2(y'_2(k)) \\ \vdots \\ g_3(y'_q(k)) \end{bmatrix}. \quad (5.7)$$

As funções $f_i(k)$ e $g_i(k)$ são funções de ativação quaisquer. Uma vez que as redes tenham sido treinadas e desde que $u(k)$ e $y(k)$ sejam conhecidos, os sinais intermediários $v'(t)$ e $y'(t)$ podem ser calculados como visto nas equações (5.4) e (5.6). Estas equações estão no formato de (5.3), portanto pode-se aplicar a taxa de redução de erro e os parâmetros mais significantes de \mathbf{W}_1 e \mathbf{W}_2 podem ser selecionados. Se a saída de um neurônio da camada oculta não é relevante para nenhuma das saídas da rede então este neurônio é retirado. O mesmo ocorre com uma entrada que não seja significativa para algum neurônio da camada oculta. Desta forma o ERR é aplicado no conjunto de entradas de cada neurônio selecionando aquelas que são relevantes e podando os parâmetros relacionados às outras entradas. A rede é então retreinada e o processo se repete até que não haja parâmetros com grau de relevância menor que um valor determinado. Resultados mostram que apenas umas poucas iterações são necessárias. A rede em que foi aplicado o err foi chamada rede HHHERR. O critério de poda foi que os pesos após a poda deveriam explicar 99,99 % da saída do neurônio. Os resultados dinâmico e estático após aplicação do ERR estão nas Figuras (5.12) e (5.13), respectivamente. A Figura (5.11) mostra a topologia da rede após a poda.

Identificação utilizando cirurgia cerebral ótimo (*Optimal Brain Surgeon*) - OBS

A poda da rede por meio do OBS foi feita utilizando a rotina `mnprune.m` que aplica a estratégia do OBS em modelos do tipo entrada e saída neurais de sistemas dinâmicos `obsurgeon.m`. Dois diferentes procedimentos são possíveis:

1. Eliminar um peso de cada vez, retreinar a rede e ir repetindo o processo.
2. Eliminar alguma percentagem do número total de pesos, retreinar a rede e ir repetindo o processo.

Não há uma maneira bem definida sobre como escolher quantos pesos são eliminados entre cada retreinamento. Neste trabalho adotou-se o primeiro procedimento. O OBS eliminou pesos referentes aos termos de ruído das redes NARMAX como pode ser visto na Figura (5.15). Como a simulação utiliza apenas os regressores de entrada e saída os resultados apresentados

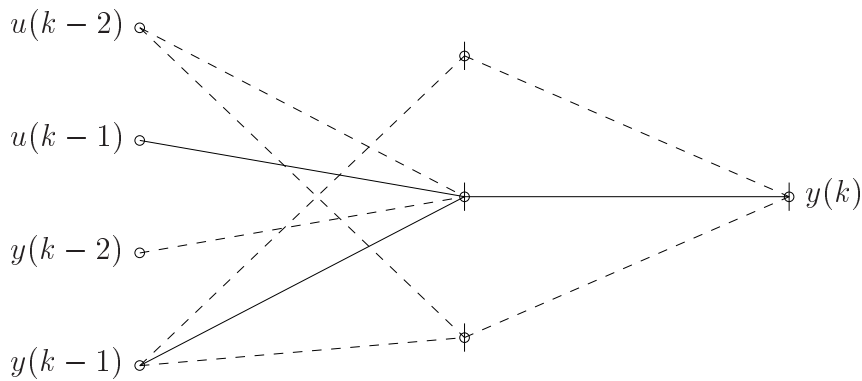


Figura 5.11: Arquitetura da rede podada pelo ERR. Esta rede consta de três neurônios com função de ativação tangente hiperbólica na camada oculta e função linear no neurônio da camada de saída. O símbolo de um pequeno círculo indica o neurônio, quando esta cortado este neurônio possui o *bias*. As linhas tracejadas indicam pesos negativos e as linhas contínuas pesos positivos.

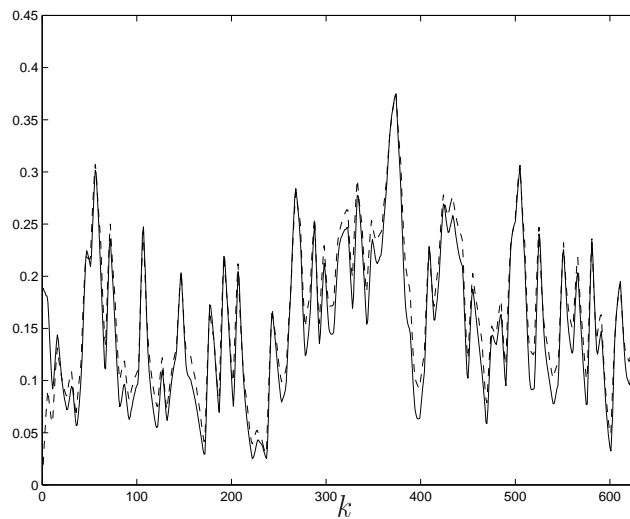


Figura 5.12: Resposta dinâmica da rede HHHERR para os dados de validação, linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-). Sendo o eixo dos x o número de amostras k e o eixo dos y o valor de saída da rede.

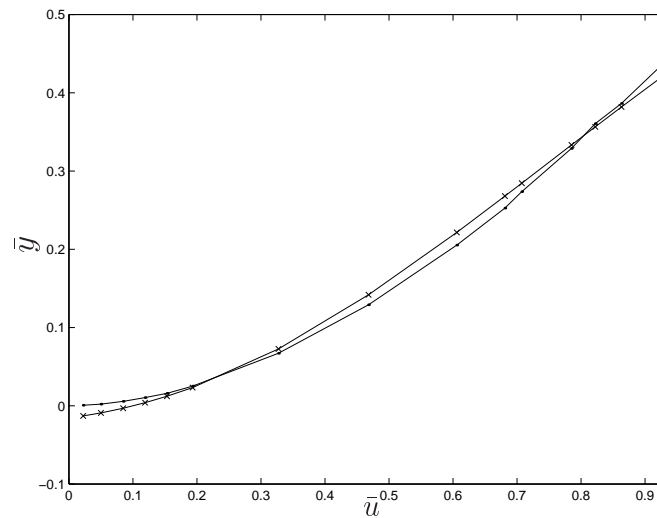


Figura 5.13: Característica estática da rede HHHERR para os dados de validação, xis traço (x-). característica estática do sistema em linha contínua (-). Sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos y os respectivos dados de saída estáticos (\bar{y}) em p.u.

não demonstram diferenças em relação aos resultados da identificação caixa preta. As Figuras 5.16 e 5.17, mostram respectivamente, o resultado da simulação dinâmica e da característica estática, para uma rede que foi podada segundo o primeiro procedimento enunciado anteriormente. A função `nnprune.m` retorna uma matriz contendo vetores, sendo que cada vetor contém todos os pesos da rede a cada poda. A Figura 5.14 mostra o erro para o conjunto de treinamento (x), o erro para o dados de validação (o) e o erro final de predição de Akaike estimado do erro de generalização médio, dentre os vários resultados o melhor é escolhido observando aquele que produz o menor erro de predição final ou o menor erro para o conjunto de validação, como pode ser observado para redes com número de parâmetros maior ou igual a 20 os erros para o conjunto de treinamento e para o conjunto de validação são aproximadamente iguais, mas o erro de predição final é menor para rede com 20 parâmetros, sendo está, portanto, a rede escolhida. A melhor rede é extraída pela função `netstruc.m`. A rede podada com o OBS foi chamada rede HHHOBS.

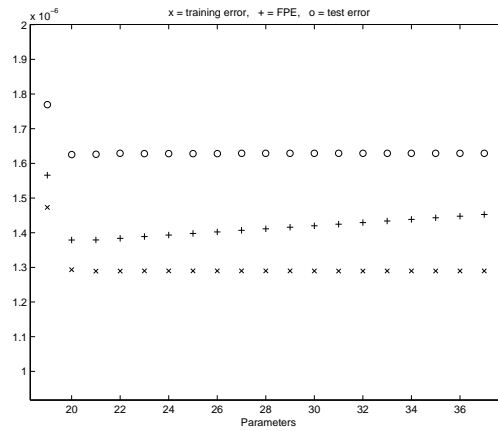


Figura 5.14: Este gráfico apresenta os critérios utilizados após aplicação do OBS para se determinar a melhor rede. o símbolo (o) representa o erro médio quadrático da rede para os dados de validação, os xis (x) representam o erro médio quadrático para os dados de treinamento e as cruces (+) indicam o erro final de predição de Akaike estimado do erro de generalização médio. O eixo dos x é o número de parâmetros (pesos e *bias*) da rede e o eixo dos y o valor dos ítems referidos acima.

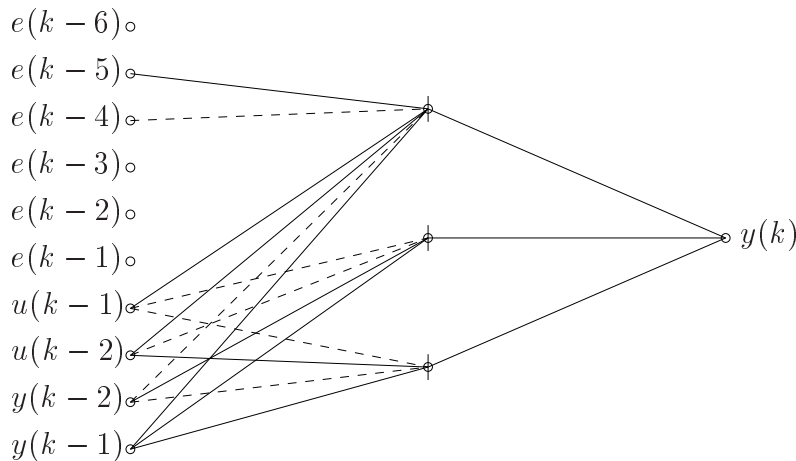


Figura 5.15: Arquitetura da rede podada pelo OBS. Esta rede consta de três neurônios com função de ativação tangente hiperbólica na camada oculta e função linear no neurônio da camada de saída. O símbolo de um pequeno círculo indica o neurônio, quando esta cortado este neurônio possui o *bias*. As linhas tracejadas indicam pesos negativos e as linhas contínuas pesos positivos. Como pode ser visto apenas os termos de ruído foram eliminados.

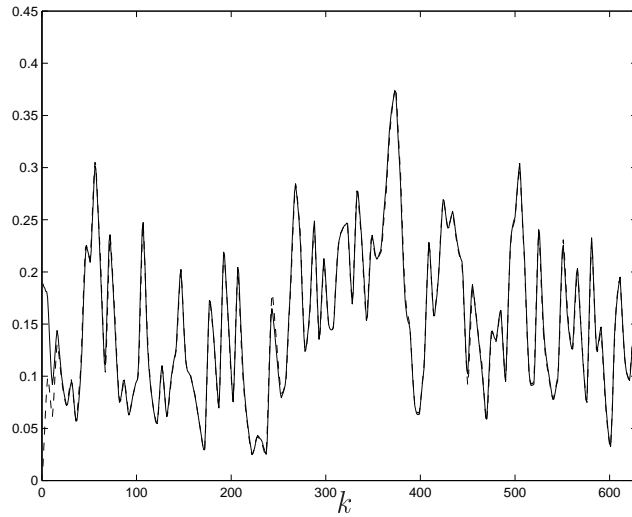


Figura 5.16: Resposta dinâmica da rede HHOBS para os dados de validação, linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-). Sendo o eixo dos x o número de amostras k e o eixo dos y o valor de saída da rede.

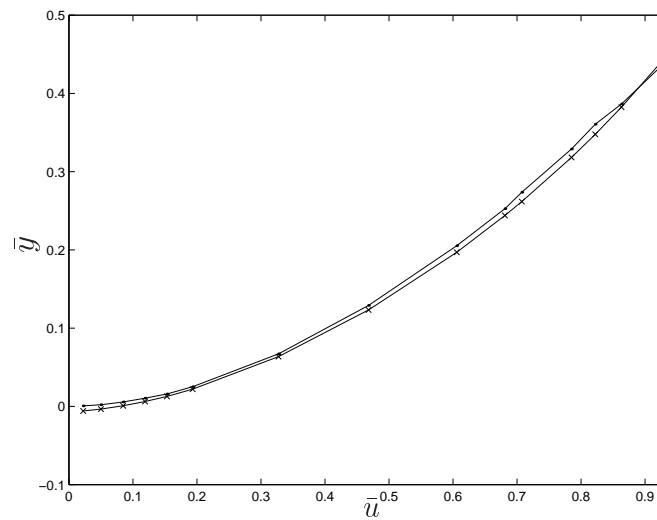


Figura 5.17: Característica estática da rede HHOBS para os dados de validação, xis traço (x-). característica estática do sistema em linha ponto traço (-). Sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos y os respectivos dados de saída estáticos (\bar{y}) em p.u.

5.3.2 Identificação caixa cinza

Nesta estratégia o conhecimento da característica estática do sistema foi utilizado no treinamento da rede, como descrito no capítulo 3. A dedução das restrições impostas a rede começa com duas observações feitas sobre a característica estática do sistema:

1. O sistema apresenta um único ponto fixo estável para cada patamar de entrada, como mostrado na Figura 5.4.
2. A característica estática pode ser aproximada por uma função quadrática. Esta função é sugerida por se tratar da função mais simples que pode reproduzir, aproximadamente aquela relação de variáveis.

Relativo a cada uma destas afirmações duas características serão impostas a rede. A primeira é que, como um único ponto fixo é associado a cada valor de entrada, a arquitetura utilizada é equivalente a ter os regressores referentes a saída do sistema, entrando em um neurônio linear e apenas os regressores referentes às entradas entrando em neurônios não lineares. Esta arquitetura é equivalente a uma das arquiteturas sugeridas em (Narendra and Parthasarathy, 1990).

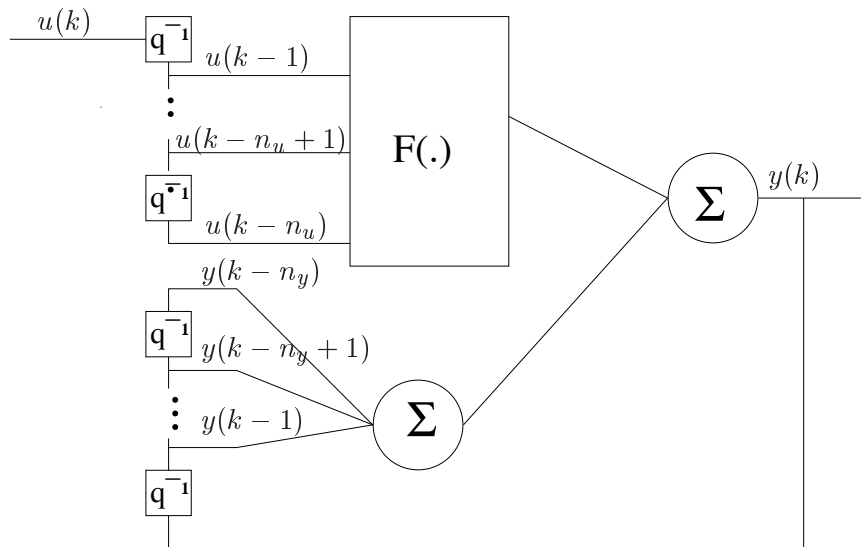


Figura 5.18: Arquitetura proposta em (Narendra and Parthasarathy, 1990). Esta rede possui a particularidade de possuir não-linearidade apenas nos regressores referentes a entrada do sistema.

A segunda observação feita sobre a característica estática fornece uma suposição que auxilia na escolha da parte não-linear da rede. Como se supõe

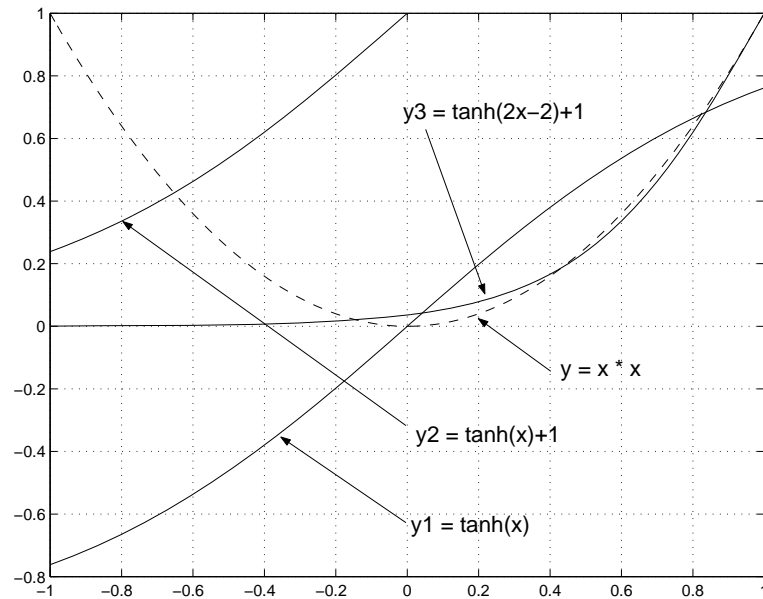


Figura 5.19: Esta figura mostra como a tangente pode aproximar uma quadrática no semi-plano direito do gráfico.

que característica estática seja quadrática pode-se optar por uma função de ativação que facilite o treinamento. Tome por exemplo a função tangente hiperbólica. Dado que uma rede possui uma única camada oculta com apenas um neurônio com função de ativação tangente hiperbólica. A equação que descreve esta rede é a seguinte:

$$y = w_{211} \tanh(w_{111} u + b_{11}) + b_{21}, \quad (5.8)$$

pode-se observar na Figura 5.19 que para que esta rede possa aproximar uma quadrática na região de semiplano direito do gráfico foi necessário fazer $b_{11} = -2$ e $b_{21} = 1$ o que resultou em y_3 . A curvatura de y_3 pode ser ajustada variando w_{211} e w_{111} . Agora será utilizado uma função de ativação gaussiana. Novamente uma rede com apenas um neurônio na camada oculta foi tomado. A equação que descreve esta rede é a que segue:

$$y = w_{211} e^{-(w_{111} u + b_{11})^2} + b_{21}. \quad (5.9)$$

Como pode ser observado na Figura 5.20 bastou fazer $b_{21} = 1$ para que a rede já desse uma aproximação da função quadrática. A curvatura de y_2 pode ser ajustada variando w_{211} e w_{111} . Aqui há um detalhe importante, o termo b_{11} , que o bias da camada oculta *não foi necessário*. Observação: os valores de peso e do termo de polarização foram encontrados empiricamente.

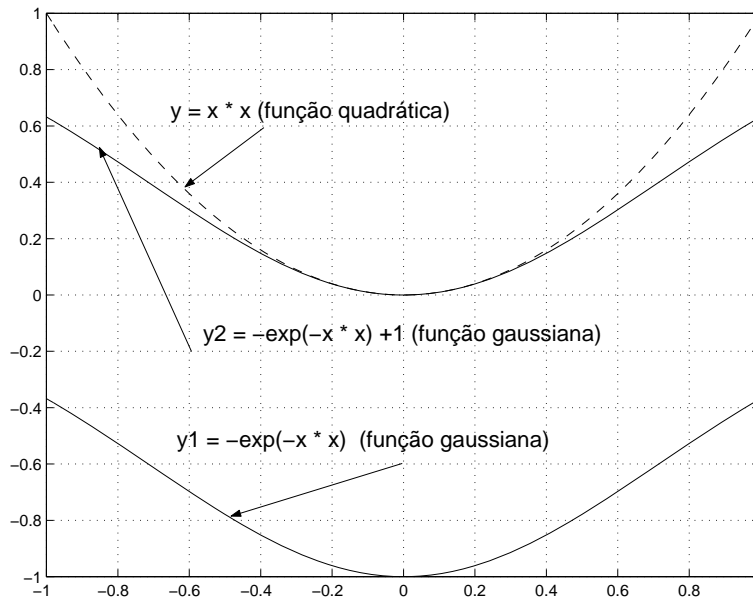


Figura 5.20: Esta figura mostra uma função quadrática sendo aproximada por uma gaussiana.

É importante ressaltar, nesta análise, que a função gaussiana dispensa o (*bias*) da camada oculta, enquanto que para funções como a tangente hiperbólica ele torna-se necessário. Assim é preferível, objetivando obter redes com menos parâmetros, utilizar a função gaussiana. A discussão pode tornar mais abrangente da seguinte forma, quando a função a ser aproximada for uma função par, isso conduziria ao uso de funções de ativação gaussianas sem os termos de polarização da camada oculta. O mesmo pode ser feito para funções ímpares, quando a função a ser aproximada for ímpar, pode se utilizar a função tangente hiperbólica (que é uma função ímpar) sem os bias da camada oculta. A rede resultante pode ser vista na Figura 5.21. Foram usados três neurônios na camada oculta e os mesmos regressores utilizados na identificação caixa preta. Apesar da função de ativação ser diferente espera-se que haja complexidade suficiente para modelar suas características dinâmica e estática.

Imposição de restrições aos pesos da rede

Partindo da rede da Figura 5.21 pode -se escrever a equação que descreve a dinâmica da rede:

$$\begin{aligned} \hat{y}(k) = & w_{211} e^{-(w_{113} u(k-1) + w_{114} u(k-2))^2} + w_{212} e^{-(w_{123} u(k-1) + w_{124} u(k-2))^2} + \\ & w_{213} e^{-(w_{133} u(k-1) + w_{134} u(k-2))^2} + w_{214} w_{141} y(k-1) + \\ & w_{214} w_{142} y(k-2) + w_{215} \end{aligned} \quad (5.10)$$

Reescrevendo a rede em sua forma estática para entrada constante, isto é, fazendo $y(k) = y(k-1) = y(k-2) = \dots = y(k-n_y)$, e a entrada $u(k) = u(k-1) = u(k-2) = \dots = u(k-n_u)$, tem-se:

$$\begin{aligned} \hat{y} = & x_1 e^{-(x_2 u)^2} + x_3 e^{-(x_4 u)^2} + \\ & x_5 e^{-(x_6 u)^2} + x_7, \end{aligned} \quad (5.11)$$

sendo os x_i 's constantes, que foram estimadas como se segue: Dado o arquivo `estat3.mat` que contém os valores medidos da característica estática do sistema e dada a equação estática da rede 5.11, pode-se utilizar a função `lsqcurvefit.m` do toolbox de otimização do programa MatLab, para minimizar o erro médio quadrático. Com os valores dos x_i 's é possível reescrever o processo de treinamento que é:

$$\min \quad E(W) = \frac{1}{2N} \sum_{i=1}^N (d_i - y_i(W, U))^2, \quad (5.12)$$

para,

$$\min \quad E(W) = \frac{1}{2N} \sum_{i=1}^N (d_i - y_i(W, U))^2 \quad (5.13)$$

sujeito a

$$\frac{w_{211}}{(1 - w_{214} w_{141} - w_{214} w_{142})} = x_1 \quad (5.14)$$

$$\frac{w_{212}}{(1 - w_{214} w_{141} - w_{214} w_{142})} = x_3 \quad (5.15)$$

$$\frac{w_{213}}{(1 - w_{214} w_{141} - w_{214} w_{142})} = x_5 \quad (5.16)$$

$$\frac{w_{215}}{(1 - w_{214} w_{141} - w_{214} w_{142})} = x_7 \quad (5.17)$$

$$w_{113} + w_{114} = x_2 \quad (5.18)$$

$$w_{123} + w_{124} = x_4 \quad (5.19)$$

$$w_{133} + w_{134} = x_6. \quad (5.20)$$

Com o problema de otimização formulado pode-se utilizar a rotina `fmincon.m` do toolbox de otimização citado anteriormente, de modo que os pesos possam ser tais que a rede aprenda a informação dinâmica contida nos dados `din3n.mat` e ao mesmo tempo preservem as relações que garantem a curva estática.

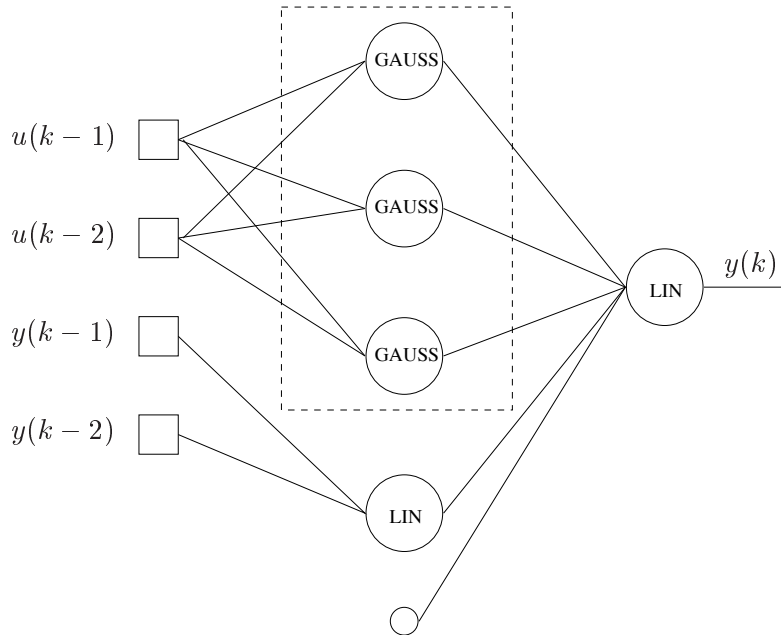


Figura 5.21: Arquitetura da Rede GGGL, que é uma rede com três neurônios com função de ativação gaussiana e um neurônio com função de ativação linear na camada oculta e um neurônio com função de ativação linear na saída. Os regressores referentes a saída do sistema, $y(k-1)$ e $y(k-2)$ passam apenas pelo neurônio linear enquanto que os regressores referentes a entrada, $u(k-1)$ e $u(k-2)$ passam apenas pelos neurônios não lineares. Os neurônios da camada oculta não possuem bias.

Para realizar a simulação dinâmica da rede a rotina `nnsimul.m` teve de ser alterada interiormente, uma vez que o toolbox de redes neurais utilizado só emprega funções de ativação lineares e tangentes hiperbólicas. O resultado da simulação, para o conjunto de validação, pode ser visualizado na Figura 5.22 e o resultado da curva estática da rede na Figura 5.23.

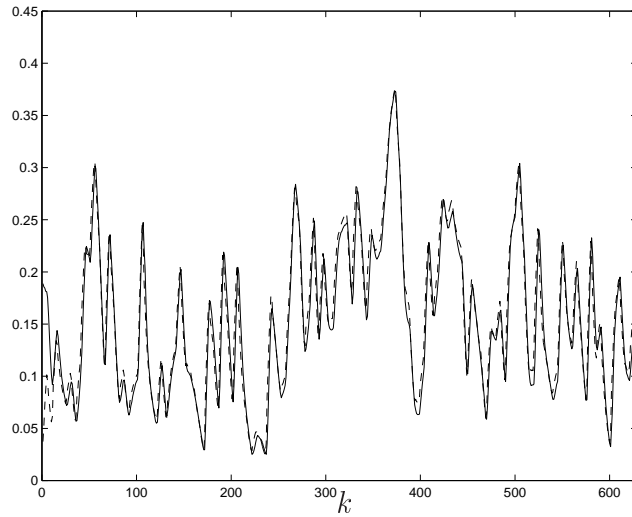


Figura 5.22: Resposta dinâmica da rede GGGL para os dados de validação em linha tracejada (- -). Resposta dinâmica do sistema em linha contínua (-). Sendo o eixo dos x o número de amostras k e o eixo dos y o valor de saída da rede.

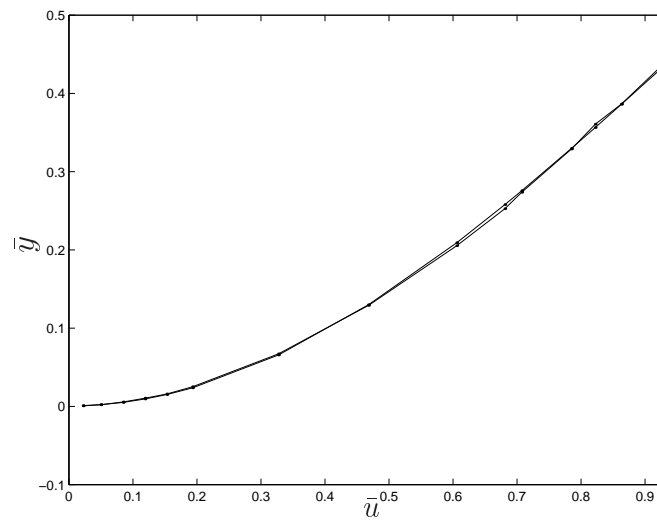


Figura 5.23: Característica estática da rede GGGL para os dados de validação em xis traço (x-) e característica estática do sistema em linha contínua (-). Sendo o eixo dos x os dados estáticos de entrada (\bar{u}) em p.u. e o eixo dos y os respectivos dados de saída estáticos (\bar{y}) em p.u.

5.4 Análise dos resultados

Na Tabela 5.1 os valores de RMSE (raiz do erro quadrático médio) das respostas dinâmica e estática das redes são apresentados. Como pode ser visto os resultados para as redes que utilizaram estratégia caixa preta com ou sem técnica de poda foram muito próximos, à exceção das redes que foram podadas via ERR, parece que a poda local, isto é, podar cada neurônio separadamente não é muito adequado, uma vez que a resposta da rede é uma composição de todos os neurônios. Para estratégia caixa cinza pode ser notado que a dinâmica teve o desempenho um pouco pior, no entanto, houve melhora na resposta estática. Este compromisso existente entre desempenho estático e dinâmico foi recentemente definido na literatura (Coelho et al., 2000). Acredita-se que o melhor desempenho estático recomendará o procedimento descrito em algumas aplicações.

Rede	Dinâmica	Estática
HHH	0,0141	0,0074
HHHDP	0,0142	0,0074
HHHOBS	0,0141	0,0073
HHHERR	0,0223	0,0102
GGGL	0,0196	0,0023

Tabela 5.1: Valores RMSE das 5 estratégias utilizadas na identificação do aquecedor elétrico.

5.5 Conclusão

Neste capítulo duas estratégias de identificação de sistemas foram utilizadas na modelagem dinâmica e estática de um aquecedor elétrico a saber: caixa-preta e caixa-cinza. Na estratégia caixa preta algumas técnicas de poda foram abordadas, decaimento de pesos, cirurgia cerebral ótimo e taxa de redução de erro. O objetivo foi verificar se a resposta já obtida pelas redes neurais sem poda, poderiam ser melhoradas. Os resultados mostraram que, estas técnicas não foram capazes de melhorar significativamente a resposta encontrada. A identificação caixa cinza foi uma estratégia capaz de melhorar a resposta estática da rede com uma pequena perda de qualidade na dinâmica. Esta rede possuía informação incorporada nas duas formas possíveis, incorporação de informação na arquitetura e no treinamento. Na arquitetura a informação auxiliou na escolha da função de ativação e nas conexões da rede. No processo de treinamento a minimização da função erro médio quadrático levou em consideração restrições lineares e não lineares que garantiram uma resposta estática adequada. Portanto, a abordagem caixa cinza mostra-se promissora, uma vez que possibilita arquiteturas mais adequadas, e além disso, que a rede aprenda dos dados a dinâmica e preservem outras características do sistema.

Capítulo 6

Conclusão

6.1 Discussão

Os resultados apresentados neste trabalho, mostram que as redes podem ser usadas de maneira mais criteriosa e não apenas sob uma perspectiva caixa-preta. A incorporação de conhecimento a priori da simetria de pontos fixos, na arquitetura das redes, é mais simples que em algumas representações, como NARMAX polinomial. Apesar dessa relativa facilidade, o caminho de caixa cinza em redes é ainda um grande caminho a ser percorrido.

Há algumas lacunas que devem ainda ser preenchidas. Por exemplo, no capítulo 2, a arquitetura que possuía além do neurônio com função de ativação tangente hiperbólica, possuía também um neurônio com função de ativação linear, conseguiu ter um treinamento mais rápido. A suposição é de que o neurônio linear permite que a rede ajuste de maneira independente a componente linear e não linear da rede. Existe a necessidade de um estudo mais detalhado que verifique os efeitos da presença desse neurônio linear na camada oculta.

6.2 Conclusão

Na identificação do oscilador de Duffing-Ueda foi verificado que a imposição de simetria de pontos fixos e ponto fixo trivial na rede neural é relativamente simples, bastando para isso que as funções de ativação da camada oculta sejam funções ímpares, e que a rede não possua nenhum *bias*. Alternativamente, em algumas situações, mesmo que na presença de *bias*, durante o treinamento a rede aprende a simetria refletindo tal aprendizado na forma de cancelamento do *bias*. Esse procedimento de cancelamento interno a estruturas matemática durante o treinamento, no contexto de modelos

polinomiais NARMAX foi chamado de agrupamento de termos (Aguirre and Billings, 1995b).

O segundo estudo de caso mostra que sistemas que possuem um único estado estacionário (um único ponto fixo para cada entrada) podem, pelo menos em princípio, ser modelados por uma estrutura de rede em que não se encontra não linearidade nos regressores de saída, podendo assim escrever-se uma forma estática em que seja possível impor restrições aos pesos da rede durante o processo de treinamento. Além disso, nos casos em que a característica estática do sistema for par, a rede pode usar funções de ativação par e dispensarem o *bias* da camada oculta. No caso de uma característica ser ímpar algo semelhante acontece, isto é, pode-se utilizar funções de ativação ímpares e dispensar os *bias* da camada oculta, pelo menos em princípio. E se não se tratar de nenhum dos dois casos em questão, redes com funções de ativação com todos os bias devem ser usadas.

6.3 Sugestões de trabalhos futuros

Generalizar os resultados obtidos especificamente para os estudos de caso em questão.

Determinar e/ou desenvolver ferramentas de análise de redes neurais que permitam implementar o item acima.

Aplicar redes que possuem as restrições que garantem a simetria, na identificação de sistemas que apresentam esta propriedade, como por exemplo o circuito de Chua que é um sistema autônomo.

Verificar como outras propriedades dinâmicas, tais como, a estabilidade de ponto fixo e a constante de tempo podem auxiliar na escolha da arquitetura e ou no treinamento das redes. No segundo estudo de caso foi verificado que apesar da abordagem caixa cinza apresentar resultados significativos, um pequeno deslocamento de fase foi verificado, o que sugere que talvez a restrição imposta a rede no sentido de garantir a característica estática, de certa forma limitou o aprendizado da dinâmica do sistema.

Pode ser verificado também como impor restrição aos pesos nos casos onde houver necessidade de não-linearidade nos regressores da saída do sistema e ou termos que relacionem de maneira não linear a entrada e a saída do sistema.

Bibliografia

- Aguirre, L. and Souza, A. (1998). An algorithm for estimating fixed points of dynamical systems from data and model structures. *International Journal of Bifurcation and Chaos*, 8(11):2203–2213.
- Aguirre, L. A. (1996a). A tutorial introduction to nonlinear dynamics and chaos, part I: Modeling and control. *Controle & Automação*, 7(1):29–49.
- Aguirre, L. A. (1996b). A tutorial introduction to nonlinear dynamics and chaos, part II: Tools and benchmarks. *Controle & Automação*, 7(1):50–66.
- Aguirre, L. A. (2000a). *Introdução à Identificação de Sistemas - Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais*. Editora UFMG, 1^a edition.
- Aguirre, L. A. (2000b). A nonlinear dynamical approach to system identification. *IEEE circuits and systems*, 11(2):10–23.
- Aguirre, L. A. and Billings, S. A. (1994). Validating identified nonlinear models with chaotic dynamics. *International Journal of Bifurcation and Chaos*, 1(4):109–125.
- Aguirre, L. A. and Billings, S. A. (1995a). Dynamical effects of over-parametrization in nonlinear models. *Physica D*, 80(1,2):26–40.
- Aguirre, L. A. and Billings, S. A. (1995b). Improved structure selection for nonlinear models based on term clustering. *International Journal of Control*, 62:569–587.
- Aguirre, L. A., Maquet, J., and Letellier, C. (2001). Induced one-parameter bifurcation in identified non-linear dynamical models. Accepted for publication in *International Journal of Bifurcation and Chaos*.
- Bagarinao, E., Pakdaman, K., Nomura, T., and Sato, S. (1999a). Generalized one-parameter bifurcation diagram reconstruction using time series. *Physica D*, 124:258–270.

- Bagarinao, E., Pakdaman, K., Nomura, T., and Sato, S. (1999b). Reconstructing bifurcation diagrams from noisy time series using nonlinear autoregressive models. *Phys. Rev. E*, 60(1):1073–1076.
- Bagarinao, E., Pakdaman, K., Nomura, T., and Sato, S. (1999c). Time series-based bifurcation diagram reconstruction. *Physica D*, 2229:1–21.
- Barroso, M. F. S. (2001). Métodos de Otimização Mono-Objetivo Aplicados à Identificação Caixa-Cinza de Sistemas Não-Lineares. Master's thesis, Universidade Federal de Minas Gerais, Programa de Pós-Graduação em Engenharia Elétrica - Centro de Pesquisa e Desenvolvimento em Engenharia Elétrica.
- Barroso, M. S., Saldanha, R. S., and Aguirre, L. A. (2001). Comparação de métodos mono-objetivos em identificação caixa-cinza. In *XXIV CNMAC*, volume 2, page 183, UniBH. SBMAC.
- Bhat, N. V. and McAvoy, T. J. (1992). Determining model structure for neural models by network pruning. *Computers in Chemical Engineering*, 16:271–281.
- Billings, S. A. and Coca, D. (1999). Discrete wavelet models for identification and qualitative analysis of chaotic systems. *International Journal of Bifurcation and Chaos*, 9(7):1263–1284.
- Braga, A. P., Carvalho, A. P. L. F., and Ludemir, T. B. (2000). *Redes Neurais Artificiais: Teoria e Aplicações*. LTC.
- Cassini, C. C. S. (1999). Estimativa recursiva de características estáticas não lineares utilizando modelos polinomiais NARMAX. Master's thesis, Universidade Federal de Minas Gerais, Programa de Pós-Graduação em Engenharia Elétrica - Centro de Pesquisa e Desenvolvimento em Engenharia Elétrica.
- Coelho, L. S., Corrêa, M. V., Cassini, C. S., and Aguirre, L. A. (2000). A comparison of representations for nonlinear system identification: A case study. *Anais do XIII Congresso Brasileiro de Automática - CBA*.
- Coelho, M. C. S. (2002). Modelos de hammerstein e wiener: conexões com modelos NARX e sua aplicação em identificação de sistemas não-lineares. Master's thesis, Programa de Pós-Graduação em Engenharia Elétrica - Centro de Pesquisa e Desenvolvimento em Engenharia Elétrica.

- Corrêa, M. V., Aguirre, L. A., and Braga, A. P. (1999). Validação de modelos neurais identificados a partir de um sistema caótico. In *IV Congresso Brasileiro de Redes Neurais*, pages 152–158, ITA São José dos Campos, SP, Brasil.
- Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control Signals and Systems*, 2:303–314.
- Dodd, M. (1992). Neural network for control and systems. *IEE control Engineering Series*, (46).
- Feigenbaum, M. J. (1983). Universal behaviour in non-linear systems. *Physica D7*, pages 16–39.
- Fiedler-Ferrara, N. and Prado, C. P. D. (1994). *Caos - Uma Introdução*. Edgard Blücher Ltda.
- Forsell, U. and Lindsjö, P. (1997). Combining semi-physical and neural network modeling: an example of its usefulness. In *SYSID'97*.
- Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural network and the bias/variance dilemma. *Neural Computation*, 4:1–58.
- Guckenheimer, J. and Holmes, P. (1983). *Nonlinear oscillations, dynamical systems, and bifurcation of vector fields*. Springer-Verlag, New York.
- Hagan, M. T. and Menhaj, M. (1994). Training feedforward networks with the Marquardt algorithm. *IEEE Transactions on neural networks*, 5(6):989–993.
- Hagiwara, M. (1990). Novel back propagation for reduction of hidden units and acceleration of convergence using artificial selection. In *IJCNN*, pages 1625–1630, San Diego.
- Hassibi, B., Stork, D. G., and Wolff, G. J. (1992). Optimal brain surgeon and general network pruning. Technical Report 9235, RICOH California Research Center, Menlo Park, CA.
- Haykin, S. (2001). *Redes Neurais. Princípios e prática*. Bookman, Porto Alegre, 2 edition. Tradução: Paulo Martins Engel.
- Henrique, H. M., Lima, E. L., and Pinto, J. C. (1998). A bifurcation study on neural network models for nonlinear dynamic systems. *Latin American Applied Research*, 28:187–200.

- Henrique, M. H., Lima, E. L., and Seborg, D. E. (2000). Model structure determination in neural network models. *Chemical Engineering Science*, 55:5457–5469.
- Hinton, G. E. (1989). Connectionist learning procedures. *Artificial Intelligence*, 40:185–234.
- Hornik, K., Stinchcombe, M., and White, H. (1988). Multilayer feedforward networks are universal approximators. discussion pap., University of California, San Diego, C.A. Dept. Economics.
- Isermann, R., Ernest, S., and Nelles, O. (1997). Identification with dynamical neural networks - architectures, comparisons, applications. In 11th *IFAC Symposium on System Identification*, volume 3, pages 997–1022.
- Kröse, B. and Smagt, P. V. D. (1996). *An Introduction to Neural Networks*. The University of Amsterdam, Kruislaan, eighth edition edition.
- Le Cun, Y., Denker, J. S., and Solla, S. A. (1990). Optimal brain damage. In *Advances in Neural Information Processing*, volume 2, pages 598–605, Denver. D.S. Touretzky.
- Leontaritis, I. and Billings, S. (1985a). Input-output parametric models for non-linear systems part I: deterministic non-linear systems. *International Journal of Control*, 41(2):303–328.
- Leontaritis, I. J. and Billings, S. A. (1985b). Input-output parametric models for non-linear systems part ii: stochastic non-linear systems. *International Journal of Control*, 41(2):329–344.
- Ljung, L. (1985). Development of identification system. Department of Electrical Engineering. Linköping University, Sweden.
- Ljung, L. (1987). *System Identification - Theory for the User*. Information and System Sciences Series. P T R Prentice Hall, New Jersey.
- Ljung, L. and Sjöberg, J. (1992). A system identification perspective on neural nets. *IEEE Workshop on Neural Networks for Signal Processing*, pages 1–13. Available by anonymous ftp 130.236.24.1.
- Mateus, R. G. and Luna, H. P. L. (1986). *Programação não-linear*. Gráfica formato Ltda, Belo Horizonte.
- Mathworks, T. (1999). *MATLAB - Optimization Toolbox*.

- Minsky, M. L. and Papert, S. A. (1969). *Perceptrons*. MIT Press, Cambridge, MA.
- Moon, F. C. (1987). *Chaotic Vibrations - An Introduction for Applied Scientists Engineers*. John Wiley and Sons, New York.
- Mozer, M. C. and Smolensky, P. (1989). Skeletonization: A technique for trimming the fat from a network via relevance assessment. In *Advances in Neural Information Processing*, volume 1, pages 170–115, Denver. D.S. Touretzky.
- Narendra, K. S. and Mukhopadhyay, S. (1997). Neural network for system identification. In *11th IFAC Symposium on System Identification*, volume 2, pages 763–770.
- Narendra, K. S. and Parthasarathy, K. (1990). Identification and control of dynamical systems using neural networks. *IEEE Trans. Neural Net.*, 1:4–27.
- Norgaard, M. (1997). Neural network based system identification - TOOLBOX. Technical report, DTU, Technical University of Denmark. Technical Report 97-E-851.
- Ogata, K. (1993). *Engenharia de Controle Moderno*. Prentice Hall.
- Parker, T. S. and Chua, L. O. (1989). *Practical Numerical Algorithms for Chaotic Systems*. Springer Verlag, Berlin.
- Reed, R. (1993). Pruning algorithms - a survey. *IEEE Transactions on Neural Networks*, 4(5):740–747.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986a). *Learning Internal Representations by Error Propagation*, volume 1. MIT press, Cambridge, MA.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1986b). Learning representations of back-propagation errors. *Nature*, 323:533–536.
- Scheinerman, E. R. (1996). *Invitation to Dynamical System*. Prentice Hall.
- Sjöberg, J. (1995). *Non-linear system identification with neural networks*. PhD thesis, Linköping University, Sweden.
- Sjöberg, J. (1997). On estimation of nonlinear black-box models: How obtain a good initialization. In *IEEE Workshop in Neural Networks for Signal Processing, Ameli Island Plantation, Florida*, pages 72–81.

- Spiegel, M. R. (1994). *Theory and problems of calculus of finite difference and difference equations*. McGraw-Hill, inc.
- Takahashi, R. H. C. (2001). Otimização multiobjetivo. Notas de aulas.
- Tsoi, A. C. and Back, A. (1997). Discrete time recurrent neural network architectures: A unifying review. *Neurocomputing*, 15:183–223.
- Ueda, Y. (1993). *The Road to Chaos*. Aerial Press, inc., P. O. box 1360, Santa Cruz, CA 95061.
- Vega, M. P., Lima, E. L., and Pinto, J. C. (2000). Use of bifurcation analysis for development of non-linear models for control applications. Technical report, Universidade federal do Rio de Janeiro, Cidade universitária, CP: 68502, Rio de Janeiro - RJ, 21945-970, Brasil.
- Wiggins, S. (1990). *Introduction to Applied Nonlinear Dynamical Systems and Chaos*. Springer-Verlag, New York.

Apêndice A

Redes Neurais

A.1 Introdução

Redes neurais são modelos matemáticos inspirados no cérebro humano. Por isso mesmo vários termos característicos da psicologia e da biologia, como adaptação, aprendizado, inteligência, reconhecimento de padrões e auto-organização, têm sido incorporados na literatura de sistemas matemáticos durante as três últimas décadas. Redes neurais apresentam-se como uma forma de processamento alternativo ao processamento realizado em computadores digitais convencionais. Computadores convencionais são extremamente rápidos e precisos na execução de instruções seqüenciais. O sistema de processamento de informação humano é composto de unidades de processamento simples (neurônios) que são ativados (excitados) com velocidades aproximadamente milhões de vezes menores que as portas lógicas que compõem os computadores. No entanto, o cérebro humano realiza rotineiramente tarefas de reconhecimento perceptivo (por exemplo reconhecer um rosto familiar numa cena não familiar) em aproximadamente 100 a 200 ms, ao passo que tarefas de complexidade muito menor podem levar dias para serem executadas em um computador digital (Haykin, 2001). Redes neurais artificiais podem ser caracterizadas como modelos computacionais com propriedades particulares tais como, capacidade de adaptar ou aprender, generalizar, agrupar ou organizar dados, operações que são baseadas em processamento paralelo (Kröse and Smagt, 1996). Pode-se definir Redes Neurais da seguinte forma:

“Redes Neurais é um sistema de processamento paralelo e distribuído, constituído de unidades de processamento simples (neurônios) que têm a capacidade de armazenar conhecimento experimental e torná-lo disponível para o uso” (Haykin, 2001).

A.2 Neurônio

Como já mencionado acima, o neurônio é a unidade de processamento, a partir da qual se constrói a rede. Pode-se dividir o neurônio em três elementos básicos, a saber:

- Um conjunto de conexões. Cada conexão é ponderada por um peso sináptico (W) que é multiplicado pelo sinal que vem do neurônio anterior ou da entrada da rede.
- Um somador. Tem a função de realizar a soma dos sinais ponderados pelos pesos sinápticos.
- Uma função de ativação. É uma função matemática estática que processa o sinal obtido pelo somador.

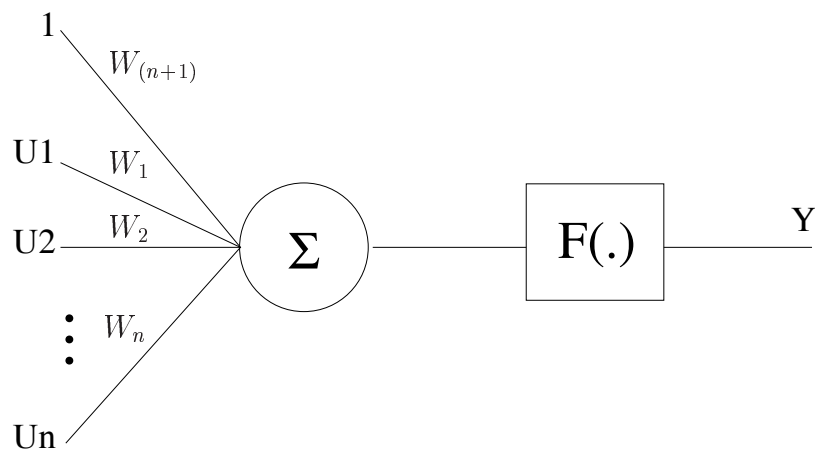


Figura A.1: Modelo de um neurônio artificial.

A.3 Função de ativação

As funções de ativação mais comuns podem ser vistas na Figura A.2 e suas equações são mostradas abaixo:

$$\text{Logística} \quad y = \frac{1}{1 + e^{-x}}, \quad (\text{A.1})$$

$$\text{Tangente Hiperbólica} \quad y = \frac{1 - e^{-x}}{1 + e^{-x}}, \quad (\text{A.2})$$

$$\text{Gaussiana} \quad y = e^{-x^2}, \quad (\text{A.3})$$

$$\text{Linear} \quad y = x, \quad (\text{A.4})$$

Nas equações acima, as variáveis x e y são, respectivamente, a saída do somador e a saída da função de ativação.

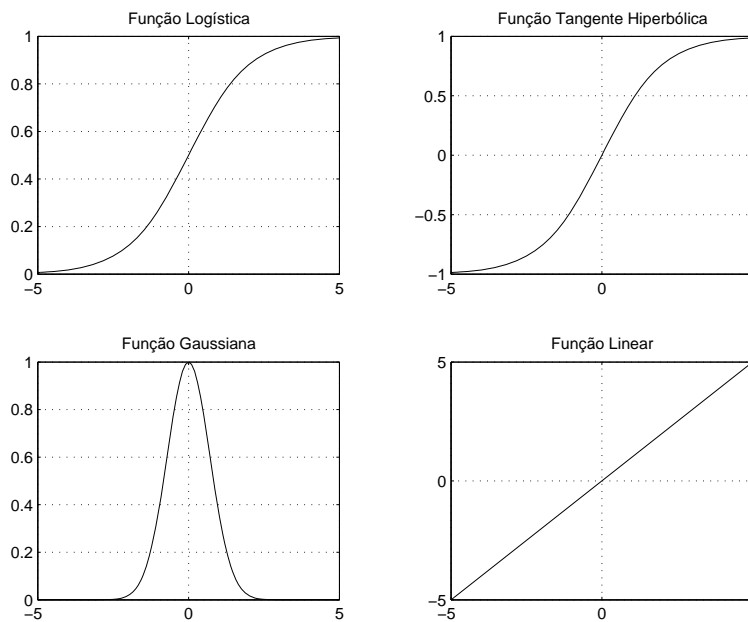


Figura A.2: Funções de ativação. O eixo das abscissas é x e o eixo das ordenadas é y .

A.4 Polarização (Bias)

O termo de polarização tem o efeito de aumentar ou diminuir a entrada líquida do neurônio dependendo se ele é positivo ou negativo. Graficamente o efeito é de transladar horizontalmente o gráfico da função de ativação.

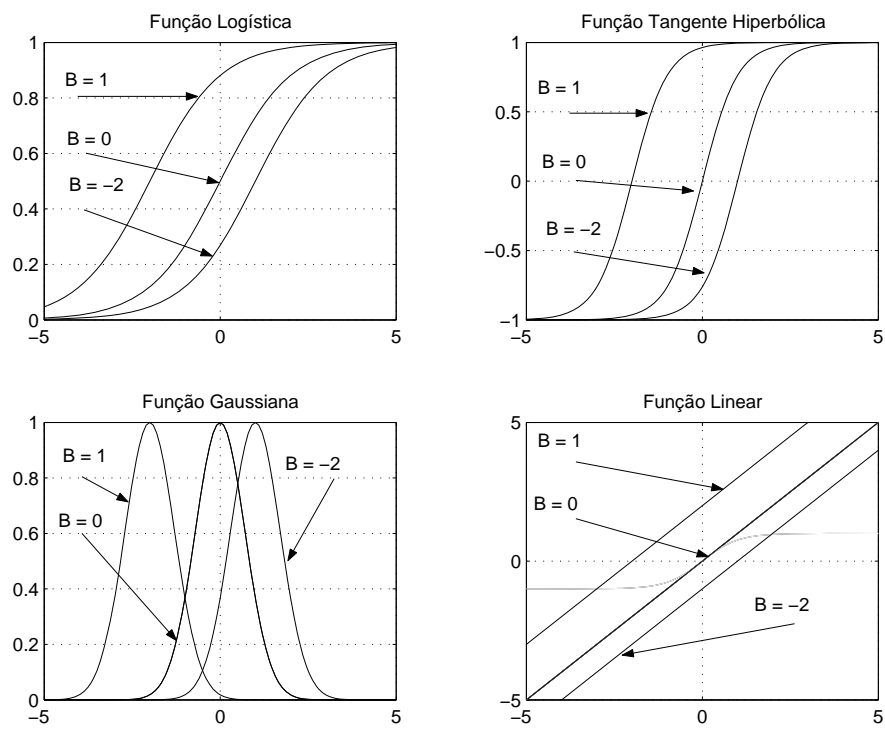


Figura A.3: Efeito do bias nas funções de ativação. O eixo das abscissas é x e o eixo das ordenadas é y .

Nos gráficos da Figura A.3 pode-se observar as funções de ativação com os seus respectivos valores de bias.

A.5 Arquitetura

Fazem parte da definição da arquitetura da rede os seguintes parâmetro (Braga et al., 2000):

- Número de camadas da rede
- Número de neurônios em cada camada
- Tipo de conexão
- Topologia da rede
- * Classificação quanto ao número de camadas
 - Redes de camada única. Entre qualquer entrada e qualquer saída só existe um único neurônio.
 - Redes de múltiplas camadas. Existe mais de um neurônio entre alguma entrada e alguma saída da rede.
- * Classificação quanto ao tipo de conexão
 - *Feedforward* ou acíclica. A saída de um neurônio na i -ésima camada não pode ser usada como entrada de neurônios em camadas de índice menor ou igual a i .
 - *Feedback* ou cíclica. A saída de um neurônio na i -ésima camada é usada como entrada de neurônios em alguma camada de índice menor ou igual a i .
- * Classificação quanto à conectividade
 - Redes fracamente ou parcialmente conectadas
 - Redes completamente conectadas

A.6 Aprendizagem

A capacidade de aprender é uma das mais importantes propriedades das redes neurais artificiais. Define-se aprendizagem no contexto de redes neurais como:

“Aprendizagem é um processo pelo qual os parâmetros livres de uma rede neural são adaptados através de um processo de estimulação pelo ambiente no qual a rede está inserida. O tipo de aprendizagem é determinado pela maneira pela qual a modificação dos parâmetros ocorre” (Haykin, 2001).

A.6.1 Processos de aprendizagem

Os vários processos de aprendizagem podem ser situados em duas categorias a saber (Kröse and Smagt, 1996):

- Aprendizagem supervisionada ou associativa. A rede é treinada para fornecer uma determinada saída para uma dada entrada. Estes pares de entrada e saída podem ser fornecidos por um “professor” externo, ou por um sistema no qual a rede esteja inserida (auto-supervisão) (Braga et al., 2000).

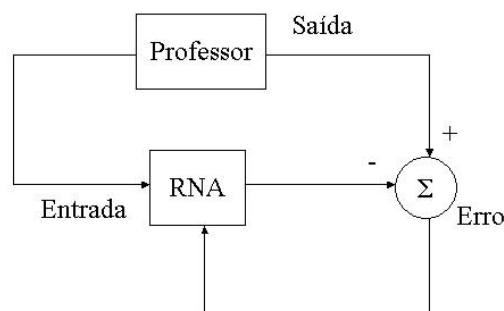


Figura A.4: Aprendizagem supervisionada.

- Aprendizagem não-supervisionada. Neste processo a rede tenta descobrir estatisticamente características salientes da população de entrada (Braga et al., 2000).

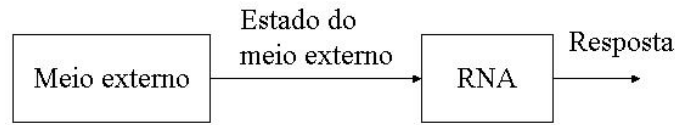


Figura A.5: Aprendizagem não-supervisionada.

A.7 Perceptron

O Perceptron foi proposto por Rosenblatt em 1958, introduzindo o conceito de aprendizagem em redes neurais. O perceptron consistia de um neurônio e um processo de aprendizagem. A topologia original descrita por Rosenblatt era composta por unidades de entrada (retina), por um nível intermediário formado por unidades de associação e por um nível de saída formado pelas unidades de resposta. Embora esta topologia possua três níveis, ela é conhecida como perceptron de uma única camada, já que somente o nível de saída (unidades de resposta) possui propriedades adaptativas (Braga et al., 2000).

A.8 Redes MLP (perceptron de múltiplas camadas)

O perceptron de uma única camada se limita à resolução de problemas linearmente separáveis. Devido a essa limitação e às críticas de Minsky e Papert em (Minsky and Papert, 1969), houve um grande desinteresse na área de redes neurais durante a década de 70 e o início dos anos 80. Em 1986 foi desenvolvido o algoritmo de retropropagação (*backpropagation*) (Rumelhart et al., 1986a,b), o que contribuiu muito para o ressurgimento do interesse em redes neurais. Tal algoritmo permite treinar redes acíclicas (feedforward) com múltiplas camadas. Segundo (Cybenko, 1989), uma rede com uma camada intermediária (oculta) pode implementar qualquer função contínua e a utilização de duas camadas ocultas, permite a aproximação de qualquer função.

A.9 O algoritmo de retropropagação (*backpropagation*)

Este processo de treinamento se enquadra dentro do paradigma supervisionado. Os dados de entrada e saída são coletados do meio ambiente. Com estes dados e a arquitetura da rede, é construída a função erro médio quadrático:

$$e(\mathbf{w}) = \frac{1}{k} \sum_{i=1}^k (d_i - y_i(\mathbf{w}))^2, \quad (\text{A.5})$$

em que, \mathbf{w} é o vetor de pesos da rede, d_i é a i -ésima saída desejada do sistema, $y_i(\mathbf{w})$ é a i -ésima saída da rede e k é o número de amostras. O algoritmo é composto de duas etapas que se repetem iterativamente até que um dos critérios de parada seja alcançado. As etapas são:

- **Propagação.** Os sinais são aplicados à entrada da rede e calcula-se o erro médio quadrático entre a saída da rede e a saída desejada equação (A.5).
- **Retropropagação.** Os pesos são ajustados segundo o método do gradiente, de modo a encontrar os pesos tais que o erro seja nulo (ou menor que uma tolerância), isto é, a saída da rede é igual (ou aproximadamente igual) à saída do sistema para uma dada entrada.

O ajuste de pesos é feito da seguinte maneira:

$$\mathbf{w}_{(i+1)} = \mathbf{w}_i + \Delta_i \mathbf{w}, \quad (\text{A.6})$$

em que:

$$\Delta_i \mathbf{w} = -\eta \frac{\partial e_i(\mathbf{w})}{\partial \mathbf{w}}, \quad (\text{A.7})$$

η é o fator de aprendizado.

A.10 O método Levenberg-Marquardt

O método Levenberg-Marquardt é um método de segunda ordem, isto é, usa não só o gradiente da função, como os métodos de primeira ordem, mas usa uma aproximação da Hessiana (matriz de derivadas segundas) da função erro médio quadrático, o que o torna bem mais robusto a mínimos locais (Hagan and Menhaj, 1994). Neste caso, a equação (A.6) é usada com:

$$\Delta_i W = -(J^T J + \lambda I)^{-1} \frac{\partial e_i(W)}{\partial W}, \quad (\text{A.8})$$

Em que, J é a matriz jacobiana da função erro médio quadrático em relação aos pesos e λ é algum valor não-negativo.

Apêndice B

Técnicas de poda (*pruning*)

B.1 Introdução

A determinação do tamanho de uma rede, isto é, qual o número de parâmetros (pesos e *bias*) mínimos necessários para a rede apresentar um desempenho requerido, não é ainda algo bem definido, mas é algo desejado, uma vez que redes neurais com um tamanho mínimo são menos propensas a aprenderem o ruído presente nos dados e além disso, modelos sobreparametrizados são propensos a apresentarem comportamentos espúrios (Aguirre and Billings, 1995a). Uma forma de tentar eliminar pesos, *bias* ou neurônios desnecessários é empregar técnicas de poda *pruning* (Reed, 1993). Algumas destas técnicas retiram neurônios desnecessários (Mozer and Smolensky, 1989; Hagiwara, 1990), outras eliminam pesos ou *bias* através de um processo de penalização (Hinton, 1989; Bhat and McAvoy, 1992) ou através de sensibilidade da função em relação aos pesos, são poda feitas com o auxílio da Hessiana (Le Cun et al., 1990)(Hassibi et al., 1992) e recentemente uma ferramenta matemática conhecida como taxa de redução de erro também tem sido aplicada (Henrique et al., 2000). Outros métodos podem ser encontrados em (Haykin, 2001).

B.2 Método de decaimento de pesos

Na medida em que o projeto de uma rede é de natureza estatística, precisa-se de um compromisso adequado entre os dados de treinamento e a qualidade do modelo (i.e., um método para resolver o dilema polarização-variância). No contexto da aprendizagem por retropropagação (Backpropagation), ou em qualquer outro procedimento de aprendizado supervisionado

pode-se realizar este compromisso minimizando a função objetivo (risco total) expressa como:

$$E(W) = \frac{1}{2N} \left(\sum_{i=1}^N (y_i - \hat{y}_i(W, U))^2 \right) + \beta \|W\|^2, \quad (\text{B.1})$$

sendo a primeira parcela da soma o erro médio quadrático, no qual, W representa os pesos e termos de polarização da rede e, U os dados de treinamento, enquanto que o segundo termo penaliza a complexidade da rede, que depende apenas da rede. Esse segundo termo força alguns dos pesos da rede a assumirem valores próximos de zero, enquanto que permite que outros mantenham seus valores relativamente elevados. Conseqüentemente, os pesos da rede podem ser grosseiramente agrupados em duas categorias: aqueles que têm grande influência sobre a rede (modelo), e aqueles que têm pequena ou nenhuma influência sobre ela. Os pesos dessa última categoria são referidos como *pesos excessivos*. O parâmetro β é o parâmetro de regularização. Quando este é zero, o processo de retropropagação (backpropagation) é irrestrito, sendo totalmente determinado pelos exemplos de treinamento. Quando β é infinitamente grande a implicação é que a restrição que foi imposta pela penalização da complexidade da rede é por si só suficiente para especificar a rede, o que é uma forma de dizer que os exemplos de treinamento não são confiáveis (Haykin, 1994). Em aplicações práticas atribue-se um valor intermediário entre esses dois extremos.

B.3 Poda estatística

A implementação feita neste trabalho foi baseado em (Henrique et al., 2000). O termo ERR vem do inglês *error reduction ratio*. Para definir a Taxa de Redução de Erro considere a seguinte equação de regressão:

$$y(k) = \sum_{i=1}^{n\theta} g_i(k)\omega_i + \xi(k), \quad (k = 1, \dots, n), \quad (\text{B.2})$$

sendo que os $\hat{g}_i(k)$ são os parâmetros e os ω_i são os regressores em (B.2). Sendo os ω_i , são ortogonais sobre os dados, ou seja,

$$\begin{aligned} \frac{1}{N} \sum_{k=1}^N \omega_i(k)\omega_{j+1}(k) &= 0, \\ \overline{\omega_i(k)\omega_{j+1}(k)} &= 0, \quad \forall i \neq j. \end{aligned} \quad (\text{B.3})$$

A operação $\langle \cdot, \cdot \rangle$ representa o produto interno. No caso do vetor \mathbf{y} , o produto interno é dado por $\mathbf{y}^T \mathbf{y}$, isto é, a soma dos valores quadráticos de \mathbf{y} . Multiplicando a equação B.2 por ela mesma tem-se:

$$y(t)^2 = \left(\sum_{i=1}^{n\theta} g_i(t)\omega_i + \xi(t) \right) \times \left(\sum_{i=1}^{n\theta} g_i(t)\omega_i + \xi(t) \right). \quad (\text{B.4})$$

Tomando-se o valor médio de (B.4) sobre os dados tem-se:

$$\begin{aligned} \langle \mathbf{y}, \mathbf{y} \rangle &= \sum_{i=1}^{n\theta} g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle + \sum_{i=1, j=1, \forall i \neq j}^{n\theta} 2g_i g_j \langle \mathbf{w}_i, \mathbf{w}_j \rangle + \\ &\quad + 2 \sum_{i=1}^{n\theta} g_i \langle \mathbf{w}_i, \boldsymbol{\xi} \rangle + \langle \boldsymbol{\xi}, \boldsymbol{\xi} \rangle \\ &= \sum_{i=1}^{n\theta} g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle + \langle \boldsymbol{\xi}, \boldsymbol{\xi} \rangle. \end{aligned} \quad (\text{B.5})$$

O último passo da equação (B.5) foi obtido observando-se que, por construção, os regressores \mathbf{w}_i são ortogonais entre si e que tais regressores não estão correlacionados com o resíduo $\boldsymbol{\xi}$. Portanto, a taxa de redução de erro pode ser definida da seguinte forma:

$$[ERR]_i = \frac{g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle}{\langle \mathbf{y}, \mathbf{y} \rangle}, \quad (\text{B.6})$$

isto é, se o i -ésimo regressor for incluído no modelo, a parcela de $\langle \mathbf{y}, \mathbf{y} \rangle$ que passa a explicar é $g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle$. O que significa, que se um dado regressor \mathbf{w}_i possuir um alto valor de ERR, o modelo "deve", pelo menos quando se tratar de caixa preta, incluir o regressor \mathbf{w}_i .

B.4 Cirurgia cerebral ótimo

A idéia do cirurgião cerebral ótimo (CCO)(Hassibi et al., 1992), é usar informação da derivada segunda da rede para verificar quais os pesos e *bias* (parâmetros) cuja eliminação causaria o menor aumento do erro quadrático médio (E_{MQ}). O ponto inicial desta técnica é uma aproximação da função custo E_{MQ} usando série de *Taylor* em torno de um ponto de operação:

$$\begin{aligned} E_{MQ}(\mathbf{W} + \Delta \mathbf{W}) &= E_{MQ}(\mathbf{W}) + g^T(\mathbf{W})\Delta \mathbf{W} + \\ &\quad + \frac{1}{2} \Delta \mathbf{W}^T H \Delta \mathbf{W} + O(\|\Delta \mathbf{W}\|^3), \end{aligned} \quad (\text{B.7})$$

em que: $\Delta \mathbf{W}$ é uma perturbação aplicada a um ponto de operação \mathbf{W} , $g(\mathbf{W})$ é o vetor gradiente calculado em \mathbf{W} , H é a matriz hessiana calculada no ponto \mathbf{W} .

Para desenvolver o algoritmo de poda, parte-se das aproximações:

1. Aproximação externa

Começa-se o processo de poda após a rede já ter sido treinada o que implica que o conjunto de parâmetros representa um mínimo local ou global da superfície de erro. Neste caso o gradiente pode ser igualado a zero podendo, portanto, ser eliminado o termo $g^T(\mathbf{W})\Delta \mathbf{W}$ da equação (B.7).

2. Aproximação quadrática

Assume-se que a superfície de erro em torno de mínimo local é “quadrática”. Com isso os termos de ordem maior que dois podem ser eliminados da série de *Taylor* na equação (B.7).

Destas suposições pode-se reescrever a equação (B.7) como:

$$\Delta E_{MQ} = E_{MQ}(\mathbf{W} + \Delta \mathbf{W}) - E_{MQ}(\mathbf{W}) - \frac{1}{2} \Delta \mathbf{W}^T H \Delta \mathbf{W}. \quad (\text{B.8})$$

O procedimento de poda baseado na matriz hessiana usado neste trabalho foi o método conhecido como cirurgia cerebral ótimo (CCO). A estratégia do método O.B.S. é fixar um dos pesos W_i para minimizar o aumento incremental de E_{MQ} dado na equação (B.8). A eliminação destes pesos é correspondente a:

$$\mathbf{W}_i + \mathbf{1}_i^T \Delta \mathbf{W}_i = 0. \quad (\text{B.9})$$

O termo $\mathbf{1}_i$ é um vetor cujos elementos são todos zero, exceto o i -ésimo elemento, que é igual à unidade. O problema pode ser reformulado da seguinte forma:

“Minimize a forma quadrática $S = \frac{1}{2} \Delta \mathbf{W}^T H \Delta \mathbf{W}$ em relação à variação incremental do vetor de pesos, $\Delta \mathbf{W}$, sujeito à restrição $\mathbf{W}_i + \Delta \mathbf{W}_i = 0$ e então minimize o resultado em relação ao índice i .”

Para resolver, monta-se o Lagrangeano $\frac{1}{2} \Delta \mathbf{W}^T H \Delta \mathbf{W} - \lambda(\Delta W_i + W_i)$, em que o λ é o multiplicador de Lagrange. Como valor ótimo do Lagrangeano

S em relação o elemento W_i é:

$$S_i = \frac{W_i^2}{2[H^{-1}]_{i,i}}, \quad (\text{B.10})$$

em que, H^{-1} é a matriz hessiana e $(H^{-1})_{i,i}$ é o ii -ésimo elemento da inversa da matriz hessiana. A esse elemento S_i dá-se o nome de Saliência. A Saliência representa o aumento no erro médio quadrático devido à eliminação do parâmetro W_i . Assim pesos com menores valores de S_i . Após pesos ser eliminado os outros pesos são reajustados segundo:

$$\Delta \mathbf{W} = \frac{W_i}{[H^{-1}]_{i,i}} [H^{-1}] \mathbf{1}_i. \quad (\text{B.11})$$

Quando não se pode mais fazer nenhuma poda sem um grande aumento no erro médio, a rede é retreinada. O algoritmo para, quando após o retreinamento, não se pode mais eliminar pesos sem que o erro aumente acima de um valor limite.

