

PONTIFÍCIA UNIVERSIDADE CATÓLICA DO PARANÁ
CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DA PRODUÇÃO E
SISTEMAS

IDENTIFICAÇÃO DE SISTEMAS NÃO-LINEARES MULTIVARIÁVEIS
USANDO REDES NEURAS PERCEPTRON MULTICAMADAS E
FUNÇÃO DE BASE RADIAL

CURITIBA, MAIO DE 2006

FABIANO LOPES ROCHA

**IDENTIFICAÇÃO DE SISTEMAS NÃO-LINEARES MULTIVARIÁVEIS
USANDO REDES NEURAS PERCEPTRON MULTICAMADAS E
FUNÇÃO DE BASE RADIAL**

Dissertação de Mestrado apresentada ao curso de Pós-Graduação em Engenharia da Produção e Sistemas, Pontifícia Universidade Católica do Paraná.

Orientador: Prof. Dr. Leandro dos Santos Coelho.

CURITIBA, MAIO DE 2006

R672i
2006 Rocha, Fabiano Lopes
Identificação de sistemas não-lineares multivariáveis usando redes neurais perceptron multicamadas e função de base radial / Fabiano Lopes Rocha ; orientador, Leandro dos Santos Coelho. – 2006.
103 p. : il. ; 30 cm

Dissertação (mestrado) – Pontifícia Universidade Católica do Paraná, Curitiba, 2006
Inclui bibliografia

1. Redes neurais (Computação). 2. Sistemas não-lineares. 3. Sistemas de energia elétrica. I. Coelho, Leandro dos Santos. II. Pontifícia Universidade Católica do Paraná. Programa de Pós-Graduação em Engenharia de Produção e Sistemas. III. Título.

CDD-20.ed. 006.32
 003.75
 621.317

Dedico este trabalho à mulher da minha vida, minha esposa, confidente, amante e mais que tudo companheira.

“De tudo, ao meu amor serei atento
Antes, e com tal zelo, e sempre, e tanto
Que mesmo em face do maior encanto
Dele se encante mais meu pensamento.”

Vinicius de Moraes

AGRADECIMENTO

Agradeço

Aos meus pais, Fernando e Célia Maria, que me deram a dádiva da vida, acreditaram em mim e deram-me incentivos e recursos para concluir meus estudos e concretizar meu sonho.

À minha esposa, Larissa, pela grande compreensão que teve nos momentos mais difíceis e, principalmente, pelo apoio, sem o qual não teria concluído este trabalho.

À minha filha, Amanda, pelas brincadeiras e *longas* conversas em todos os momentos que precisei descansar, além do grande amor que recebo todos os dias.

Ao meu irmão, Jaime Luís, que sempre se fez presente em minha vida e contribuiu com seu incentivo na vitória de mais uma etapa da minha formação.

Ao meu orientador, Prof. Dr. Leandro dos Santos Coelho, que acreditou na minha proposta e orientou-me na realização desse trabalho.

A todos os colegas, professores e amigos que de alguma forma me ajudaram a chegar na conquista deste título.

Muito Obrigado!

RESUMO

A identificação de sistemas dinâmicos não-lineares multivariáveis é uma área importante em várias áreas da Engenharia. Esta dissertação apresenta o estudo de uma metodologia baseada em redes neurais artificiais para identificação de sistemas não-lineares com várias entradas e saídas, motivado principalmente pelo fato de as redes neurais artificiais apresentarem potencialidades para identificação de sistemas não-lineares, ou seja, habilidade de tratar sistemas complexos, representação de conhecimento quantitativo, processamento paralelo, aprendizado, adaptabilidade e generalização. As redes neurais artificiais avaliadas neste estudo são: (i) rede neural de base radial e (ii) rede perceptron multicamadas. As simulações foram realizadas para dois estudos de caso de identificação não-linear multivariável. Nos estudos de casos, são utilizados sistemas não-lineares de geração de energia elétrica. Os resultados obtidos pelas redes neurais artificiais foram promissores, motivando futuras pesquisas em identificação de sistemas baseadas em novas configurações de redes neurais em aplicações em sistemas de potência.

Palavras-chave: redes neurais artificiais, identificação não-linear, sistemas multivariáveis, sistemas de potência.

ABSTRACT

The identification of multivariable nonlinear dynamic systems is an important area in Engineering. This dissertation presents a methodology based on artificial neural networks for identification of nonlinear system with some inputs and outputs. This study it is mainly motivated by artificial neural networks to present potentialities for identification of nonlinear systems, such as: ability to treat with complex systems, representation of quantitative knowledge, parallel processing, learning, adaptability and generalization. The evaluated artificial neural networks in this study are: (i) radial basis function neural network and (ii) multilayer perceptron neural network. Simulations had been carried through for two case studies of multivariable nonlinear identification. The studies of cases are of nonlinear systems of generation of electric energy. The results gotten for artificial neural networks had been promising, motivating future research in identification of systems based on new configurations of neural networks in applications in power systems.

Keywords: artificial neural networks, nonlinear identification, multivariable systems, power systems.

SUMÁRIO

	pág.
AGRADECIMENTO	4
RESUMO.....	5
ABSTRACT.....	6
LISTA DE SIGLAS.....	9
LISTA DE FIGURAS	10
LISTA DE TABELAS	13
1 INTRODUÇÃO.....	14
1.1 MOTIVAÇÃO	14
1.2 ESTADO DA ARTE.....	14
1.3 OBJETIVO	18
1.4 JUSTIFICATIVA.....	19
1.5 ORGANIZAÇÃO DA DISSERTAÇÃO.....	20
2 IDENTIFICAÇÃO DE SISTEMAS.....	22
2.1 EXPERIMENTOS E A DETECÇÃO DE NÃO-LINEARIDADES.....	24
2.2 REPRESENTAÇÕES MATEMÁTICAS PARA SISTEMAS NÃO-LINEARES	25
2.3 DETECÇÃO DE ESTRUTURA DE MODELOS NARMAX POLINOMIAIS.....	27
2.4 ESTIMAÇÃO DE PARÂMETROS.....	28
2.5 CRITÉRIOS DE INFORMAÇÃO	31
2.6 VALIDAÇÃO DE MODELOS	31
2.7 SELEÇÃO DE ESTRUTURA DE MODELOS NÃO-LINEARES.....	33
2.8 USO DE CONHECIMENTO <i>A PRIORI</i> NA SELEÇÃO DE ESTRUTURAS	34
3 REDES NEURAIS ARTIFICIAIS	36
3.1 INTRODUÇÃO À INTELIGÊNCIA ARTIFICIAL	36
3.2 REDES NEURAIS ARTIFICIAIS.....	38
3.3 ARQUITETURA DE REDES NEURAIS	40
3.3.1 <i>Perceptron</i> de camada única.....	40
3.3.2 <i>Perceptron</i> multicamadas.....	42
3.3.3 Arquitetura do MLP	43
3.3.4 Definição de uma topologia ideal para o MLP	44
3.3.5 Treinamento de RNs do tipo MLP	45

3.3.6	Rede neural função de base radial.....	52
3.3.7	Método k -médias.....	56
3.3.8	RNs para identificação de sistemas não-lineares.....	56
4	ESTUDO DE CASOS E ANÁLISE DE RESULTADOS.....	58
4.1	CASO 1: SISTEMA DE GERAÇÃO DE PONT-SUR-SAMBRE	58
4.2	CASO 2: SISTEMA DE GERAÇÃO DE ABBOTT.....	59
4.3	RESULTADOS DE SIMULAÇÃO	61
5	CONCLUSÃO E FUTURA PESQUISA	92
	REFERÊNCIAS BIBLIOGRÁFICAS.....	95

LISTA DE SIGLAS

ABNT – Associação Brasileira de Normas Técnicas

AIC – Critério de Informação de Akaike

ATP – *Alternative Transient Program*

BIC – Critério de Informação de Bayes

ERR – *Error Reduction Ratio*

FAM – *Fuzzy Adaptive Memory*

FDA – *Fault Detection and Accomodation*

IA – Inteligência Artificial

IEEE – *Institute of Electrical and Electronics Engineers*

MLP – Perceptron Multi-Camadas

NARI – *Nonlinear AutoRegressive Integrated*

NARMA – *Nonlinear AutoRegressive Moving Average inputs*

NARMAX – *Nonlinear AutoRegressive Moving Average eXogeneous inputs*

PDP – *Parallel Distributed Processing*

PRESS - Soma dos quadrados dos erros de previsão

PSS – *Power System Stabilizer*

RBF – *Radial Basis Function*

RMSE – *Root Mean Square Error*

RN – Rede Neural

RNA – Rede Neural Artificial

SCADA – Sistema Digital de Supervisão e Controle

SNNS – *Stuttgart Neural Network Simulator*

SVM – Máquina Vetorial

LISTA DE FIGURAS

	pág.
Figura 1.1 – Acoplamento de sinais.	18
Figura 3.1 - Topologia do <i>perceptron</i> simples.	42
Figura 3.2 - Rede MLP com camada intermediária.	43
Figura 3.3 - Fluxo de processamento do algoritmo retropropagação do erro.	46
Figura 3.4 - Superfície do erro de uma rede MLP.	49
Figura 3.5 – Estrutura básica da função base radial.	53
Figura 4.1 – Modelo da entradas e saídas do sistema Pont-sur-Sambre.	59
Figura 4.2 - Modelo da entradas e saídas do sistema Abbott.	60
Figura 4.3 - Saída $y_1(t)$ para 3 neurônios na camada oculta da RN MLP.	63
Figura 4.4 - Saída $y_2(t)$ para 3 neurônios na camada oculta da RN MLP.	63
Figura 4.5 - Saída $y_3(t)$ para 3 neurônios na camada oculta da RN MLP.	63
Figura 4.6 - Saída $y_1(t)$ para 4 neurônios na camada oculta da RN MLP.	64
Figura 4.7 - Saída $y_2(t)$ para 4 neurônios na camada oculta da RN MLP.	64
Figura 4.8 - Saída $y_3(t)$ para 4 neurônios na camada oculta da RN MLP.	65
Figura 4.9 - Saída $y_1(t)$ para 5 neurônios na camada oculta da RN MLP.	65
Figura 4.10 - Saída $y_2(t)$ para 5 neurônios na camada oculta da RN MLP.	65
Figura 4.11 - Saída $y_3(t)$ para 5 neurônios na camada oculta da RN MLP.	66
Figura 4.12 - Saída $y_1(t)$ para 6 neurônios na camada oculta da RN MLP.	66
Figura 4.13 - Saída $y_2(t)$ para 6 neurônios na camada oculta da RN MLP.	66
Figura 4.14 - Saída $y_3(t)$ para 6 neurônios na camada oculta da RN MLP.	67
Figura 4.15 - Saída $y_1(t)$ para 7 neurônios na camada oculta da RN MLP.	67
Figura 4.16 - Saída $y_2(t)$ para 7 neurônios na camada oculta da RN MLP.	67
Figura 4.17 - Saída $y_3(t)$ para 7 neurônios na camada oculta da RN MLP.	68
Figura 4.18 - Saída $y_1(t)$ para 8 neurônios na camada oculta da RN MLP.	68
Figura 4.19 - Saída $y_2(t)$ para 8 neurônios na camada oculta da RN MLP.	69
Figura 4.20 - Saída $y_3(t)$ para 8 neurônios na camada oculta da RN MLP.	69
Figura 4.21 - Saída $y_1(t)$ para 3 centros da RN RBF.	70
Figura 4.22 - Saída $y_2(t)$ para 3 centros da RN RBF.	70
Figura 4.23 - Saída $y_3(t)$ para 3 centros da RN RBF.	70
Figura 4.24 - Saída $y_1(t)$ para 4 centros da RN RBF.	71

Figura 4.25 - Saída $y_2(t)$ para 4 centros da RN RBF.....	71
Figura 4.26 - Saída $y_3(t)$ para 4 centros da RN RBF.....	71
Figura 4.27 - Saída $y_1(t)$ para 5 centros da RN RBF.....	72
Figura 4.28 - Saída $y_2(t)$ para 5 centros da RN RBF.....	72
Figura 4.29 - Saída $y_3(t)$ para 5 centros da RN RBF.....	72
Figura 4.30 - Saída $y_1(t)$ para 6 centros da RN RBF.....	73
Figura 4.31 - Saída $y_2(t)$ para 6 centros da RN RBF.....	73
Figura 4.32 - Saída $y_3(t)$ para 6 centros da RN RBF.....	73
Figura 4.33 - Saída $y_1(t)$ para 7 centros da RN RBF.....	74
Figura 4.34 - Saída $y_2(t)$ para 7 centros da RN RBF.....	74
Figura 4.35 - Saída $y_3(t)$ para 7 centros da RN RBF.....	74
Figura 4.36 - Saída $y_1(t)$ para 8 centros da RN RBF.....	75
Figura 4.37 - Saída $y_2(t)$ para 8 centros da RN RBF.....	75
Figura 4.38 - Saída $y_3(t)$ para 8 centros da RN RBF.....	75
Figura 4.39 - Saída $y_1(t)$ para 3 neurônios na camada oculta da RN MLP.....	76
Figura 4.40 - Saída $y_2(t)$ para 3 neurônios na camada oculta da RN MLP.....	76
Figura 4.41 - Saída $y_3(t)$ para 3 neurônios na camada oculta da RN MLP.....	77
Figura 4.42 - Saída $y_4(t)$ para 3 neurônios na camada oculta da RN MLP.....	77
Figura 4.43 - Saída $y_1(t)$ para 4 neurônios na camada oculta da RN MLP.....	77
Figura 4.44 - Saída $y_2(t)$ para 4 neurônios na camada oculta da RN MLP.....	78
Figura 4.45 - Saída $y_3(t)$ para 4 neurônios na camada oculta da RN MLP.....	78
Figura 4.46 - Saída $y_4(t)$ para 4 neurônios na camada oculta da RN MLP.....	78
Figura 4.47 - Saída $y_1(t)$ para 5 neurônios na camada oculta da RN MLP.....	79
Figura 4.48 - Saída $y_2(t)$ para 5 neurônios na camada oculta da RN MLP.....	79
Figura 4.49 - Saída $y_3(t)$ para 5 neurônios na camada oculta da RN MLP.....	79
Figura 4.50 - Saída $y_4(t)$ para 5 neurônios na camada oculta da RN MLP.....	80
Figura 4.51 - Saída $y_1(t)$ para 6 neurônios na camada oculta da RN MLP.....	80
Figura 4.52 - Saída $y_2(t)$ para 6 neurônios na camada oculta da RN MLP.....	80
Figura 4.53 - Saída $y_3(t)$ para 6 neurônios na camada oculta da RN MLP.....	81
Figura 4.54 - Saída $y_4(t)$ para 6 neurônios na camada oculta da RN MLP.....	81
Figura 4.55 - Saída $y_1(t)$ para 7 neurônios na camada oculta da RN MLP.....	81
Figura 4.56 - Saída $y_2(t)$ para 7 neurônios na camada oculta da RN MLP.....	82
Figura 4.57 - Saída $y_3(t)$ para 7 neurônios na camada oculta da RN MLP.....	82

Figura 4.58 - Saída $y_4(t)$ para 7 neurônios na camada oculta da RN MLP.....	82
Figura 4.59 - Saída $y_1(t)$ para 8 neurônios na camada oculta da RN MLP.....	83
Figura 4.60 - Saída $y_2(t)$ para 8 neurônios na camada oculta da RN MLP.....	83
Figura 4.61 - Saída $y_3(t)$ para 8 neurônios na camada oculta da RN MLP.....	83
Figura 4.62 - Saída $y_4(t)$ para 8 neurônios na camada oculta da RN MLP.....	84
Figura 4.63 - Saída $y_1(t)$ para 4 centros da RN RBF.....	84
Figura 4.64 - Saída $y_2(t)$ para 4 centros da RN RBF.....	85
Figura 4.65 - Saída $y_3(t)$ para 4 centros da RN RBF.....	85
Figura 4.66 - Saída $y_4(t)$ para 4 centros da RN RBF.....	85
Figura 4.67 - Saída $y_1(t)$ para 5 centros da RN RBF.....	86
Figura 4.68 - Saída $y_2(t)$ para 5 centros da RN RBF.....	86
Figura 4.69 - Saída $y_3(t)$ para 5 centros da RN RBF.....	86
Figura 4.70 - Saída $y_4(t)$ para 5 centros da RN RBF.....	87
Figura 4.71 - Saída $y_1(t)$ para 6 centros da RN RBF.....	87
Figura 4.72 - Saída $y_2(t)$ para 6 centros da RN RBF.....	87
Figura 4.73 - Saída $y_3(t)$ para 6 centros da RN RBF.....	88
Figura 4.74 - Saída $y_4(t)$ para 6 centros da RN RBF.....	88
Figura 4.75 - Saída $y_1(t)$ para 7 centros da RN RBF.....	88
Figura 4.76 - Saída $y_2(t)$ para 7 centros da RN RBF.....	89
Figura 4.77 - Saída $y_3(t)$ para 7 centros da RN RBF.....	89
Figura 4.78 - Saída $y_4(t)$ para 7 centros da RN RBF.....	89
Figura 4.79 - Saída $y_1(t)$ para 8 centros da RN RBF.....	90
Figura 4.80 - Saída $y_2(t)$ para 8 centros da RN RBF.....	90
Figura 4.81 - Saída $y_3(t)$ para 8 centros da RN RBF.....	90
Figura 4.82 - Saída $y_4(t)$ para 8 centros da RN RBF.....	91

LISTA DE TABELAS

Tabela 4.1 – RMSE para rede MLP no caso de Point-sur-sambre.....	61
Tabela 4.2 – RMSE para rede RBF no caso de Point-sur-sambre.....	61
Tabela 4.3 – RMSE para rede MLP no caso de Abbott.....	62
Tabela 4.4 - RMSE para rede RBF no caso de Abbott.....	62

1 INTRODUÇÃO

1.1 MOTIVAÇÃO

O segmento de identificação de sistemas não-lineares é importante para diversas áreas da Engenharia. A capacidade de se ter um modelo matemático de um sistema permite vários estudos, tais como detecção de falhas, controle de processos, análise de comportamento complexo, entre outros.

Um aspecto útil sobre identificação de sistemas é a capacidade de se prever o comportamento de um sistema dinâmico com algum tempo de antecedência. Neste contexto, é possível tomar decisões preventivas sobre um sistema quanto sujeito a alguma falha ou mesmo mudança dinâmica.

Por outro lado, as redes neurais artificiais, com sua capacidade de aprendizado e adaptabilidade, tornam-se uma ferramenta potencial no estudo de sistemas não-lineares complexos.

1.2 ESTADO DA ARTE

Existem diversos estudos relacionados à modelagem e identificação não-linear em engenharia, principalmente na área de sistemas de potência, este é o foco dos dois estudos de casos desta dissertação. Nesse contexto, a literatura é rica em pesquisas quanto à utilização de redes neurais artificiais e outros sistemas inteligentes em aplicações vinculadas a identificação de sistemas de potência.

Ting *et al.* (1995) realizaram um estudo e a simulação de uma rede neural artificial aplicado a um problema de identificação em um sistema de potência. Nesse trabalho, foi desenvolvida uma técnica de identificação que precisa pouco conhecimento da estrutura do

modelo. Nesse estudo foi obtido um método com convergência rápida dos parâmetros e com procedimentos simples de treinamento.

Tan *et al.* (1995), propuseram uma técnica recursiva de identificação para sistemas dinâmicos multivariáveis e não-linear. A técnica aproxima um problema não-linear da identificação do sistema em dois estágios, onde um estágio constrói uma rede neural função de base radial (RBF) e o outro projeta uma ponderação recursiva estável para atualizar os pesos da RBF.

Gillard & Bollinger (1996) investigaram o uso de uma rede neural multicamadas na medição da função de transferência de um sistema de potência para o uso no estabilizador do sistema de potência (PSS). O objetivo desta pesquisa é medir a função de transferência que relaciona a saída de potência elétrica à entrada da tensão de referência em um sistema de potência que opera em circunstâncias normais.

Chow & Leung (1996) desenvolveram uma técnica de previsão de carga através da utilização de redes neurais. Nesse trabalho foi identificado um modelo auto-regressivo não-linear integrado (NARI) como sendo apropriado para incluir a compensação temporal na previsão da carga elétrica.

Bostanci *et al.* (1997) mostram a influência das cargas no estudo da estabilidade da tensão elétrica através de uma metodologia de redes neurais para identificar a dinâmica da carga num sistema de potência. O modelo matemático desenvolvido foi capaz de prever com razoável sucesso o comportamento futuro da carga do sistema de potência para entradas arbitrárias.

Hiyama *et al.* (1997) investigaram o uso de redes neurais artificiais com realimentação para modelar cargas dinâmicas de um sistema de potência usando dados coletados em campo. Também foi obtida uma resposta de frequência da carga através de uma

rede neural artificial do tipo perceptron multicamadas utilizada para identificar a dinâmica da carga. Com isso, conseguiram emular a dinâmica da carga com boa aproximação.

Souza *et al.* (1997) propuseram um método de distinguir erros analógicos e topológicos em sistemas de potência. Neste contexto, uma rede neural foi capaz de identificar as anomalias e falhas do sistema.

Kumar & Srivastava (1999) apresentaram um método de previsão do estados de um sistema de potência usando redes neurais artificiais. Nesse método, a previsão do estado do sistema ocorre em duas etapas: (i) filtragem e (ii) previsão.

Oleskovicz *et al.* (2000) demonstraram a aplicação de redes neurais artificiais no reconhecimento de padrões para a classificação de falhas num sistema de transmissão de energia elétrica. Os autores desenvolveram comparações de diferentes topologias de redes neurais do tipo MLP (Multilayer Perceptron), que foram validadas usando o *software Neural Works*. Quanto ao treinamento, utilizaram o algoritmo supervisionado *Norm-Cum-Delta Learning Rule*. Este algoritmo é uma variação do algoritmo da retropropagação do erro. Os dados de treinamento da rede neural foram gerados usando o *software ATP (Alternative Transient Program)*. A rede estudada obteve resultados satisfatórios na detecção de falhas no sistema de transmissão de energia elétrica.

Lu & Hogg (2000) descreveram um modelo não-linear de um sistema de potência baseado em princípios físicos e modelo neural. Nas comparações dos resultados da simulação dos modelos da rede neural e do modelo físico não-linear em circunstâncias treinadas ou não foram satisfatórios dentro do espaço analisado e se operados em condições normais, mas insatisfatória em algumas circunstâncias extremas.

Venayagamoorthy & Harley (2001) apresentaram o projeto de um sistema de controle baseado em redes neurais artificiais. Neste trabalho foi desenvolvida e treinada uma rede para identificar a dinâmica não-linear de um sistema de potência e outra rede neural para

controlar um turbogerador. Os resultados foram melhores que qualquer outro controlador, com isso conseguiu-se apropriados identificadores e controladores para um turbogerador, mas em situações que extrapolem seu espaço já não satisfazem todas as condições.

Lin *et al.* (2001) estudaram o conceito da identificação de falhas em uma linha de transmissão de energia elétrica. Para essa finalidade utilizaram uma rede neural RBF, tendo como algoritmo de treinamento o método dos mínimos quadrados ortogonais. Esse trabalho aborda a rede neural RBF em simulações usando como dados de testes de tensão e corrente elétrica. Obtiveram como resultado uma rede neural com convergência rápida e capaz de detectar falhas em um tempo curto, em alguns casos podendo ser usado em tempo real no sistema.

Park *et al.* (2002) realizaram uma comparação de desempenho entre as redes neurais MLP e as RBF. Essas redes foram submetidas a identificação de um sistema dinâmico não-linear, um gerador síncrono. Os experimentos demonstraram que a RBF converge mais rápido e necessita de menos memória que a MLP.

Stankovic *et al.* (2003) desenvolveram uma rede neural artificial para a identificação de um modelo equivalente de ordem reduzida de um sistema de potência. Esse modelo do sistema é formulado em tempo contínuo e com isso, torna compatível com modelos padrões de componentes do sistema de potência.

Ramaswamy *et al.* (2003) propuseram um método automático de identificação de falhas em um sistema de potência. Neste método é utilizado o sinal transiente da forma de onda de um sistema de potência e uma rede neural probabilística.

Magalhães (2004) apresentou duas abordagens de previsão. A primeira aborda modelos matemáticos de previsão de séries temporais baseados em agrupamento nebuloso de dados, que capacitam o sistema explorar os dados históricos disponíveis. A segunda abordagem se refere à combinação (método híbrido) de modelos preditivos que apresentem

desempenho individual promissor em determinados pontos da série temporal, de forma que o método híbrido seja capaz de determinar quais modelos locais são relevantes para uma dada previsão.

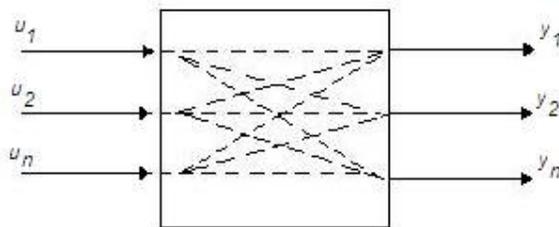
Cardoso *et al.* (2004) desenvolveram um sistema capaz de auxiliar operadores de um centro de controle de uma subestação de energia elétrica na tarefa de localização de uma falha na rede elétrica. O sistema foi concebido com a modelagem dos barramento de proteção, das linhas de transmissão e dos transformadores. Essas modelagens foram realizadas utilizando dois tipos redes neurais: a rede neural de regressão generalizada e a rede neural MLP.

1.3 OBJETIVO

O principal objetivo deste trabalho é a aplicação de redes neurais na identificação de sistemas não-lineares, multivariáveis e em casos onde ocorre acoplamento de sinais tanto de entrada quanto de saída.

Pode-se dizer que sinais estão acoplados quando um sinal influencia um outro conforme apresentado na figura 1.1, na qual as linhas tracejadas representam a interdependência dos sinais.

FIGURA 1.1 – ACOPLAMENTO DE SINAIS.



O acoplamento de sinais pode ocorrer de três formas:

- (i) Acoplamento de Entrada;

- (ii) Acoplamento de Saída;
- (iii) Acoplamento de Entrada e Saída.

O acoplamento de entrada ocorre quando mais de um sinal de entrada influenciam o sinal de saída resultante, ficando a equação da seguinte forma:

$$y_1 = f(u_1, u_2, \dots, u_n) \quad (1.1)$$

Já no caso no do acoplamento de saída a variante da equação fica da seguinte forma:

$$y_1 = f[y_1(t-1), y_2(t-1), \dots, y_n(t-1)] \quad (1.2)$$

Por fim tem-se o acoplamento de entrada e saída, que nada mais que uma união da equação 1.1 com a 1.2, obtendo:

$$y_1 = f[u_1, u_2, \dots, u_n, y_1(t-1), y_2(t-1), \dots, y_n(t-1)] \quad (1.3)$$

As equações (1.1) a (1.3) para y_1 podem ser generalizadas para sistemas com múltiplas saídas, tal que y_1 a y_n podem apresentar acoplamentos.

1.4 JUSTIFICATIVA

A crescente modernização dos sistemas computacionais viabilizou a implementação de abordagens de sistemas inteligentes, pois o aumento das capacidades de processamento e armazenamento de memória faz com que essas abordagens sejam eficientes para aplicações na indústria. Este aspecto fez com que fossem realizados, em diversos centros de pesquisa estudos na área de sistemas de potência [Ting *et al.* (1995), Gillard & Bollinger (1996), Bostanci *et al.* (1997), Souza *et al.* (1997), Ramaswamy *et al.* (2003), entre outros] usando técnicas de inteligência artificial, atingindo, deste modo, um crescimento de conhecimento na área da inteligência artificial e identificação de sistemas.

As redes neurais artificiais são técnicas eficientes da inteligência artificial para lidar com sistemas complexos, não-lineares e multivariáveis, pois as redes neurais artificiais

possuem características como a capacidade de adaptação, aprendizado, reconhecimento de padrões, memorização e aproximação de funções não-lineares que são habilidades fundamentais para tratar de sistemas complexos.

As redes neurais artificiais, por terem a potencialidade de adaptar e aprender, tornam-se ferramentas apropriadas à previsão de comportamentos dinâmicos. Essa habilidade de apreensão e adaptação são capacidades próprias que a rede neural possui para lidar com situações complexas. Uma rede satisfatoriamente treinada é capaz de generalizar o sistema quando são apresentados, nas entradas, dados novos, não utilizados na fase de treinamento. Com essa capacidade, as redes neurais artificiais se tornam técnicas ideais para estimar qualquer sistema não-linear com um grande grau de precisão.

Outra potencialidade das redes neurais que justifica sua escolha é a capacidade de lidar com sistemas multivariáveis de forma análoga a abordagens para lidarem com sistemas menos complexos. Isso é possível devido à potencialidade de uma rede neural processar várias informações de forma paralela e distribuída.

Com isso, a rede neural é capaz de processar várias entradas simultaneamente (com acoplamentos) e de forma independente, o que é fundamental na identificação de sistemas de multivariáveis. Pelo fato de processar as entradas de forma paralela, a rede neural torna o processamento do sistema de multivariáveis possível e promissor.

1.5 ORGANIZAÇÃO DA DISSERTAÇÃO

Esta dissertação está dividida em 5 capítulos.

No capítulo 2 são abordados os fundamentos não-lineares e da identificação de sistemas. Estes fundamentos são relevantes para melhor contextualização das redes neurais artificiais.

No capítulo 3 são detalhadas a fundamentação teórica e potencialidades das redes neurais artificiais. Além disso, as formas de aprendizado e aplicações das redes neurais artificiais são também comentadas.

O capítulo 4 apresenta uma descrição de dois estudos de casos de identificação multivariável abordados nesta dissertação. Estes estudos de casos foram validados em identificação multivariável de curto prazo (previsão um passo à frente) usando dois tipos de redes neurais, a MLP e a RBF.

Por fim, no capítulo 5 são apresentadas a conclusão e as perspectivas de futura pesquisa.

2 IDENTIFICAÇÃO DE SISTEMAS

A identificação de sistemas é uma forma de construir modelos matemáticos de sistemas dinâmicos com base nos dados experimentais.

Os tipos de sistemas usados em identificação de sistemas podem ser classificados pelo conhecimento dos sistemas a serem identificados. Os grupos de sistemas são os seguintes:

- (i) sistemas caixa preta;
- (ii) sistemas caixa cinza;
- (iii) sistemas caixa branca.

Nos sistemas caixa branca existe o total conhecimento de suas entradas, saídas e do processo de funcionamento dos sistemas. Esse tipo de sistema é facilmente identificável e modelável.

No segundo sistema, o sistema caixa cinza, se possui o conhecimento das entradas e das saídas, mas o processo de funcionamento é pouco conhecido, ou seja, não se sabe exatamente todos os parâmetros que influenciam o sistema.

Quando se tem pouco conhecimento sobre os sistemas a serem identificados é utilizado o modelo conhecido como caixa preta. Um sistema caixa preta pode servir para representar tanto sistemas lineares como não-lineares, sistemas dinâmicos ou estáticos. Entretanto, é difícil uma análise algébrica das saídas obtidas pelos sistemas caixa preta.

Os sistemas caixa branca utilizam a física do processo (modelagem). Os sistemas caixa preta, em contra partida, utilizam a identificação baseada em dados.

A diferença básica entre esses dois sistemas relaciona-se à quantidade de conhecimento sobre o fenômeno. O primeiro exige um conhecimento amplo sobre o

fenômeno (caixa branca) e o segundo não pressupõe nenhum conhecimento prévio (caixa preta).

Recentemente, o estudo dos sistemas não-lineares tem sido direcionado a abordar um terceiro grupo de sistemas, os sistemas caixa cinza. Nesse grupo de modelagem é possível que o projetista utilize conhecimento prévio, mesmo que superficial do processo, porém não exige que o mesmo possua um detalhado conhecimento do comportamento estático e dinâmico do processo.

Nos casos de pouco conhecimento ou de nenhum conhecimento do sistema, é interessante utilizar a teoria de sistemas lineares para se aproximar o comportamento de um sistema real, uma vez que a identificação de sistemas não-lineares é uma tarefa mais árdua que a identificação dos sistemas lineares.

A obtenção de modelos matemáticos aceitáveis geralmente engloba a utilização de procedimentos de otimização (minimização ou maximização) e índices de desempenho que podem considerar tanto a complexidade quanto a precisão do modelo encontrado para representar um determinado sistema.

Uma possível classificação de sistemas é dividi-los em sistemas lineares e não-lineares. Os sistemas lineares caracterizam-se por representar adequadamente sistemas de baixa ordem. Os sistemas não-lineares são utilizados para a modelagem de situações complexas, como as que acontecem em vários sistemas de potência, indústria química e sistemas com comportamento caótico.

Sob este contexto, e visto a necessidade de expressão dos fenômenos complexos, tais como acoplamentos entre variáveis e não-linearidades presentes em sistemas de potência sob a forma numérica, aborda-se, neste trabalho, a identificação de sistemas não-lineares usando redes neurais artificiais.

Sob o contexto das etapas para a identificação dos sistemas, Ljung (1987) menciona que o problema de identificação de sistemas pode ser dividido nas seguintes etapas principais:

- (i) obtenção de dados de experimentação do sistema que se deseja modelar;
- (ii) aplicação de testes aos dados obtidos para detecção de não-linearidades;
- (iii) escolha da estrutura que será utilizada para representar o modelo;
- (iv) estimativa dos parâmetros do modelo matemático;
- (v) validação do modelo obtido matemático.

2.1 EXPERIMENTOS E A DETECÇÃO DE NÃO-LINEARIDADES

Segundo Aguirre *et al.* (1997), a experimentação do sistema constitui-se na observação do sinal de saída do sistema e/ou das suas variáveis de estado observáveis. Os dados de identificação assim obtidos são utilizados na detecção de não-linearidades e no ajuste dos parâmetros do modelo escolhido.

Com relação aos dados utilizados na alimentação do sistema, Leontaritis & Billings (1987) mencionam que os dados de entrada utilizados na identificação devem conter informações sobre o sistema a ser modelado. Assim, o sinal de excitação deve apresentar espectro suficientemente amplo em frequência e amplitude, de tal forma que excursions o sistema pelas regiões de operação de interesse. Estes autores expõem que, no caso de sistemas não-lineares, isto requer que os efeitos não-lineares sejam instigados por tais sinais e estejam presentes nos dados.

Conforme detalhado em Aguirre *et al.* (1997), os procedimentos de identificação com base em dados coletados são aptos em lidar com modelos dinâmicos discretos, existindo vários métodos práticos para a determinação do período de amostragem de um sinal. Sob este contexto, Aguirre (1995) apresenta um método de seleção de período de amostragem que

utiliza a autocorrelação e uma correlação não-linear para detectar o tipo de interações presentes naquele sinal. Com relação à motivação básica do método, Aguirre (1995) comenta que é simples usar este método para amostras a uma taxa superior à necessária, onde há redundância entre amostras vizinhas. Para quantificar a “redundância” pode-se utilizar funções de autocorrelação.

A novidade em Aguirre (1995) reside em mostrar que, se os dados tiverem informação não-linear presente, a função de autocorrelação linear em alguns casos é insuficiente para quantificar a “redundância” mencionada. Neste caso, tal função é inadequada para a escolha do tempo de amostragem.

Com relação aos procedimentos de identificação de não-linearidades, Billings & Voon (1983), Haber (1985) e Haber & Unbehauen (1990) expõem que os algoritmos de detecção de não-linearidades são utilizados para quantificar o nível de interações não-lineares encontradas nos dados de identificação.

Aguirre *et al.* (1997) finalizam seu estudo expondo que o sistema que gerou os dados deve ser representado por um modelo não-linear quando a correlação calculada não permanece dentro do intervalo de confiança. Nesse caso, as interações não-lineares nos dados de identificação são consideráveis e devem ser modeladas.

2.2 REPRESENTAÇÕES MATEMÁTICAS PARA SISTEMAS NÃO-LINEARES

Conforme Aguirre *et al.* (1997), uma questão importante na modelagem é a escolha da estrutura representativa do comportamento do sistema dinâmico. As redes neurais (Elsner, 1992; Masri *et al.*, 1993), funções de base radial (Casdagli, 1989), séries de Volterra (Billings, 1980), *wavelets* (Strang, 1989), funções polinomiais e racionais (Chen & Billings, 1989; Haber & Unbehauen, 1990; Foss & Johansen, 1992; Noshiro *et al.*, 1993; Jang & Kim,

1994) e equações diferenciais polinomiais (Gouesbet & Letellier, 1994) são representações utilizadas na modelagem de sistemas não-lineares.

Segundo Chen & Billings (1989), as estruturas "não-lineares auto-regressivas com média móvel e entrada exógena ou NARMAX (*Nonlinear AutoRegressive Moving Average eXogeneous inputs*) constituem uma representação adequada para uma grande variedade de sistemas não-lineares.

A forma do modelo matemático, isto é, ordem atraso de transporte e não-linearidade, não é conhecida *a priori*. Assim, a dinâmica do sistema deve ser reconstruída, utilizando-se uma aproximação para representar o sistema identificado. Segundo Chen & Billings (1989), as possíveis aproximações para esta função são os modelos polinomiais e racionais.

Aguirre & Mendes (1996) mencionam que os modelos polinomiais apresentam algumas vantagens sobre as demais representações para dinâmicas não-lineares. Geralmente, é possível obter modelos NARMAX polinomiais que ajustem dados com boa exatidão. Esses dados, porém, não podem apresentar variações abruptas. Outra vantagem da representação polinomial é a facilidade com que a informação analítica sobre a dinâmica do modelo pode ser obtida.

Davis & Vinter (1985), Korenberg *et al.* (1988) e Chen *et al.* (1989) expõem que os polinômios não-lineares são funções lineares nos parâmetros, o que permite a utilização de algoritmos de avaliação de parâmetros para modelos lineares, tal como o método dos mínimos quadrados.

A identificação de modelos NARMAX multivariáveis foi analisada em Billings *et al.* (1989), enquanto que diversas representações não-lineares para séries temporais incluindo redes neurais artificiais e modelos NARMA foram analisadas por Çinar (1995).

2.3 DETECÇÃO DE ESTRUTURA DE MODELOS NARMAX POLINOMIAIS

Segundo Korenberg *et al.* (1988), o número de termos possíveis em modelos polinomiais cresce bastante com o aumento do grau de não-linearidade e dos atrasos máximos, sendo que esse número pode ser determinado para modelos monovariáveis.

Conforme Aguirre *et al.* (1997), a união de todos os termos possíveis em um modelo polinomial é denominado conjunto de termos candidatos, o qual possui, na maior parte das vezes, muitos elementos. Entretanto, as representações polinomiais concisas podem ser obtidas para uma vasta diversidade de sistemas não-lineares, garantindo que os termos importantes no modelo possam ser corretamente encontrados. O procedimento de seleção dos termos a ser incluído em um modelo matemático não-linear é denominado *deteção de estrutura*.

Çinar (1995) comenta que a utilização de uma estrutura não compatível com os tipos de não-linearidades existentes nos dados tem um efeito significativo sobre o esforço de identificação e a qualidade do modelo gerado. Assim, o processo de deteção da estrutura de um modelo dinâmico para sistemas não-lineares deve receber uma atenção especial durante o procedimento de identificação.

Com relação aos modelos de identificação de estruturas, Thouverez & Jezequel (1996) propuseram o modelo NARMAX baseado em análise modal. Wang & Cluett (1996) utilizaram uma ferramenta estatística (os resíduos PRESS) para avaliar, nesse modelo, a qualidade da estrutura não-linear.

Segundo Korenberg *et al.* (1988), Billings *et al.* (1989) e Chen *et al.* (1989) a taxa de redução do erro ("*Error Reduction Ratio*" ou ERR) é um critério utilizado na deteção de estrutura de modelos NARMAX polinomiais. O ERR de cada termo candidato é um número que indica a melhoria obtida na representação do sistema através da sua inclusão no modelo.

2.4 ESTIMAÇÃO DE PARÂMETROS

Conforme Billings & Voon (1984), Chen *et al.* (1989) e Zhu & Billings (1996), após determinada a estrutura do modelo deve-se estimar seus parâmetros para aproximar o comportamento dinâmico apresentado pelo sistema original. Os modelos NARMAX polinomiais são estruturas lineares nos parâmetros, podendo ser estimados através do algoritmo de mínimos quadrados lineares. A estrutura mostrada na equação (2.1) pode ser representada na forma do erro de previsão:

$$\hat{y}(t) = \sum_{i=1}^{n_{\theta}} p_i(t) \hat{\theta}_i + \xi(t), \quad (2.1)$$

onde $\hat{y}(t)$ é a saída estimada $p_i(t)$ são os regressores do modelo e correspondem aos diferentes termos no polinômio, os $\hat{\theta}_i$ são as polarizações (*bias*) estimadas e o $\xi(t)$ é o resíduo de identificação, este definido como:

$$\xi(t, \theta) = y(t) - \hat{y}(t, \theta), \quad (2.2)$$

$$\hat{y}(t, \hat{\theta}) = \sum_{i=1}^{n_{\theta}} p_i(t) \hat{\theta}_i, \quad (2.3)$$

Com relação aos erros de modelagem, o preditor e os parâmetros, Aguirre *et al.* (1997) expõem:

O vetor de resíduos $\{\xi(t), t=1, \dots, N\}$ representa os erros de modelagem, o ruído aditivo do sistema e as incertezas de ordem qualquer. A equação (2.4) é denominada preditor de um passo-a-frente e $\hat{y}(t)$ é a predição de “um-passo-a-frente” de $y(t)$. Os parâmetros θ_i do modelo podem ser escolhidos, dentro de um espaço de busca, de modo a minimizar a função de custo do algoritmo de mínimos quadrados Aguirre *et al.* (1997), tal que:

$$J_N(\theta) = \frac{1}{N} \sum_{t=1}^N \xi^t \xi(t, \theta), \quad (2.4)$$

Sendo assim, o vetor de parâmetros estimados é omitido na representação dos resíduos de identificação e da saída prevista do modelo para simplificação de notação.

Conforme Davis & Vinter (1985) a estimativa obtida é dita não-polarizada se os resíduos forem brancos e não apresentarem correlação com os regressores. Quando isso não é verificado, os resíduos apresentam alguma dinâmica que não foi devidamente explicada pelo modelo. Novos termos devem ser incluídos neste modelo para que as estimativas se tornem não-polarizadas e toda a dinâmica dos dados seja absorvida pelo modelo.

O critério do ERR pode ser formalizado através da estrutura do modelo ortogonal. A variância dos resíduos de identificação $\xi(t)$ é igual ao erro quadrático médio da saída $y(t)$ quando nenhum termo é incluído no modelo ortogonal. A cada novo termo colocado no modelo ortogonal, a variância de $\xi(t)$ é decrescida de um fator $1/N(g_i^2 w_i^T w_i)$ onde w_i indica o termo incluído e g_i o seu respectivo parâmetro. Assim, o ERR de cada termo é definido por:

$$[ERR]_i = \frac{g_i^2 w_i^T w_i}{Y^T Y}, 1 \leq i \leq n_\theta. \quad (2.5)$$

Conforme Korenberg *et al.* (1988) e Billings *et al.* (1989), o ERR indica a porção da variância pela inclusão de um novo termo no modelo ortogonal. Ele é usado na detecção de estrutura de modelos não-lineares polinomiais. Escolhe-se o número de termos desejados para o modelo e consideram-se aqueles que possuem os maiores valores de ERR.

Segundo Luo *et al.* (1994), os algoritmos de estimação de parâmetros podem ser feitos em batelada (de maneira *off-line*), de estimação recursiva (*on-line*) e de estimação de parâmetros *on-line* baseado na ortogonalização. Os algoritmos *on-line* são mais complexos dos *off-line*, sendo que sua utilidade prática ainda precisa ser estabelecida.

Conforme Pröl & Karim (1994), na prática, a detecção de estrutura de modelos não-lineares é feita de maneira *off-line*, enquanto que a avaliação de parâmetros pode ser feita *on-line*.

Aguirre & Mendes (1995) expõem, resumidamente, o procedimento de detecção de estruturas não-lineares. Assim segue:

- 1º passo: formar uma equação matricial com os M possíveis termos candidatos;
- 2º passo: tomar o i -ésimo regressor original para compor o 1º regressor ortogonal;
- 3º passo: estimar por mínimos quadrados o respectivo coeficiente;
- 4º passo: determinar o ERR de cada possível regressor, candidato ao 1º regressor ortogonal;
- 5º passo: escolher para ser o 1º regressor ortogonal, aquele com maior ERR;
- 6º passo: completar o número de termos desejado no modelo, para todos os regressores que ainda não foram escolhidos;
- 7º passo: estimar por mínimos quadrados os coeficientes dos regressores ortogonais escolhidos até a presente iteração (k);
- 8º passo: determinar o próximo regressor ortogonal eliminando de um regressor original o efeito dos $k-1$ regressores ortogonais escolhidos até o presente;
- 9º passo: calcular o coeficiente do regressor ortogonal determinado no passo anterior;
- 10º passo: determinar o valor do respectivo ERR;
- 11º passo: escolher para ser o k -ésimo regressor ortogonal aquele regressor ortogonal, dentre os restantes, com maior ERR.

2.5 CRITÉRIOS DE INFORMAÇÃO

É possível mensurar a ordem de modelos de sistemas dinâmicos a partir de dados medidos. Entre tais procedimentos destacam-se os critérios de informação de Akaike ou AIC (Akaike, 1974), Bayes ou BIC (Kashyap, 1977), Schwarz *et al.* (1987), “entropia do modelo” (Mees, 1993) e critério de Rissanen (Rissanen, 1978). O trabalho de Gooijer *et al.* (1985) apresenta alguns métodos estatísticos para determinar a ordem de modelos lineares auto-regressivos.

O critério de informação de Akaike é um dos métodos mais utilizados para estimar o número de termos em modelos dinâmicos. De acordo com o critério de Akaike, o número ótimo de termos deve minimizar uma função custo. A função de custo de Akaike estabelece um compromisso entre a qualidade do ajuste aos dados de identificação.

Segundo Aguirre (1994), é importante ressaltar que o uso de critérios de informação pressupõe que o conjunto de termos candidatos esteja pré-ordenado segundo uma certa hierarquia. No caso de modelos não-lineares polinomiais, a hierarquia entre os termos possíveis é definida pelo critério do ERR.

2.6 VALIDAÇÃO DE MODELOS

Conforme Box & Jenkins (1976) e Ljung (1987), uma vertente da validação de modelos dinâmicos utiliza funções de correlação. Tal validação é utilizada para detectar possíveis dinâmicas não modeladas nos resíduos de identificação.

Nos sistemas lineares a validação verifica se os resíduos são brancos (média zero e variância σ) e não-correlacionados com a entrada. Esta verificação pode ser realizada através do cálculo das funções de autocorrelação dos resíduos e da correlação cruzada dos resíduos com a entrada.

Tratando-se de sistemas não-lineares esses testes não são suficientes. É importante verificar se os resíduos (lineares e não-lineares) são brancos com relação à entrada e à saída, bem como se são não autocorrelacionados. É necessário avaliar se existem correlações estatísticas entre os resíduos.

Segundo Billings & Voon (1986) e Billings & Tao (1991), os testes de verificação dos resíduos são:

$$\Phi_{\xi\xi}(\tau) = E\{\xi(t)\xi(t-\tau)\} = \delta(\tau), \quad (2.6)$$

$$\Phi_{\xi u}(\tau) = E\{\xi(t)u(t-\tau)\} = 0 \forall \tau, \quad (2.7)$$

$$\Phi_{\xi\xi u}(\tau) = E\{\xi(t)\xi(t-\tau)u(t-\tau)\} = 0 \forall \tau, \quad (2.8)$$

$$\Phi_{u^2\xi}(\tau) = E\{u^2(t) - E\{u^2(t)\}\xi(t-\tau)\} = 0 \forall \tau, \quad (2.9)$$

$$\Phi_{u^2\xi^2}(\tau) = E\{u^2(t) - E\{u^2(t)\}\xi^2(t-\tau)\} = 0 \forall \tau, \quad (2.10)$$

$$\Phi_{\xi\xi u}(\tau) = E\{\xi(t)\xi(t-\tau)u(t-\tau)\} = 0 \forall \tau, \quad (2.11)$$

$$\Phi_{\xi\xi u}(\tau) = E\{\xi(t)\xi(t-\tau)u(t-\tau)\} = 0 \forall \tau, \quad (2.12)$$

$$\Phi_{\xi\xi u}(\tau) = E\{\xi(t)\xi(t-\tau)u(t-\tau)\} = 0 \forall \tau, \quad (2.13)$$

$$\Phi_{\xi\xi u}(\tau) = E\{\xi(t)\xi(t-\tau)u(t-\tau)\} = 0 \forall \tau, \quad (2.14)$$

$$\Phi_{(y\xi)u^2}(\tau_1) = E\{(y(t)\xi(t) - E\{y(t)\xi(t)\})(u^2(t-\tau_1) - E\{u^2(t)\})\} = 0, \forall \tau_1 \quad (2.15)$$

onde E é a soma do erro médio quadrático da(s) saída(s), $u(t)$ é a entrada do sistema, $\xi(t)$ é o resíduo, $\delta(\tau)$ é delta de Dirac e Φ é o fator de aceleração do coeficiente de aprendizado

Nas equações (2.6) a (2.15) determina-se possíveis correlações não-lineares existentes nos resíduos. Se as correlações não forem detectadas, os resíduos são (linear e não linearmente) brancos. O procedimento descrito é denominado validação estatística de um

modelo dinâmico. Serve para garantir que não existem correlações não-modeladas nos resíduos de identificação.

Segundo Aguirre & Billings (1994,1995), um modelo estatisticamente válido pode não reproduzir uma ou mais propriedades dinâmicas do sistema original. Por esse motivo, um procedimento de validação dinâmica deve estar sempre associado com a validação estatística.

A validação dinâmica deverá verificar se o modelo identificado reproduz as principais características dinâmicas originais. Conforme Aguirre & Billings (1994), Fiedler-Ferrara & Prado (1994) e Letellier & Gouesbet (1995), algumas propriedades dinâmicas que podem ser utilizadas para validar modelos são:

- (i) expoentes de Lyapunov;
- (ii) mapas e seções de Poincaré;
- (iii) dimensão de correlação;
- (iv) diagramas de bifurcação;
- (v) características topológicas de atratores reconstruídos;
- (vi) constantes de tempo, ganhos e características estáticas conhecidas *a priori*.

2.7 SELEÇÃO DE ESTRUTURA DE MODELOS NÃO-LINEARES

Segundo Aguirre *et al.* (1997), a seleção da estrutura de modelos não-lineares polinomiais pode ser implementada utilizando o critério do ERR.

O ERR é um critério estatístico e não apresenta relações claras com aspectos dinâmicos do sistema a ser modelado. O desempenho do ERR cai com o aumento do ruído nos dados de identificação.

O conceito de agrupamentos de termos possui características que permitem relacionar a estrutura de um modelo com algumas de suas características dinâmicas. Assim,

este conceito pode ser utilizado para incrementar o desempenho do ERR na seleção da estrutura de modelos NARMAX.

Um algoritmo de seleção de estrutura que utilize a idéia de agrupamentos de termos deverá ser capaz de determinar quais são os agrupamentos que devem compor um modelo matemático para reproduzir um dado comportamento dinâmico. Após a determinação dos agrupamentos importantes, os termos do modelo poderão ser escolhidos através do critério do ERR.

Aguirre & Billings (1995) mostraram que os coeficientes dos agrupamentos de termos espúrios normalmente possuem valores reduzidos em relação aos coeficientes dos agrupamentos de termos efetivos.

Um agrupamento é dito espúrio quando os seus termos não são necessários para representar a dinâmica do sistema original. Por outro lado, um agrupamento de termos efetivo é essencial para modelar aquela dinâmica. A importância de um agrupamento de termos em um modelo NARMAX pode ser quantificada pelo seu coeficiente de agrupamento.

Conforme Jácome (1996), o coeficiente de um agrupamento espúrio é sensível ao número de termos do modelo. Existe uma situação na qual um pequeno valor de um coeficiente de agrupamento não indica que tal agrupamento é espúrio. Um modelo com termos pertencentes a agrupamentos espúrios pode apresentar regimes dinâmicos não refletidos nos dados de identificação. Portanto, os modelos identificados a partir do conjunto de agrupamentos efetivos têm melhores chances de reproduzir as propriedades dinâmicas do sistema original.

2.8 USO DE CONHECIMENTO *A PRIORI* NA SELEÇÃO DE ESTRUTURAS

Conforme Jácome (1996), a seleção da estrutura dos modelos é uma das etapas importantes do procedimento de identificação de sistemas não-lineares. Por esse motivo, seria

interessante que qualquer conhecimento prévio do sistema pudesse ser utilizado para reduzir o conjunto de termos candidato na identificação. Essa eliminação de termos aumentaria as chances de obter modelos NARMAX polinomiais concisos e dinamicamente válidos.

3 REDES NEURAIIS ARTIFICIAIS

3.1 INTRODUÇÃO À INTELIGÊNCIA ARTIFICIAL

Segundo Maia (1998), há tempos procura-se implementar em sistemas computacionais métodos que emulem o raciocínio humano, na pretensão de torná-los inteligentes o suficiente e capacitá-los a tomarem decisões em situações complexas.

Os estudos na área se intensificaram em 1956, nos Estados Unidos, quando John McCarthy reuniu em uma conferência proferida ao *Darmouth College*, na Universidade de *New Hampshire*, vários pesquisadores para estudar o que foi denominada por Minsky, McCarthy, Newell e Simon de Inteligência Artificial (IA) (expressão utilizada para designar um tipo de inteligência construída pelo homem para dotar a máquina de comportamentos inteligentes) (Maia, 1998).

Com a aceitação do paradigma baseado em conhecimento, em 1970, foram concebidos com sucesso vários protótipos de sistemas especialistas. Esses sistemas poderiam interpretar espectrogramas de massa para identificar constituintes químicos (DENDRAL), análise de dados geológicos para detecção de óleo (DIPMETER) e minerais (PROSPECTOR) e sistema de configuração de computadores (XCON/R1).

O sistema especialista MYCIN foi importante por diversas razões. Primeiro, ele demonstrou que a Inteligência Artificial poderia ser usada para a solução de problemas práticos do mundo real. Esse sistema testou novos conceitos, tais como facilidade de explanação, aquisição automática de conhecimento e tutor inteligente. Esse último, que fez de MYCIN uma experiência importante, foi a demonstração da exeqüibilidade da construção de *shell* para o desenvolvimento de sistemas especialistas (Maia, 1998).

Atualmente, os sistemas especialistas têm tomado a atenção de pesquisadores e analistas de sistemas, por propor novos sistemas capazes de facilitar algumas atividades que antes só eram realizadas pelo homem. Isso tem se mostrado, na maioria das vezes, através de redes neurais artificiais.

Fischler & Firschein (1987) sugerem três requisitos para que um sistema computacional seja considerado inteligente, quais sejam:

- (i) o sistema deve ser capaz de resolver uma ampla gama de problemas, incluindo formulações para as quais não foi projetado;
- (ii) o sistema deve ser capaz de explicitar informações implícitas, ou seja, de uma informação qualquer ele deve ser capaz de derivar todas as suas representações equivalentes;
- (iii) o sistema deve ter uma estrutura de controle que determine que transformações aplicar, quando obteve uma informação ou quando um esforço adicional será inútil.

A inteligência computacional, portanto, compreende todas as técnicas que buscam imitar a natureza, ou seja, buscam otimizar soluções, não buscar tão somente soluções globais para o problema.

No entanto, o grande desafio está em buscar a melhor solução no menor tempo e custo computacional. Para isso, existem várias técnicas computacionais tais como: algoritmos evolutivos, sistemas fuzzy e redes neurais artificiais. Todavia, foi somente na década de 80 que surgiram essas técnicas mais modernas e flexíveis. Dentre elas se destacam as redes neurais, pela sua grande potencialidade na aproximação de mapeamentos não-lineares. Essa e outras características das redes neurais serão abordados a seguir.

3.2 REDES NEURAIIS ARTIFICIAIS

As redes neurais artificiais são paradigmas computacionais que apresentam um modelo matemático inspirado, rudimentarmente, na estrutura neuronal de organismos inteligentes e que adquirem conhecimento através da experiência.

Haykin (2001) menciona que uma rede neural é uma “máquina” que é projetada para modelar, de forma simplificada, a maneira como o cérebro realiza uma tarefa particular ou função de interesse; a rede é normalmente implementada utilizando-se componentes eletrônicos ou é simulada por programação computacional.

Além disso, Haykin (2001) cita que uma rede neural é um processador maciço e paralelamente distribuído, constituída de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso.

Da mesma maneira que no cérebro, as redes neurais artificiais são organizadas na forma de um número de elementos individuais simples (os neurônios) que se interconectam uns aos outros, formando redes capazes de armazenar e transmitir informação provinda do exterior. Outra capacidade importante das redes neurais artificiais é a auto-organização ou plasticidade, ou seja, através de um processo de aprendizado é possível alterar-se os padrões de interconexão entre seus elementos. Por este motivo, as redes neurais artificiais são um tipo de sistema conexionista, no qual as propriedades computacionais são resultado dos padrões de interconexão da rede, como acontece também no sistema nervoso biológico (Tafner, 1996).

Essas capacidades e outras, que serão citadas adiante, tornam as redes neurais sistemas adequados para a utilização em detecções de falhas em sistemas elétricos, uma vez que a rede elétrica de distribuição possui uma característica de comportamento que pode ser

aprendido por um sistema de reconhecimento de padrões, como são as redes neurais. Após esse aprendizado, o sistema passa a perceber alterações de comportamento da rede elétrica.

Segundo Murakami *et al.* (2004), as redes neurais (RNs) possuem as seguintes características:

- (i) processamento paralelo e endereçamento pelo conteúdo, inspirado no funcionamento do cérebro que não possui endereço de memória nem procura uma informação sequencialmente;
- (ii) aprendizado, dotando a rede da capacidade de aprender certo conhecimento por sucessivas apresentações de padrões (experiência), sem a necessidade de explicitar os algoritmos para executar uma tarefa;
- (iii) associação, permitindo que a RN possa associar diferentes padrões;
- (iv) generalização, habilitando a RN a lidar com ruídos e distorções e responder corretamente a uma entrada nunca vista, por similaridade com outros padrões já apresentados;
- (v) abstração, dotando a rede da capacidade de abstrair a essência de um conjunto de entradas;
- (vi) tolerância a falhas, característica que permite à rede continuar a apresentar resultados aceitáveis no caso de falhas de alguns neurônios (unidades computacionais básicas das redes neurais artificiais);
- (vii) robustez, devido ao paralelismo que, mesmo com a perda de elementos processadores, não permite o funcionamento inapropriado da rede.

As RNs podem ser utilizadas na resolução de uma grande classe de problemas encontrados em Haykin (2001) e Sjöberg (1995), tais como: classificação, identificação, diagnóstico, análise de sinais e de imagens, otimização e controle.

As RNs são, muitas vezes, robustas no tratamento de dados com ruídos e eficientes, especialmente nos problemas em que a não formulação analítica não está disponível, um

conhecimento explícito não está acessível, os dados “estão contaminados” por ruído, ou o próprio problema modifica-se com o tempo.

O emprego de uma RN depende da habilidade em adaptá-la ao problema considerado, por meio de mudanças nos pesos sinápticos (aprendizagem), de modo a aumentar sua eficiência.

3.3 ARQUITETURA DE REDES NEURAIAS

Segundo Menezes (2004), a forma de organização dos neurônios em uma rede neural está intrinsecamente ligada ao problema que se quer solucionar, e é fator importante para a definição dos algoritmos de aprendizagem a serem utilizados.

Para melhor compreensão de aspectos relativos a sistemas não-lineares que são fundamentais neste trabalho, primeiramente é preciso compreender os fundamentos das RNs.

3.3.1 *Perceptron* de camada única

Segundo Haykin (2001), o *perceptron* é a forma mais simples de uma rede neural usada para classificação de padrões ditos linearmente separáveis, como, por exemplo, os padrões que se encontram em lados opostos de um hiperplano.

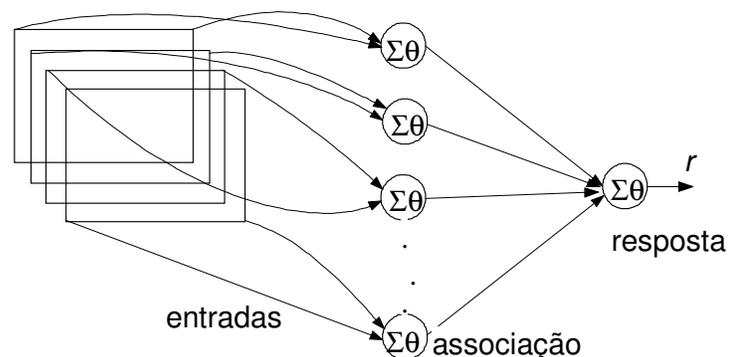
O *perceptron* consiste de um único neurônio com pesos sinápticos ajustáveis e *bias*. O algoritmo usado para ajustar os parâmetros livres dessa rede neural apareceu primeiro em um procedimento de aprendizagem por Rosenblatt (1958), em que desenvolveu seu modelo cerebral de *perceptron*.

Rosenblatt (1958) provou que se os padrões (vetores) usados para treinar o *perceptron* são retirados de duas classes linearmente separáveis; então, o algoritmo do *perceptron* converge e posiciona a superfície de decisão na forma de um hiperplano entre as duas classes.

O *perceptron* construído em torno de um único neurônio é limitado a realizar a classificação de padrões com apenas duas classes (hipóteses). Expandindo a camada de (computação) saída do *perceptron* para incluir mais de um neurônio, pode-se correspondentemente realizar a classificação com mais de duas classes. Entretanto, as classes devem ser linearmente separáveis para que o *perceptron* funcione adequadamente.

Segundo Fabri (2004), o modelo proposto por Rosenblatt (1958), conhecido como *perceptron*, é composto por uma estrutura de rede tendo como unidades básicas nodos MLP e uma regra de aprendizado. Rosenblatt (1958) demonstrou o teorema de convergência do *perceptron*, o qual mostra que um nodo MLP treinado com o algoritmo de aprendizado do *perceptron* sempre converge, caso o problema em questão seja linearmente separável (Sycara, 1988).

A topologia original do *perceptron* era composta por unidades de entrada, por um nível intermediário formado pelas unidades de associação e por um nível de saída formado pelas unidades de resposta. Embora esta topologia original possua três níveis, ela é conhecida como *perceptron* de uma única camada, já que somente o nível de saída (unidade de resposta) possui propriedades adaptativas. A seguir, tem-se um esboço da topologia do *perceptron*, conforme apresentado na figura 3.1.

FIGURA 3.1 – TOPOLOGIA DO *PERCEPTRON* SIMPLES.

3.3.2 *Perceptron* multicamadas

Segundo Haykin (2001), com uma estrutura bem definida de nós e camadas de entrada e saída, a RN consiste de um conjunto de unidades sensoriais (nós de fonte) que constituem a camada de entrada, uma ou mais camadas ocultas de nós computacionais intermediárias e uma camada de saída de nós computacionais. O sinal de entrada se propaga para frente através da rede, camada por camada. Estas redes neurais são normalmente chamadas de *perceptron* de múltiplas camadas, as quais representam uma generalização do *perceptron* de camada única.

A solução de problemas não linearmente separáveis passa pelo uso de redes com uma ou mais camadas intermediárias, ou escondidas. Segundo Cybenko (1989), uma rede com uma camada intermediária pode implementar qualquer função contínua. A utilização de duas camadas intermediárias permite a aproximação de qualquer função.

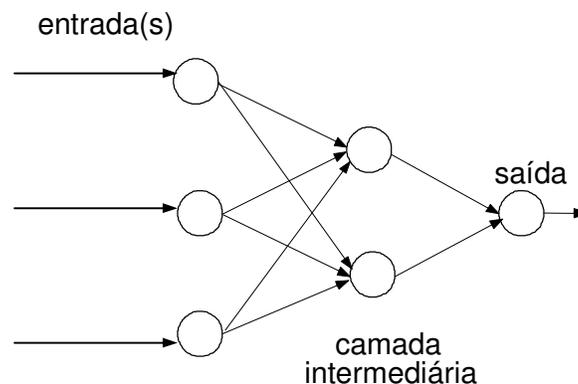
Com a utilização desta arquitetura seria necessária a divisão da rede em conjunto de sub-rede, com treinamento independente. Uma RN com uma camada intermediária pode ser formada por um conjunto de redes *perceptron*, uma rede para cada grupo de entradas

linearmente separáveis. Na camada de saída, uma outra rede combina as saídas produzidas pelas redes da primeira camada, gerando, assim, a classificação final (Castro & Castro, 2000).

3.3.3 Arquitetura do MLP

As redes MLP apresentam um poder computacional maior do que aquele apresentado pelas redes sem camadas intermediárias. As RNs do tipo MLP podem tratar com dados que não são linearmente separáveis. A precisão obtida e a implementação de uma rede neural MLP dependem do número de nodos utilizados nas camadas intermediárias, e um dos problemas é a definição de uma função de ativação. Na figura 3.2 é apresentada uma rede MLP típica.

FIGURA 3.2 - REDE MLP COM CAMADA INTERMEDIÁRIA.



Segundo Castro & Castro (2000), nota-se que duas características desta estrutura podem ser citadas:

- (i) rede progressiva (*feedforward*): o sinal se propaga camada a camada em sentido progressivo, conectando-se as saídas dos neurônios de uma camada às entradas dos neurônios da camada seguinte, sem laços de realimentação;

(ii) com relação à conexão: a rede pode ser completamente conectada, onde cada nó de uma camada é conectado aos nós da camada adjacente, ou parcialmente conectado, onde algumas sinapses podem estar faltando, o que ocorre com as redes localmente conectadas.

3.3.4 Definição de uma topologia ideal para o MLP

A definição de uma topologia de uma RN do tipo MLP está intimamente ligada com a definição dos nodos (nós) de camada intermediária; este número é, geralmente, definido empiricamente e depende, fortemente, da distribuição dos padrões de treinamento e validação da rede.

Segundo Castro & Castro (2000), o projeto de uma rede MLP requer a consideração dos aspectos de:

- (i) a determinação do número de camadas escondidas;
- (ii) a determinação do número de neurônios em cada uma das camadas escondidas;
- (iii) a especificação dos pesos sinápticos que interconectam os neurônios nas diferentes camadas da rede.

Porém, existem problemas que necessitam apenas de uma unidade de entrada e uma unidade de saída, e outros que podem precisar de muitas unidades intermediárias. O número de unidades intermediárias pode, também, em alguns casos, crescer exponencialmente com o número de entradas. A solução neural mais eficiente é aquele onde o número de unidades cresce apenas polinomialmente com o aumento do número de unidades de entrada.

Para solução de problemas práticos de reconhecimento de padrões, aloca-se para a RN um número de unidades intermediárias suficiente para a solução do problema. Deve-se ter cuidado para:

- (i) não utilizar unidades demais, o que pode levar a rede a memorizar padrões de treinamento, ao invés de extrair as características gerais que permitem a generalização ou

reconhecimento de padrões não vistos durante o treinamento (este problema é denominado de *overfitting*);

(ii) não utilizar um número muito pequeno, que possa forçar a rede a gastar tempo em excesso tentando encontrar uma representação ótima (as unidades utilizadas podem ficar sobrecarregadas, tendo que lidar com um elevado número de restrições).

3.3.5 Treinamento de RNs do tipo MLP

Existem, atualmente, vários algoritmos para treinar rede MLP, sendo eles na sua maioria do tipo supervisionado. De acordo com os parâmetros que eles atualizam, os algoritmos para treinamento de redes MLP podem ser classificados em estáticos ou dinâmicos.

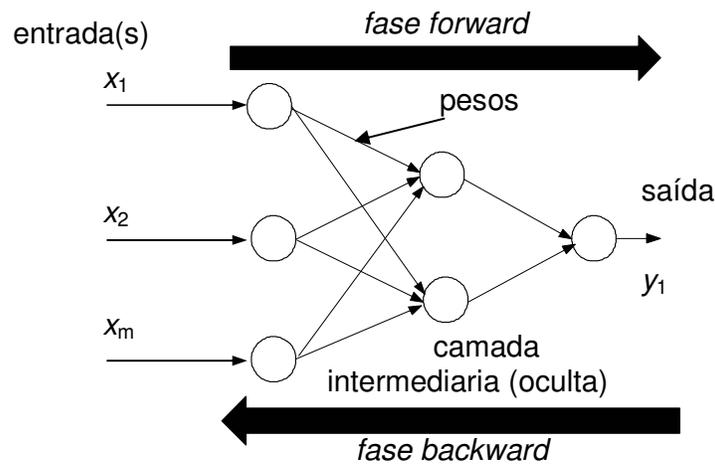
Os algoritmos estáticos não alteram a estrutura da rede, variando apenas os valores de seus pesos, já os algoritmos dinâmicos podem tanto reduzir quanto aumentar o tamanho da rede (número de camadas, número de nodos nas camadas intermediárias e número de conexões). Quando se utiliza o aprendizado estático, a mesma regra de aprendizado é empregada para redes do tipo MLP com diferentes tamanhos e formatos.

O algoritmo de aprendizado mais conhecido para treinamento destas redes é o *back-propagation*. A maioria dos métodos de aprendizado para RNAs do tipo MLP utiliza variações deste algoritmo.

O *back-propagation* foi um dos principais responsáveis pelo ressurgimento de interesse em RNAs, por ocasião da publicação do livro *Parallel Distributed Processing*, mais conhecido como PDP, em 1986 (Rumelhart & McClelland, 1986). Embora a popularização deste algoritmo tenha surgido a partir de 1986, ele foi proposto muito antes, com diferentes propósitos e por diferentes pesquisadores, tais como Bryson & Ho (1969), Werbos (1974), Parker (1985) e Le Cun (1985).

O *back-propagation* é um algoritmo supervisionado, que utiliza pares (entrada, saída) para, através de um mecanismo de correção de erros, ajustar os pesos da rede. O treinamento ocorre em duas fases, onde cada fase percorre a rede em um sentido. Estas duas fases, denominadas de fase *forward* e *backward*, são ilustradas na figura 3.3.

FIGURA 3.3 - FLUXO DE PROCESSAMENTO DO ALGORITMO RETROPROPAGAÇÃO DO ERRO.



A fase *forward* envolve os seguintes passos:

- (i) a entrada é apresentada à primeira camada da rede, a camada C^0 .
- (ii) para cada camada C^i a partir da camada de entrada. Após os nodos da camada C^i ($i > 0$), calcular os sinais de saída, estes servem como entrada para a definição das saídas produzidas pelos nodos da camada C^{i+1} .
- (iii) as saídas produzidas pelos nodos da última camada são comparadas às saídas desejadas.

A fase *backward* envolve as etapas listadas a seguir:

- (i) a partir da última camada, até chegar na camada de entrada;

(ii) os nodos da camada atual ajustam seus pesos de forma a reduzir seus erros. O erro de um nodo da camada intermediária é calculado, utilizando os erros dos nodos da camada seguinte conectados a ele, ponderado pelos pesos das conexões entre eles.

O algoritmo da retropropagação do erro (*back-propagation*), que faz uso destas duas fases, é apresentado a seguir:

- (i) inicia os pesos e parâmetros da rede neural;
- (ii) repete até o erro ser mínimo ou a realização de um dado número de ciclos:
 - (a) para cada padrão de treinamento X :

definir saída da rede através de fase *forward*;

comparar saídas produzidas com as saídas desejadas;

atualizar pesos dos nodos através da fase *backward*.

O *back-propagation* é baseado na regra proposta por Widrow & Hoff (1988), sendo também denominado de regra delta generalizada.

A derivação da regra delta generalizada é simples e semelhante à derivação da regra delta. Também neste caso, a função a ser minimizada é uma função erro, definida pela soma dos erros quadráticos e representada por:

$$E = \frac{1}{2} \sum_p \sum_{i=1}^k (d_i^p - y_i^p)^2 \quad (3.1)$$

onde

E é a medida de erro total;

p é o número de padrões;

k é o número de unidades de saída;

d_i é a i -ésima saída desejada;

y_i é a i -ésima saída gerada pela RN.

A equação (3.1) define o erro total cometido pela rede, ou a quantidade que, para todos os padrões p de um dado conjunto, as saídas geradas pela rede diferem das saídas desejadas.

A regra delta generalizada requer que as funções de ativação utilizadas pelos nós sejam contínuas, diferenciáveis, geralmente, não decrescentes da entrada total recebida pelo nó. Estas funções são denominadas de funções semi-lineares. A equação (3.2) ilustra o cálculo do valor de ativação.

$$y_j^p = f_j(\text{rede}_j^p) \quad (3.2)$$

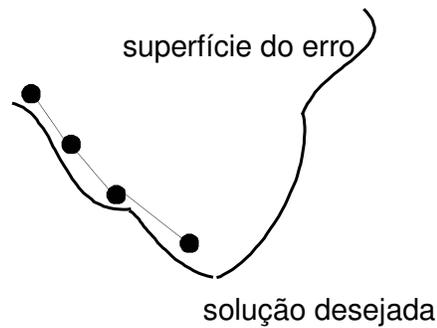
onde

$$\text{rede}_j^p = \sum_{i=1}^n x_i^p w_{ji} \quad (3.3)$$

A constante n representa o número de conexões de entrada do nó j e w_{ij} é o peso da conexão entre a entrada x_i^p e o nó j .

Há garantia de convergência se a superfície do erro for simples. Uma forma de visualizar a influência do treinamento da rede no valor do erro está ilustrada pela superfície de erro na figura 3.4. Esta figura ilustra um caso simples de uma rede com apenas um nó e duas conexões. Os valores dos pesos da RN definem a coordenada de um ponto da superfície do erro. O erro produzido pela RN, para cada combinação de valores de pesos, é determinado pela altura da superfície naquele ponto.

FIGURA 3.4 - SUPERFÍCIE DO ERRO DE UMA REDE MLP.



As equações (3.1) a (3.3) mostram como obter o erro global da rede e como encontrar o erro local de um determinado padrão de entrada. Após processadas as duas equações anteriores, deve-se aplicar a equação seguinte que ajusta os pesos das diversas camadas:

$$\Delta w_{ji} = \eta \delta_j x_i \quad \text{ou} \quad (3.4)$$

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j(t) x_i(t) \quad (3.5)$$

onde

$$\delta_j = (d_j - y_j) \text{ para camada de saída e}$$

$$\delta_j = \sum w_{ij} \delta_i \text{ para camadas intermediárias, sendo } i \text{ a unidade em questão e } j \text{ a camada}$$

em questão.

Um dos problemas enfrentados no treinamento de rede MLP diz respeito à definição de seus parâmetros. A seleção dos parâmetros de treinamento do algoritmo da retropropagação do erro é um processo pouco conhecido. Pequenas diferenças nestes parâmetros podem levar à divergência tanto no tempo de treinamento como na generalização obtida.

Porém pergunta-se: “Em que momento parar de treinar a rede?” Esta é uma dúvida que surge naturalmente. Existem vários métodos para a determinação do momento que o treinamento deva ser encerrado. Estes métodos são chamados de *critérios de parada*.

Os critérios de paradas mais utilizados são:

- (i) encerrar o treinamento após n ciclos de treinamento;
- (ii) encerrar o treinamento após o erro quadrático médio ficar abaixo de uma constante c ;
- (iii) encerrar o treinamento quando a percentagem de classificações corretas estiverem acima de uma constante c (mais indicados para saídas binárias).

Outro aspecto que precisa ser observado é a frequência das atualizações dos pesos. A frequência de ajustes dos pesos influencia o desempenho obtido durante o treinamento. Duas abordagens diferentes têm sido utilizadas quanto à frequência (periodicidade) para o ajuste de pesos pelo algoritmo *back-propagation*:

- (i) por padrão (*on-line*);
- (ii) por ciclo (*batch*).

Existem vantagens e desvantagens em cada uma destas abordagens. Na abordagem por padrão, os pesos são atualizados após a apresentação de cada padrão de treinamento. Esta abordagem é estável se a taxa de aprendizado for pequena (é aconselhável reduzir progressivamente esta taxa). Quando taxas elevadas são utilizadas, o treinamento da RN pode se tornar instável. A abordagem por padrão é mais rápida, principalmente, se o conjunto de treinamento for grande e redundante. Outra vantagem desta técnica é que ela requer menos memória.

Na abordagem por ciclo, por sua vez, os pesos são atualizados após todos os padrões terem sido apresentados. Esta técnica é mais estável, mas ela pode ser lenta, se o conjunto de treinamento for grande e redundante. Esta abordagem apresenta uma estimativa mais precisa

do vetor gradiente, ao custo da necessidade de mais memória. A escolha da abordagem a ser utilizada depende da aplicação e da distribuição estatística dos dados.

O principal problema diz respeito à lentidão do algoritmo para superfícies complexas. Uma forma de minimizar este problema é considerar efeitos de segunda ordem (taxa de momento) para a máxima descida (*descent gradient*). Não é raro o algoritmo convergir para mínimos locais.

Os mínimos locais são pontos na superfície de erro que apresentam uma solução estável, embora não seja correta. Algumas técnicas são utilizadas tanto para acelerar o algoritmo da retropropagação do erro como para reduzir a incidência de mínimos locais:

- (i) utilizar taxa de aprendizado decrescente;
- (ii) adicionar nós intermediários;
- (iii) utilizar um termo de momento.

Entre as várias técnicas utilizadas para acelerar o processo de treinamento e evitar mínimos locais, a adição de um termo de momento é uma das mais frequentes. O termo de momento é representado por:

$$w_{ij}(t+1) = w_{ij}(t) + \eta \delta_j(t) x_i(t) + \alpha [w_{ij}(t) - w_{ij}(t-1)] \quad (3.6)$$

onde α é taxa de *momentum* a ser utilizada no processamento do aprendizado da RN.

A inclusão da taxa de momento na fórmula de ajuste dos pesos aumenta a velocidade de aprendizado (aceleração), reduzindo a incidência de instabilidade. A taxa de momento pode acelerar o treinamento em regiões muito planas para a superfície de erro. Além disso, o termo de momento suprime a oscilação de pesos em valores de ravinas.

3.3.6 Rede neural função de base radial

Esta rede neural artificial é inspirada em neurônios que têm ativações localmente sintonizadas ou neurônios seletivos, que respondem para determinadas faixas de sinais de entrada. Em princípio, as redes RBF podem ter multicamadas e terem funções de ativação na saída não-lineares. Contudo, as redes RBF têm tradicionalmente sido associadas a funções radiais em uma única camada escondida e a funções de saída lineares.

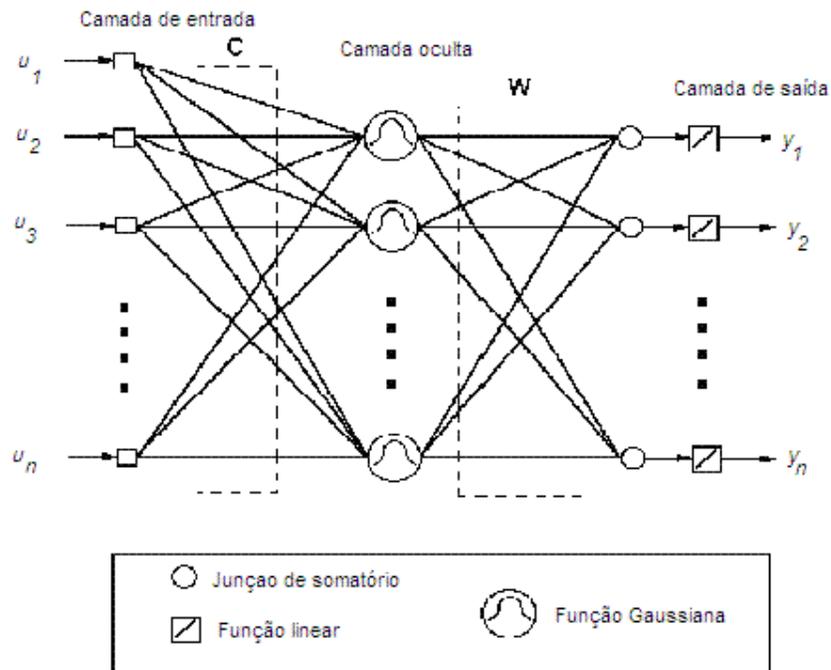
Segundo Castro & Castro (2000), as redes neurais RBF são redes supervisionadas, consideradas funções aproximadores universais, assim como as RNs do tipo MLP.

Todesco (1995) afirma que a construção de uma RBF em sua forma mais básica, com três camadas, seria como a demonstrada na figura 3.5, cujos nodos de saída formam uma combinação linear das funções de base radial calculados pelos nodos da camada escondida.

As funções de base radial na camada escondida produzem uma resposta localizada para o estímulo (padrão) de entrada, isto é, eles produzem uma resposta significativamente diferente de zero somente quando o padrão de entrada está dentro de uma região pequena localizada no espaço de entradas. Por esta razão, esta categoria de rede algumas vezes é referenciada na literatura como redes de campos receptivos localizados.

A entrada é feita dos nodos fontes (unidades sensoriais). Cada função de ativação requer um "centro" e um parâmetro escalar. Uma função que pode ser utilizada como ativação é a função Gaussiniana, sendo que esta rede pode ser usada para tomar decisões, sendo que, dos vários centros, é mais similar com o vetor de entrada.

FIGURA 3.5 – ESTRUTURA BÁSICA DA FUNÇÃO BASE RADIAL



Os campos receptivos são centros que fazem a saída do nodo j da função-radial, tal que:

$$y_j = f_k(\|x - m_j\|/\sigma) \quad (3.7)$$

onde f_k é uma função-radial, $\|\cdot\|$ pode ser a norma Euclidiana, m_j é o vetor representando o centro do nodo j e σ é o parâmetro livre que determina a largura (raio) da função radial.

A equação f_k pode ser uma função Gaussiniana, ou mesmo uma função *sino*, que tem o valor alto quando a entrada está próxima do centro do nodo. À medida que a entrada se distancia do centro de um nodo, seu valor de saída decresce monotonicamente.

O raio de atuação de cada campo receptivo está intimamente ligado ao parâmetro σ que determina a região de influência de cada nodo e a quantia de alinhamento. As decisões de classificação são baseadas nos nodos com maior saída, que são formados pelas somas ponderadas das saídas dos nodos da função-radial.

Dado um vetor de entrada, a saída de um nodo simples fica:

$$y = f(x - c) \quad (3.8)$$

Onde, por exemplo, a função pode ser tomada como:

$$f = (x - c) = \frac{1}{(2\pi)^{n/2} \sigma_1 \sigma_2 \dots \sigma_n} \exp \left\{ -\frac{1}{2} \sum_{j=1}^n \left(\frac{x_j - c_j}{\sigma_j} \right)^2 \right\} \quad (3.9)$$

Os valores $\sigma_1 \sigma_2 \dots \sigma_n$, $j=[1, n]$, são usados da mesma maneira que na distribuição de probabilidade normal para determinar a dispersão escalar em cada direção.

É interessante comentar algumas diferenças importantes entre as redes neurais MLP e RBF:

- (i) Uma função de ativação comumente aplicada às unidades escondidas das redes neurais MLP é a função sigmoideal, que não é linear e continuamente diferenciável. Duas formas da função sigmoideal utilizadas são: a função logística e a função tangente hiperbólica. Em contraste, em uma rede neural RBF, a ativação de uma unidade escondida é determinada por uma função não-linear da distância entre o vetor de entrada e um vetor de referência. As funções localizadas são as quais apresentam base radial definida sobre seu domínio.
- (ii) Uma rede MLP forma uma representação distribuída no espaço de valores de ativação para as unidades escondidas, já que, para um dado vetor de entrada, muitas unidades escondidas contribuem para a determinação do valor de saída, razão pela qual as redes neurais MLP tendem a resultar em aproximações globais. Em contraste, as redes neurais RBF, com funções de base localizadas, formam uma representação no espaço de unidades escondidas, as quais são locais e devem respeitar o espaço de entrada porque, para um dado vetor de entrada, tipicamente apenas algumas unidades escondidas apresentarão ativações significantes. Por esta razão, as redes neurais RBF tendem a produzir aproximações locais.

(iii) Uma rede neural MLP pode possuir duas ou mais camadas de pesos e um complexo padrão de conectividade. Uma rede neural RBF, no entanto, geralmente tem uma arquitetura simples, consistindo de duas camadas de pesos.

(iv) Todos os parâmetros em uma rede neural MLP são usualmente determinados ao mesmo tempo, como parte de uma única estratégia global de treinamento, envolvendo treinamento supervisionado. Uma rede neural RBF é tipicamente treinada em dois estágios, com as funções de base radial sendo determinadas primeiramente por meio de técnicas não-supervisionadas.

No contexto de aproximação de funções, de uma forma geral, pode-se afirmar que:

- (i) o erro final atingido por uma rede neural RBF é menor do que o erro final atingido por uma rede neural MLP;
- (ii) a convergência de uma rede neural RBF pode chegar a ser uma ordem de grandeza mais rápida do que a convergência de uma rede neural MLP;
- (iii) a capacidade de generalização de uma rede neural MLP é, em geral, superior à capacidade de generalização de uma rede neural RBF.

As redes neurais RBF são ferramentas flexíveis em um ambiente dinâmico. Elas têm a capacidade de aprender rapidamente padrões complexos e tendências presentes nos dados e de se adaptar rapidamente às mudanças. Estas características as tornam especialmente adequadas para predição de séries temporais, especialmente aquelas séries regidas por processos não lineares e/ou estacionários.

Os dados representados através de redes neurais RBF são, portanto, expandidos com referência a um conjunto finito de funções de ativação neurais, denominadas de funções de base radial.

3.3.7 Método k -médias

O algoritmo de agrupamento k -médias (*k-means clustering algorithm*) envolve um processamento simples de estimação dos parâmetros. Suponha que exista N amostras (dados) x^n no total e deseja-se encontrar K vetores c_j onde $j=1, \dots, K$. O algoritmo procura particionar às amostras x^n em K subconjuntos S_j contendo N_j amostras, de tal maneira a minimizar a função da soma dos quadrados dada por:

$$J = \sum_{j=1}^k \sum_{n \in S_j} \|x^n - c_j\|^2 \quad (3.10)$$

onde c_j é o centro das amostras do conjunto S_j e é dado por:

$$c_j = \frac{1}{N_j} \sum_{n \in S_j} x^n \quad (3.11)$$

3.3.8 RNs para identificação de sistemas não-lineares

A identificação de processos é uma área relevante em muitos campos da engenharia como é o caso de sistemas de potência sendo esta identificação a modelagem matemática de um processo desconhecido, para propósitos de previsão e/ou compreensão do comportamento do processo. A complexidade inerente a muitos processos reais (não-lineares, variantes no tempo e multivariáveis) dificulta a aplicação de técnicas convencionais de identificação. Para isso, são desenvolvidas técnicas de identificação avançadas, muitas delas baseadas em inteligência computacional.

Segundo Coelho (2000), a RN-MLP é muito utilizada para os propósitos de identificação e controle de processos. O modelo matemático do processo, obtido através da RN-MLP, tem representação estrutural, ou seja, modelo caixa-preta. Esta característica é proibitiva quanto à análise das propriedades do modelo aprendidas.

A capacidade de identificação não-linear das RNs pode ser explorada para aprimorar as metodologias de controle preditivo, baseadas em modelo, pois um modelo preciso do processo é parte essencial na aplicação eficiente desta metodologia.

Em identificação é importante o papel das RNs em modelos *black-box* (identificação estrutural) e *grey-box* de sistemas dinâmicos não-lineares. As RNs podem ser aplicadas, com diferentes regressores, em modelos dinâmicos *black-box* não-lineares, tais como (Coelho, 2000):

- (i) RN-NFIR (*Non-linear Finite Impulse Response model*): usa somente medidas passadas do vetor $u(t-i)$ como regressores;
- (ii) RN-NARX (*Non-linear AutoRegressive model structure eXogenous inputs*): emprega o vetor das entradas $u(t-i)$ e das saídas $y(t-j)$ passadas como regressores;
- (iii) RN-NOE (*Non-linear Output Error model*): utiliza o vetor das entradas passadas $u(t-i)$ e das saídas previstas passadas $\hat{y}_u(t-j|\theta)$, como entradas da RN. Este modelo é denominado modelo paralelo;
- (iv) RN-NARMAX (*Non-linear AutoRegressive Moving Average model structure with eXogenous inputs*): utiliza os vetores $e(t-i)$, $u(t-i)$ e $y(t-j)$;
- (v) RN-BJ (*Non-linear Box-Jenkins model structure*): utiliza os quatro tipos de regressores mencionados nos itens anteriores;
- (vi) modelos não-lineares, representados no espaço de estados: utiliza os componentes passados “virtualmente” das saídas.

A identificação de um processo não linear, utilizando-se RNs, pode ser dividido em problemas básicos:

- (i) a seleção do sinal de treinamento;
- (ii) o cálculo dos parâmetros; e
- (iii) a validação e a seleção da configuração apropriada.

4 ESTUDO DE CASOS E ANÁLISE DE RESULTADOS

Para se comparar as redes neurais MLP com as RBF foram utilizados dois estudos de casos. Os dados utilizados foram obtidos da base de dados Daisy (2005) (*Database for the Identification of Systems*).

Para a comparação das redes neurais MLP e RBF foi obtida a raiz do erro quadrado médio (RMSE) representada pela equação 4.1.

$$RMSE_i = \sqrt{\frac{\sum (y_i(t) - \hat{y}_i(t))^2}{n}} \quad 4.1$$

onde n é número de amostras, $y_i(t)$ é saída real do sistema, $\hat{y}_i(t)$ é saída estimada pela rede neural, $i=1, \dots, n$ saídas.

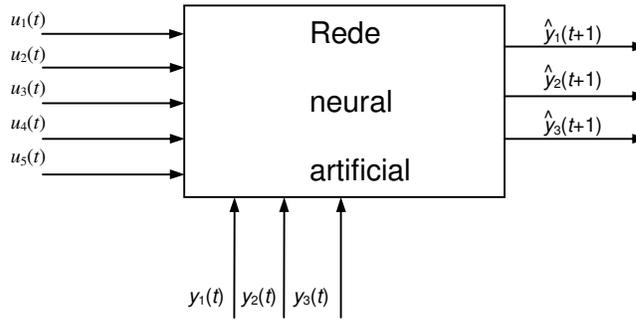
Além disso, foram simulados dois casos para as duas redes neurais em diferentes configurações de número de neurônios na camada intermediária (oculta). No caso da MLP foi simulada a rede neural de 3 a 8 neurônios nas camadas ocultas, sempre para um critério de parada 100 épocas de treinamento, sendo o treinamento realizado pelo método Levenberg-Merquardt. Já no caso da RBF foi simulada de 3 a 8 centros, usando função de ativação Gaussiana, com abertura igual a 1 para todas as Gaussianas e o treinamento utilizado foi pelo método k -médias (ajuste dos centros das Gaussianas) e pseudo-inversa (determinação dos pesos da camada de saída). Em ambas, foram utilizadas 50% das amostras no treinamento e as outras 50% na validação do procedimento de identificação não-linear.

4.1 CASO 1: SISTEMA DE GERAÇÃO DE PONT-SUR-SAMBRE

No primeiro caso foi utilizado dado de um sistema de geração de energia elétrica de 120 MW, que fica em Pont-sur-Sambre, na França. Esses dados são compostos por 200

amostras com 5 entradas e 3 saídas encontrados em Daisy (2005). Sendo que metade das amostra foram utilizadas para estimação e a outra metade para validação.

FIGURA 4.1 – MODELO DA ENTRADAS E SAÍDAS DO SISTEMA PONT-SUR-SAMBRE.



As entradas são:

- (i) vazão do gás – $u_1(t)$;
- (ii) abertura da válvula da turbina – $u_2(t)$;
- (iii) vazão do pulverizador do calefator – $u_3(t)$;
- (iv) amortizadores do gás – $u_4(t)$;
- (v) vazão do ar – $u_5(t)$.

As saídas (reais) são:

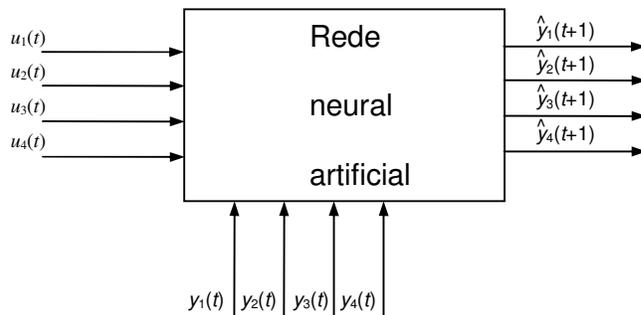
- (i) pressão do vapor – $y_1(t)$;
- (ii) temperatura principal da haste – $y_2(t)$;
- (iii) temperatura do vapor de reaquecimento – $y_3(t)$.

4.2 CASO 2: SISTEMA DE GERAÇÃO DE ABBOTT

Neste segundo caso, os dados são de um modelo de gerador a vapor de um sistema de geração de energia elétrica de Abbott, que fica na cidade de Champaign/IL, na França.

Essas amostras são divididas em 4 entradas e 4 saídas com 3650 amostras, também encontradas em Daisy (2005). Neste caso também foram utilizadas metade das amostras para estimação e a outra metade para validação.

FIGURA 4.2 - MODELO DA ENTRADAS E SAÍDAS DO SISTEMA ABBOTT.



As entradas para este caso são:

- (i) combustível – $u_1(t)$;
- (ii) ar – $u_2(t)$;
- (iii) nível – $u_3(t)$;
- (iv) distúrbio pelo nível da carga – $u_4(t)$.

As saídas do sistema são:

- (i) pressão do cilindro – $y_1(t)$;
- (ii) excesso de oxigênio em gases de exaustão – $y_2(t)$;
- (iii) nível de água no cilindro – $y_3(t)$;
- (iv) fluxo do vapor – $y_4(t)$.

4.3 RESULTADOS DE SIMULAÇÃO

No caso Point-sur-Sambre pode-se observar que a configuração da rede MLP que obteve um menor erro foi a rede neural com 5 neurônios na camada oculta, conforme apresentado na tabela 4.1.

TABELA 4.1 – RMSE PARA REDE MLP NO CASO DE POINT-SUR-SAMBRE.

RN	neurônios na camada oculta	RMSE ₁	RMSE ₂	RMSE ₃
MLP	3	6,0391	1,9913	2,5939
MLP	4	2,2153	0,7491	1,1246
MLP	5	0,9238	1,0709	0,7302
MLP	6	11,0259	2,2494	1,5664
MLP	7	1,6591	0,4641	0,9997
MLP	8	47,6845	10,3218	1,7184

Na tabela 4.2 tem-se a rede neural com três centros como sendo a configuração da RBF que obteve menor erro entre as rede RBF e MLP para esse caso.

TABELA 4.2 – RMSE PARA REDE RBF NO CASO DE POINT-SUR-SAMBRE.

RN	número de centros	RMSE ₁	RMSE ₂	RMSE ₃
RBF	3	$3,2774 \times 10^{-10}$	$3,5587 \times 10^{-10}$	$1,6950 \times 10^{-10}$
RBF	4	$1,0766 \times 10^{-9}$	$6,1532 \times 10^{-10}$	$1,0664 \times 10^{-9}$
RBF	5	$9,7903 \times 10^{-8}$	$4,0952 \times 10^{-7}$	$8,1122 \times 10^{-8}$
RBF	6	$5,3493 \times 10^{-7}$	$1,8765 \times 10^{-7}$	$8,9710 \times 10^{-8}$
RBF	7	$3,9654 \times 10^{-6}$	$8,0778 \times 10^{-7}$	$5,7450 \times 10^{-6}$
RBF	8	$1,2396 \times 10^{-7}$	$1,6446 \times 10^{-7}$	$4,0129 \times 10^{-7}$

Na comparação das duas redes neurais, nota-se que a rede RBF obteve erros menores que os da rede MLP neste caso.

No caso Abbott pode-se observar que a configuração da rede MLP que obteve menor erro foi a RN com 8 neurônios na camada oculta, conforme apresentado na tabela 4.3.

TABELA 4.3 – RMSE PARA REDE MLP NO CASO DE ABBOTT.

RN	neurônios na camada oculta	RMSE ₁	RMSE ₂	RMSE ₃	RMSE ₄
MLP	3	0,1496	0,1525	0,2383	0,1245
MLP	4	0,1302	0,1166	0,0669	0,0323
MLP	5	0,1320	0,1179	0,0916	0,0611
MLP	6	0,1349	0,0981	0,0393	0,0343
MLP	7	0,1358	0,0956	0,0361	0,0325
MLP	8	0,1403	0,0940	0,0286	0,0294

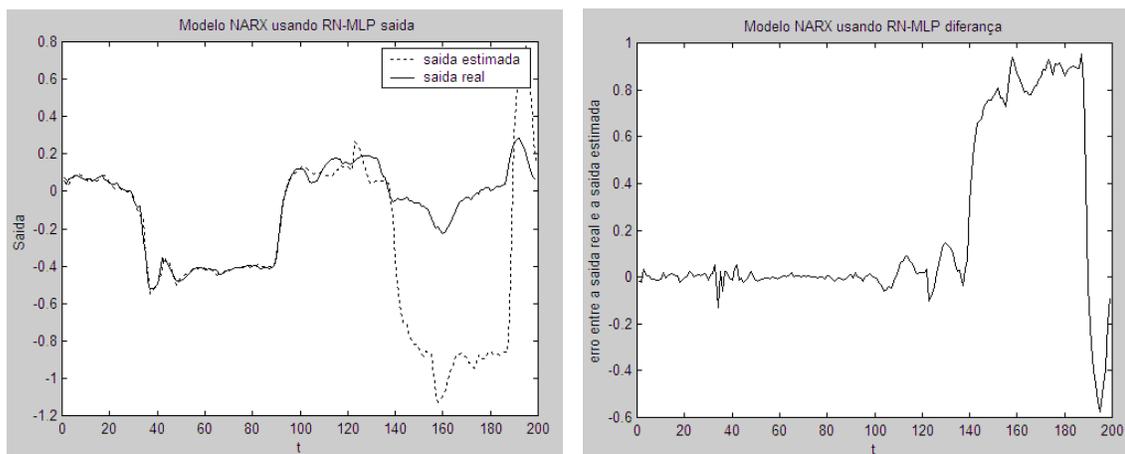
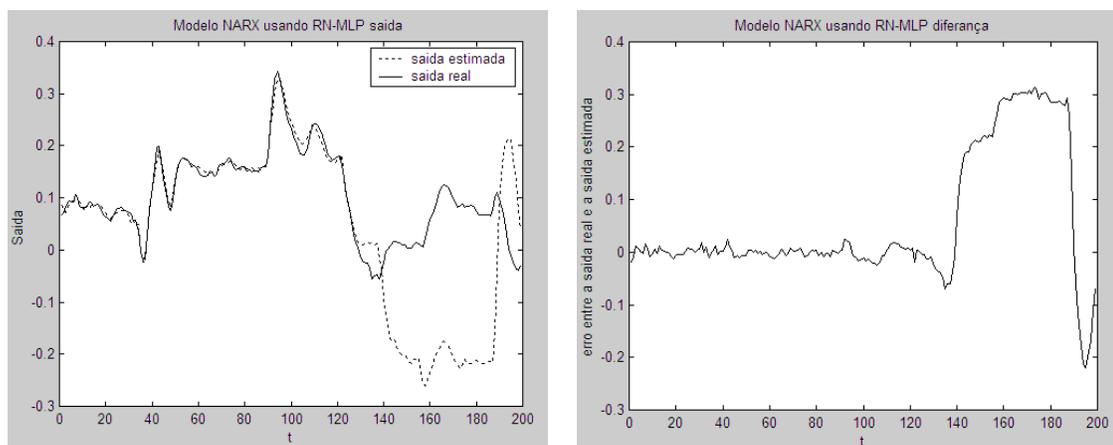
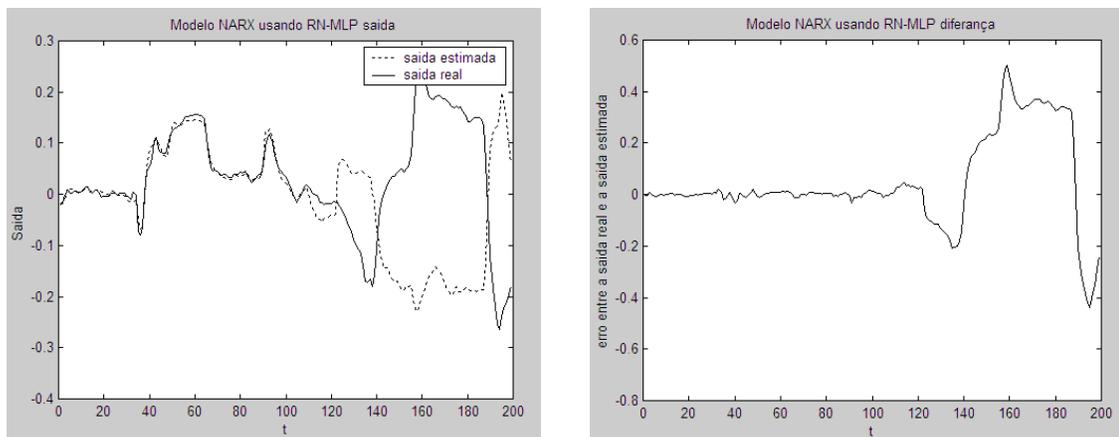
Na tabela 4.4 tem-se a rede neural com quatro centros como sendo a configuração da RBF que obteve menor erro.

TABELA 4.4 - RMSE PARA REDE RBF NO CASO DE ABBOTT.

RN	número de centros	RMSE ₁	RMSE ₂	RMSE ₃	RMSE ₄
RBF	4	0,2039	0,0094	0,0225	0,1084
RBF	5	0,7977	0,4554	1,4009	1,1107
RBF	6	1,5615	0,2894	0,6176	2,6193
RBF	7	4,0061	1,5518	3,9628	8,6788
RBF	8	30,8676	4,0849	26,0004	9,7583

Para o caso Abbott, as duas redes neurais obtiveram uma aproximação semelhante do sistema proposto. Tendo a rede RBF um desempenho melhor nas saídas $y_2(t)$ e $y_3(t)$ e pior nas saídas $y_1(t)$ e $y_4(t)$.

Nas figuras 4.3 a 4.20 são apresentados os resultados da saída estimada das RN MLP para o estudo de caso 1.

FIGURA 4.3 - SAÍDA $y_1(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.4 - SAÍDA $y_2(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.5 - SAÍDA $y_3(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

Ao observar as saídas estimadas nas figuras 4.3 a 4.5, nota-se que a RN MLP com 3 neurônios na camada oculta, no período de treinamento, responde adequadamente, quase zerando o erro de estimativa. Já em um segundo momento, quando é feita a validação, a RN diferencia o valor estimado do real, obtendo um resultado ruim na saída.

FIGURA 4.6 - SAÍDA $y_1(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

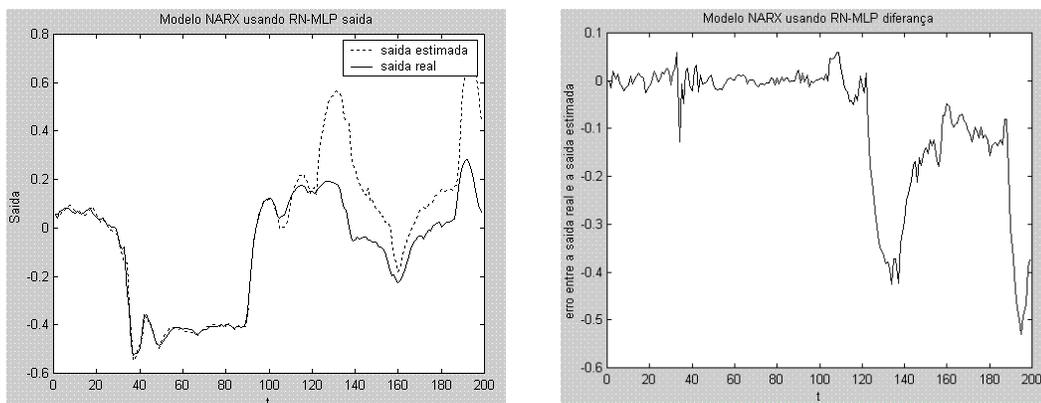
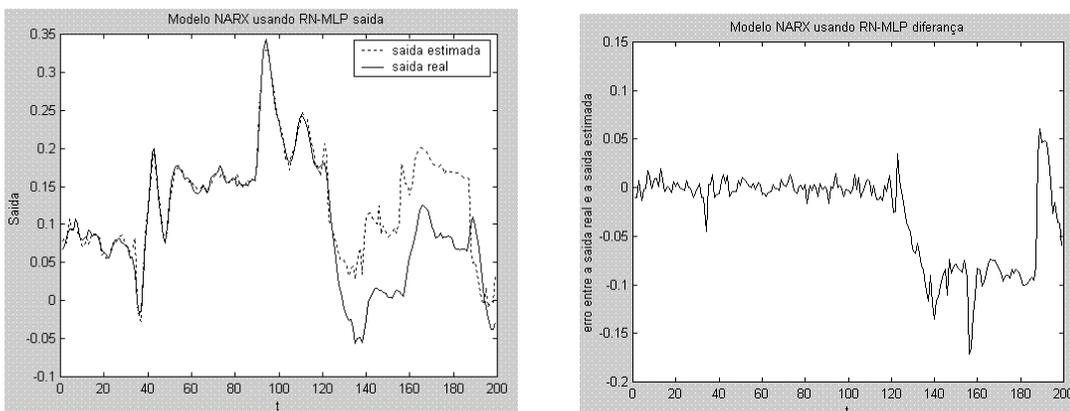


FIGURA 4.7 - SAÍDA $y_2(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.



Nas figuras 4.6, 4.7 e 4.8, nota-se que a RN MLP com 4 neurônios na camada oculta, no período de treinamento, responde satisfatoriamente, quase zerando o erro de estimativa; nos dados de validação obtém um valor razoável, apesar de ainda apresentar um erro de estimativa relativamente grande.

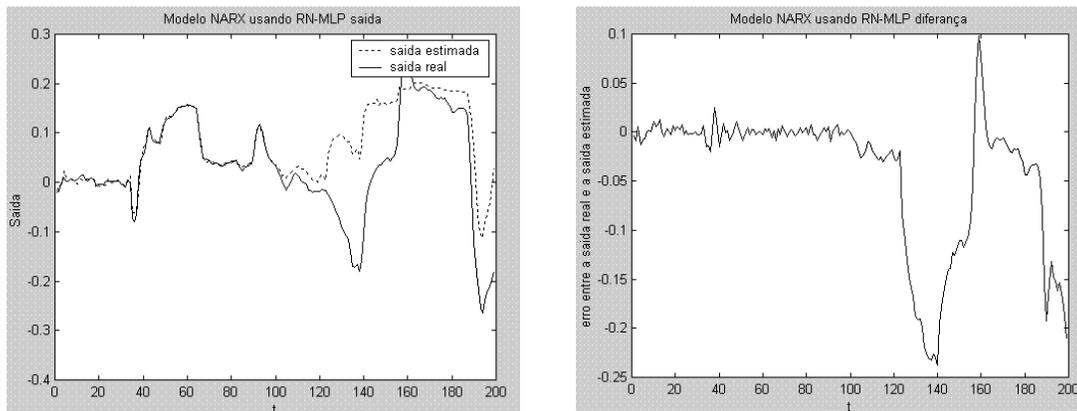
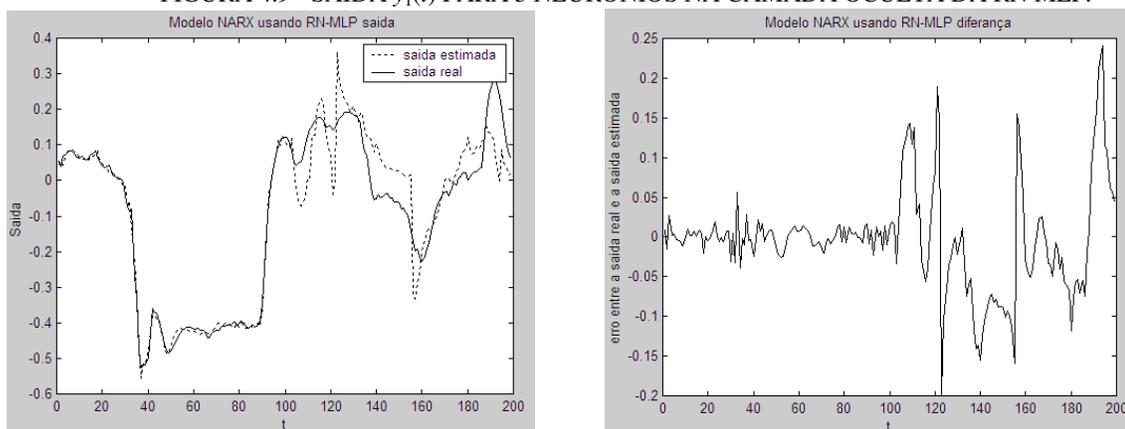
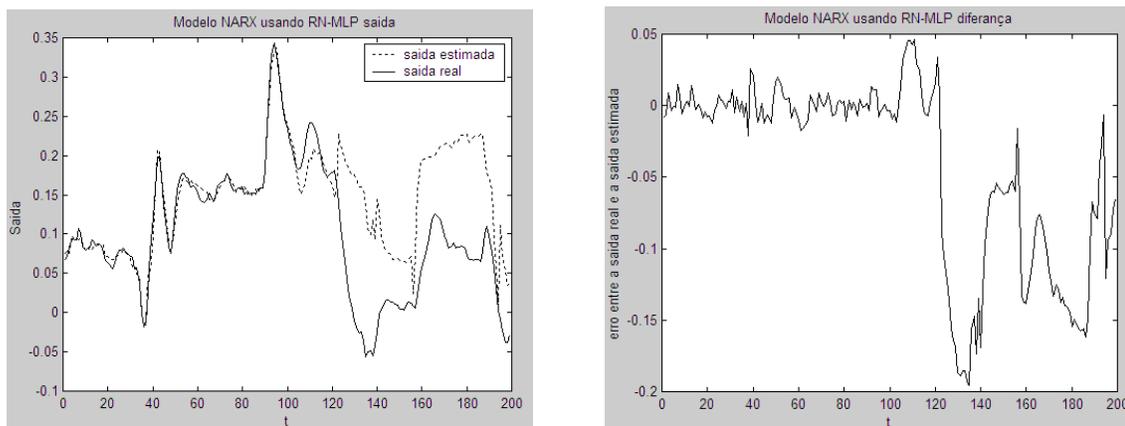
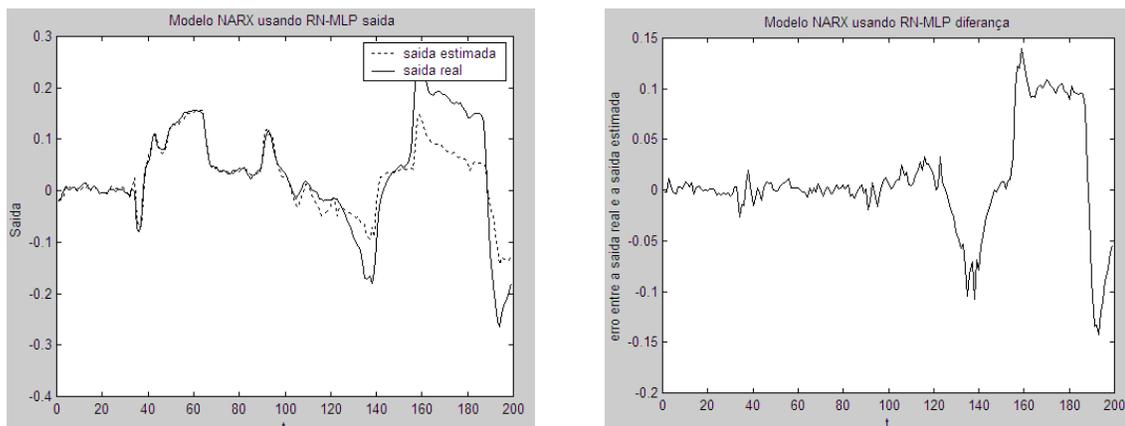
FIGURA 4.8 - SAÍDA $y_3(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.9 - SAÍDA $y_1(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.10 - SAÍDA $y_2(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

FIGURA 4.11 - SAÍDA $y_3(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP

As figuras 4.9, 4.10 e 4.11 obtiveram uma resposta satisfatória, tanto no período de treinamento quanto no período de validação.

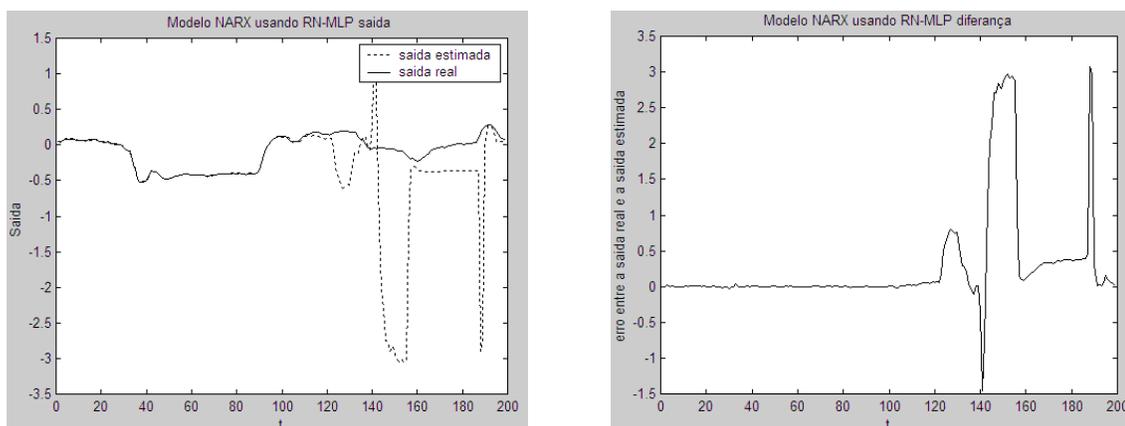
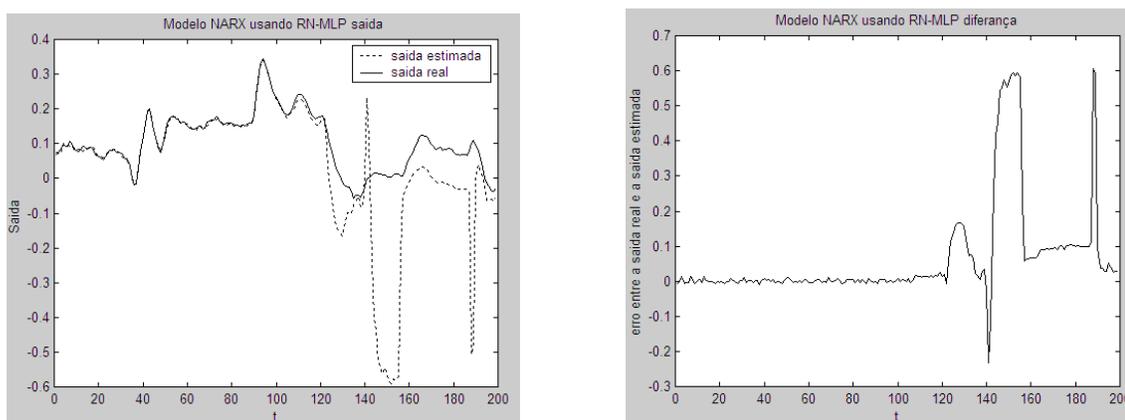
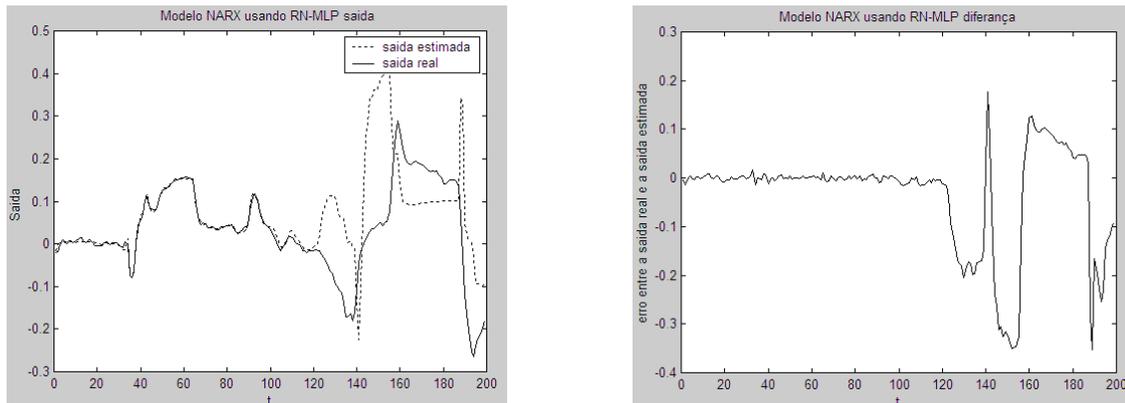
FIGURA 4.12 - SAÍDA $y_1(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.13 - SAÍDA $y_2(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

FIGURA 4.14 - SAÍDA $y_3(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

Na configuração de 6 neurônios na camada oculta, as saídas estimadas mostradas nas figuras 4.12, 4.13 e 4.14 obtiveram uma resposta satisfatória no período de treinamento, mas no período de validação o valor estimado teve momentos com grandes erros.

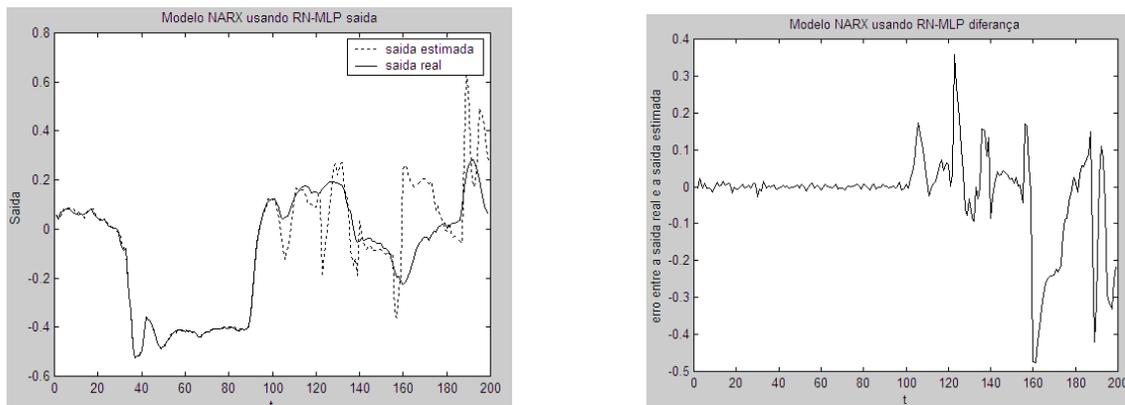
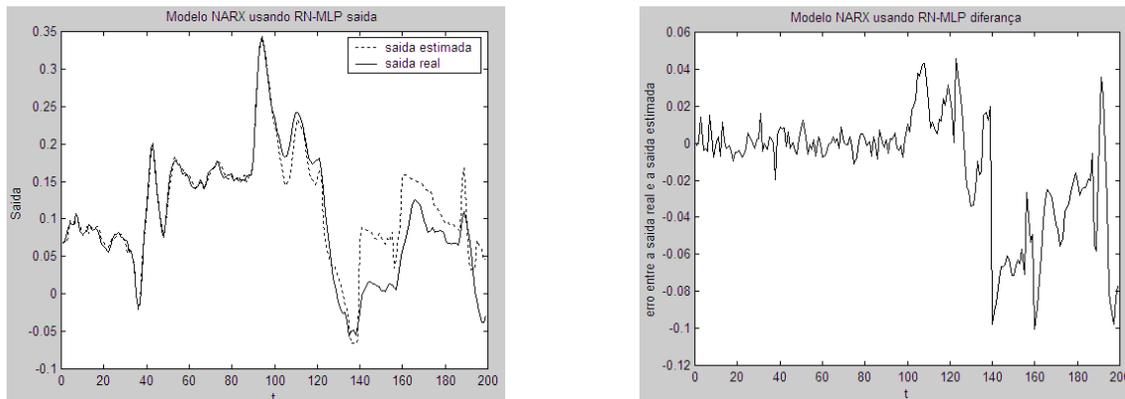
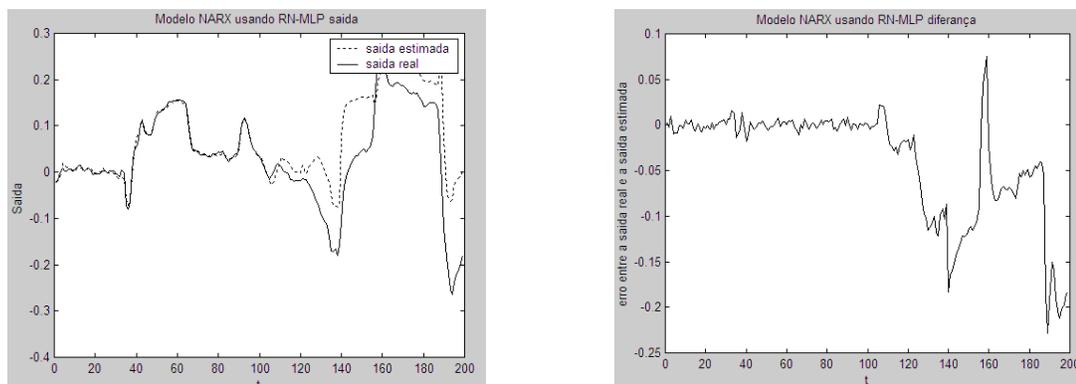
FIGURA 4.15 - SAÍDA $y_1(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.16 - SAÍDA $y_2(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

FIGURA 4.17 - SAÍDA $y_3(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.



As figuras 4.15 a 4.17, obtiveram uma resposta satisfatória tanto no período de treinamento quanto no período de validação, sendo apenas um pouco pior que na configuração de 5 neurônios na camada oculta.

FIGURA 4.18 - SAÍDA $y_1(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

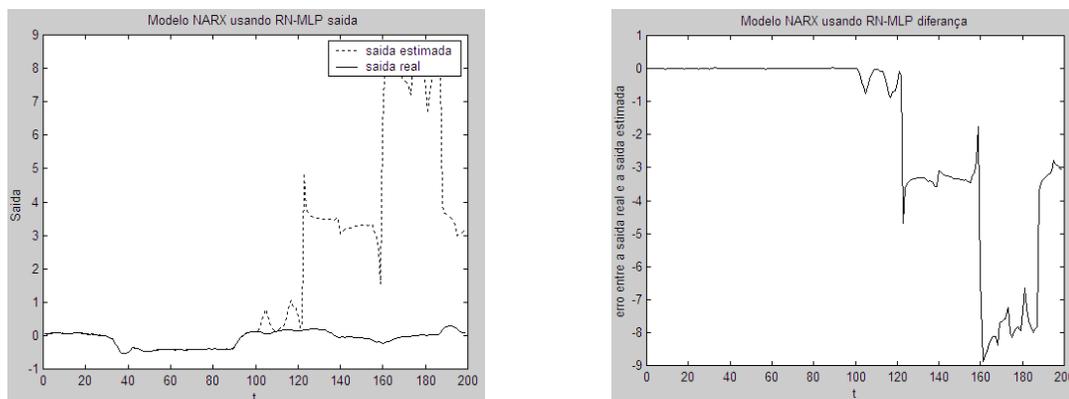
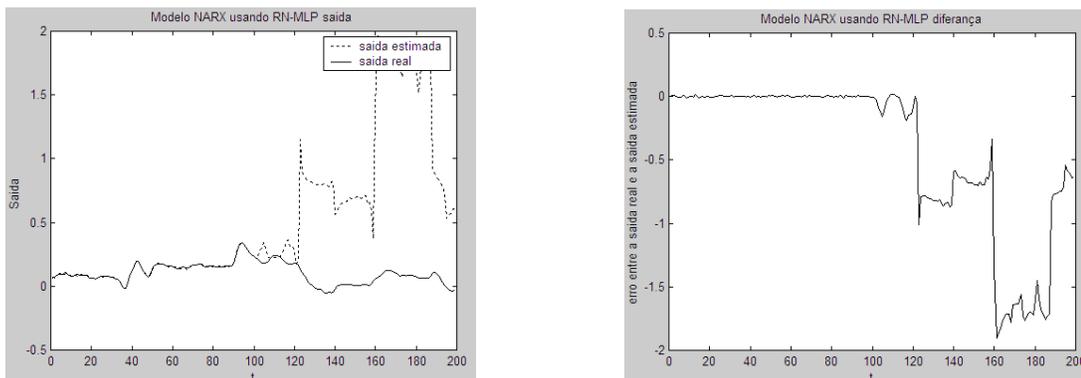
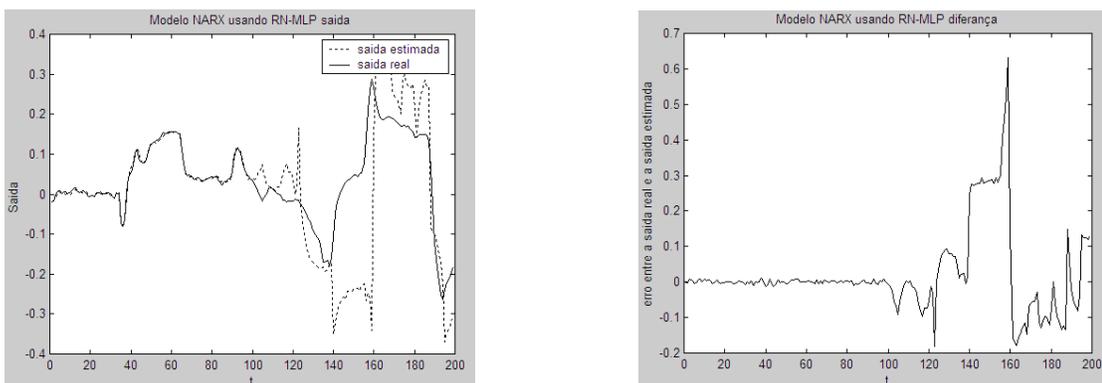


FIGURA 4.19 - SAÍDA $y_2(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.20 - SAÍDA $y_3(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

Na configuração de 8 neurônios na camada oculta, mostrada nas figuras 4.18 a 4.20, observa-se que a saída estimada não está de acordo com a saída real no período de validação.

Nas figuras 4.3 a 4.20 nota-se que as redes MLP só tiveram um erro reduzido para os dados treinados e no restante dos dados de entrada a rede estimou a saída com grandes diferenças. Além disso, as redes com um número maior de neurônios na camada oculta fez com que o erro obtido fosse menor, mas isso só ocorreu até 5 neurônios na camada oculta. Após isso, os erros voltaram a aumentar.

Nas figuras 4.21 a 4.38 são apresentados os resultados da saída estimada das RN RBF para o caso 1.

FIGURA 4.21 - SAÍDA $y_1(t)$ PARA 3 CENTROS DA RN RBF.

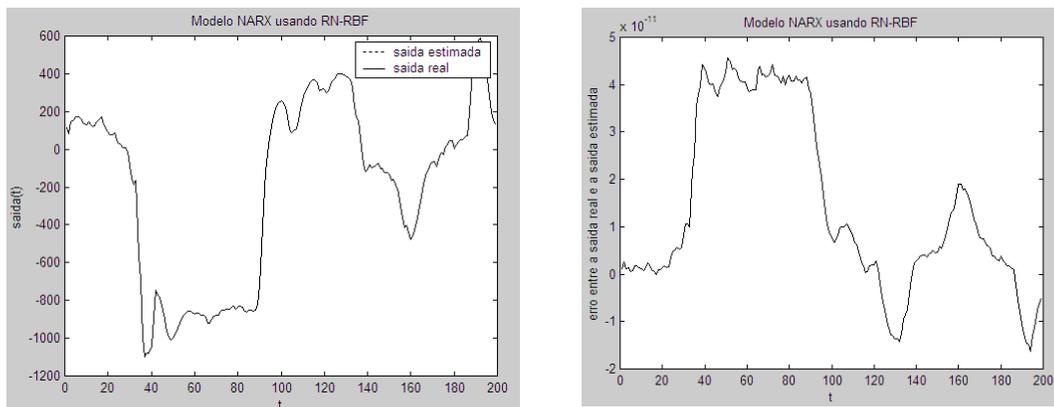


FIGURA 4.22 - SAÍDA $y_2(t)$ PARA 3 CENTROS DA RN RBF.

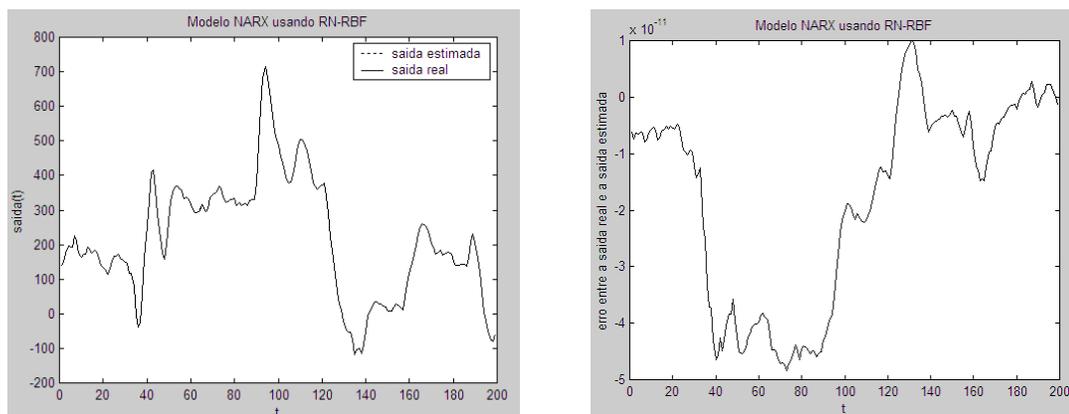


FIGURA 4.23 - SAÍDA $y_3(t)$ PARA 3 CENTROS DA RN RBF.

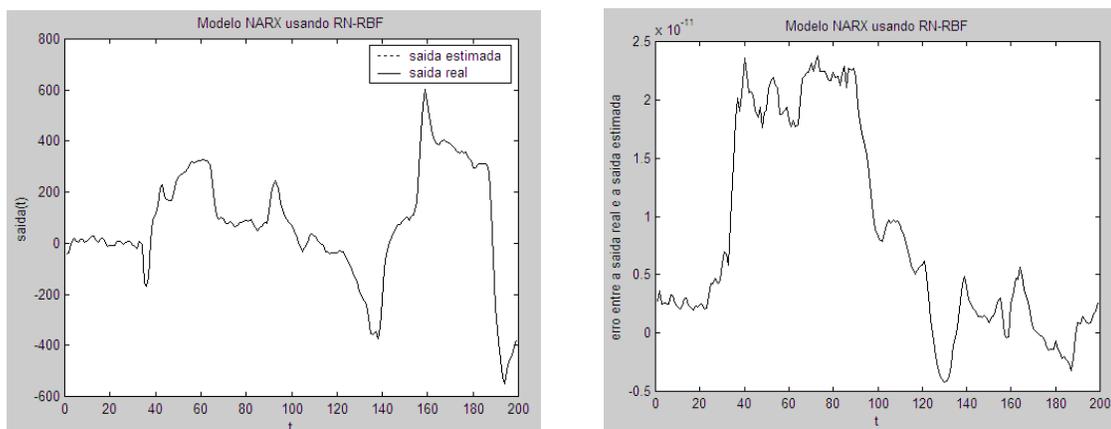


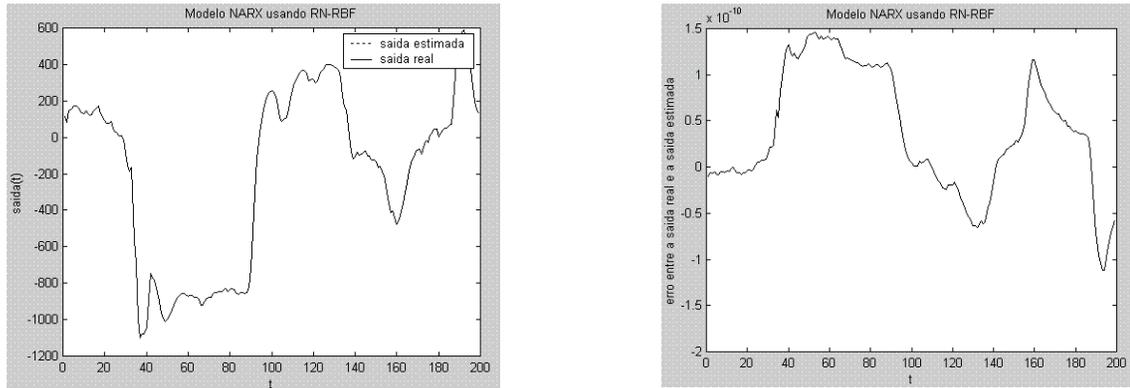
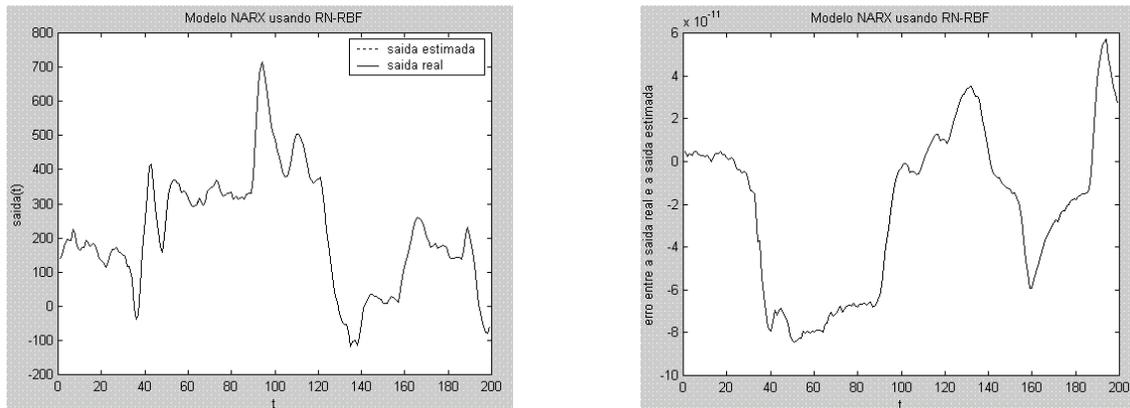
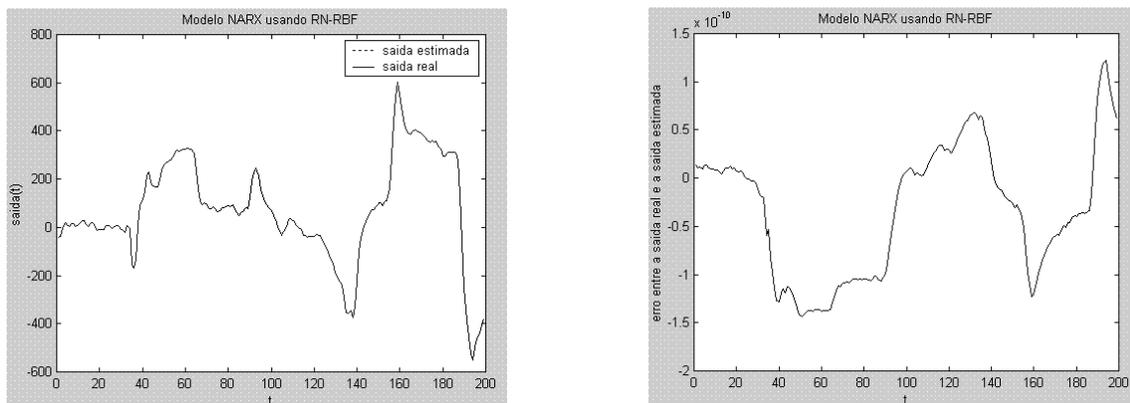
FIGURA 4.24 - SAÍDA $y_1(t)$ PARA 4 CENTROS DA RN RBF.FIGURA 4.25 - SAÍDA $y_2(t)$ PARA 4 CENTROS DA RN RBF.FIGURA 4.26 - SAÍDA $y_3(t)$ PARA 4 CENTROS DA RN RBF.

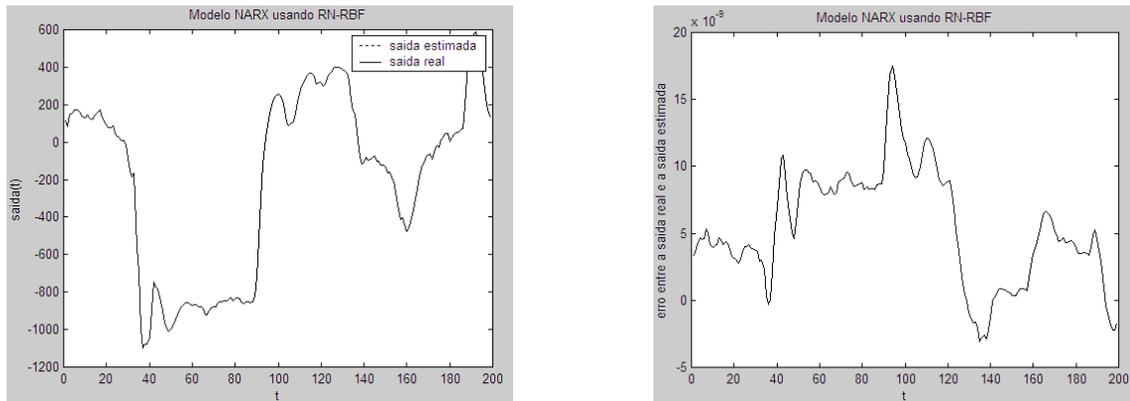
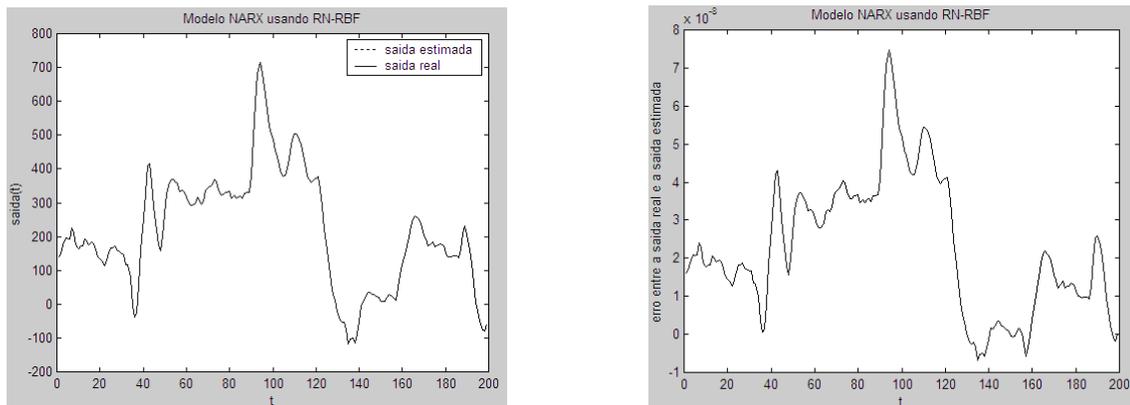
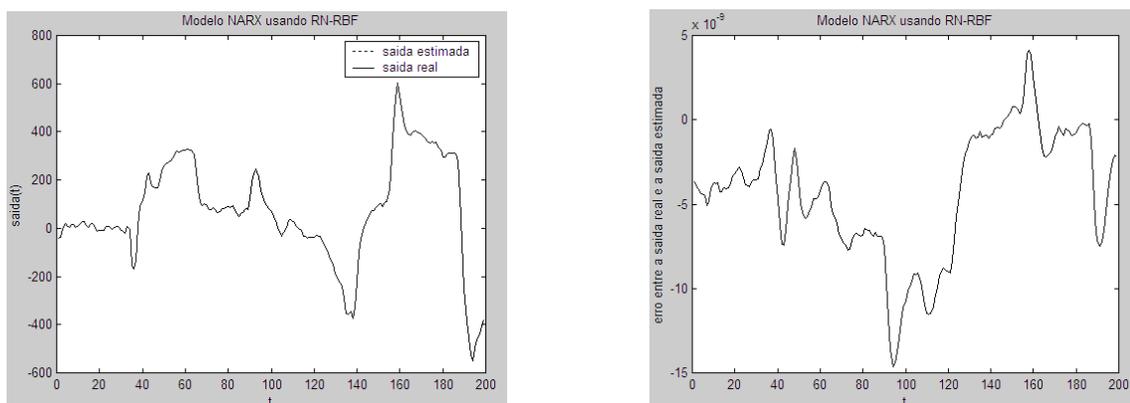
FIGURA 4.27 - SAÍDA $y_1(t)$ PARA 5 CENTROS DA RN RBF.FIGURA 4.28 - SAÍDA $y_2(t)$ PARA 5 CENTROS DA RN RBF.FIGURA 4.29 - SAÍDA $y_3(t)$ PARA 5 CENTROS DA RN RBF.

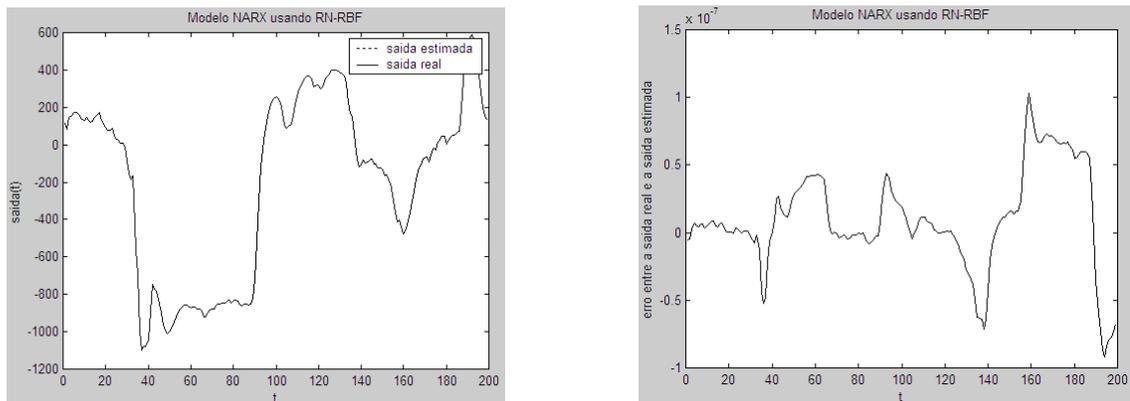
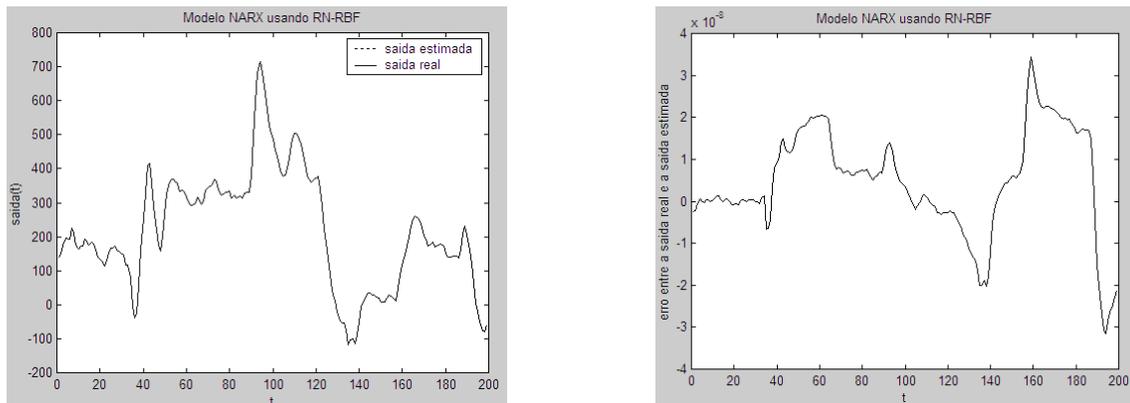
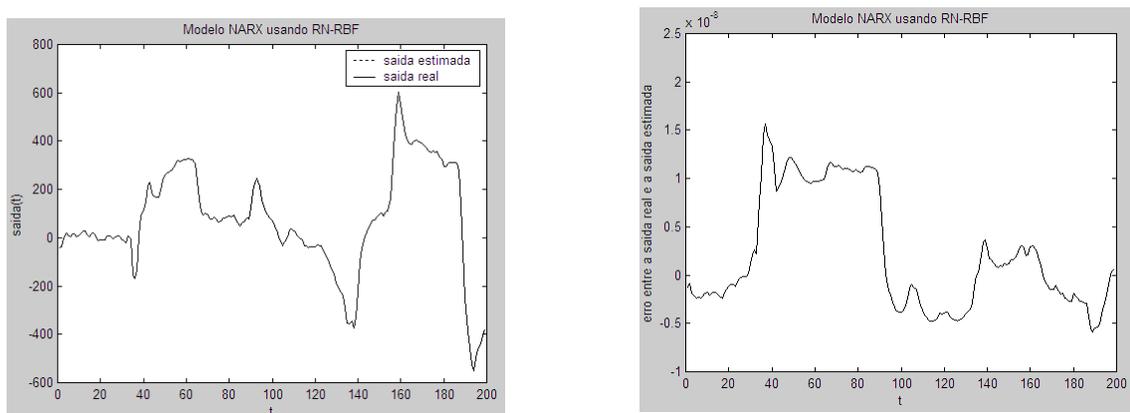
FIGURA 4.30 - SAÍDA $y_1(t)$ PARA 6 CENTROS DA RN RBF.FIGURA 4.31 - SAÍDA $y_2(t)$ PARA 6 CENTROS DA RN RBF.FIGURA 4.32 - SAÍDA $y_3(t)$ PARA 6 CENTROS DA RN RBF.

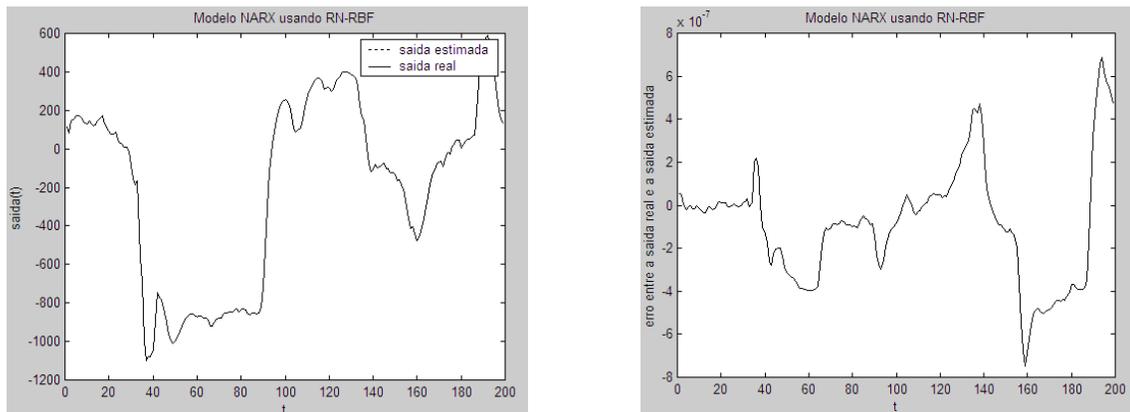
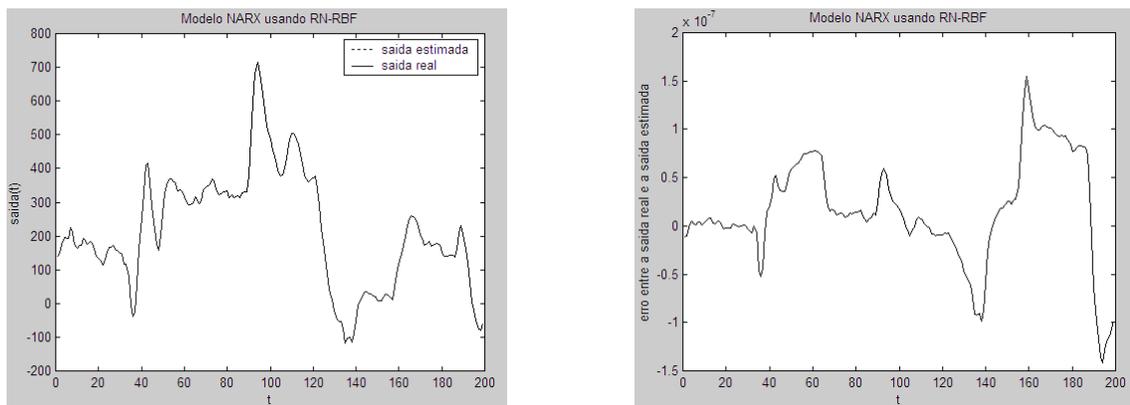
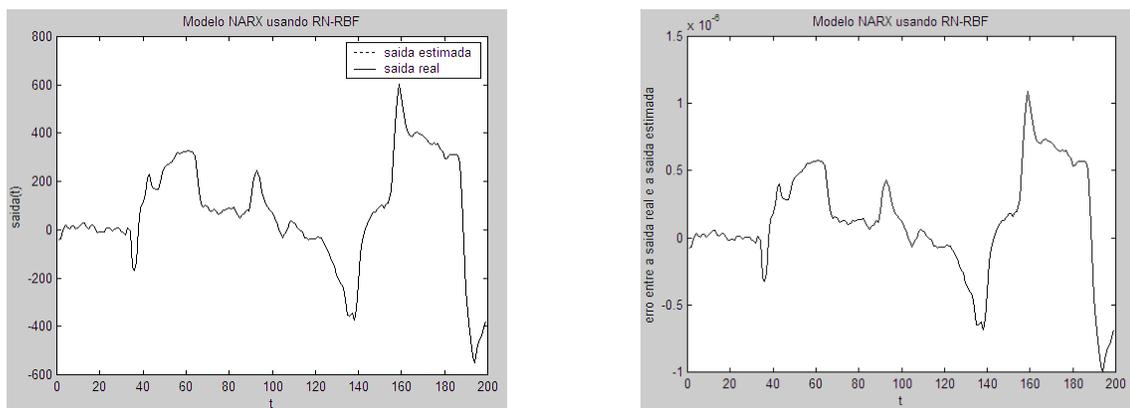
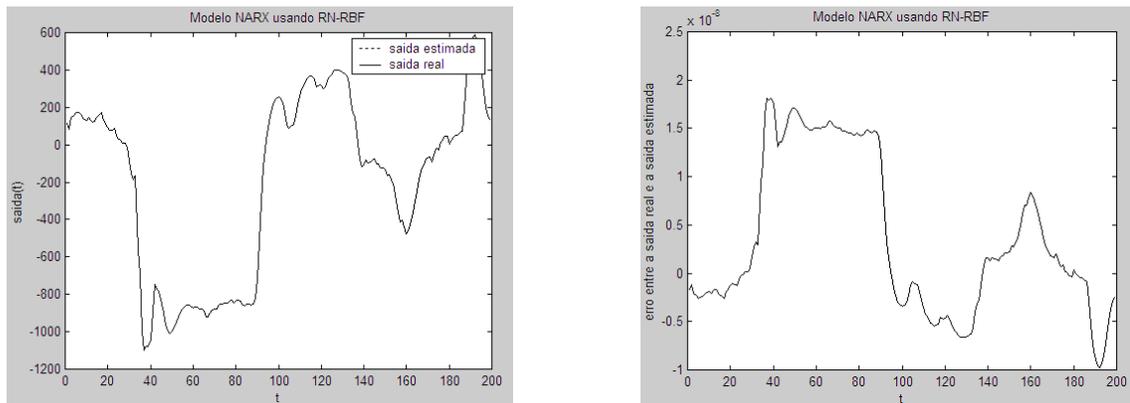
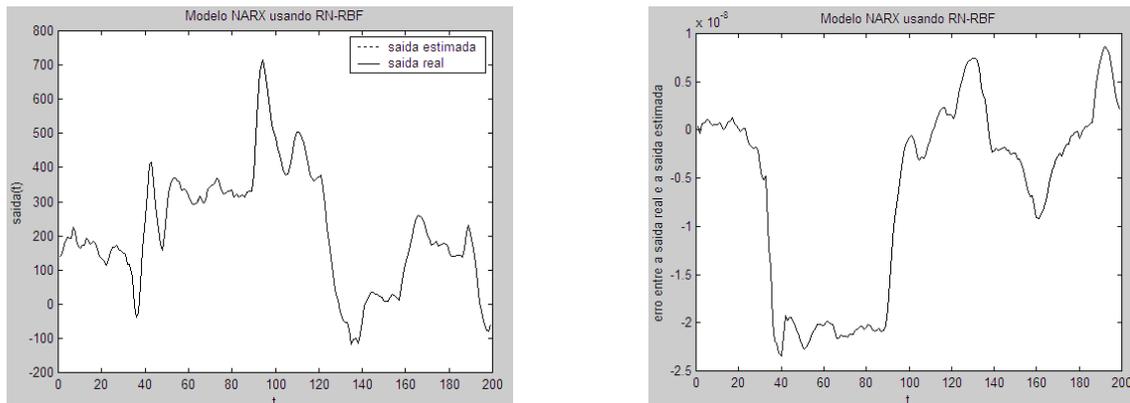
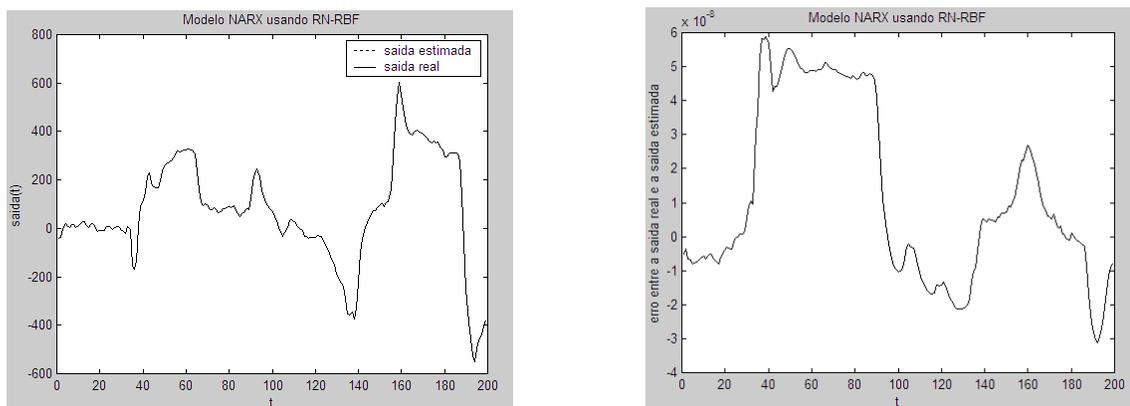
FIGURA 4.33 - SAÍDA $y_1(t)$ PARA 7 CENTROS DA RN RBF.FIGURA 4.34 - SAÍDA $y_2(t)$ PARA 7 CENTROS DA RN RBF.FIGURA 4.35 - SAÍDA $y_3(t)$ PARA 7 CENTROS DA RN RBF.

FIGURA 4.36 - SAÍDA $y_1(t)$ PARA 8 CENTROS DA RN RBF.FIGURA 4.37 - SAÍDA $y_2(t)$ PARA 8 CENTROS DA RN RBF.FIGURA 4.38 - SAÍDA $y_3(t)$ PARA 8 CENTROS DA RN RBF.

Nas figuras 4.21 a 4.38 nota-se que as RN RBF obtiveram um resultado satisfatório em todas as configurações de 3 a 8 centros. Ao analisar o gráfico do erro dessas figuras nota-se um valor bem baixo, o que demonstra a eficiência da técnica. Ao ver essa eficiência percebe-se que a configuração de 3 centros obteve uma estimativa melhor que todas as outras configurações da RN RBF.

Nas figuras 4.39 a 4.62 são apresentados os resultados da saída estimada das RN MLP para o caso 2.

FIGURA 4.39 - SAÍDA $y_1(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

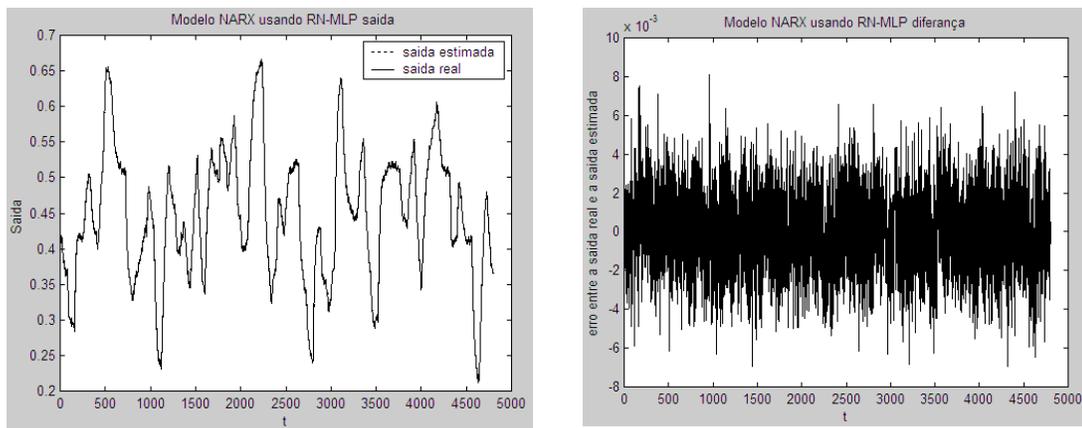


FIGURA 4.40 - SAÍDA $y_2(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

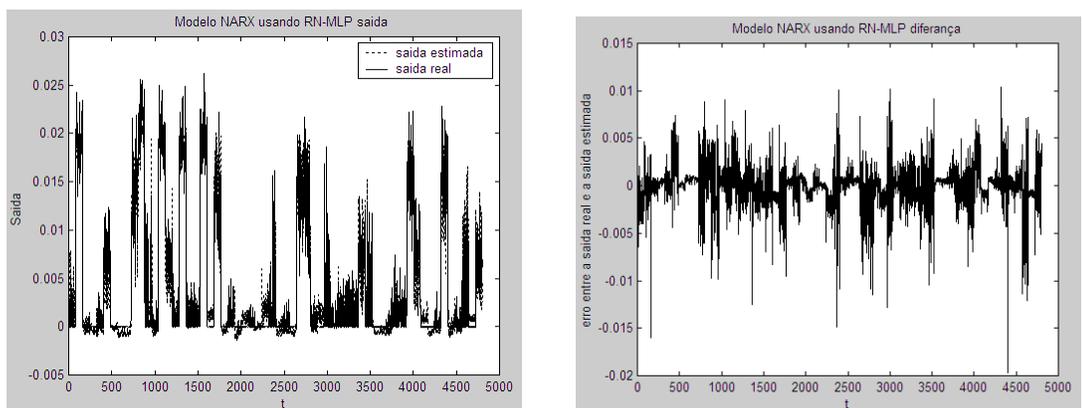


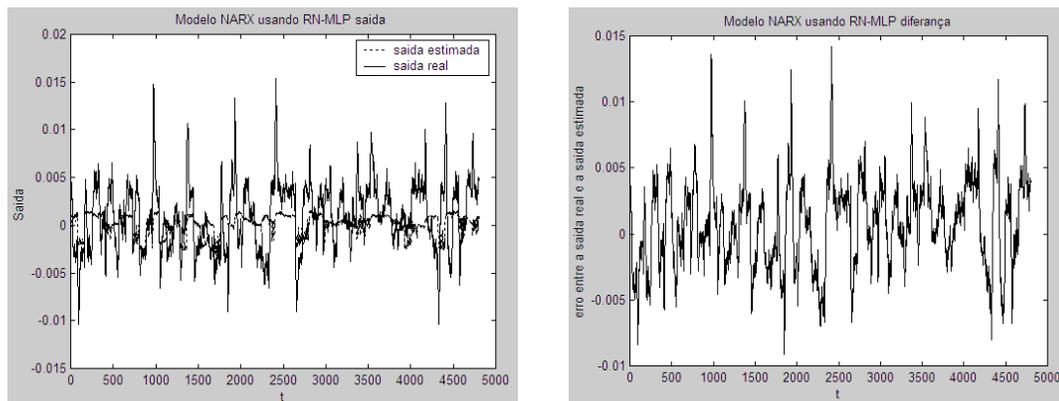
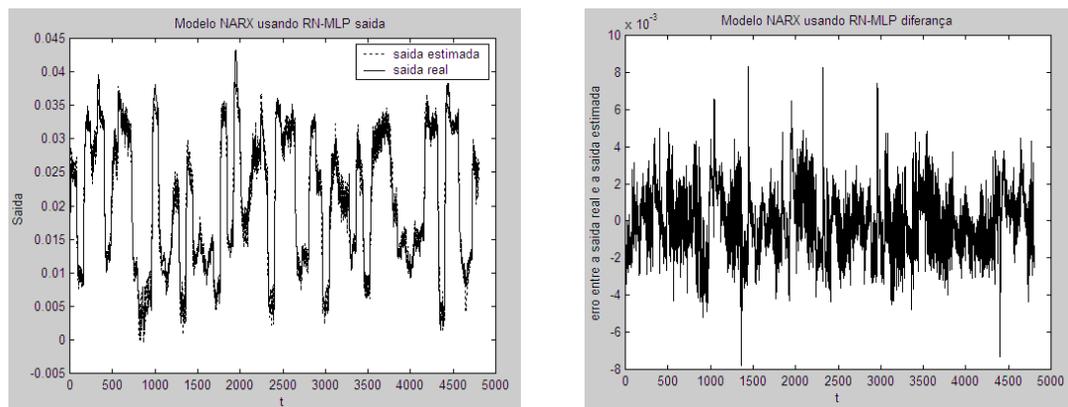
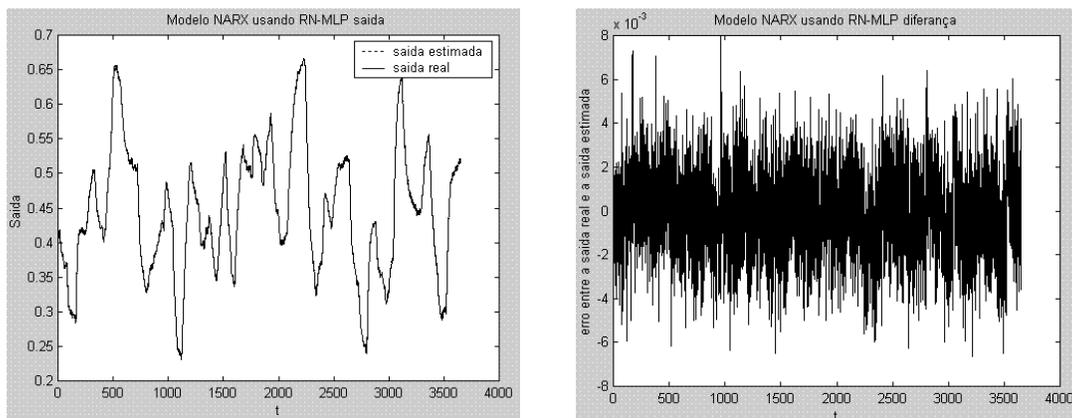
FIGURA 4.41 - SAÍDA $y_3(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.42 - SAÍDA $y_4(t)$ PARA 3 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.43 - SAÍDA $y_1(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

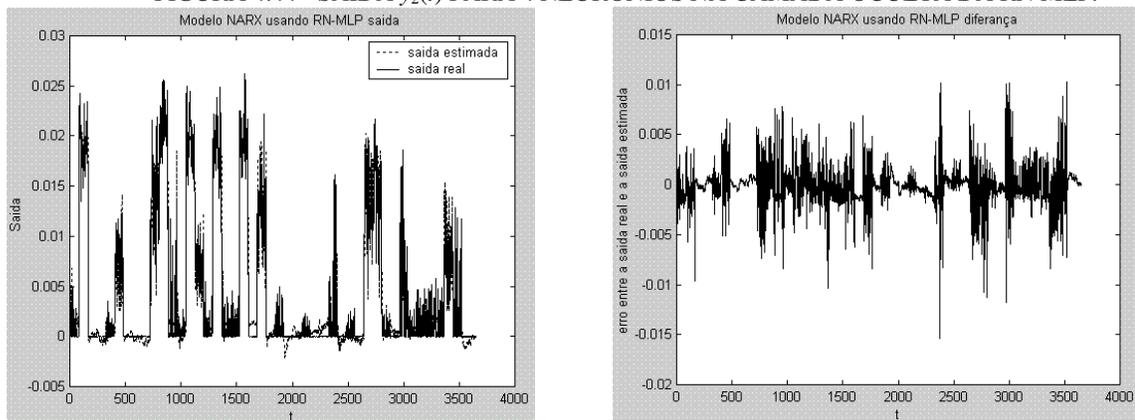
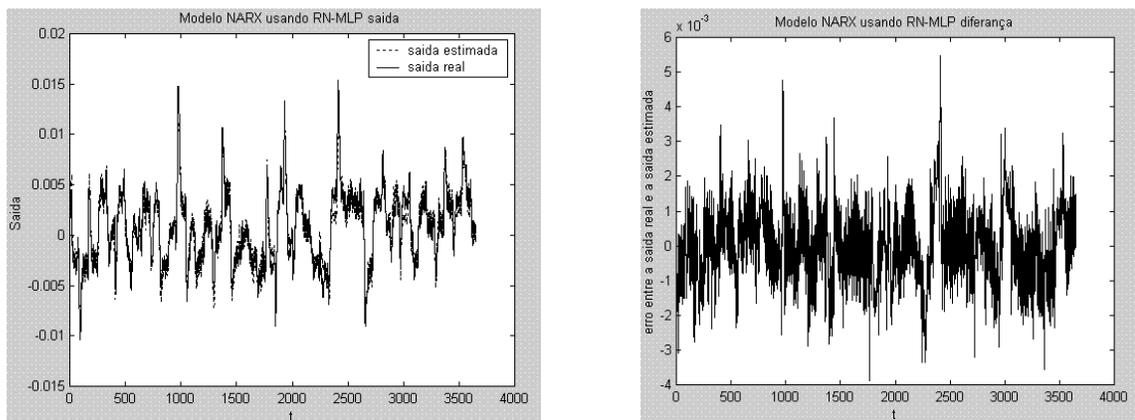
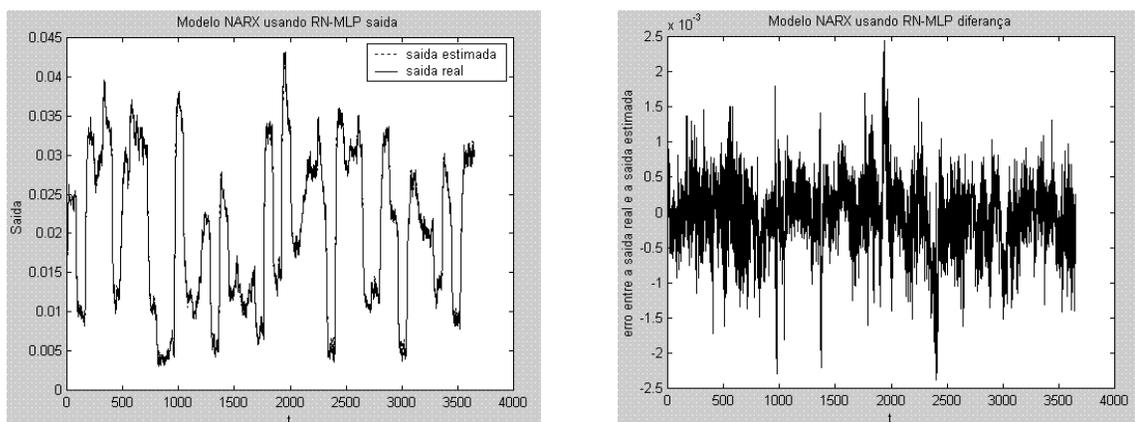
FIGURA 4.44 - SAÍDA $y_2(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.45 - SAÍDA $y_3(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.46 - SAÍDA $y_4(t)$ PARA 4 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

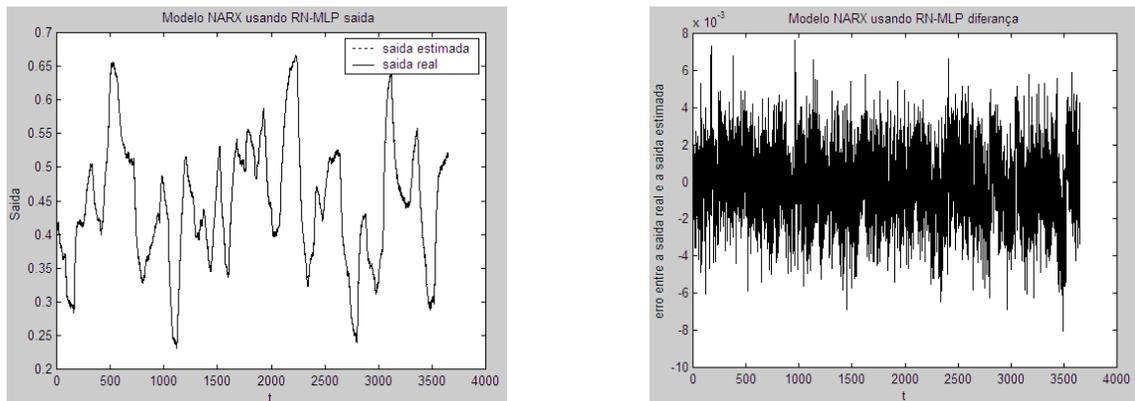
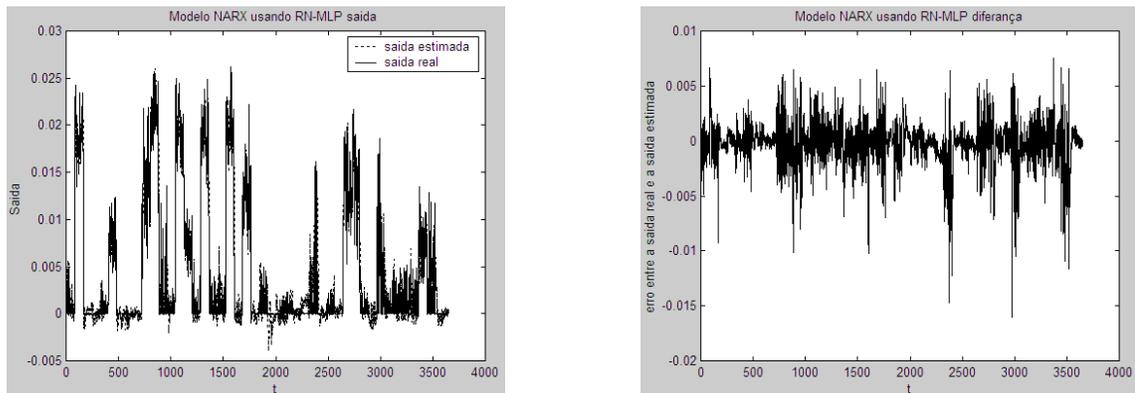
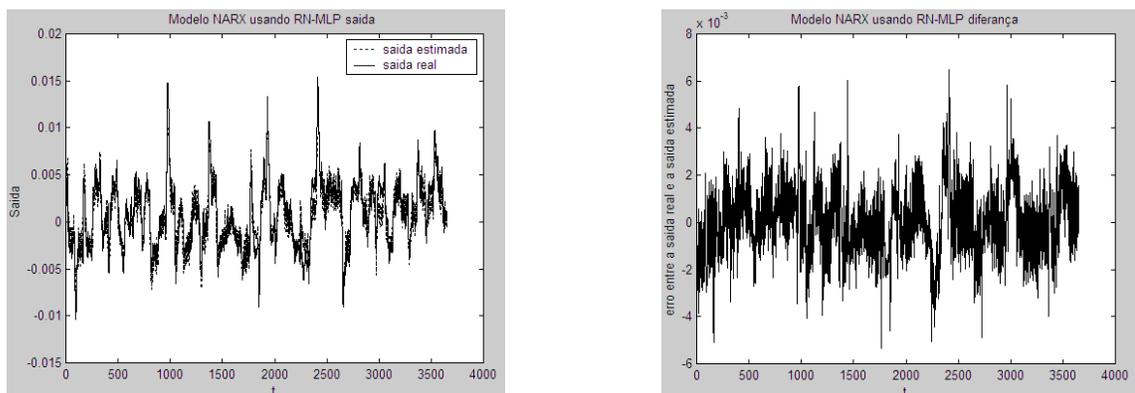
FIGURA 4.47 - SAÍDA $y_1(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.48 - SAÍDA $y_2(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.49 - SAÍDA $y_3(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

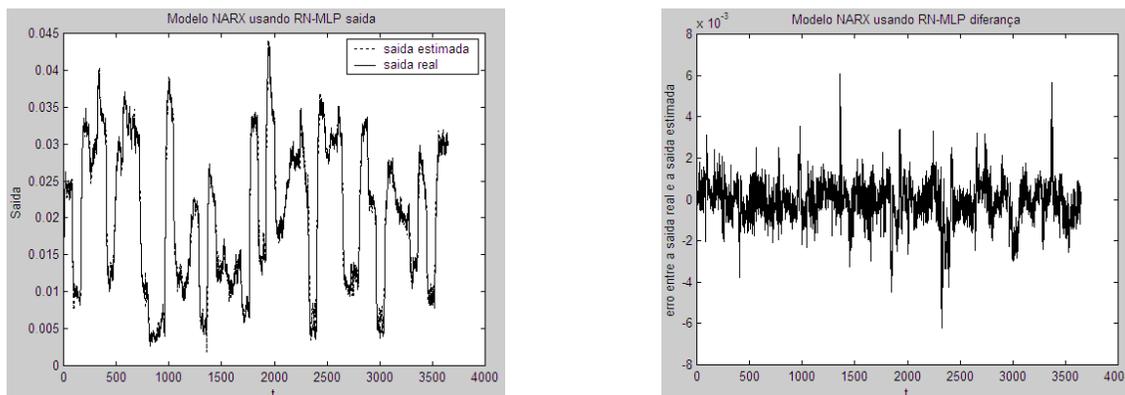
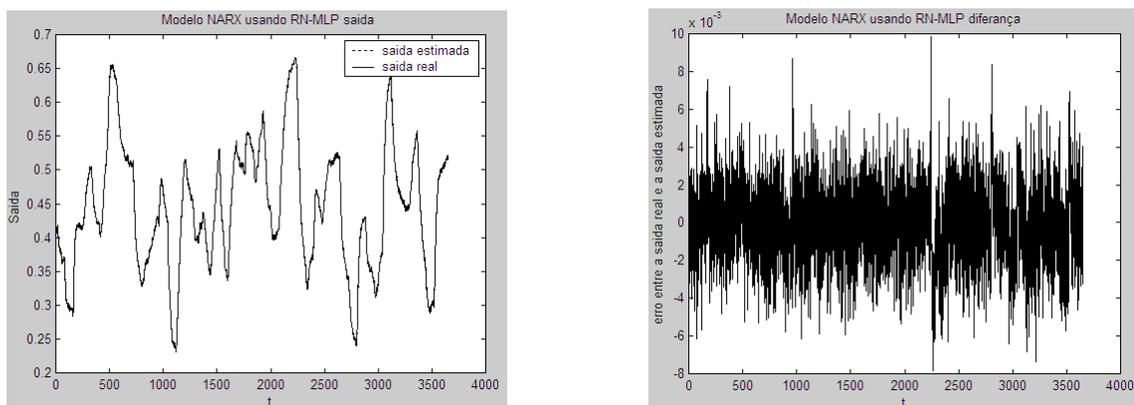
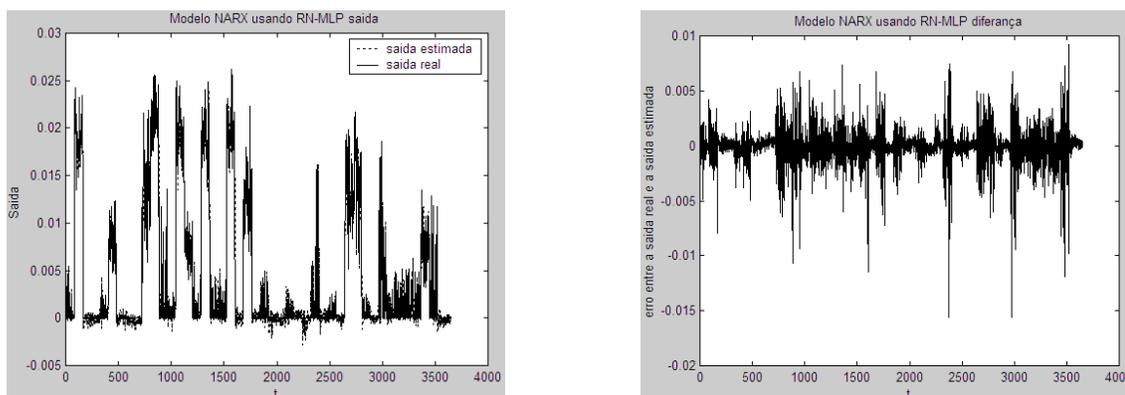
FIGURA 4.50 - SAÍDA $y_4(t)$ PARA 5 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.51 - SAÍDA $y_1(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.52 - SAÍDA $y_2(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

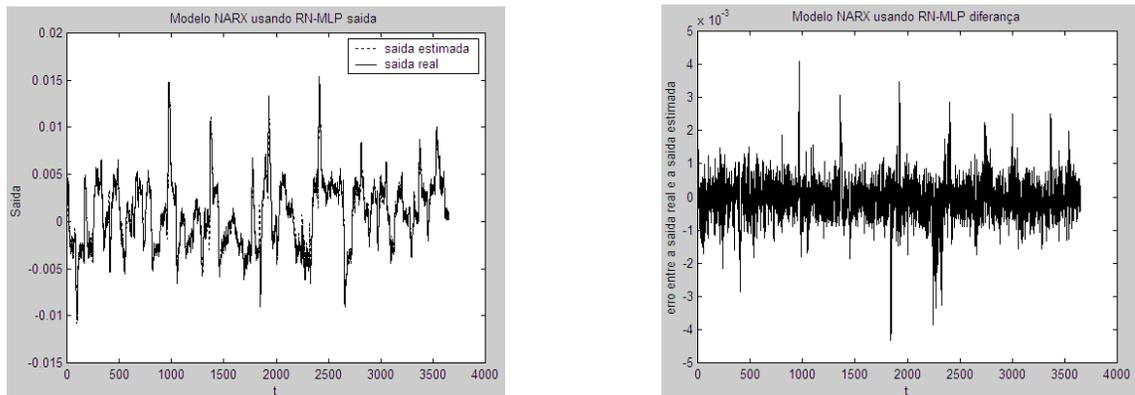
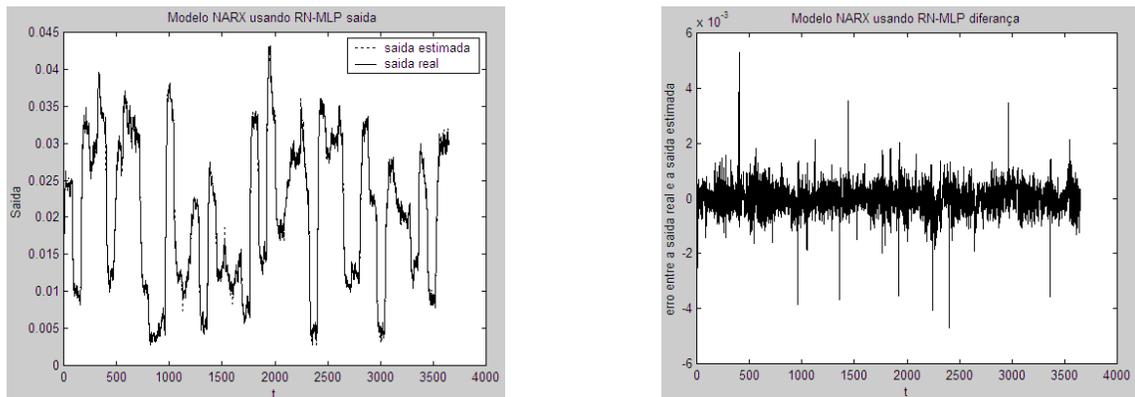
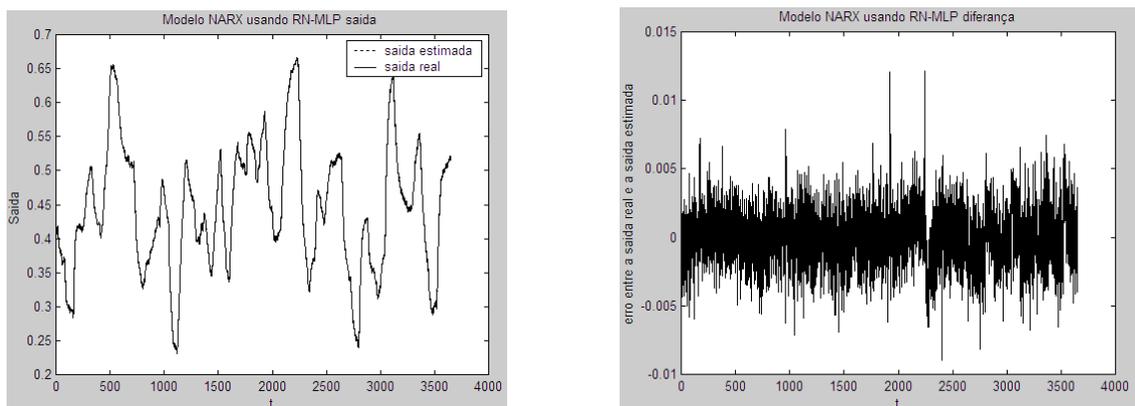
FIGURA 4.53 - SAÍDA $y_3(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.54 - SAÍDA $y_4(t)$ PARA 6 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.55 - SAÍDA $y_1(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

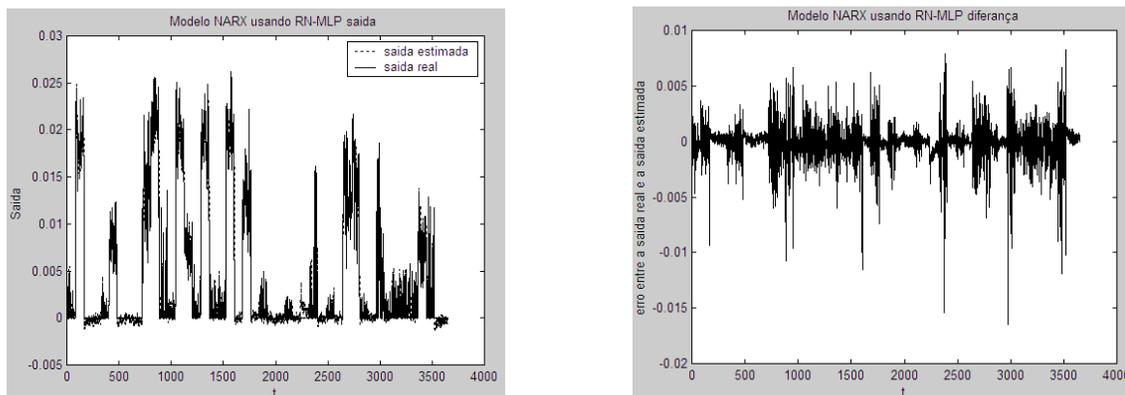
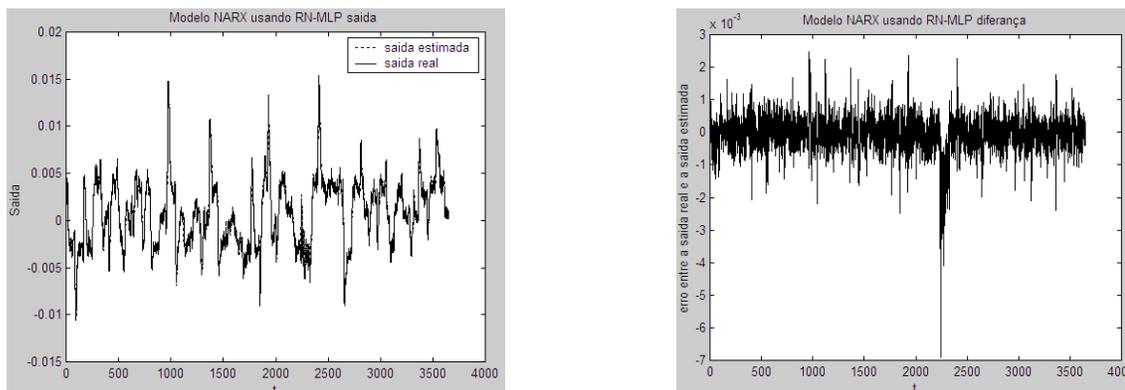
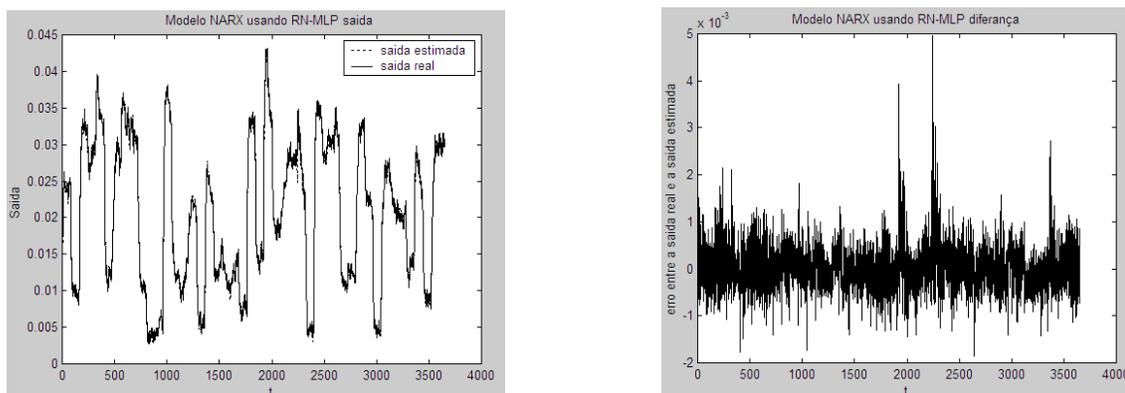
FIGURA 4.56 - SAÍDA $y_2(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.57 - SAÍDA $y_3(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.58 - SAÍDA $y_4(t)$ PARA 7 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

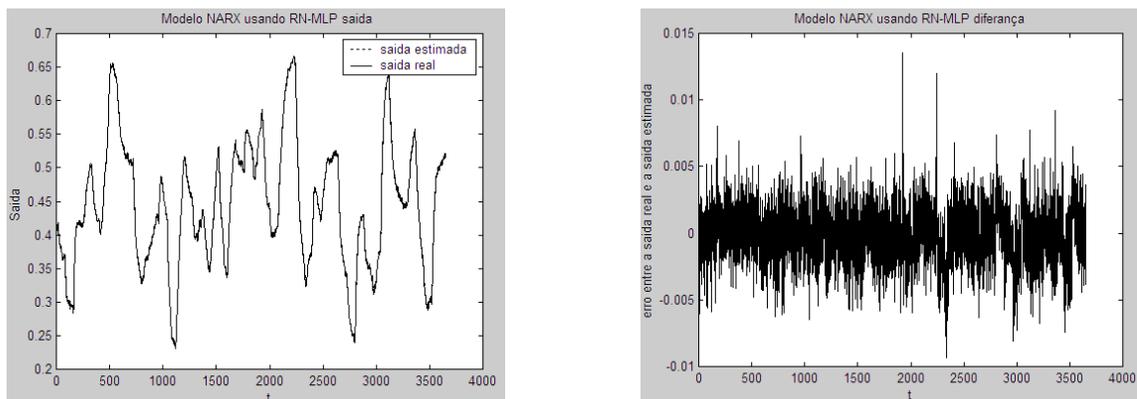
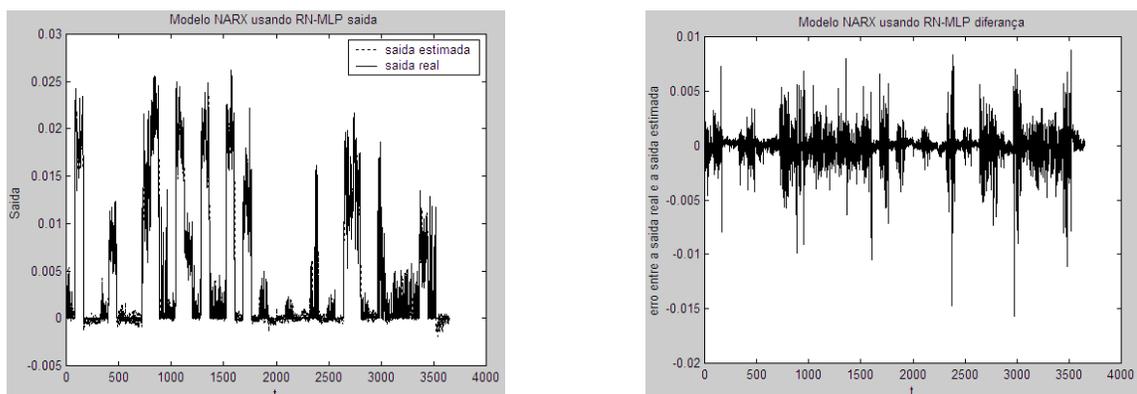
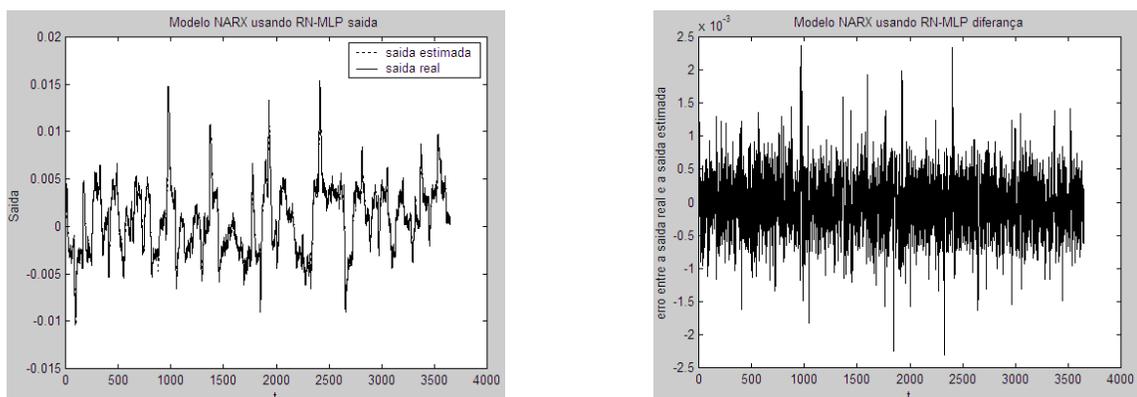
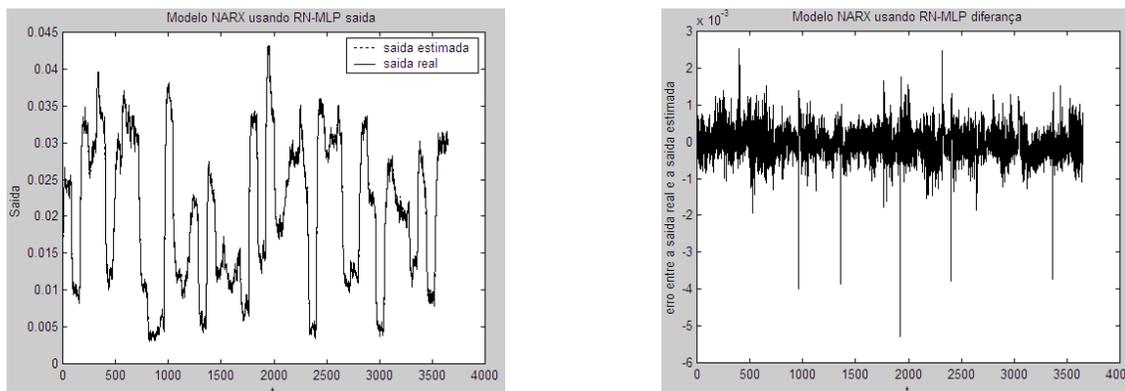
FIGURA 4.59 - SAÍDA $y_1(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.60 - SAÍDA $y_2(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.FIGURA 4.61 - SAÍDA $y_3(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

FIGURA 4.62 - SAÍDA $y_4(t)$ PARA 8 NEURÔNIOS NA CAMADA OCULTA DA RN MLP.

Nas figuras 4.39 a 4.62 nota-se que as RN MLP obtiveram um resultado satisfatório em quase todas as saídas e em todas as configurações de 3 a 8 neurônios na camada oculta. Ao analisar essas figuras nota-se que a saída $y_3(t)$ da figura 4.41 obteve um valor ruim de estimativa, porém, o fato de a saída $y_3(t)$ ter valores baixos o seu erro médio acaba ficando baixo conforme observado na tabela 4.3.

Nas figuras 4.63 a 4.82 são apresentados os resultados da saída estimada das RN RBF para o caso 2.

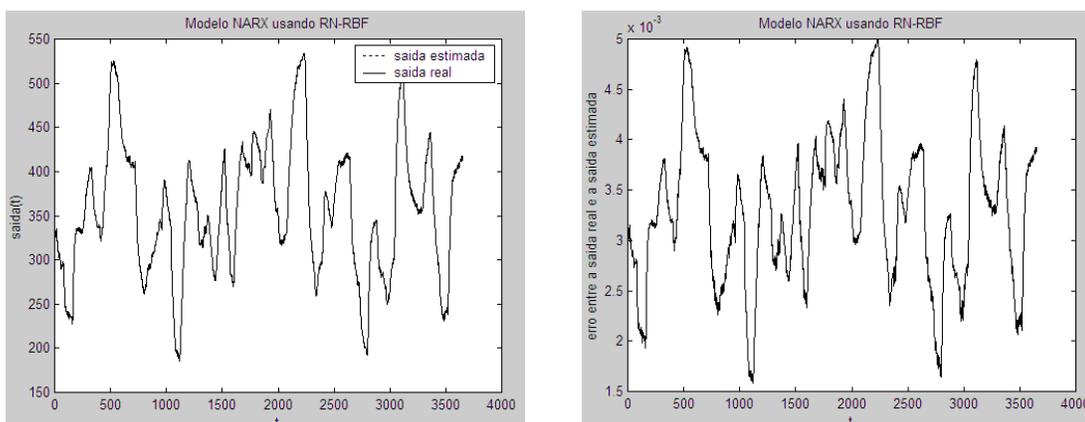
FIGURA 4.63 - SAÍDA $y_1(t)$ PARA 4 CENTROS DA RN RBF.

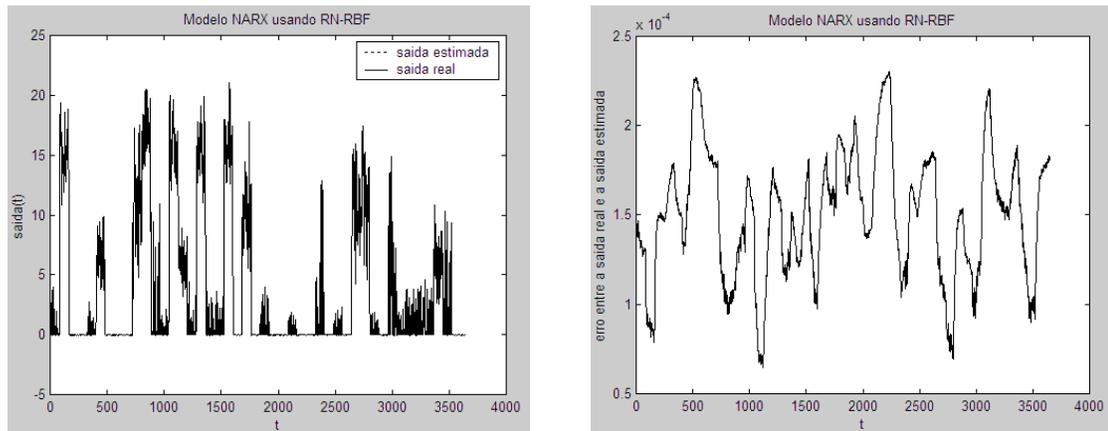
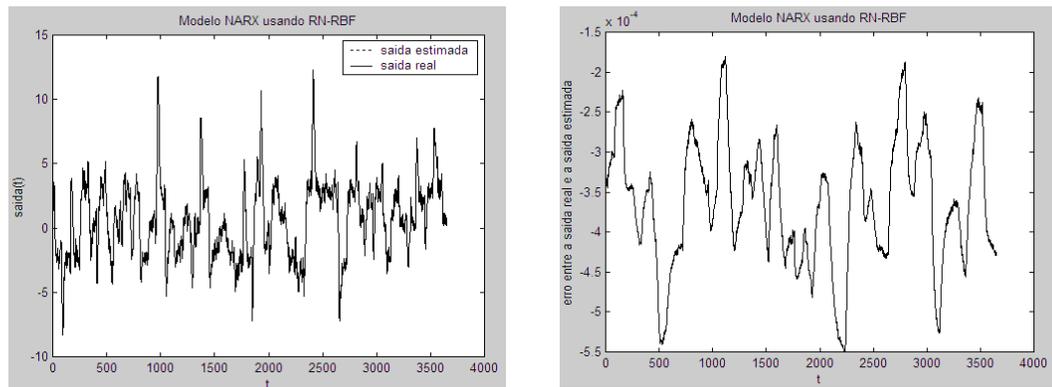
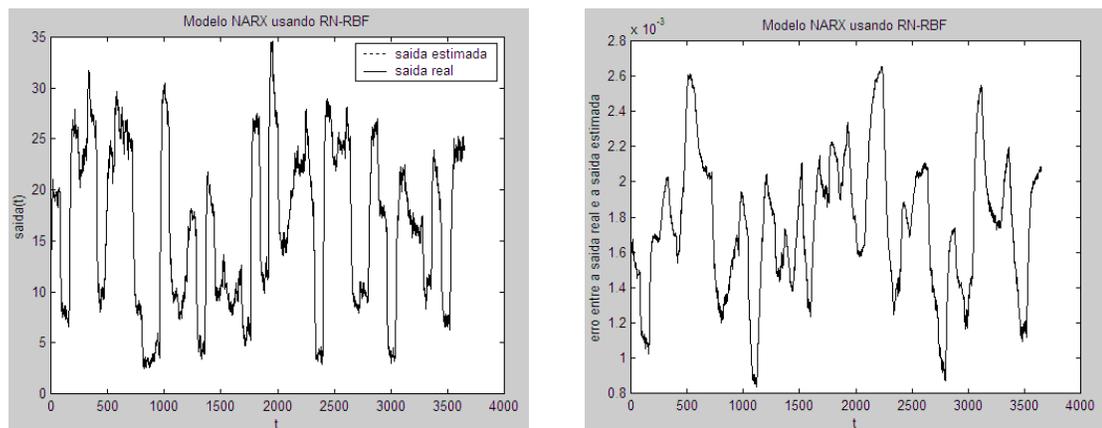
FIGURA 4.64 - SAÍDA $y_2(t)$ PARA 4 CENTROS DA RN RBF.FIGURA 4.65 - SAÍDA $y_3(t)$ PARA 4 CENTROS DA RN RBF.FIGURA 4.66 - SAÍDA $y_4(t)$ PARA 4 CENTROS DA RN RBF.

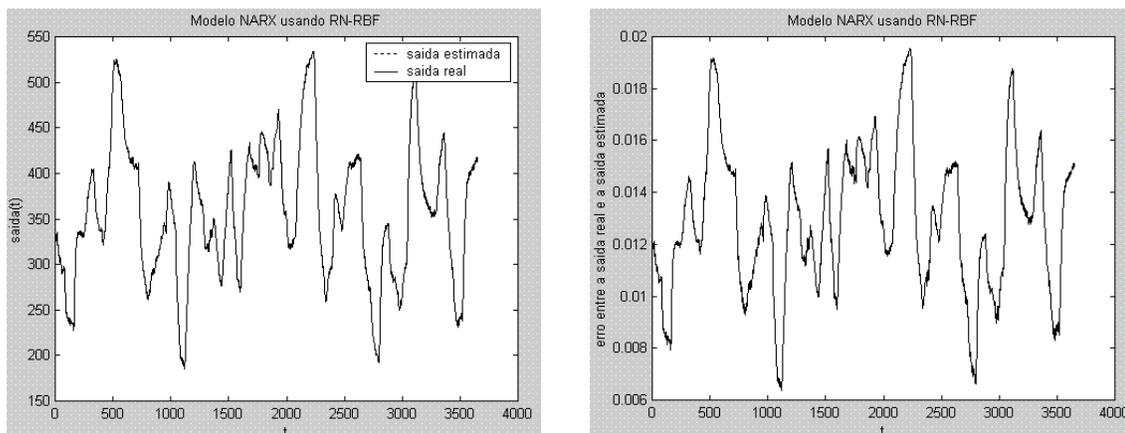
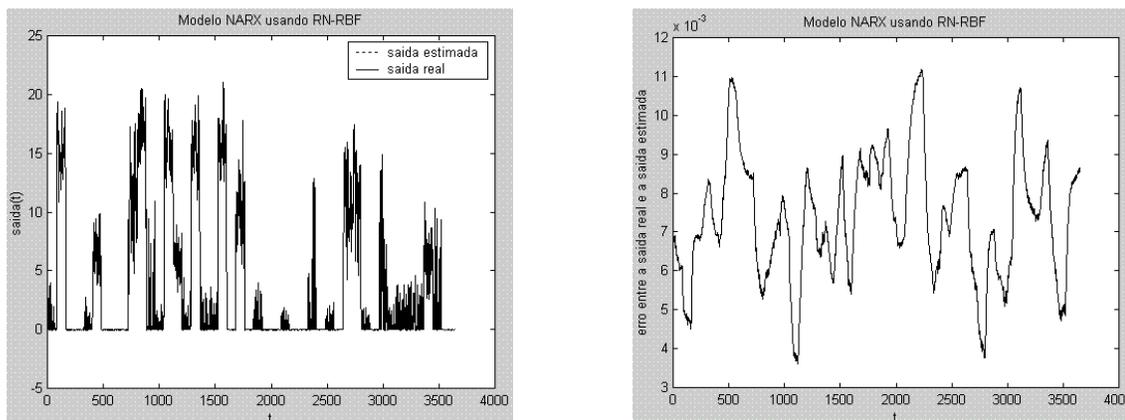
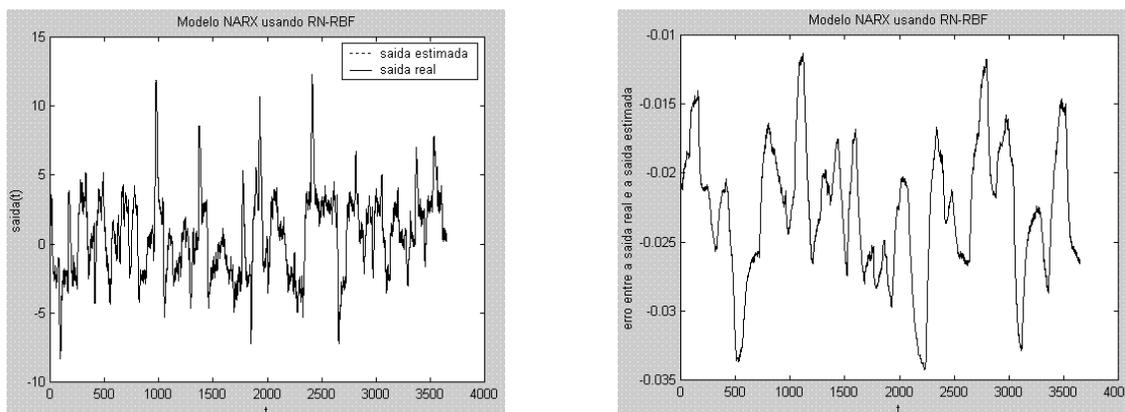
FIGURA 4.67 - SAÍDA $y_1(t)$ PARA 5 CENTROS DA RN RBF.FIGURA 4.68 - SAÍDA $y_2(t)$ PARA 5 CENTROS DA RN RBF.FIGURA 4.69 - SAÍDA $y_3(t)$ PARA 5 CENTROS DA RN RBF.

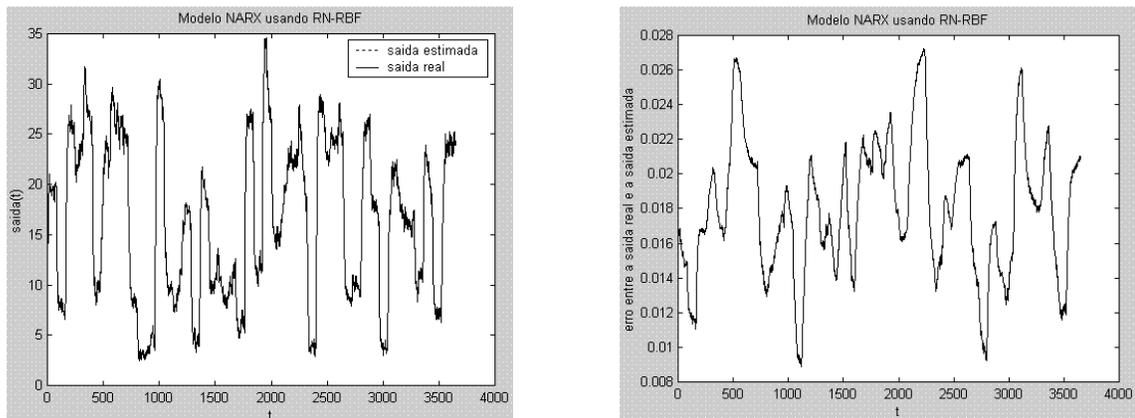
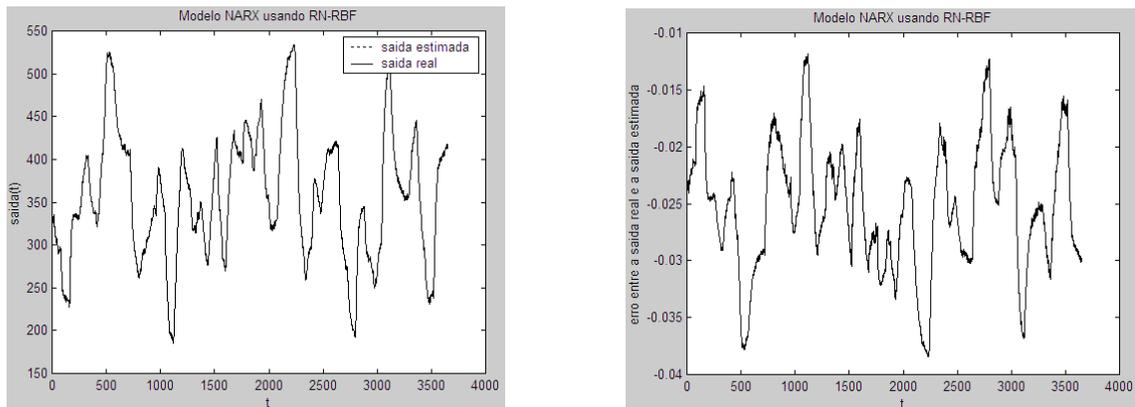
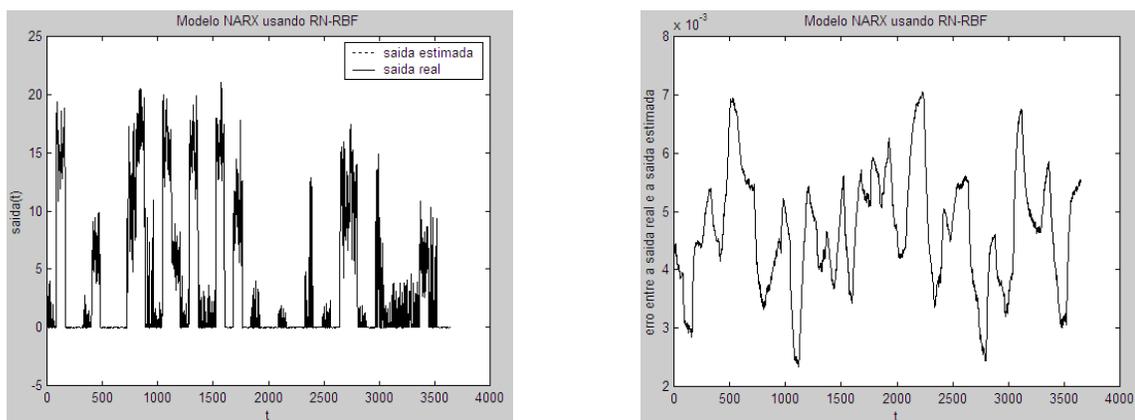
FIGURA 4.70 - SAÍDA $y_4(t)$ PARA 5 CENTROS DA RN RBF.FIGURA 4.71 - SAÍDA $y_1(t)$ PARA 6 CENTROS DA RN RBF.FIGURA 4.72 - SAÍDA $y_2(t)$ PARA 6 CENTROS DA RN RBF.

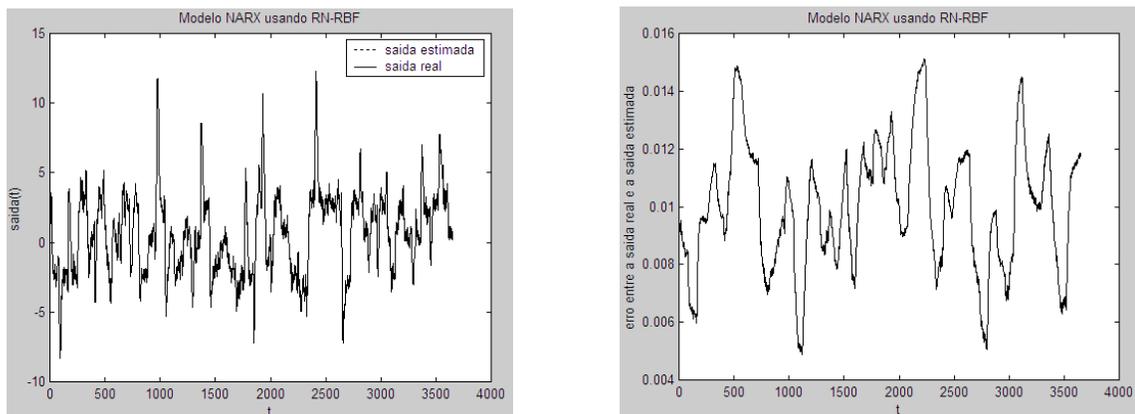
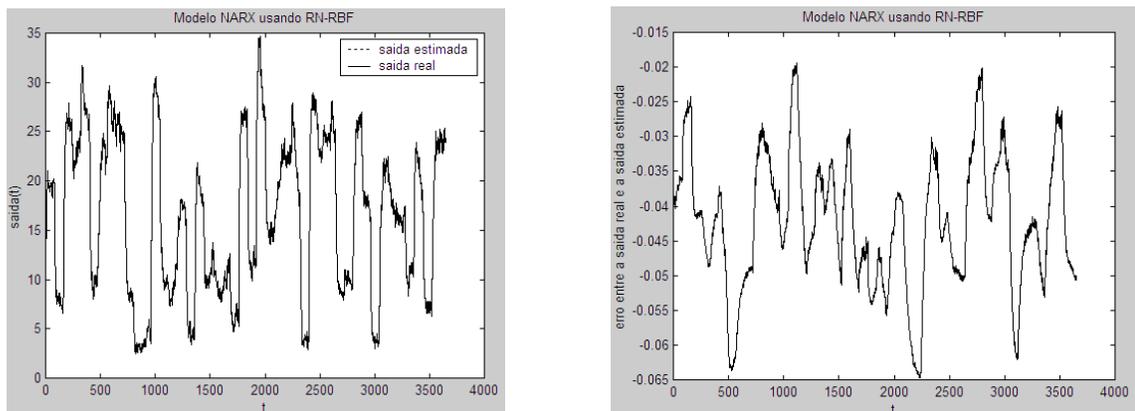
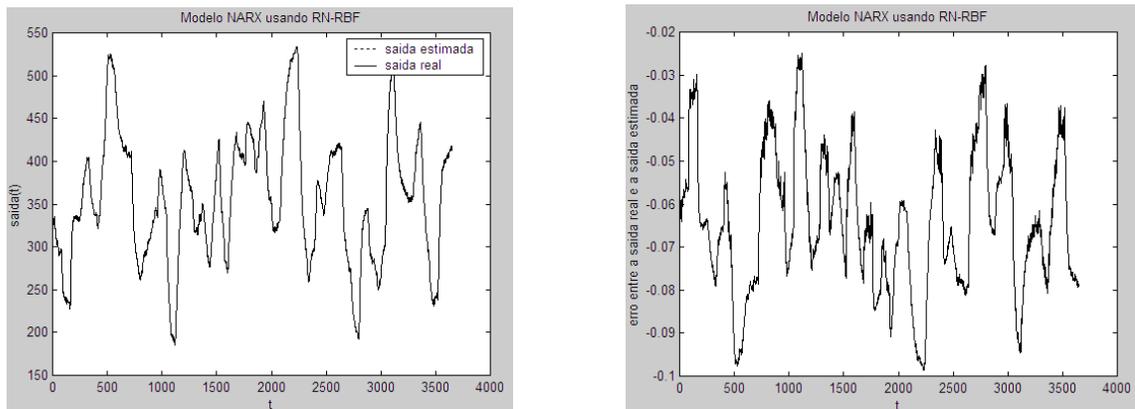
FIGURA 4.73 - SAÍDA $y_3(t)$ PARA 6 CENTROS DA RN RBF.FIGURA 4.74 - SAÍDA $y_4(t)$ PARA 6 CENTROS DA RN RBF.FIGURA 4.75 - SAÍDA $y_1(t)$ PARA 7 CENTROS DA RN RBF.

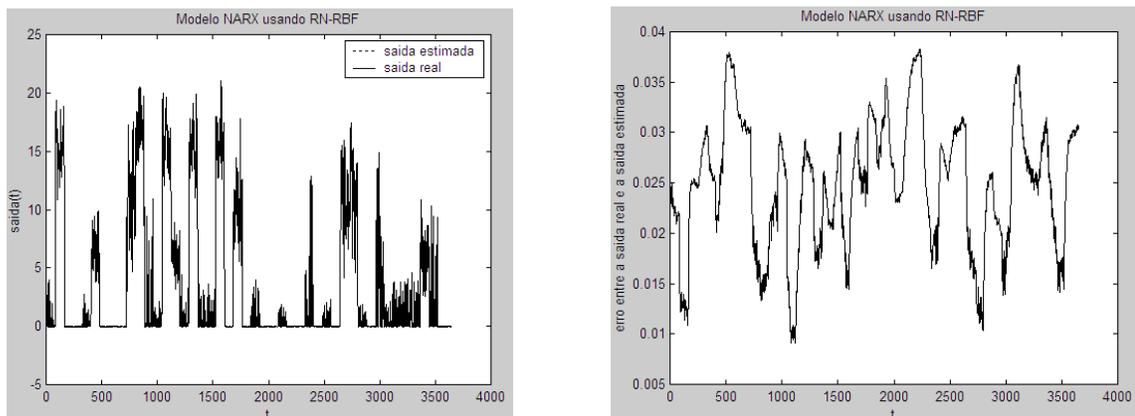
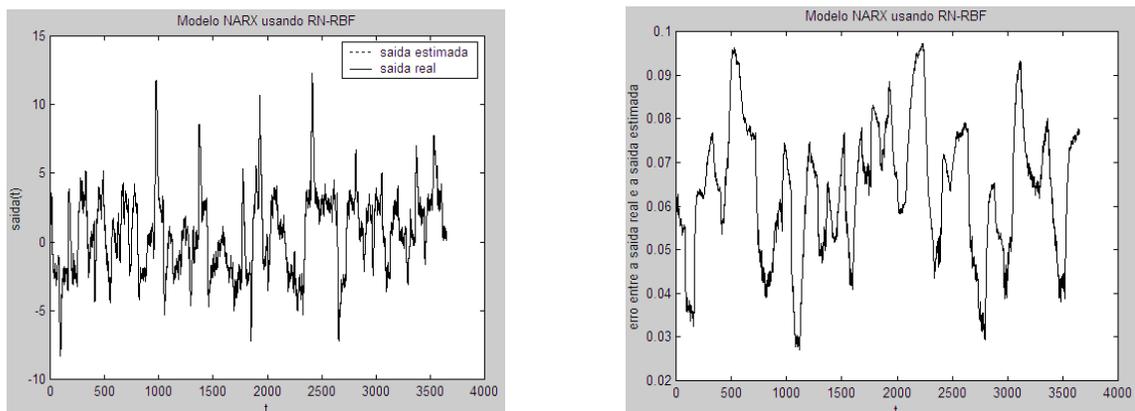
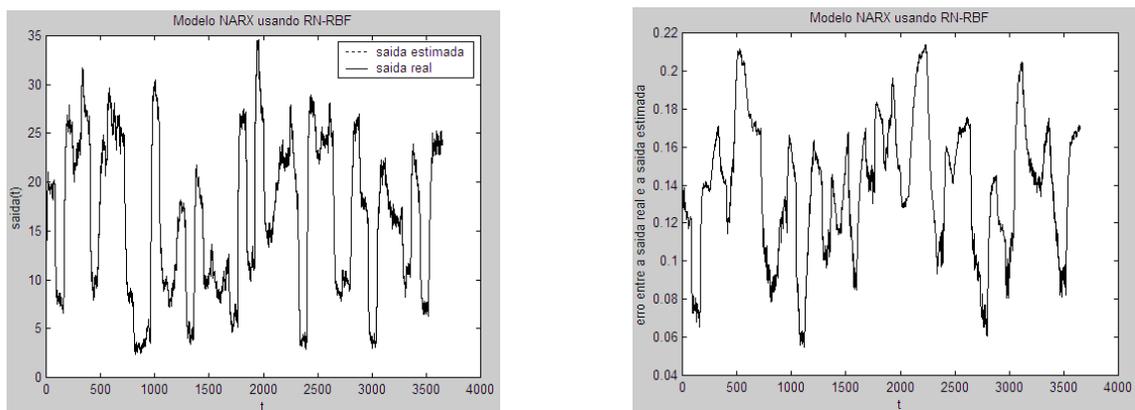
FIGURA 4.76 - SAÍDA $y_2(t)$ PARA 7 CENTROS DA RN RBF.FIGURA 4.77 - SAÍDA $y_3(t)$ PARA 7 CENTROS DA RN RBF.FIGURA 4.78 - SAÍDA $y_4(t)$ PARA 7 CENTROS DA RN RBF.

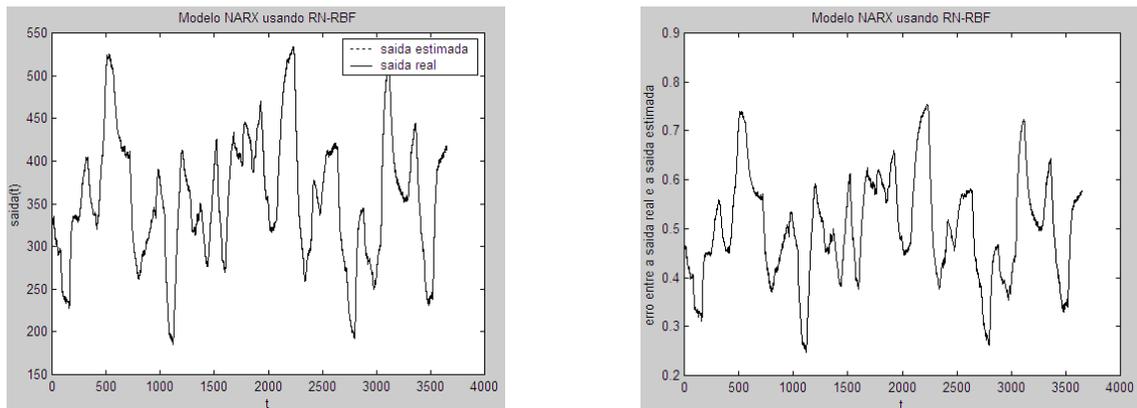
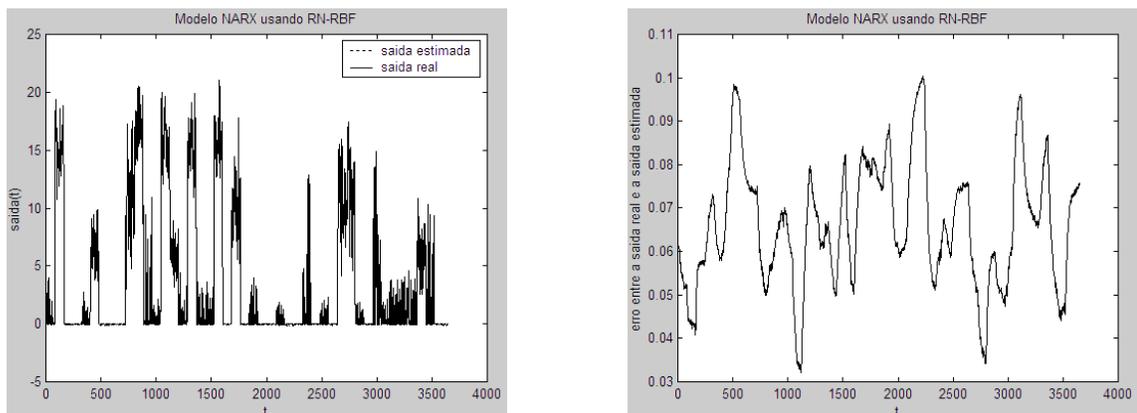
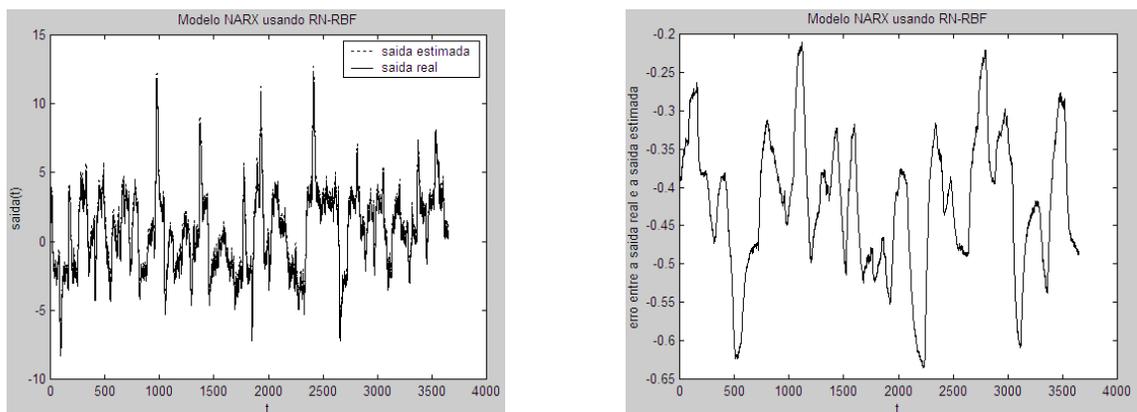
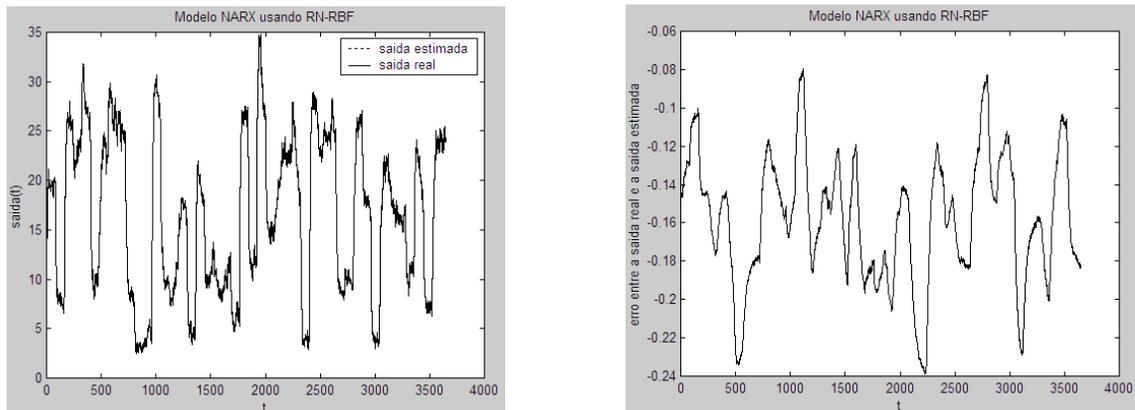
FIGURA 4.79 - SAÍDA $y_1(t)$ PARA 8 CENTROS DA RN RBF.FIGURA 4.80 - SAÍDA $y_2(t)$ PARA 8 CENTROS DA RN RBF.FIGURA 4.81 - SAÍDA $y_3(t)$ PARA 8 CENTROS DA RN RBF.

FIGURA 4.82 - SAÍDA $y_4(t)$ PARA 8 CENTROS DA RN RBF.

Nas figuras 4.63 a 4.82 nota-se que as RN RBF obtiveram um resultado satisfatório em todas as configurações de 3 a 8 centros. Ao analisar o gráfico do erro dessas figuras nota-se um valor aceitável, o que demonstra a eficiência da técnica. Ao ver essa eficiência percebe-se que a configuração de 4 centros obteve uma estimativa melhor que todas as outras configurações da RN RBF.

5 CONCLUSÃO E FUTURA PESQUISA

Nesta dissertação foi apresentado um estudo comparativo de duas técnicas de RNs na identificação de sistemas de potência não-lineares e multivariáveis, obtendo resultados promissores na previsão do comportamento dinâmico desses sistemas.

Com base nos resultados apresentados no capítulo 4, nota-se que as RNs são eficientes na identificação de sistemas não-lineares com várias entradas e saídas. Além disso, as RNs são capazes de prever o comportamento do sistema com um passo à frente sem erros significativos. No entanto, estes resultados são dependentes do projeto da RN, ou seja, números de entradas da RN, número de neurônios na camada oculta, método de treinamento usado e também que dados são utilizados nas fases de estimação e validação do modelo neural.

Outro comentário em relação aos resultados é a qualidade superior do projeto adotado para a rede RBF sobre a rede MLP na identificação dos sistemas para os dois casos apresentados no capítulo 4. Isto pode ser observado nas tabelas 4.1 e 4.2 onde foi apresentado uma análise de desempenho da MLP e também da RBF. Nota-se que a RBF apresentou um RMSE menor que o da MLP.

Entretanto, as tabelas 4.3 e 4.4 apresentam as redes RBF e MLP com RMSE similares, mostrando que com uma maior quantidade de amostras, que é o caso 2, as redes MLP tornam-se mais eficientes e as RBF perdem sua qualidade de aproximação porque as redes do tipo RBF são aproximadores locais e por isso não necessitam de grandes quantidades de dados para o seu treinamento e funcionamento. No caso das redes do tipo MLP, quanto maior for o número de dados qualitativos previamente conhecidos dos sistemas, melhor será o treinamento e a validação dessas redes, pois são aproximadores globais de mapeamento não-lineares.

No primeiro caso, no qual possuem-se poucos dados amostrados, a rede MLP teve uma dificuldade para prever e acompanhar o sinal de saída, enquanto a rede RBF previu e acompanhou o sinal de saída quase sem erros, conforme apresentado nas tabelas 4.1 a 4.4.

Quanto à implementação computacional das duas técnicas, a rede RBF é mais simples e com treinamento mais rápido. A MLP, por sua vez, possui um algoritmo de treinamento (não-linear) mais complexo, maior e mais lento para o processamento dos dados amostrado.

Ainda nesta questão, a RN RBF utilizou um treinamento baseado no método de k -médias, que é um método de agrupamento, aplicando logo em seguida a pseudo-inversa, pois, após o agrupamento, obtém-se um problema linear nos parâmetros, tornando o problema de identificação de um sistema não-linear mais simples e de rápido processamento, o que torna essa técnica de RN mais rápida e eficiente.

Na RN MLP, em contra-partida, o problema de treinamento é um problema não-linear e o método usado, o método de Levenberg-Marquardt, apesar de ser eficiente, consome muita memória do computador.

Desse modo, conclui-se que a RN RBF mostrou-se ser uma técnica melhor que a MLP ao tratar de identificação de sistemas não-lineares multivariáveis, porque as estimativas obtidas pela RBF foram mais satisfatórias que as da MLP. Além disso, a RN RBF obteve um processamento dos dados mais rápido em relação a MLP, possuindo um custo computacional mais econômico.

Portanto, os resultados validaram a aplicabilidade das redes neurais em sistemas nos quais ocorrem acoplamentos de sinais tanto de entrada quanto de saída, com isso atingindo o objetivo deste estudo.

Para futura pesquisa, o autor deseja testar novos métodos de treinamento para a RN MLP e RBF usando métodos híbridos dos utilizados nesta dissertação com algoritmos evolutivos e da inteligência coletiva para problemas de identificação multivariável.

REFERÊNCIAS BIBLIOGRÁFICAS

- Aguirre, L. A. (1994). Some remarks on structure of nonlinear systems. *International Journal of Bifurcation and Chaos*, vol. 4, n° 6, p. 1707-1714.
- Aguirre, L. A. (1995) A nonlinear correlation function for selecting the delay time in dynamical reconstructions. *Physics Letters A*, vol. 203, n° 2, p. 88-94.
- Aguirre, L. A.; Billings, S. A. (1994). Validating identified nonlinear models with chaotic dynamics. *International Journal of Bifurcation and Chaos*, vol. 4, p. 109-125.
- Aguirre, L. A.; Billings, S. A. (1995). Improved structure selection based on term clustering. *International Journal of Control*, vol. 62, n° 3, p. 569-587.
- Aguirre, L. A.; Mendes, E. M. A. M. (1995). The least squares padé method for model reduction of multivariable systems. *International Journal of Systems Science*, vol. 26, n° 4, p. 819-839.
- Aguirre, L. A.; Mendes, E. M. A. M. (1996). Global nonlinear polynomial models: Structure, term clusters and fixed points. *International Journal of Bifurcation and Chaos*, vol. 6, n° 2, p. 279-294.
- Aguirre, L. A.; Rodrigues, G. G.; Mendes, E. M. A. M. (1997). Nonlinear identification and cluster analysis of chaotic attractors from a real implementation of Chua's circuit. *International Journal of Bifurcation and Chaos*, vol. 7, n° 6, p.1411-1423.
- Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, vol. 19, p. 716-723.
- Amjady, N. (2001). Short-term hourly load forecasting using time-series modeling with peak load estimation capability. *IEEE Transactions on Power Systems*, vol. 16, n° 3, p. 498-505.
- Ayoubi, M.; Isermann, R. (1997). Neuro-fuzzy systems for diagnosis, fuzzy sets and systems, vol. 89, p. 289-307.
- Billings, S. A. (1980). Identification of nonlinear systems – A survey. *IEE Proceedings, Part D*, vol. 127, p. 272-285.
- Billings, S. A.; Chen, S.; Korenberg, M. J. (1989). Identification of MIMO non-linear systems using a forward-regression orthogonal estimator. *International Journal of Control*, vol. 49, p. 2157-2189.
- Billings, S. A.; Tao, Q. H.(1991). Model validation tests for nonlinear signal processing applications. *International Journal of Control*, vol. 54, n° 1, p. 157–194.
- Billings, S. A.; Voon, W. S. F. (1983). Structure detection and model validity tests in the identification of nonlinear systems. *IEE Proceedings, Part D*, vol. 127, p. 193-199.

- Bostanci, M.; Koplowitz, J.; Taylor, C. W. (1997). Identification of power system load dynamics using artificial neural networks. *IEEE Transactions on Power Systems*, vol. 12 n° 4, p 1468 – 1473.
- Bouthiba, Tahar. (2003). Artificial neural network-based fault location in EHV transmission lines. *International Conference on Eletrical and Electronics Engineering*. Bursa, Turkey.
- Box, G. E. P.; Jenkins, G. M. (1976). *Time series analysis: Forecasting and control*. 2nd edition, Holden-Day.
- Bretas, A. S.; Phadke, A. G. (2003). Artificial neural networks in power system restoration. *IEEE Transactions on Power Delivery*, vol. 18, n° 4, p. 1181-1186.
- Bryson, A. E.; Ho, Y. (1969). *Applied optimal control*. Blaisdell. Blaisdell, Waltham, MA, USA.
- Cardoso Jr., G.; Rolim, J. G.; Zürn, H. H. (2004). Application of neural-network modules to electric power system fault section estimation. *IEEE Transactions on Power Delivery*, vol. 19, n° 3, p. 1034-1041.
- Casdagli, M. (1989). Nonlinear prediction of chaotic time series. *Physica D*, vol. 35 p. 335-356.
- Castro, F. C. C.; Castro, M. C. F. (2000). *Redes neurais artificiais. Notas de aula*. Pontifícia Universidade Católica do Rio Grande do Sul, Porto Alegre, RS.
- Chen, S.; Billings, S. A. (1989). Representation of non-linear systems: the NARMAX model. *International Journal of Control*, vol. 49, p. 1013-1032.
- Chen, S.; Billings, S. A.; Luo, W. (1989). Orthogonal least squares methods and their application to non-linear system identification. *International Journal of Control*, vol. 50, p. 1873-1896.
- Chen, Y. M.; Lee, M. L. (2001). Neural networks-based scheme for system failure detection and diagnosis, *Mathematics and Computers in Simulation*, vol. 58, p. 101-109.
- Chow, T. W. S.; Leung C.-T. (1996). Nonlinear autoregressive integrated neural network model for short-term load forecasting. *IEE Proc.-Gener. Transm. Distrib*, vol. 143, n° 5, p. 500-506.
- Çinar, A. (1995) Nonlinear time series models for multivariable dynamic processes. *Chemometrics and Intelligent Laboratory Systems*, p. 147-158.
- Coelho, L. S. (2000). *Identificação e controle de processos multivariáveis via metodologias avançadas e inteligência computacional*. Dissertação de doutorado, Universidade Federal de Santa Catarina, Florianópolis, UFSC.

- Coelho, L. S. (2004). Introdução às redes neurais artificiais. Notas de aula da disciplina Sistemas Inteligentes, PPGEPS. Curitiba, Pontifícia Universidade Católica do Paraná, Curitiba, PR.
- Cybenko, G. (1989). Approximation by superpositions of a sigmoid function. *Mathematics of Control, Signals, and Systems*, vol. 2, p. 303-314.
- Cybenko, G V. (1996). Neural networks in computational science and engineering. *IEEE Computational Science and Engineering*, Vol. 3. p. 36-42.
- Daisy (2005): Database for the identification of systems, Departamento de Engenharia elétrica, K.U.Leuven, Belgium, URL: <http://www.esat.kuleuven.ac.be/sista/daisy/>, visitado em março de 2005.
- Davis, M. H. A.; Vinter, R. B. (1985). *Stochastic modeling and control*. London: Chapman and Hall.
- Duarte, A. N. (2003). Tratamento de eventos em redes elétricas: uma ferramenta. Dissertação de mestrado, Universidade Federal de Campina Grande. Campina Grande, PB.
- Elbuluk, M. E.; Chan, W. H.; Husain, I. (1998). Neural network controllers for power factor correction of AC/DC switching converters. *IEEE IAS Annual Conference*, St. Louis, MO.
- Elsner, J. B. (1992). Predicting time series using a neural network as a method of distinguishing chaos from noise. *Journal of Physics, Mathematical Gen.*, vol. 25, p. 843-850.
- Fabri, J. A. Redes Neurais Artificiais. Apostila, disponível em: <http://users.femanet.com.br/~fabri/RNAs.htm> acesso em 26/04/2004.
- Fernandes, C. M. A. P. (2003). Redução de perdas técnicas em redes primárias de distribuição de energia elétrica. Tese de doutorado, Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e de Computação, Campinas, SP.
- Fiedler-Ferrara, N.; Prado, C. P. C. (1994). *Caos : Uma introdução*. São Paulo, Edgara Blücher.
- Fischler, Martin A.; Firschein, Oscar (1987). *Intelligence: the eye, the brain, and the computer*. Massachusetts: Addison-Wesley.
- Foss, B. A.; Johansen, T. A. (1992) A NARMAX model representation for adaptive control based on local models. *Modeling, Identification and Control*, vol. 13, p. 25- 39.
- Garcez, A. S. d'Avila; Zaverucha, G.; Silva, V. N. A. L.. (1997). Applying the connectionist inductive learning and logic programming system to power system diagnosis. *ICNN97*, Houston, Texas, TX, USA.
- Giarratano; R. (1993). *Expert systems, principles and programming*. 2nd Edition. PWS Publishing Company.

- Gillard, D. M.; Bollinger, K. E. (1996). Neural network identification of power system transfer functions. *IEEE Transactions on Energy Conversion*, vol. 11, n° 1, p. 104 - 110.
- Giovanini, R.; Coury, D. V. (1999). Classificação rápida de faltas em sistemas elétricos utilizando redes neurais artificiais. IV Congresso Brasileiro de Redes Neurais, São José dos Campos, SP.
- Gooijer, J. G.; Abraham, B.; Gould, A.; Robinson, L. (1985). Methods for determining the order of an autoregressive-moving average process: a survey. *International Statistical Review*, vol. 53, p. 301-329.
- Gouesbet, G.; Letellier, C. (1994). Global vector field reconstruction by using a multivariate polynomial L2 approximation on nets. *Physical Review E*, vol. 49, n° 6, p. 4955-4972.
- Haber, R. (1985). Nonlinearity tests for dynamic processes. *IFAC Identification and System Parameter Estimation*. York, UK, p. 409-413.
- Haber, R.; Unbehauen, H. (1990). Structure identification of nonlinear dynamic systems. *Automatica*, vol. 26, n° 4, p. 651-677.
- Haykin, S. (2001). *Redes neurais: princípios e prática*. Trad. Paulo Martins Engel. 2ª ed. Porto Alegre: Bookman, RS.
- Hiyama, T.; Tokieda, M.; Hubbi, W. ; Andou, H. (1997). Artificial neural network based dynamic load modeling. *IEEE Transaction on Power Systems*, vol. 12, n° 4, p. 1576-1583.
- Hippert, H. S.; Pedreira, C. E.; Souza, R. C. (2001). Neural networks for short-term load forecasting: a review and evaluation. *IEEE Transactions on Power Systems*, vol. 16, n° 1, p. 44-55.
- Huang, Y. C. (2002). Abductive reasoning network based diagnosis system for fault section estimation in power systems. *IEEE Transactions on Power Delivery*, vol. 17, n° 2, p. 369-374.
- Ibrahim, W. R. A.; Morcos, M. M. (2003). A power quality perspective to system operational diagnosis using fuzzy logic and adaptive techniques. *IEEE Transactions on Power Delivery*, vol. 18, n° 3, p. 903 - 909.
- Jácome, C. R. F. (1996). *Uso de conhecimento prévio na identificação de modelos polinomiais NARMAX*. Dissertação de mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Minas Gerais, Universidade Federal de Minas Gerais, MG.
- Jácome, C.R.F.; Aguirre, L.A. (1996). Uso de técnicas de identificação não-linear na modelagem de válvulas para controle de processos. I Congresso Mineiro de Automação e IV Simpósio Regional de Instrumentação, Belo Horizonte, MG, p. 37-45.
- Jang, H.; Kim, K. (1994). Identification of loudspeaker nonlinearities using the NARMAX modeling technique. *Journal of Audio Engineering Society*. vol. 42, n° 1/2, p. 50-59.

Jorge, D. C.; Coury, D. V.; Carvalho, A. C. P. L. F. (1999). Localização de faltas em linhas de transmissão de energia elétrica utilizando reconhecimento de padrões. IV Congresso Brasileiro de Redes Neurais, São José dos Campos, SP.

Kashyap, R. L. (1977). A bayesian comparison of different classes of dynamical models using empirical data. IEEE Transactions on Automatic Control, vol. 22, n° 5, p. 715-727.

Korenberg, M.; Billings, S. A., Liu, Y. P.; Mcilroy, P. J. (1988). Orthogonal parameter estimation algorithm for non-linear stochastic systems. International Journal of Control, vol. 48, n° 1, p. 193-210.

Kumar, D. M. V.; Srivastava, S. C. (1999). Power system state forecasting using artificial neural networks. Electric Machines and Power Systems, vol. 27, p. 653-664.

Le Cun, Y. (1985). A learning procedure for asymmetric threshold network. Proceedings of Cognitiva 85, Paris, France, p. 599-604.

Leontaritis, I. J.; Billings, S. A. (1987). Model selection and validation methods for nonlinear systems. International Journal of Control, vol. 45, n° 1, p. 311-341.

Letellier, C.; Gouesbet, G.(1995). Topological characterization of a system with high-order symmetries. Physical Review E, vol. 52, n° 5, p. 4754-4761.

Lin, W. M.; Yang, C. D.; Lin, J. H.; Tsay, M. T. (2001). A fault classification method by RBF neural network with OLS learning procedure. IEEE Transactions on Power Delivery, vol. 16, n° 4, p. 473-477.

Lin, W. M.; Zhan, T. S.; Yang, C. D. (2003). Distribution system reliability worth analysis with the customer cost model based on RBF neural network. IEEE Transactions on Power Delivery, vol. 18, n° 3, p. 1015-1021.

Ljung, L.(1987). System identification - theory for the user. Prentice-Hall International, New Jersey, USA.

Luo, W.; Billings, S. A., Tsang, K. M. (1994). On-line structures detection and parameter estimation with exponential windowing for nonlinear systems. Research Report n° 503, Department of Automatic Control and Systems Engineering, University of Sheffield.

Lu, S.; Hogg, B.W. (2000). Dynamic nonlinear modeling of power plant by physical principles and neural networks. Electrical Power and Energy Systems, vol. 22, p. 67-78.

Magalhães, M. H. (2004). Redes neurais, metodologias de agrupamento e combinação de previsores aplicados à previsão de vazões naturais. Dissertação de mestrado, Faculdade de Engenharia Elétrica e de Computação da Universidade Estadual de Campinas, SP.

Maia, W. U. L. A. (1998). Sistema integrado de operação e diagnóstico de falhas para sistemas de energia elétrica - S O D F. Dissertação de mestrado, Universidade Federal de Santa Catarina, Florianópolis, SC.

- Masri, S.F.; Chassiakos, A.G.; Caughey, T. K. (1993). Identification of nonlinear dynamic systems using neural networks. *Transactions ASME - Journal of Applied Mechanics*, vol. 60, p. 123-133.
- Mazon, A. J.; Gracia, I. Z., J.; Sagastabeitia, K. J.; Saenz, J. R. (2001). Selecting ANN structures to find transmission faults. *IEEE Computer Applications in Power*, vol. 14, n^o 3, p. 44-48.
- McGhee, J.; Henderson, I. A.; Baird, A. (1997). Neural networks applied for the identification and fault diagnosis of process valves and actuators. *Measurement*, vol. 20, p. 267-275.
- Mees, A. (1993). Parsimonious dynamical reconstruction. *International Journal of Bifurcation and Chaos*, vol. 3, n^o 3, p. 669-675.
- Mendel, J. M.; McLaren, R. W. Adaptive, learning, and pattern recognition systems: theory and applications, chapter reinforcement-learning control and pattern recognition systems. New York: Academic Press, 70.
- Menezes, C. C. (2004). Redes neurais. Disponível em: http://www.inf.ufrgs.br/procpar/disc/cmp135/trabs/012/T1/redes_neurais/redes_neurais.htm. Acesso em: 10 Jun.
- Murakami, A.; Campos, A. L.; Costa, C. S.; Borges, R. B. (2004). Redes neurais artificiais (RNAs). Disponível em: <http://www.terravista.pt/Meco/5704/RedesNeurais.htm> Acesso em: 02 mai.
- Neto, L. B. (1997). Sistema híbrido de apoio a decisão para detecção e diagnóstico de falhas em redes elétricas. Dissertação de mestrado, Pontifícia Universidade Católica do Rio de Janeiro, RJ.
- Noshiro, M.; Furuya, M.; Linkens, D.; Goode, K. (1993). Nonlinear identification of PCO2 control system in man. *Computer Methods and Programs in Biomedicine*, p. 189-202.
- Oleskovicz, M.; Coury, D. V.; Aggarwal, R. K. (2000). Redes neurais artificiais aplicadas à classificação rápida de faltas em sistemas elétricos de potência. *SBA Controle & Automação*, Vol. 11, n^o 2, p. 160-168.
- Parker, D. (1985). Learning logic: casting the cortex of the human brain in silicon Technical Report, Center for Computational Research in Economics and Management Science, MIT, USA.
- Park, J.-W.; Harley, R. G.; Venayagamoorthy, G. K. (2002). Comparison of MLP and RBF neural networks using deviation signals for on-line identification of a synchronous generator. *IEEE Power Engineering Society Winter Meeting*, New York, USA.
- Pasand, M. S.; Zadeh, H. K. (2003). Transmission line fault detection & phase selection using ANN. *International Conference on Power Systems Transients – IPST*, New Orleans, USA.

- Peyton-Jones, J. C.; Billings, S. A. (1989). Recursive algorithm for computing the frequency response of a class of non-linear difference equation models. *International Journal of Control*, vol. 50, p. 1925-1940.
- Pröl, T.; Karim, M. N. (1994). Model-predictive PH control using real-time NARX approach. *AICHE Journal*, vol. 40, n° 2, p. 269-282.
- Ramaswamy, S. V.; Kiran, B.; Kashyap, K. Harish; Shenoy, U. J. (2003). Classification of Power System Transients using Wavelet transforms and Probabilistic Neural Networks. *TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region*.
- Rengaswamy, R.; Mylaraswamy, D.; Arzén, K. E.; Venkatasubramanian, V. (2001). A comparison of model-based and neural network-based diagnostic methods. *Engineering Applications of Artificial Intelligence*, vol. 14, p. 805-818.
- Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, vol. 14, p. 467-471.
- Rodrigues, G. G. (1996). Identificação de sistemas dinâmicos não-lineares utilizando modelos NARMAX polinomiais - aplicação a sistemas reais. Dissertação de mestrado, Programa de Pós-Graduação em Engenharia Elétrica, Universidade Federal de Minas Gerais, MG.
- Rodrigues, G.G.; Aguirre, L.A.; Freitas, U.S.; Pena, R.T. (1996). Identificação de um forno elétrico não-linear utilizando modelos NARMAX polinomiais. *XI Congresso Brasileiro de Automática*, São Paulo, SP, p. 953-958.
- Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychol. Rev.*, n° 65, p. 386-408.
- Rumelhart, David E.; McClelland, James (1986) *Parallel distributed processing*. Bradford Book, MIT Press, Massachusetts, USA.
- Salat, R.; Osowski, S. (2004). Accurate fault location in the power transmission line using support vector machine approach. *IEEE Transactions on Power Systems*, vol. 19, n° 2, p. 979-986.
- Schwarz, M. W.; Cowan, W. B.; Beatty, J. C. (1987). An experimental comparison of rgb, yiq, lab, hsv and opponent color models. *ACM Transactions on Graphics* vol. 6, n° 2, p. 123-158.
- Senjyu, T.; Takara, H.; Uezato, K.; Funabashi, T. (2002). One-hour-ahead load forecasting using neural network. *IEEE Transactions on Power Systems*, vol. 17, n° 1, p. 113-118.
- Sidhu, T. S.; Mital, L.; Sachdev, M. S. (2004). A comprehensive analysis of an artificial neural-network-based fault direction discriminator. *IEEE Transactions on Power Delivery*, vol. 19, n° 3, p. 1042-1048.
- Sjoberg, J. (1995). Non-linear system identification with neural networks. Electrical Engineer Department, Linköping University. Linköping, Sweden.

- Souza, J. C. S.; Silva, A. M. L.; Silva, A. P. A. (1997). Data visualization and identification of anomalies in power system state estimation using artificial neural networks. *IEE Proc.-Gener. Transm. Distrib.*, vol. 144, n° 5, p. 445-455.
- Stankovic', A. M.; Saric', A. T.; Sevic, M. M. (2003). Identification of nonparametric dynamic power system equivalents with artificial neural networks. *IEEE Transactions on Power Systems*, vol. 18, n° 4, p. 1478-1486.
- Strang, G. (1989). Wavelets and dilation equations: a brief introduction. *SIAM Review*. vol. 31, p. 614-627.
- Swarup, K. S.; Corthis, P. B. (2002). ANN approach assesses system security. *IEEE Computer Applications in Power*, vol. 15, n° 3, p. 32-38.
- Sycara, K. (1988). Using case-based reasoning for plan adaptation and repair. In *Case Based Reasoning Workshop, DARPA*. Morgan Kaufmann.
- Tan, S.; Hao, J.; Vandewalle, J. (1995). Efficient identification of RBF neural net models for nonlinear discrete-time multivariable dynamical systems. *Neurocomputing*, vol. 9, p. 11-26.
- Tan, S. C.; Lim, C. P. (2004). Application of an adaptive neural network with symbolic rule extraction to fault detection and diagnosis in a power generation plant. *IEEE Transactions on Energy Conversion*, vol. 19, n° 2, p. 369-377.
- Ting, K. L.; Berger, C. S.; Conlon, M. F. (1995). Neural network approach to signal modelling in power systems. *IEE Proc.-Control Theory Appl.*, Vol. 14, n° 4, p. 257-264.
- Thouverez, F.; Jezequel, L. (1996) Identification of NARMAX models on a modal base. *Journal of Sound and Vibration*, vol. 189, n° 2, p. 193-213.
- Tafner, M. A. (1996). Reconhecimento de palavras isoladas usando redes neurais artificiais. *Dissertação de mestrado, Universidade Federal de Santa Catarina. Florianópolis, SC.*
- Todesco, J. L. (1995). Reconhecimento de padrões usando rede neural artificial com uma função de base radial: uma aplicação na classificação de cromossomos humanos. *Dissertação de doutorado, Universidade Federal de Santa Catarina. Florianópolis, SC.*
- Venayagamoorthy, G. K.; Harley, R. G. (2001). A continually online trained neurocontroller for excitation and turbine control of a turbogenerator. *IEEE Transactions on Energy Conversion*, vol. 16, n° 3, p. 261-269.
- Vieira, L. A. (1996). Detecção e acomodação de falhas em sistemas elétricos usando redes neurais artificiais. *Dissertação de mestrado, Curso de Mestrado em Engenharia Elétrica, Centro Tecnológico, Universidade Federal do Pará, Belém, PA.*
- Wang, L.; Cluett, W. R. (1996). Use of press residuals in dynamic system identification. *Automatica*, vol. 32, n° 5, p. 781-784.

Werbos, B. (1974). Regression: new tools for prediction and analysis in the behavioral sciences. PhD thesis, Harvard University, USA.

Widrow, B.; Hoff, M. E. (1960). Adaptive switching circuits. Institute of Radio Engineers. Western Electronic Show and Convention.

Widrow, B.; Hoff, M. E. (1988). Adaptive Switching Circuits. Neurocomputing: foundations of research. Cambridge, MA, USA. p. 123-134.

Zhu, Q. M.; Billings, S. A. (1996). Fast orthogonal identification of nonlinear stochastic models and radial basis function neural networks. International Journal of Control, vol. 64, n° 5, p. 871-886.