

Denis Fabrício Sousa de Sá

MODELOS BASEADOS EM REDES NEURAS ARTIFICIAIS COM
APLICAÇÃO EM CONTROLE INDIRETO DE TEMPERATURA

São Luis
2015

Universidade Federal do Maranhão
Pós Graduação em Engenharia Elétrica

Denis Fabrício Sousa de Sá

MODELOS BASEADOS EM REDES NEURAS ARTIFICIAIS COM APLICAÇÃO EM
CONTROLE INDIRETO DE TEMPERATURA

Dissertação de mestrado apresentada à Pós Graduação de Engenharia Elétrica como parte dos requisitos exigidos para a obtenção do título de Mestre em Engenharia Elétrica. Área de concentração: Automação e Controle.

Orientador: João Viana da Fonseca Neto

São Luis
2015

Sá, Denis Fabrício Sousa de.

Modelos baseados em redes neurais artificiais com aplicação em controle indireto de temperatura/ Denis Fabrício Sousa de Sá. – São Luís, 2015.

141 f.

Impresso por computador (fotocópia).

Orientador: João Viana da Fonseca Neto.

Dissertação (Mestrado) – Universidade Federal do Maranhão, Programa de Pós-Graduação em Engenharia Elétrica, 2015.

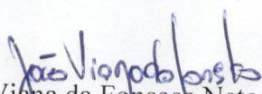
1. Controle adaptativo. 2. Controle Proporcional-Integral-Derivativo. 3. Redes Neurais Artificiais. 4. Treinamento Online de RNA. 5. Identificação de Sistemas. 6. Sistemas Térmicos. 7. Medição Indireta. 8. Célula Peltier. 9. DLQR. 10. Sistemas Embarcados. I. Título.

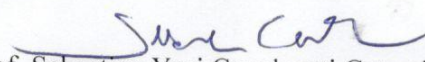
CDU 621.3.052.4

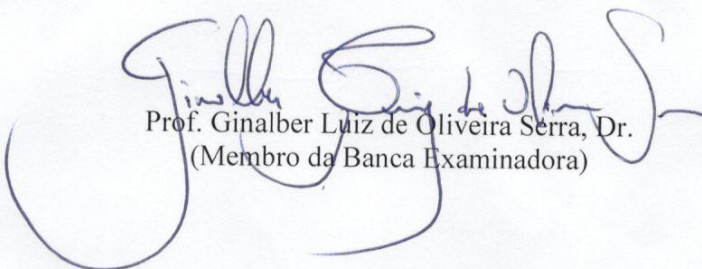
**MODELOS BASEADOS EM REDES NEURAIS ARTIFICIAIS COM
APLICAÇÃO EM CONTROLE INDIRETO DE TEMPERATURA**

Denis Fabrício Sousa de Sá

Dissertação aprovada em 10 de abril de 2015.


Prof. João Viana da Fonseca Neto, Dr.
(Orientador)


Prof. Sebastian Yuri Cavalcanti Catunda, Dr.
(Membro da Banca Examinadora)


Prof. Ginalber Luiz de Oliveira Serra, Dr.
(Membro da Banca Examinadora)

DEDICO ESTE TRABALHO AOS FA-
MILIARES E AMIGOS QUE CONTRI-
BUÍRAM COM MEU DESENVOLVI-
MENTO PESSOAL E PROFISSIONAL.

Agradecimentos

Agradeço,

a Deus e ao Prof. Dr. João viana da Fonseca Neto pelos cinco anos de orientação, pela oportunidade para desenvolvimento deste trabalho, pela contribuição na meu desenvolvimento profissional.

aos meus familiares, em especial, minha mãe, Núbia Gorette Pereira de Sousa, irmã, Dayana J. Sousa de Sá, a nova irmã Larissa, a minha namorada Liliane, que me proporcionaram um condições necessários, como estrutura familiar, alegrias, educação e suporte, para que eu pudesse persistir e seguir minha carreira escolhida.

aos colegas de laboratório mais próximos: Gustavo, Marcio, Luisa, Sarah, Watson, Cleydson, Trabulsi, Larissa, Moises, Ernesto, Batista, Madson e Vinícius a boa convivência e as trocas de experiências.

aos professores da Universidade Federal do Maranhão (UFMA) de Graduação e da Pós Graduação que acrescentaram bastante aos meus conhecimentos.

a UFMA e a PPGEE/UFMA, pela estrutura e oportunidade, juntamente a CAPES e todos que de alguma forma contribuíram com este estudo.

O único lugar onde o sucesso vem antes do trabalho
é no dicionário.

Albert Einstein

Resumo

A representação de sistemas dinâmicos ou plantas por meio de modelos matemáticos ocupa uma posição relevante no projeto de sistemas de controle, permitindo que o projetista avalie o desempenho dos controladores durante a fase de desenvolvimento do projeto. Estes modelos também são utilizados para resolver o problema da dificuldade ou impossibilidade da inserção de sensores em plantas para medição de variáveis controladas, onde os modelos viabilizam a medição não invasiva destas variáveis, fornecendo uma alternativa para realização do controle indireto adaptativo e ótimo de um dado processo. Nesta dissertação apresenta-se o desenvolvimento de modelos propostos baseados em redes neurais artificiais recorrentes para o controle sensorless ou indireto da planta. Os modelos propostos estão associados com as abordagens de Identificação de Sistemas e de RNA's recorrentes. Os algoritmos desenvolvidos para o treinamento das RNAs são do tipo *Backpropagation* acelerado e RLS, que são comparados com estratégias e métodos clássicos, para obtenção *online* dos parâmetros do sistema de controle indireto de uma planta térmica, tendo como atuador uma célula Peltier. Para fins de avaliação de desempenho do sistema de controle indireto da planta, os modelos paramétricos e controladores digitais adaptativos do tipo PID e regulador linear quadrático (DLQR) são avaliados por meio de simulações híbridas, sendo os algoritmos dos controladores implementados em microcontroladores e a planta representada por modelos matemáticos. Os resultados apresentados são promissores, não só no sentido do desempenho do sistema de controle, mas também nos custos reduzidos para seu desenvolvimento, operação e flexibilidade de aplicação em outros sistemas dinâmicos.

Palavras-chave: Controle Adaptativo, Controle Proporcional-Integral-Derivativo (PID), Redes Neurais Artificiais, Treinamento on line de RNAs, Identificação de Sistemas, Sistema Térmico, Medição Indireta, Célula Peltier, Regulador Linear Quadrático Discreto (DLQR), Sistemas Embarcados.

Abstract

The representation of dynamic systems or plants via mathematical models occupies an important position in control system design that allow the performance evaluation of the controller during his development stage.

These models are also used as an alternative to solve the problem of the hardness or impracticability to install sensors that measure the controlled variables, the dynamic systems representations enable non-invasive measurement of these variables. As consequence the designer has an alternative way to perform adaptive and optimal sensorless control for a given process.

In this dissertation is presented a proposal for control systems schemas and algorithms, based on recurrent neural networks (ANN) and Box-Jenkins models, that are dedicated to sensorless or indirect control of dynamic systems.

The proposed models and algorithms are associated with the systems identification and recurrent ANN approaches. The algorithms developed for the AAN training are Backpropagation Accelerated and RLS types that are compared with classical methods and strategies to obtain it online parameters of indirect control of system for a thermal plant, where the actuator is Peltier cell.

The performance the parametric models of the plant and adaptive PID digital controllers and linear quadratic regulator (DLQR) that are the main elements of the sensorless temperature control system, are evaluated by means of hybrid simulations, where the algorithms implemented in micro controllers and the plant represented by mathematical models.

The performance results of the proposed sensorless control algorithms are promissory, not only, in terms of the control system performance, but also due to the reflexibility to deploy it in other dynamic systems.

Key-words: Adaptive Control, Proportional-Integral-Derivative Control (PID), Artificial Neural Networks, Online training of ANN's, Systems Identification, Thermal System, Indirect Measurement, Discrete Linear Quadratic Regulator (DLQR), Peltier cell and Embedded systems.

Lista de Figuras

2.1	Blocos conectados de Hammerstein.	24
2.2	Blocos interconectados de Wiener.	25
2.3	Neurônio Biológico.	28
2.4	Neurônio Artificial	28
2.5	RNA de uma camada	30
2.6	RNA com múltiplas camadas	30
2.7	RNA realimentada	31
2.8	Exemplo de uma RNA como estimador paramétrico	32
2.9	Exemplo de uma RNA como modelo	32
2.10	Modelo RNA/ARX obtido de (HAYKIN, 2001)	33
2.11	Aprendizado com professor (HAYKIN, 2001)	35
3.1	Arquitetura de uma rede multicamadas	44
4.1	Câmara térmica	51
4.2	Dados para estimação do núcleo do SMI, medidos de 4 em 4 segundos	54
4.3	Medição para verificação das diferenças de temperatura entre ar interno e objeto	54
4.4	Característica estática para o modelo do SMI	55
4.5	Estimação de primeira e segunda ordem para o modelo do SMI	56
4.6	Validação para o modelo do SMI	57
4.7	Característica estática real e estimada do sistema não linear	58
4.8	Estimação para temperatura no ar em relação a tensão na célula	60
4.9	Estimação para temperatura no objeto em relação a tensão na célula	61
4.10	Validação para os modelos referentes ao ar e objeto	61
4.11	Diagrama de Bode para verificação da influência do SMI	63
5.1	Estrutura RNA/ARX para controle	65
5.2	Estrutura RNA no Espaço de Estados	66
5.3	Temperatura do objeto, Erro e Pesos com treinamento BP - RNA/ARX para uma iteração por amostra	68
5.4	Temperatura no objeto, Autovalores e Energia com treinamento BP- RNA/ARX para uma iteração por amostra	69

5.5	Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/ARX para 10 iterações por amostra	69
5.6	Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/ARX para 10 iterações por amostra	70
5.7	Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/ARX com iterações por amostra limitado por condição de erro	70
5.8	Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/ARX com iterações por amostra limitado por condição de erro	71
5.9	Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/Espaço de Estados com uma iteração por amostra	72
5.10	Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/Espaço de Estados com uma iteração por amostra	72
5.11	Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/Espaço de Estados com 10 iterações por amostra	73
5.12	Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/Espaço de Estados com 10 iterações por amostra	73
5.13	Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/Espaço de Estados com número de iterações limitado por critério de erro	74
5.14	Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/Espaço de Estados com número de iterações limitado por critério de erro	74
5.15	Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/ARX com uma iteração por amostra	76
5.16	Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/ARX com uma iteração por amostra	76
5.17	Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/ARX com 10 iterações por amostra	77
5.18	Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/ARX com 10 iterações por amostra	77
5.19	Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/ARX com iterações por amostra limitadas por critério de erro	78
5.20	Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/ARX com iterações por amostra limitadas por critério de erro	78
5.21	Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/Espaço de Estados com uma iteração por amostra	79
5.22	Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/Espaço de Estados com uma iteração por amostra	80
5.23	Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/Espaço de Estados com 10 iterações por amostra	80
5.24	Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/Espaço de Estados com 10 iterações por amostra	81
5.25	Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro	81

5.26	Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro	82
5.27	Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/ARX com 1 iteração por amostra	83
5.28	Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/ARX com 1 iteração por amostra	84
5.29	Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/ARX com 10 iterações por amostra	84
5.30	Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/ARX com 10 iterações por amostra	85
5.31	Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/ARX com iterações por amostra limitada por critério de erro	85
5.32	Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/ARX com iterações por amostra limitada por critério de erro	86
5.33	Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/Espaço de Estados com 1 iteração por amostra	87
5.34	Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/Espaço de Estados com 1 iteração por amostra	87
5.35	Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/Espaço de Estados com 10 iterações por amostra	88
5.36	Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/Espaço de Estados com 10 iterações por amostra	88
5.37	Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro	89
5.38	Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro	89
6.1	Estrutura para aplicação do PID Digital adaptativo	94
6.2	Estrutural geral para controle digital, com base em (LANDAU, 1938)	95
6.3	Planta com PID digital para $\tau = 300$, $\zeta = 0.8$ e $T_{ref} = 70$	98
6.4	Estrutural geral para DLQR adaptativo	100
6.5	Regulador DLQR adaptativo aplicado a planta	104
6.6	Regulador DLQR adaptativo acoplado com controlador PI	104
6.7	Diagrama para implementação do sistema de controle	106
6.8	Diagrama de tarefas do sistema de controle associado a Fig.6.7	107
6.9	Resposta em malha fechada com estimação inteligente e controle adaptativo embarcados utilizando BP	109
6.10	Resposta em malha fechada com estimação inteligente e controle adaptativo embarcados utilizando BP-AG	110
6.11	Resposta em malha fechada com estimação inteligente e controle adaptativo embarcados utilizando RLS	111
6.12	Variação dos parâmetros de controle	111

Lista de Tabelas

4.1	Polos dos modelos	62
5.1	BP caso 1	71
5.2	BP caso 2	75
5.3	BP-AG caso 1	79
5.4	BP-AG caso 2	82
5.5	Análise dos algoritmos para o caso 1	90
5.6	Análise dos algoritmos para o caso 2	91
6.1	Tabela de desempenho do controle adaptativo digital	99
6.2	Tabela de desempenho do LQR com $T_{ref} = 0^{\circ}C$ para o LQR puro e $T_{ref} = 80^{\circ}C$ para o LQR/PI	105

Lista de Siglas e Abreviações

ARX	Modelo Auto Regressivo com Entradas Exógenas
ARMA	Modelo Auto Regressivo com Média Móvel
A/D	Conversor Analógico Digital
BP	Backpropagation
BP-AG	Backpropagation com Aceleração de Gradiente
DLQR	Regulador Linear Quadrático Discreto
FIR	Modelo de Resposta ao Impulso Finito
LS	Mínimos Quadrados (Least Square)
LQR	Regulador Linear Quadrático
MIMO	Múltiplas Entradas e Múltiplas Saídas
MLP	Rede com Multicamadas de Perceptrons
NARX	Modelo Não Linear Auto Regressivo com Entradas Exógenas
NARMAX	Modelo Não Linear Auto Regressivo com Média Móvel e Entradas Exógenas
PWM	Sinal Modulado por Largura de Pulso
PID	Controlador Proporcional, Integral e Derivativo
RLS	Mínimos Quadrados Recursivo
RNA	Rede Neural Artificial
RMLP	Rede com Multicamadas de Perceptrons Recursiva
SISO	Sistema de Uma Entrada e Uma Saída
SMI	Sistema de Medição Indireta

Sumário

1	Introdução Geral	1
1.1	Motivação da Pesquisa	2
1.2	Problema e Solução proposta	3
1.3	Objetivos	4
1.3.1	Objetivos Gerais	4
1.3.2	Objetivos Específicos	4
1.4	Estado da Arte	5
1.4.1	Identificação não Linear	5
1.4.2	Redes Neurais Artificiais	8
1.4.3	Sistema de Medição Indireta	10
1.4.4	Controle Adaptativo	12
1.5	Organização dos Próximos Capítulos	13
2	Métodos Clássicos de Identificação e Redes Neurais Artificiais	15
2.1	Conceitos Básicos	15
2.2	Metodologias para Identificação de Sistemas e Estimação	16
2.3	Descrição de Modelos Lineares	17
2.3.1	Função de Transferência	17
2.3.2	Espaço de Estados	18
2.3.3	Modelo Box-Jenkins	19
2.3.4	Modelo ARMAX	19
2.3.5	Modelo ARX	19
2.3.6	Modelo FIR	20
2.4	Representação de Sistemas Não Lineares	21
2.4.1	Modelo NARX	22
2.4.2	Modelo NARMAX	23
2.4.3	Modelo de Hammerstein	23
2.4.4	Modelo de Wiener	24
2.5	Método dos Mínimos Quadrados	26
2.6	Redes Neurais Artificiais	27
2.6.1	Neurônio Biológico e Artificial	28

2.7	Arquiteturas de Rede Neurais Artificiais	29
2.7.1	Redes de Camada Única	29
2.7.2	Redes com Múltiplas Camadas	30
2.7.3	Redes Recorrentes	31
2.8	Redes Neurais e Identificação	31
2.9	Processos de Aprendizagem	34
2.9.1	Aprendizagem com e sem professor	35
2.9.2	Aprendizado por Correção de Erro	35
2.9.3	Aprendizagem Hebbiana	36
3	Treinamento Online de RNA's	38
3.1	Princípios de Otimização	38
3.2	Método do Gradiente	40
3.3	Método do Gradiente Acelerado	42
3.4	Treinamento Backpropagation	44
3.5	Treinamento Backpropagation Acelerado	46
3.6	Mínimos Quadrados Recursivos (RLS)	47
4	Sistema Térmico e Medição Indireta	50
4.1	O Sistema Térmico	50
4.1.1	Modelo da Célula	51
4.1.2	Modelo da Planta	52
4.2	Modelo para o Núcleo do Sistema de Medição Indireta (SMI)	53
4.2.1	Estimação do Núcleo do SMI	55
4.2.2	Validação do Modelo do SMI	57
4.3	Modelo Não Linear do Sistema Térmico	57
4.3.1	Estimação paramétrica da planta	59
4.3.2	Validação	61
4.4	Análise dos Modelos Levantados	62
5	Estimação Paramétrica Online Inteligente	64
5.1	Estruturas das RNA's Online	64
5.1.1	Estrutura RNA/ARX	65
5.1.2	Estrutura RNA no Espaço de Estados	66
5.2	Estimação com Algoritmos de Treinamento Backpropagation	67
5.3	Estimação Utilizando Backpropagation Acelerado	75
5.4	Estimação paramétrica utilizando RLS	83
5.5	Comparação entre os métodos.	90
6	Sistema de Controle Adaptativo e Ótimo	93
6.1	Controle PID Digital Adaptativo	94
6.2	Regulador Linear Quadrático Adaptativo	99
6.3	Implementação do PID Adaptativo mais Estimação Paramétrica Inteligente em Microcontrolador	105

7	Considerações Finais	112
7.1	Conclusão	112
7.2	Publicações	113
7.3	Trabalhos Futuros	114
	Bibliografia	115

Introdução Geral

A identificação de Sistemas é de extrema importância para qualquer estudo que envolve sistemas físicos, (PASQUALETTI; DORFLER; BULLO, 2013), e até biológicos como em (LIU et al., 2010) e (MARMARELIS; NAKA, 1974) que dizem que a identificação começa com simples atitudes humanas. Já em (NORGAARD O. RAVN; HANSEN, 2000), diz que é a detecção de padrões de comportamento de sistemas físicos, como o movimento de um carro, por exemplo. Assim, identificar é uma prática natural do complexo sistema do corpo humano e que começou a ser transcrito de forma matemática para representar sistemas físicos.

A necessidade de representar esses sistemas fez com que várias técnicas fossem desenvolvidas para se encontrar modelos matemáticos que os representassem bem e para diversos fins, como controle e predição, por exemplo. Em muitos casos, sejam lineares ou não lineares, tem como desafio analisar e determinar variáveis desconhecidas com o uso de dados disponíveis de um dado sistema (MILANESE; VICINO., 1989).

A solução para esses problemas que envolvem identificação, podem variar do complexo até um simples fator multiplicativo. Dentre os casos mais complexos, devem-se levar em consideração incertezas, dadas por um ruído ou variáveis aleatórias com uma certa densidade de probabilidade. Deve-se ter uma grande quantidade de informação sobre o sistema (LJUNG, 1998) e conhecimento sobre os condições sobre a qual o sistema está sujeito para o levantamento de modelos matemáticos.

Existem diversas formas de determinar esses modelos com a utilização de métodos como o de tentativa e erro, determinísticos e regressão linear. Dentre essas formas, tem-se os modelos clássicos que podem ser encontrados facilmente nas bibliografias como o ARX (modelo auto regressivo com entradas exógenas), o ARMAX (modelo auto regressivo com média móvel e entradas exógenas), aplicado a sistemas lineares e os que derivam destes como o NARX e o NARMAX, que são representações matemáticas não lineares, assim como os

modelos de Wiener e Hammerstein.

Além das estruturas clássicas citadas, desenvolve-se métodos de identificação utilizando inteligência computacional para representar sistemas físicos não lineares, como Redes Neurais. Esta ferramenta tem uso predominante para sistemas não lineares devido a sua potencialidade.

Muitos exemplos comprovam a importância do uso de inteligência computacional em identificação. Em (KO, 2012) uma rede de base radial é utilizada para identificar um sistema não linear com aprendizado variável no tempo. Já (TURCHETTI et al., 2008) utiliza inteligência computacional para identificar um sistema estacionário não linear e em (JURADO et al., 2011a) é utilizada uma rede recorrente de ordem elevada para a identificação em um robô manipulador.

Nesse contexto, tem-se os métodos de otimização como uma grande ferramenta na determinação desses parâmetros, como o método dos mínimos quadrados (LS), mínimos quadrados recursivo (RLS) e os métodos envolvendo gradiente. Além disso existem duas abordagens: a *online*, que a identificação é realizada durante a evolução de um processo e a *offline*, que é realizada de forma separada e posteriormente utilizada.

Dentro dessas abordagens, o desenvolvimento de algoritmos e técnicas para identificação e estimação tem sido foco de várias pesquisas científicas, principalmente na estimação *online* de sistemas não lineares e no estudo de convergência para aplicações de tempo real (BAZAEI; MOALLEM, 2007).

Com base nesses tipos de estudo é que o desenvolvimento desta pesquisa está fundamentada, unindo a busca por melhorias dos algoritmos e a utilização de inteligência computacional para estimação *online* de parâmetros aplicados a sistemas térmicos, visto que representar matematicamente este tipo de sistema impõe vários desafios como a variabilidade dos ganhos e do tempo de resposta, dentre outros fatores. Assim utiliza-se o controle adaptativo com uma realimentação indireta (Medição indireta de temperatura) que tem como vantagens a redução de custo e a medição de variáveis que antes não podiam ser mensuradas.

1.1 Motivação da Pesquisa

Vista a importância da identificação, no contexto industrial e científico, este trabalho tem como principal motivação o estudo de métodos e desenvolvimento de algoritmos na tentativa de se obter melhorias no processo de estimação paramétrica utilizando inteligência computacional, tendo como aplicação sistemas térmicos e controle adaptativo para abordagens do tipo *online*.

Outro fator importante que motiva este trabalho é a necessidade de levantar modelos que satisfaçam uma determinada necessidade, que de forma geral, podem ser obtidos tanto de forma *online* quanto *offline*.

Pode-se citar como fator importante, a relevância das aplicações com redes neurais, como em (IBTISSEM; NOUREDINE, 2013) que trata de algoritmos de treinamento de uma rede para sistemas não lineares e em (TITTI; SQUARTINI; PIAZZA, 2005) que trata de uma rede multicamada variante no tempo na identificação da não estacionaridade de sistemas não lineares.

Também pode-se enquadrar neste estudo, o desafio que sistemas térmicos impõem, como o comportamento atípico e variações nos parâmetros gerando alterações na região não estacionária. Junto a isso, têm-se a estimação e aplicação de um sistema de medição indireta de temperatura, que possui grande apelo industrial e inúmeras aplicações.

Com relação ao controle adaptativo, têm-se como aplicação os sistemas não lineares como em (BASRI et al., 2011), e que permite a utilização de várias diretrizes, como fuzzy adaptativo, controle neural adaptativo, podendo citar até mesmo os controles chaveados.

As formulações e estudos abrangem os seguintes temas:

- Estudo de métodos para estimação inteligente;
- Estudos e elaboração de algoritmos na tentativa de se obter melhorias;
- Estimação e controle *online*;
- Modelos neurais adaptativos;
- Medição indireta de temperatura;
- Controle adaptativo.

Os desafios que se apresentam a seguir podem ser considerados relevantes, mas acima de tudo, um impulso para o desenvolvimento da pesquisa em busca de melhorias e possíveis novas soluções.

1.2 Problema e Solução proposta

A obtenção de modelos para um sistema térmico, que seja satisfatório, pode apresentar certa dificuldade por fatores já citados nas seções anteriores, como mudanças atípicas, por ser um processo que apresenta alterações de dinâmica por consequência da mudança em seus

parâmetros quando se muda de ponto de operação. Além de que, associados a esse tipo de sistema, há casos em que se necessita medir uma dada temperatura mas o custo dos sensores são altos ou é inviável medi-la diretamente, como pode ocorrer em algumas indústrias.

Esses tipos de problemas, associados à necessidade de se monitorar ou controlar o sistema, demanda um modelo e com isso a identificação e estimação dos seus parâmetros.

Visto as características desse sistema, um modelo neural é o ideal, juntamente a um controle adaptativo. Com isso há a necessidade do estudo dos métodos para estimação paramétrica *online* que podem apresentar vantagens neste caso. Junto a isso, pode-se encontrar alguns problemas como o caso em que o sistema adaptativo não consiga acompanhar de forma efetiva a dinâmica da planta podendo gerar instabilidade, o que pode ter como consequência a não controlabilidade do sistema.

Como solução, apresenta-se o controle adaptativo e o modelo neural. É possível incluir também o projeto de um sistema de medição indireta de temperatura, que pode reduzir custos e permitir que este trabalho apresente maiores aplicações em escalas industriais. No âmbito dos problemas que envolvem a estimação paramétrica, propõe-se como solução o estudo e desenvolvimento de algoritmos que utilizam técnicas modernas e clássicas que possibilite atingir os objetivos, que são apresentados na seção posterior.

1.3 Objetivos

Os objetivos desta pesquisa são destacados como gerais e específicos para mostrar uma melhor visibilidade dos pontos que se pretendem alcançar e os avanços que podem ser introduzidos em um contexto específico do trabalho.

1.3.1 Objetivos Gerais

Realizar a estimação paramétrica e controle *online* do sistema térmico com aplicação e desenvolvimento de um sistema de medição indireta para fornecer o sinal de realimentação.

Estudar métodos e estratégias para o melhoramento das técnicas existentes na área de identificação voltadas para aplicações em controle utilizando métodos clássicos conhecidos e redes neurais.

1.3.2 Objetivos Específicos

Neste trabalho são citados como objetivos específicos os seguintes itens:

- Desenvolver algoritmos para treinamento *online* de redes neurais;
- Caracterizar um modelo para o sistema de medição indireta de temperatura;
- Realizar comparações entre um sistema com medição direta e com indireta;
- Sintetizar e realizar o controle adaptativo de temperatura;

1.4 Estado da Arte

Apresenta-se nesta seção o estado da arte que busca apresentar as áreas do conhecimento que envolvem este trabalho, como sistemas não lineares, medição indireta, inteligência computacional com aplicações de redes neurais artificiais, identificação de sistemas e controle adaptativo.

1.4.1 Identificação não Linear

A identificação de sistemas abrange vários aspectos e abordagens como a exploração da adaptabilidade. Neste contexto, os métodos foram evoluindo ao longo das décadas, onde pode-se citar a utilização de formas polinomiais para identificar características não lineares, como em (LORY; LAI; HUGGINS, 1959) que descreve um método para determinar uma função polinomial de terceira ordem para um sistema estático não linear. Em 1962 o conceito de adaptabilidade foi utilizado por (ROY; DERUSSO, 1962), onde um dado processo foi dividido em diversas classes e os parâmetros do modelo são ajustados de acordo com o sinal de saída, com base no erro quadrático. Como pode-se ver, este tipo de modelo adaptativo tem grande utilização, sendo visto como uma solução eficaz para estimação de sistemas não lineares.

Os métodos iterativos também tem um papel fundamental na evolução deste tipo de pesquisa, como em 1966 que (NARENDRA; GALLMAN, 1966) propôs um método iterativo para a identificação de um sistema não-linear com base em dados de entrada e saída na presença de ruído, onde o modelo utilizado consistiu de um polinômio seguido por um sistema linear, se caracterizando como um modelo de Hammerstein. Em 1968 o conceito de modelo adaptativo foi novamente utilizado em (BUBNICKI, 1968), assim como em (FURUTA; PAQUET, 1970) que utilizou processos iterativos para estimar os estados de um sistema discreto não linear com base no erro quadrático e na convergência deste, levando-o a resultados satisfatórios, e em 1972, (GOVINDAN; HIGGINS, 1972) estudou a convergência do erro através de um critério de área mínima.

Uma outra diretriz que envolve a identificação é a abordagem no domínio da frequência, como em 1975 que (SHANMUGAM; JONG, 1975) utilizou técnicas no domínio da frequência para identificação de uma classe de sistemas não lineares, onde uma série de potências não lineares foi colocada entre duas funções de transferência e que tinha como solução os parâmetros dessas funções e da série de potências, utilizando dados de resposta em frequência. Utilizando o mesmo tipo de estrutura, (COOPER; FALKNER, 1975) utilizou impulsos na entrada e observando a saída de um sistema, realizou a estimação paramétrica através de simulação digital. Em 1976, (GALLMAN, 1976) compara dois algoritmos que visam minimizar erros quadráticos na estimação de parâmetros, analisando a covariância e convergência para modelos de elevada ordem.

Com o desenvolvimento e pesquisas, novos métodos de identificação surgiram e assim como os modelos de Hammerstein, os de Wiener também foram bastante utilizados, como em (BILLINGS; FAKHOURI, 1977) que utilizou o algoritmo para identificação de um sistema não linear.

Outra estrutura que surge no contexto de identificação, é a do tipo ARMA, um modelo auto regressivo com média móvel, que é utilizado em (PARKER; PERRY, 1981) que apresenta um conjunto de modelos lineares nos parâmetros do tipo ARMA, interconectados formando o sistema não linear.

É notório a importância e aplicação das técnicas de estimação paramétrica, como as várias já citadas anteriormente. Além das técnicas adaptativas, os métodos estatísticos também estão presentes, como em (KORENBERG, 1983) que propõe uma estimação paramétrica para uma equação a diferença, que no caso se trata de uma estrutura do tipo NARMAX, utilizando mínimos quadrados e em (NAKAMORI, 1986), propondo uma identificação *online* para modelar um sinal através de dados de covariância aplicado a um sistema estocástico não estacionário.

A identificação de sistemas não lineares sempre foi um desafio e dentre as diversas técnicas, como já foi explicitado nos trabalhos apresentados, está a identificação *online* como em (HELFERTY et al., 1988), onde o sistema é caracterizado por um conjunto de equações diferenciais parciais e utiliza-se um algoritmo recursivo baseado no erro quadrático para estimar os parâmetros do modelo.

Outros estudos apresentam a utilização de lógica *fuzzy* como identificador para sistemas não lineares, como em (WANG, 1995), onde o autor mostra a eficácia deste método, desenvolvendo uma lei de adaptação paramétrica.

Assim como a inferência *fuzzy* é utilizada como identificador paramétrico, as redes neurais artificiais também podem, como em (CHON; COHEN, 1997), onde uma RNA estima os parâmetros de uma estrutura tipo ARMA (Modelo Auto Regressivo com Média Móvel)

para análises de dados experimentais de frequência cardíaca e flutuações do volume pulmonar.

Com a evolução das pesquisas e das tecnologias, a inteligência computacional, como redes neurais e fuzzy ganharam mais espaço como métodos para identificação de sistemas não lineares, como em (LI; SUNDARARAJAN; SARATCHANDRAN, 2000) que fez uso de uma rede artificial de base radial (RBF) para identificação *online* de um sistema, realizando análises com relação a quantidade de neurônios utilizados reduzindo a esforço computacional.

Para evidenciar ainda mais o estudo sobre a identificação de sistemas não lineares, mais trabalhos podem ser apresentados e mostram a abrangência de conteúdos que envolvem esta dissertação, como em (ZHAO; CHEN, 2006) que evidencia os métodos de Hammerstein tendo por base um modelo ARX (Auto Regressivo com Entradas Exógenas) e (CHEN; WANG; HARRIS, 2008) que utiliza funções de base radial para uma estrutura do tipo NARX (ARX não Linear) sendo realizados estudos da variância do erro quadrático. Em (PEREZ; MATOS; FERNANDEZ, 2009) é utilizado algoritmos genéticos para levantar um modelo não linear da temperatura do óleo em transformadores e compara-los com os resultados do método dos mínimos quadrados. Em (STANTON et al., 2010) é proposto a obtenção e validação de um modelo para um piezoelétrico não linear, enfatizando a importância da modelagem das não linearidades inerentes ao sistema que incluem efeitos elásticos, acoplamento, amortecimento, ganho e largura de banda.

Dentre os trabalhos mais recentes que abordam o tema identificação não linear, tem-se (NING; JING; CHENG, 2011) que realiza a identificação das não linearidades de sistemas com variações espaciais no tempo, transformando-os em uma classe de sistemas MIMO (Múltiplas Entradas e Múltiplas Saídas) parcialmente lineares. Em (HAMETNER; JAKUBEK, 2011) são utilizados modelos de RNA's para aproximar um sistema não linear por uma classe de modelos no qual faz uso dos mínimos quadrados para estimação dos parâmetros. Em (SCHOUKENS; PINTELON; ROLAIN, 2011) é proposto um método de identificação para sistemas de Hammerstein paralelos, com abordagens no tempo contínuo e discreto, (LI; WEN, 2012) utiliza tanto o modelo de Wiener quanto o chaveamento de modelos lineares na presença de ruído onde a classificação dos modelos é realizada por uma máquina de vetor suporte (SVM), (ZENG et al., 2012) aborda a filtragem de partículas, onde tem por objetivo identificar um modelo no espaço de estados para o fluxo dessas partículas através da medição de séries temporais disponíveis, sendo classificado como um modelo não linear estocástico, (FU et al., 2013) trata da identificação adaptativa via rede neural multicamada para o rastreamento de trajetórias com diferentes escalas de tempo, (BHASIN et al., 2013) realiza uma identificação robusta onde é utilizada uma rede neural recorrente multicamada para aproximar a dinâmica do sistema de forma *online*, (KRANJCIC; STUMBERGER, 2014) propõe novos modelos para turbinas, sendo eles não lineares, tendo dados obtidos nas condições normais de funcionamento, (YU; ZHANG; XIE, 2014) investiga os sistemas determi-

nísticos de Hammerstein apresentando uma nova abordagem para estes modelos utilizando sobre-amostragem, aumentando a riqueza de informação e (MEZGHANI-MARRAKCHI et al., 2014), que realiza a identificação de um sistema de áudio não linear que se baseia na gaussianidade e estacionaridade.

1.4.2 Redes Neurais Artificiais

As Redes Neurais Artificiais (RNA's), que se define como uma tecnologia segundo (HAYKIN, 1999), ou ferramenta bio-inspirada, possui uma vasta aplicação, dentre as quais, algumas serão citadas neste trabalho. Assim, como primeiro exemplo, (HILTZ, 1962) realiza um estudo das características de adaptação, comportamento e tratamento de dados de redes neurais biológicas e artificiais onde foi possível simular as características de um neurônio como o sinal de ação, polarização, dentre outros. Nesse mesmo trabalho foi desenvolvido um circuito com amplificadores operacionais mostrando o que mais tarde vem a ser o foco de muitos pesquisas.

As RNA's possuem uma diversidade de aplicações, tanto na área médica quanto nas ciências exatas, como em (MINAMITANI; HAGITA, 1981) que utiliza uma rede como modelo para a descrição numérica dos mecanismos da dor com base em dados obtidos através de estudos fisiológicos e anatômicos. Já em (GEVINS; MORGAN, 1988), a rede neural é utilizada para a detecção de padrões neurológicos, tida como uma excelente forma de incorporar o conhecimento sobre o cérebro na forma matemática. Em (KING; HWANG, 1989) é proposto a utilização de RNA's no processamento de dados em robótica considerando os vários tipos de estrutura de redes, como a de camada única e multicamadas, demonstrando o seu alto grau de aplicabilidade no planejamento de tarefas, controle e planejamento de trajetórias.

Encontra-se também aplicações em detecção de falhas e padrões, como em (CHOW; YEE, 1991) onde uma nova abordagem na detecção *online* de falhas em motores de indução é realizada com o uso de uma RNA para filtrar ruídos e outra para detecção de falhas através de dados obtidos do motor. Já em (WEYMAERE; MARTENS, 1993) é realizado um treinamento supervisionado em uma rede para classificar padrões.

No ramo da robótica, em (HESSELROTH et al., 1994) um controle neural é utilizado para controlar um braço robótico pneumático no estudo das características mecânicas do sistema muscular e esquelético com um total de duzentos neurônios. Em (ZHANG et al., 1996) é apresentada uma RNA para detecção de falhas em transformadores de potência à óleo com base nos gases presentes em seu interior. Em (RAOL; MADHURANATH, 1996) são propostas várias arquiteturas de redes recorrentes para resolver um problema de estimação paramétrica em sistemas dinâmicos. Em (ORLOWSKA-KOWALSKA; KOWALSKI,

1997) redes neurais são utilizadas para estimar o vetor de fluxo do rotor com base, somente, nos dados de corrente e fase, obtendo-se, com precisão, as estimativas sendo considerada uma técnica viável com relação à outras técnicas de estimação de fluxo e velocidade. Em (BHARADWAJ; PARLOS; TOLIYAT, 1999) uma rede neural adaptativa é utilizada como um *sensorless* (medição indireta de uma variável) no monitoramento *online* das condições de um motor de indução, funcionando como um filtro de velocidade, onde é treinada utilizando dados de corrente e tensão no motor.

Em (MAO; SHANG; LIU, 2002) as redes artificiais foram utilizadas como preditor do nível em um lençol freático apresentando uma estrutura auto regressiva e considerando diversos fatores que influenciam no nível do lençol, sendo o modelo considerado adequado para a previsão de situações futuras. Em (GE; ZHANG; LEE, 2004) é realizado um controle adaptativo com redes neurais em um sistema não linear representado por uma classe de modelos MIMO, tendo sua estabilidade comprovada de acordo com os conceitos de Lyapunov. Em (ERTUNC, 2006) uma RNA serve como preditor do fluxo de calor para líquidos dielétricos sob uma variedade de condições de operação através dos dados de pressão, temperaturas de saturação do fluido, aquecimento e da superfície, sendo o resultado fornecido pela RNA, comparado com dados experimentais. Em (CHIANG; HO, 2008) uma rede neural faz a seleção de recursos na tentativa de se encontrar características relevantes para classificar a expressão genética e comparar com outros métodos.

Já pode-se ver as diversas aplicações de RNA's, tanto na área médica, identificação de sistemas, controle adaptativo, classificação de dados, dentre outros. Ainda existem uma série de outros trabalhos, mais atuais, a serem mostrados, como em (ZHOU; ANNAKAGE; RAJAPAKSE, 2010), que tem como aplicação os sistemas de alimentação de ônibus elétricos na estimação da margem de estabilidade de tensão, através de uma RNA, sob condições normais e adversas que é utilizada como referencia para se realizar uma ação de controle. Em (ALANIS et al., 2011), um observador neural recorrente de estados, não linear e no tempo discreto na presença de perturbações externas e incertezas, é proposto. Em (REN; LV, 2011) é proposta uma rede neural dinâmica auto-otimizada para uma classe de sistemas de Hammerstein na presença de ruídos gaussianos, onde a rede tem como entrada a ordem do sistema e a camada intermediária é determinada de forma *online*, sendo seu tamanho determinado pela dinâmica da planta, reduzindo assim o esforço computacional. Em (MANITSAS et al., 2012) é apresentado um sistema de medição indireta através de redes artificiais, utilizando mínimos quadrados para o processo de aprendizagem.

Uma aplicação interessante, que também faz parte deste trabalho, é apresentada em (MAITI et al., 2012) que propõe um estimador adaptativo para a velocidade de um motor de indução, com aplicação de redes neurais em medição indireta, comprovando a versatilidade desta tecnologia. Em (LIN; LIANG; CHEN, 2013) uma rede probabilística estima o estado

de baterias com o intuito de evitar acidentes ou inconveniências devido ao mau funcionamento destas. Da mesma forma, aplicações relacionadas ao estudo dos sinais emitidos pelo corpo humano é de grande relevância, como em (KHOSROWABADI et al., 2014) que usa a rede bio-inspirada para discriminar as emoções humanas através dos sinais cerebrais e em (ALFARO-PONCE; CRUZ; CHAIREZ, 2014) é apresentada uma rede, em tempo contínuo, para estimação de sistemas não lineares via construção de um algoritmo adaptativo para o treinamento da RNA.

Uma das vertentes, presente neste trabalho, trata da utilização das redes bio-inspiradas no processo de identificação de sistemas, portanto vale ressaltar alguns dos trabalhos nesta linha de pesquisa, como (YAMADA; YABUTA, 1993) que propõe um método prático para o projeto de uma RNA para a identificação de função de transferência, (CINCOTTI; DANERI, 1997) que faz uso da rede para identificar os parâmetros para um circuito não-linear no estudo de fenômenos de histerese, (WON et al., 2010) apresenta uma classe de redes recorrentes, provando sua capacidade de identificar sistemas dinâmicos no tempo discreto, apresentando desempenho melhor em comparação aos métodos convencionais.

Baseado em inteligência computacional e no reconhecimento de padrões, (SHI; GAO, 2013) utiliza esse tipo de técnica para identificar uma fonte eletromagnética de ruído baseado em diferenças de frequência. Em (MINGXUAN; HAIGANG; YING, 2013) redes variáveis no tempo modelam e identificam sistemas lineares e variantes no tempo (LTV). Já em (ZHAO et al., 2014) é apresentada, juntamente a inteligência computacional, a tecnologia *pipeline*, onde um método de processamento de informação em computadores é utilizado para reduzir a carga de cálculo dispendida na identificação de sistemas dinâmicos e (NAWROCKI; NAWROCKA, 2014) que apresenta um braço manipulador e uma RNA de uma camada para identifica-lo, tendo sua utilização justificada devido a não linearidade do sistema.

1.4.3 Sistema de Medição Indireta

Um dos grandes problemas envolvendo sistemas físicos, como em processos industriais, ou até mesmo sistemas voltados para a própria pesquisa, trata da existência de variáveis que apresentam difícil acesso para instalação de sensores ou devido ao alto custo. Baseado neste paradigma, a ideia da medição indireta foi abordada. Assim, com base nessa prerrogativa, são apresentados aqui, trabalhos que envolvem este tema e que o evidencia.

Como primeiro exemplo, mostrando que a ideia já era existente, nos anos 70 é (CLYMER, 1970) apresentou métodos matemáticos e técnicas computacionais para medição indireta de variáveis de alguns processos físicos. Em (MAYHAN; BAILEY, 1975) foi realizada a medição indireta do efeito dielétrico em linhas de transmissão ao longo de estradas em situ-

ações secas e molhadas. Já (STIBITZ; MCCANN, 1977) trata de uma aplicação em estudo de células, ou seja, mede-se indiretamente a condutividade intracelular que é necessária para a determinação da resistência da membrana celular.

Tomando como referência o intervalo entre 1960 e 1980 não foram encontrados muitos outros trabalhos que abordam este tema, o que começou a se intensificar a partir dos anos 90, com trabalhos mais voltados para os estudos propostos neste trabalho. Como em (KANMACHI; TAKAHASAH, 1995) que mede a velocidade de um motor de indução indiretamente por dois métodos, um de forma direta e outra através de uma estimativa com base no valor de resistência do secundário. Em (BOILLEREAUX; FLAUS, 2000) é proposto um sensor baseado em modelo que permite a medição indireta de variáveis não mensuráveis explicitando a simplicidade de implementação e aplicações em bio-processos. Em (KAWABATA; MIYATA; ARAKAWA, 2001) e (KAWABATA; FUKUSHIMA, 2005) é realizada a medição indireta de corrente AC em sistemas de distribuição. Além das aplicações em sistemas de potência, a medição do fluxo de água em tubos de forma indireta é proposta em (VAZQUEZ; FLEURY; WALTER, 2006) com base na velocidade em determinados pontos de uma seção transversal e tem como observador dessa variável, uma rede neural.

Um trabalho que está diretamente ligado a linha de pesquisa deste trabalho são (MOURIZINHO et al., 2006) e (NETO J.V.F.; SA, 2010), onde são desenvolvidos e estimados observadores de estado para a medição indireta de temperatura em um forno elétrico, tendo como atuadores uma resistência elétrica e célula Peltier, respectivamente.

Voltado novamente para sistemas de potência, (PRAMANIK; SATISH, 2011) utiliza um SMI para determinar a capacitância série de um enrolamento de um transformador com base na sua resposta em frequência. Já (KOSTUR, 2012) aborda a variável temperatura em fornos com impossibilidades da medição direta, onde cita as questões das incertezas que envolvem este tema bem como a importância da precisão da medição. Em (DURDAN; KACUR, 2013) a medição indireta de temperatura é novamente trabalhada no processo gaseificação subterrânea do carvão, ambiente que a medição via sensor físico é difícil devido ao ambiente agressivo.

Dentre a várias aplicações envolvendo sistema de medição indireta, em (PENNELLI; NANNINI; MACUCCI, 2014) é relatada a medição indireta da condutividade térmica em nanoestruturas, baseado no auto-aquecimento dos nanofios que os compõe. Baseado no baixo rendimento de sensores de velocidade para baixos níveis de rotação do motor, (TAHERI; MOHAMMADBEIGI, 2014) utiliza um filtro de Kalman para medir indiretamente esta grandeza, obtendo resultados satisfatórios. Por fim, levando em consideração as aplicações em robótica, no auxílio do dia-a-dia humano, a realização do controle da força em suas partes causa um custo elevado, devido aos sensores e ao consumo de energia, viabilizando o estudo

sobre a utilização de medição indireta.

1.4.4 Controle Adaptativo

Nesta seção é apresentado um conjunto de trabalhos envolvendo controle adaptativo, que se trata de uma aplicação visto o sistema de identificação *online* e o sistema de medição indireta deste trabalho. Assim, tem-se que dados os trabalhos pesquisados, a ideia do controle adaptativo não é nova, como se pode ver pela data de um dos inúmeros artigos científicos existentes na área, como em (BELLMAN; KALABA, 1959). Mas o que evoluiu, nesse tempo, foram as técnicas de projeto, algoritmos, metodologias e os controladores bio-inspirados. Dentre os diversos exemplos, (MORRIS; NEUMAN, 1981) apresenta o controle adaptativo baseado em modelo de referência para o ajuste dos parâmetros do controlador. Em (KRAFT; CAMPAGNA, 1990), uma comparação entre controladores com base em redes neurais de auto-ajuste e em modelo de referência são comparados, conceitualmente, através de simulação para um sistema não linear na presença de ruído, analisando a velocidade de adaptação.

Este tipo de controle possui uma diversidade de aplicações, como em (BISHOP; PAYNTER; SUNKEL, 1992) onde é investigado a abordagem de controle adaptativo para uma estação espacial, utilizando o Regulador Linear Quadrático (LQR), onde o processo de identificação se dá por um filtro de Kalman recursivo. Em (SANNER; SLOTINE, 1995) é proposto um controlador adaptativo baseado em modelo biológico, como RNA, para sistemas não lineares aplicados em sistemas mecânicos multivariados e robôs manipuladores. Em (DJUKANOVIC et al., 1997) é apresentado o estudo de caso de uma hidroelétrica onde pretende-se controlar a pressão nas turbinas, assim tratando-se de um sistema não linear onde um controlador neuro-fuzzy é utilizado devido a sua capacidade de auto-aprendizagem. Em (GE; HANG; ZHANG, 1999) é proposto um controlador adaptativo, tendo por base uma RNA, para uma classe de sistemas não lineares onde foi verificado a eficiência do uso de inteligência computacional para este fim.

Dentre as diversas abordagens voltada para controle adaptativo, o uso de inteligência computacional como RNA's e inferência *fuzzy* são as mais utilizadas. Em (GE; WANG, 2002) um projeto de controlador neural robusto é proposto para um sistema não-linear na presença de ruído e em (YANG; REN, 2003) a inferência *fuzzy* realiza o mesmo papel da rede neural. Além dessas formas, a utilização de simples regras que ponderam os diversos valores para os parâmetros de controle é uma alternativa, como é realizado em (SISHUBA; REDFERN, 2005) com geradores de energia. Já em (KAHVECI; IOANNOU; MIRMIRANI, 2008) é apresentado o controle adaptativo linear quadrático de trajetória de um planador, abordando a questão do melhor desempenho através das teorias de otimização.

Existem diversas vertentes para controle adaptativo utilizando-se os princípios de otimização, aprendizagem por reforço, chaveamento de ganhos com modelos de referência, os bio inspirados, ou seja, uma grande quantidade de trabalhos relacionados. Assim, serão apresentados agora alguns trabalhos mais atuais, como (KUIPERS; IOANNOU, 2010) que fala que apesar das bem sucedidas aplicações físicas e teóricas do controle adaptativo, a ideia ainda não está maturado para aplicações que exijam alto desempenho e robustez, onde salienta a adaptação, com base em múltiplos modelos lineares e invariantes no tempo (LTI) de forma unificada, através de controladores que tem por base o erro médio quadrático, obtendo robustez, adaptabilidade e desempenho. Já (SANG; TAO, 2010) realiza um estudo sistemático das margens de ganho em sistema que fazem uso desta metodologia de controle.

A adaptabilidade baseada em múltiplos modelos em controle adaptativo pode ser observada em (GIOVANINI, 2011) sobre uma ótica do tipo *online*, permitindo estabelecer um dado desempenho e robustez. Em (CHAOUÏ; SICARD, 2012) é trabalhado o problema de regulação da velocidade em motores síncronos à imã permanente que está sujeito à grandes variações de inercia, propondo um controlador adaptativo *fuzzy* para resolver o problema. Em (SHEN; LIU; DOWELL, 2013) um projeto de controle adaptativo robusto utilizando H_∞ para longitudinal de uma aeronave F-18, onde a falha nos atuadores é considerada através de uma identificação adaptativa desta para que o controlador possa compensa-las. Em (XU; LI, 2014) é trabalhado a questão da estabilidade da tensão através do fornecimento de potência reativa, onde um controlador PI (Proporcional Integral - (OGATA, 2011)) adaptativo é utilizado, mostrando excelência em diversos pontos de operação. Já (CHEN et al., 2014) utiliza a ideia de neuro-controle adaptativo para uma classe de sistemas não lineares.

Muitos outros trabalhos podem ser citados como exemplos que utilizam abordagem em controle ótimo adaptativo, como em (LEWIS; VRABIE; VAMVOUDAKIS, 2012), utilizando aprendizagem por reforço com base nas equações de Riccati e HJB (Hamilton-Jacobi-Bellman). Em (LIU; WANG; LI, 2014) é utilizado um controle ótimo *online* com base em redes neurais para estabilizar uma classe de sistemas não lineares. E por fim, (NODLAND; ZARGARZADEH; JAGANNATHAN, 2013) que também utiliza RNA no contexto de controle ótimo no rastreamento de trajetória de um helicóptero não tripulado.

1.5 Organização dos Próximos Capítulos

No Capítulo 2 deste trabalho, são apresentados alguns conceitos importantes que envolve a identificação de sistemas, bem como alguns tipos de modelos, tanto lineares quanto não lineares. É introduzido algumas teorias e conceitos relacionados a redes neurais artificiais, bem como alguns tipos de redes voltadas para identificação e estimação paramétrica de

sistemas e por fim alguns métodos de treinamento de redes.

No Capítulo 3 são apresentados métodos de treinamento *online* de redes neurais, onde destacam-se três algoritmos: o Backpropagation, o Backpropagation Acelerado e o RLS. É apresentado alguns conceitos de otimização e conceitos de convexidade bem como um estudo sobre gradiente de funções.

No Capítulo 4, tem-se o sistema térmico, onde são apresentados as equações físicas e diagrama esquemático do mesmo. É mostrado o projeto do sistema de medição indireta com base em estruturas apresentadas no Capítulo 2, bem como uma análise ao eliminar o sensor e introduzir o sensor virtual. Apresenta, também, a estimação paramétrica para planta utilizando modelos clássicos para futuras comparações.

No Capítulo 5, é realizado o treinamento *online* de duas estruturas de redes propostas, utilizando o sistema térmico, bem como a avaliação dos resultados, verificando os tempos de convergência e dos parâmetros obtidos a cada intervalo de tempo para os três algoritmos propostos e uma comparação com os modelos clássicos não lineares.

No Capítulo 6, são apresentadas aplicações em controle para as duas estruturas de rede propostas, onde são utilizados duas estruturas de controle. Um PID digital adaptativo e um DLQR adaptativo são apresentados e simulados, unindo a estimação *online* e controle]. Posteriormente é apresentada implementação de alguns desses algoritmos em microcontroladores, através de um simulador de circuito para validação do funcionamento dos mesmos e por fim a conclusão.

Métodos Clássicos de Identificação e Redes Neurais Artificiais

Estudos em identificação de sistemas cresceram bastante no meio científico devido as necessidades presentes nos meios industriais, como nas áreas de engenharia química, elétrica, mecânica e até mesmo nos ramos da biologia e medicina.

Com base nesse contexto, este capítulo apresenta a teoria clássica sobre identificação, bem como sua importância e algumas metodologias. Assim, são apresentadas tanto estruturas lineares do tipo ARX, ARMAX e ARMA, quanto as não lineares do tipo NARX, NARMAX e modelos de Wiener e Hammertein. Juntamente, é apresentada a teoria de redes neurais artificiais, mostrando algumas estruturas básicas, assim como os métodos dos treinamento clássicos que são a base deste trabalho de pesquisa.

2.1 Conceitos Básicos

A Identificação, segundo (LJUNG, 1998), trata da inferência de modelos a partir da observação e do estudo das propriedades de um sistema, onde a necessidade de se obter tais representações matemáticas vem, segundo (AGUIRRE, 2006), acompanhada da precisão destes e da rapidez com que se pode obtê-los.

Dentre as diversas necessidades, principalmente industrial, uma diversidade de estruturas e métodos de estimação surgiram, abrangendo desde os métodos determinísticos até os bio inspirados. Assim, como conceito importante neste tipo de estudo, está a modelagem matemática, que é englobada no estudo de identificação de sistemas.

Segundo as bibliografias, modelagem é a área do conhecimento que estuda maneiras de

implementar modelos matemáticos para um dado sistema real (AGUIRRE, 2006), onde existem três técnicas básicas: a *modelagem caixa branca* a *modelagem caixa cinza* e a *modelagem caixa preta*.

Na *modelagem caixa branca* é necessário conhecer totalmente o sistema, como as leis físicas que o regem e seus parâmetros, na *modelagem caixa cinza* se tem um conhecimento parcial do sistema, como alguns parâmetros e leis e na *modelagem caixa preta* ocorre quando não se tem conhecimento sobre o sistema, tendo acesso a somente dados de entrada e saída (AGUIRRE, 2006).

A identificação de sistemas é considerado como um conjunto de métodos e técnicas alternativas a modelagem caixa branca, podendo ser referida como modelagem caixa preta e parcialmente presente na modelagem caixa cinza.

Baseado nesse contexto, são citados um conjunto de conceitos relevantes para o entendimento deste estudo, como o de modelo matemático, linearidade, variância e invariância no tempo, modelos discreto e contínuos, monovariáveis e multivariáveis que podem ser facilmente encontrados em (AGUIRRE, 2006). Além desses conceitos, existem várias metodologias aplicadas a identificação, desde a obtenção de dados, escolha do modelo, métodos de estimação e validação, que são aplicadas de forma direta neste trabalho.

2.2 Metodologias para Identificação de Sistemas e Estimação

Antes de se determinar qualquer estrutura ou técnica para se obter um modelo, primeiramente, deve-se determinar as necessidades associadas a aplicação. Como por exemplo, para controle de um sistema não há necessidade de um modelo com extrema exatidão para atingir o alvo, mas para casos, como uma medição indireta, em que um modelo assume o papel de um sensor, o grau de aproximação do modelo com relação ao sistema real deve ser maior. Assim, o processo de identificação pode ser dividido em etapas.

A primeira etapa trata de analisar o sistema para que se possa identificar um modelo apropriado, onde este pode ser obtido através das representações matemáticas baseadas nas leis da física, podendo se enquadrar na modelagem caixa branca ou cinza, onde a ordem e estrutura do modelo são determinadas. Uma outra forma de se escolher o modelo, pode ser através de testes utilizando medições obtidas do sistema, onde várias estruturas matemáticas, com formas e ordens diferentes, são analisadas para que se escolha a mais apropriada para uma dada aplicação.

Caso haja necessidade de se realizar medições, tem-se a segunda etapa a ser realizada,

que é a aquisição de dados. Neste momento é determinado o sinal pelo qual o sistema será excitado (sinal de entrada) e a frequência em que os sinais de entrada e saída serão medidos (frequência de amostragem). Assim, a frequência de amostragem é definida através da teoria de Nyquist e o sinal de entrada deve ser tal que possa varrer o máximo de características diferentes, como variação em frequência e de ganho.

O próximo passo, é a separação dos dados para estimação paramétrica do modelo escolhido e validação deste. Dessa forma, são utilizados de 70 a 90% dos dados para estimação e o restante para validar os resultados.

Realizada a estimação paramétrica, passa-se a validação, onde além dos 10 a 30% dos dados, podem ser utilizadas medições diferentes das utilizadas na estimação dos parâmetros. Caso o resultado não tenha sido satisfatório, novas estimações, com outras estruturas ou até mesmo novos dados, devem ser realizadas. Assim, dentre os métodos de identificação e estimação de parâmetros de modelos, utiliza-se métodos recursivos para abordagens *online/offline* e determinísticos para abordagens *offline*.

2.3 Descrição de Modelos Lineares

Existem diversas formas de se representar sistemas dinâmicos por meio de descrições matemáticas lineares, como *função de transferência* e modelos no *espaço de estados*. Estes modelos podem ser caracterizados como contínuos e discretos, com destaque aos modelos discretos do tipo FIR, ARX, ARMAX, ARMA, Box Jenkins, dentre outros.

2.3.1 Função de Transferência

A descrição por *função de transferência* é uma das alternativas mais utilizadas para representar sistemas dinâmicos, podendo ser encontrada em diversas bibliografias, como (LANDAU, 1938), (LJUNG, 1998), (AGUIRRE, 2006), (FRANKLIN; POWELL; WORKMAN, 1990), (OGATA, 2011) e (CHEN, 2012).

Esta representação é definida como a transformada de Laplace da resposta ao impulso $h(t)$ de um sistema, com condições iniciais nulas e definida por

$$H(s) = \int_0^{\infty} h(t)e^{-st} dt, \quad (2.1)$$

sendo $s = \sigma + j\omega$.

Para o tempo discreto, a função de transferência é definida como a *transformada Z* da resposta ao impulso discreto, $h(k)$ e com condições iniciais nulas, é dada por

$$H(z) = \sum_{n=-\infty}^{\infty} h(n)z^{-n} \quad (2.2)$$

De forma generalizada, a função de transferência é a relação entrada/saída de um sistema, na forma $Y(s)/U(s)$ para o tempo contínuo e $Y(z)/U(z)$ para o tempo discreto.

2.3.2 Espaço de Estados

Outra descrição bastante importante, é o *espaço de estados*. Esta é base para os projetos de controladores ótimos e robustos, assim como o *LQR* e o *H-infinito*, podendo ser tanto de tempo contínuo quanto discreto.

De forma generalizada, o *espaço de estados* é dado por $\dot{x} = f(x, u(t))$, sendo x o estado do sistema, com $x \in \mathbb{R}^n$, $\dot{x} = \frac{dx}{dt}$, e $u(t) \in \mathbb{R}^r$ um conjunto de entradas exógenas.

Ao se assumir que o sistema representado é linear e invariante no tempo (LTI), pode-se escrever $\dot{x} = f(x, u(t))$ em sua forma linearizada, dada por (CHEN, 2012):

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t), \end{aligned} \quad (2.3)$$

sendo A , B , C e D as matrizes constantes de ordem $n \times n$, $n \times r$, $q \times n$ e $q \times r$, respectivamente.

A representação no espaço de estados pode ser escrita na sua forma discreta, que é dada por:

$$\begin{aligned} x(n+1) &= A_d x(n) + B_d u(n) \\ y(n) &= C_d x(n) + D_d u(n) \end{aligned} \quad (2.4)$$

Pode-se observar através das equações de estado e funções de transferência, que a representação no espaço de estados apresenta certas vantagens, como maior acesso as informações do sistema e na formulação adequada para projeto de controladores mais modernos.

2.3.3 Modelo Box-Jenkins

A estrutura do tipo *Box-Jenkins*, dentre as estruturas clássicas, é a mais completa. A partir desta, todas as outras estruturas podem ser obtidas. Este modelo possui quatro graus de liberdade, dados por $B(z^{-1})$, $A(z^{-1})$, $C(z^{-1})$ e $D(z^{-1})$ e possui forma dada por

$$y(n) = \frac{B(z^{-1})}{A(z^{-1})}u(n) + \frac{C(z^{-1})}{D(z^{-1})}\varepsilon(n), \quad (2.5)$$

sendo $A(z^{-1}) = 1 + a_1z^{-1} + \dots + a_nz^{-n}$, $B(z^{-1}) = 1 + b_1z^{-1} + \dots + b_mz^{-m}$, $C(z^{-1}) = 1 + c_1z^{-1} + \dots + c_pz^{-p}$ e $D(z^{-1}) = 1 + d_1z^{-1} + \dots + d_kz^{-k}$.

Os polinômios que compõem o modelo não possuem parâmetros em comum, proporcionando os quatros graus de liberdade mencionados.

2.3.4 Modelo ARMAX

O modelo ARMAX (Auto-Regressivo com Média Móvel e entradas Exógenas) é um caso particular da Eq.(2.5), onde a diferença está no menor grau de liberdade com $D(z^{-1}) = A(z^{-1})$, possuindo três polinômios e estrutura dada por

$$y(n) = \frac{B(z^{-1})}{A(z^{-1})}u(n) + \frac{C(z^{-1})}{A(z^{-1})}\varepsilon(n). \quad (2.6)$$

Este tipo de representação é bastante utilizada em séries temporais lineares e tem como modelo subsequente o ARX e ARMA.

2.3.5 Modelo ARX

A evolução das estruturas para representação de sistemas, começou com o modelo FIR (Modelo de Resposta ao Impulso Finito), que leva em consideração somente o sinal de entrada, passando para o ARX (Modelo Auto-Regressivo com entradas externas) que leva em consideração entradas externas e saída em sua estrutura e uma dinâmica igual com relação ao ruído $\varepsilon(n)$. A forma do ARX pode ser observada na equação dada por

$$y(n) = \frac{B(z^{-1})}{A(z^{-1})}u(n) + \frac{1}{A(z^{-1})}\varepsilon(n). \quad (2.7)$$

É possível reescrever a Eq.(2.7) na forma de equação a diferença, que é dada por

$$A(z^{-1})y(n) = B(z^{-1})u(n) + \varepsilon(n), \quad (2.8)$$

$$y(n) - a_1y(n-1) - \dots - a_ky(n-k) = b_1u(n-1) + \dots + b_pu(n-p) + \varepsilon(n), \quad (2.9)$$

onde a Eq.(2.9) pode ser representada por $y(n) = x(n)^T\theta$, sendo θ o vetor de parâmetros e $x(n)$ o de regressores, respectivamente, dadas por

$$\theta = [a_1 \cdots a_{l_a} \quad b_1 \cdots b_{l_b}]^T, \quad (2.10)$$

$$x(n) = [-y(n-1) \cdots -y(n-l_a) \quad u(n-1) \cdots u(n-l_b)]^T. \quad (2.11)$$

Por fim, será apresentado o modelo mais simples, o modelo FIR.

2.3.6 Modelo FIR

O modelo do tipo FIR (JANCZAK, 2004)(*Resposta ao Impulso Finita*), é uma das representações discretas mais simples não recursivas e pode ser escrita na forma dada por

$$y(n) = B(z^{-1})u(n) + \varepsilon(n), \quad (2.12)$$

sendo $y(n)$ a saída e $u(n)$ uma entrada externa, esta pode ser reescrita na forma de equação a diferença, dada por

$$y(n) = b_1u(n-1) + b_2u(n-2) + \dots + b_lu(n-l) + \varepsilon(n). \quad (2.13)$$

Com a equação a diferença do modelo FIR, a forma matricial pode ser obtida, onde

tem-se o vetor de parâmetros θ , que é dado por

$$\theta = [b_1 \cdots b_l]^T, \quad (2.14)$$

e o vetor de regressores $x(n)$,

$$x(n) = [u(n-1) \cdots u(n-l)]^T. \quad (2.15)$$

Logo a Eq.(2.13), na forma compacta é dada por

$$y(n) = x^T(n)\theta. \quad (2.16)$$

2.4 Representação de Sistemas Não Lineares

Na natureza, de forma generalizada, os sistemas são do tipo não lineares, onde se pode escolher uma dada região de operação e se obter um modelo linearizado que represente tal região. Mas quando o interesse está no funcionamento em toda faixa de operação do sistema, um modelo não linear é necessário.

Ao se analisar a dinâmica desses sistemas e verificar se não atendem ao princípio da superposição, é possível definir se o sistema é não linear (CHEN, 2012). Nesse contexto, segundo (KHALIL, 2002), tem-se os modelos não lineares para se representar fenômenos não lineares, embora seja possível representar sistemas não lineares com um conjunto de modelos lineares, divididos por pontos de operação.

Muitas das representações não lineares presentes nas bibliografias, como em (LJUNG, 1998) e (AGUIRRE, 2006), tem estrutura parecidas com as lineares, onde podem ser citadas o NARX, NARMAX e os modelos de Wiener e Hammerstein.

De forma geral, as estruturas não lineares são representadas por

$$y(n) = g(x(n), \theta) + \varepsilon(n), \quad (2.17)$$

sendo $x(n)$ uma entrada qualquer, θ os parâmetros e $\varepsilon(n)$ o ruído.

Muitos modelos não lineares são obtidos com base nas estruturas das representações lineares, como o ARX e ARMAX, com a inclusão de um funcional $g(x(n), \theta)$. Assim, nesta seção são apresentados alguns dos modelos mais clássicos, como o NARX, NARMAX e modelos de Wiener e Hammerstein.

2.4.1 Modelo NARX

O modelo não linear auto-regressivo (NARX) possui duas formas básicas: a polinomial e a racional (AGUIRRE, 2006). A forma racional é representada por

$$y(n) = g[y(n-1), \dots, y(n-l_y), u(n-d), \dots, u(n-d-l_u)] + e(n), \quad (2.18)$$

sendo $g(\cdot)$ uma função não linear.

Esta estrutura é bastante utilizada e serve de base para outras estruturas como a de Wiener e Hammerstein que serão apresentadas nos tópicos posteriores. Esta, tem por característica a linearidade nos parâmetros. Já a forma polinomial é bastante utilizada na determinação das funções não lineares $g(\cdot)$, como pode ser visto em (AGUIRRE, 2006). Assim, de forma geral, o NARX polinomial pode ser representado por

$$y(n) = \sum_{m=0}^a \sum_{p=0}^{a-m} \sum_{l_1, l_m}^{l_y, l_u} c_{p,m}(l_1, \dots, l_m) y(n-l_y)^p u(n-l_u)^m. \quad (2.19)$$

Como exemplo, tem-se a equação desenvolvida para $a = 2$, $l_u = 2$ e $l_y = 1$ dada por

$$\begin{aligned} y(n) = & c_{0,0} + c_{0,1}(1)y(n-1) + c_{2,0}(1)y(n-1)^2 + \\ & c_{0,1}(1)u(n-1) + c_{0,1}(2)u(n-2) + \\ & c_{1,1}(1)y(n-1)u(n-1) + c_{1,1}(1,2)y(n-1)u(n-2) + \dots \\ & c_{0,2}(1)u(n-1)^2 + c_{0,2}(1,2)u(n-1)u(n-2) + \\ & c_{0,2}(2)u(n-2)^2 \end{aligned} \quad (2.20)$$

Com este tipo de estrutura, pode-se obter facilmente a característica estática de um sistema, sendo que é através desta que se obtêm a função $g(\cdot)$.

Um problema presente neste tipo de estrutura, é que para a determinação do regime

permanente onde $\bar{y} = y(n-l)$, é que surgem termos quadráticos da própria saída, dificultando a obtenção da função inversa de $g(\cdot)$, podendo dificultar a obtenção do modelo de Wiener pela possibilidade de $g(\cdot)$ escolhida não admitir inversa.

Uma solução é utilizar a sua estrutura de forma mais simples possível, ou seja, $a = 1$, ou omitir os termos que possuem potência quadrática ou maior de $y(n-l)$ e fatores que tem multiplicação de y e u .

Caso se possa identificar claramente a característica estática do sistema através dos dados obtidos, fica mais fácil a obtenção de $g(\cdot)$, utilizando-se, simplesmente, uma identificação dessa curva através de funções polinomiais ou exponenciais, por exemplo.

2.4.2 Modelo NARMAX

O modelo NARMAX (Não linear Auto-Regressivo com Média Móvel e entradas Exógenas), assim como o NARX tem por base o ARX, baseia-se no ARMAX com a presença de um funcional $g(\cdot)$.

Os modelos baseados em equações diferenciais, tem um papel importante na representação de sistema lineares e não-lineares e tem como representação geral para o NARMAX a expressão dada por

$$\begin{aligned} y(n) = & g[y(n-1), \dots, y(n-l_y), u(n-d), \dots, \\ & u(n-d-l_u), e(n-1), \dots, e(n-l_e)] + e(n), \end{aligned} \quad (2.21)$$

sendo $y(n)$, $u(n)$ e $e(n)$, respectivamente, a saída, a entrada e o ruído do sistema.

A forma de determinação de $g(\cdot)$ é a mesma utilizada para o modelo NARX que inicialmente não é conhecida, podendo ser aproximada por funções polinomiais e racionais.

2.4.3 Modelo de Hammerstein

O modelo de Hammerstein (OGUNFUNMI, 2007) (BOLSTAD et al., 2011), é uma representação dinâmica não linear no tempo discreto, através de blocos interconectados de acordo com a Fig.2.1.

Desta forma, pode-se determinar a equação geral da estrutura, sendo $g(\cdot)$ uma estru-

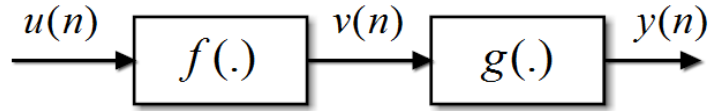


Figura 2.1: Blocos conectados de Hammerstein.

tura autoregressiva linear e $f(\cdot)$ uma função não linear, que é dada por

$$\begin{aligned} v(n) &= f(u(n)), \\ v(n - q) &= f(u(n - q)), \end{aligned} \quad (2.22)$$

$$y(n) = \sum_{i=1}^{l_y} a_i y(n - i) + \sum_{j=1}^{l_u} b_j v(n - j), \quad (2.23)$$

$$y(n) = \sum_{i=1}^{l_y} a_i y(n - i) + \sum_{j=1}^{l_u} b_j f(u(n - j)). \quad (2.24)$$

A Eq.(2.24) mostra como a não linearidade está expressa em termos de $u(n)$, sendo assim, um modelo linear em seus parâmetros. Uma vantagem em relação ao modelo de Wiener está na questão de não se utilizar função inversa.

A resposta em estado estacionário, ou seja, quando nem a entrada e nem a saída variam, é dada por

$$\bar{y} = \frac{\left(\sum_{i=1}^{l_u} b_i \right) f(\bar{u})}{1 - \left(\sum_{j=1}^{l_y} a_j \right)}, \quad (2.25)$$

sendo \bar{y} e \bar{u} são, respectivamente, a saída e entrada em regime permanente.

2.4.4 Modelo de Wiener

Enquanto no modelo de Hammerstein, a função não linear mapeia o sinal $u(n)$, no modelo de Wiener é mapeado o sinal intermediário $v(n)$. Isso pode ser visto claramente na Fig.2.2.

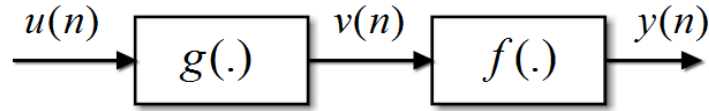


Figura 2.2: Blocos interconectados de Wiener.

Para este caso, como não é possível medir $v(n)$, a função não linear $f(\cdot)$ precisa ser inversível, de tal forma que

$$y(n) = f(v(n)), \quad (2.26)$$

$$v(n) = f^{-1}(y(n)), \quad (2.27)$$

sendo $g(\cdot)$ uma estrutura autoregressiva que assumindo a forma ARX, obtêm-se

$$v(n) = \sum_{i=1}^{l_y} a_i v(n-i) + \sum_{j=1}^{l_u} b_j u(n-j) \quad (2.28)$$

e

$$y(n) = f \left(\sum_{i=1}^{l_y} a_i v(n-i) + \sum_{j=1}^{l_u} b_j u(n-j) \right). \quad (2.29)$$

Com o desenvolvimento matemático apresentado, a equação final que representa o modelo de Wiener é dada por

$$y(n) = f \left(\sum_{p=1}^{l_v} a_p f^{-1}(y(n-p)) + \sum_{q=1}^{l_u} b_q u(n-q) \right). \quad (2.30)$$

Utilizando o mesmo procedimento realizado no modelo de Hammerstein, para determinar o valor DC para uma dada entrada, para o modelo de Wiener, tem-se

$$\bar{y} = f \left(\frac{\left(\sum_{q=1}^{l_u} b_q \right) \bar{u}}{1 - \left(\sum_{p=1}^{l_y} a_p \right)} \right). \quad (2.31)$$

2.5 Método dos Mínimos Quadrados

O método dos mínimos quadrados, teve seu princípio formulado por Karl Friedrich Gauss (ASTROM; WITTENMARK, 2008) que diz que os parâmetros desconhecidos de um modelo matemático, devem ser escolhidos de forma que a soma das diferenças dos quadrados entre valores atuais e estimados, multiplicados por números que medem o grau de exatidão, é um mínimo.

O método dos mínimos quadrados (LS ou *least squares* na língua inglesa) tem uma larga aplicação na estimação paramétrica de modelos e sua formulação parte de um modelo similar a da Eq.(2.16), dada de uma outra forma representada por

$$y(n) = \varphi_1(n)\theta_1^0 + \varphi_2(n)\theta_2^0 + \dots + \varphi_n(n)\theta_n^0, \quad (2.32)$$

sendo y a variável observada, $\theta_1^0, \dots, \theta_n^0$ os parâmetros a serem determinados e $\varphi_1(n), \dots, \varphi_n(n)$ funções conhecidas.

Tendo em vista a Eq.(2.32) e colocando na forma matricial, $y(n) = \varphi^T(n)\theta$, tem-se

$$\varphi^T(n) = [\varphi_1(n) \quad \varphi_2(n) \quad \dots \quad \varphi_n(n)],$$

$$\theta^0 = [\theta_1^0 \quad \theta_2^0 \quad \dots \quad \theta_n^0],$$

com n denotado no tempo, define-se $\varphi(n)$ como sendo a variável de regressão ou regressores, onde $\{(y(n), \varphi(n)), n = 1, 2, \dots, t\}$ são obtidos experimentalmente.

Para se determinar os parâmetros θ , de tal maneira que a saída calculada se aproxime o quanto for possível do valor medido, aplica-se as teorias de otimização. No caso do LS, trata-se de um problema de otimização irrestrita para se encontrar os valores de θ que minimize o erro quadrático, que é dado por

$$\min V(\theta, t) = \frac{1}{2} \sum_{n=1}^t (y(n) - \varphi^T(n)\theta)^2. \quad (2.33)$$

Baseado nessa teoria, a solução do problema de otimização dado pela Eq.(2.33) segundo (ASTROM; WITTENMARK, 2008), é dada por

$$\hat{\theta} = (\phi^T \phi)^{-1} \phi^T Y. \quad (2.34)$$

A Eq.(2.34) é chamada de equação normal do LS e $(\phi^T \phi)^{-1} \phi^T$ é chamada de pseudo inversa.

2.6 Redes Neurais Artificiais

Aplicações com RNA's tem crescido de forma espantosa nos últimos anos. As Redes Neurais, segundo (HAYKIN, 2001), é uma tecnologia que tem por base o cérebro humano, que é altamente complexa, não-linear e paralela. E para (SILVA; SPATTI; FLAUZINO, 2010), são ferramentas computacionais para resolução de problemas complexos do mundo real, compostas de elementos simples de processamento densamente interconectados tendo por base o neurônio biológico.

As aplicações são desenvolvidas, principalmente, nas áreas de Computação e Engenharia, como na identificação de sistemas (JURADO et al., 2011b), controle (XIAO; LONG; ZHAO, 2013) e classificação de dados (NIRKHI, 2010), e associados com outras tecnologias, como Fuzzy por exemplo (BARUCH et al., 2012).

Dentre as características do neurônio biológico, tem-se a não linearidade, o alto grau de paralelismo, robustez, alta capacidade de aprendizado e a capacidade de lidar com informações imprecisas e confusas (HAYKIN, 2001).

Já o neurônio artificial, possui boa adequação no tratamento de dados não lineares e lineares, buscando-se boa precisão nos resultados e robustez, tanto relacionado aos ruídos quanto falha de hardware, adaptabilidade e generalização, que trata de sua utilização para dados ao qual não foi treinado.

Para que se possa obter tais características, a rede neural passa por um processo de treinamento, que é realizado por determinados algoritmos, que são apresentados no Capítulo 3, para o sistema térmico apresentado no Capítulo 4, com resultados presentes no Capítulo 5, onde os modelos baseados nas redes neurais artificiais são utilizados no controle adaptativo da planta no Capítulo 6. Com isso, apresenta-se o conhecimento necessário para o entendimento das RNA's.

2.6.1 Neurônio Biológico e Artificial

O Neurônio é responsável pelo gerenciamento de todas as atividades do corpo humano, estando localizados no cérebro, que contabiliza bilhões de neurônios, onde cada elemento é dividido em três partes básicas: dendritos, corpo celular e axônio (HAYKIN, 2001), como descreve a Fig.2.3.

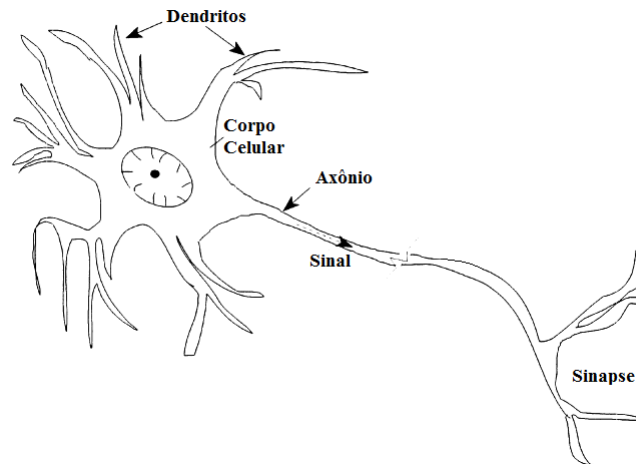


Figura 2.3: Neurônio Biológico.

O corpo celular possui toda a informação, ou seja, o núcleo. Já os dendritos recebem informações de outros neurônios e as transmite para o núcleo e por fim os axônios recebem informações do corpo celular e as transporta através das sinapses.

Analogamente ao neurônio biológico, o perceptron é de extrema importância na compreensão das redes neurais artificiais, como mostra (HAYKIN, 2001) e (BASHEER; HAJ-MEER, 2000), onde as redes artificiais tem por base o perceptron da Fig.2.4. Esta estrutura possui entradas, que equivalem aos dendritos, um somador que equivale ao núcleo e concentra toda a informação, e tem uma saída que transmite um sinal para um determinado fim ou para outro perceptron.

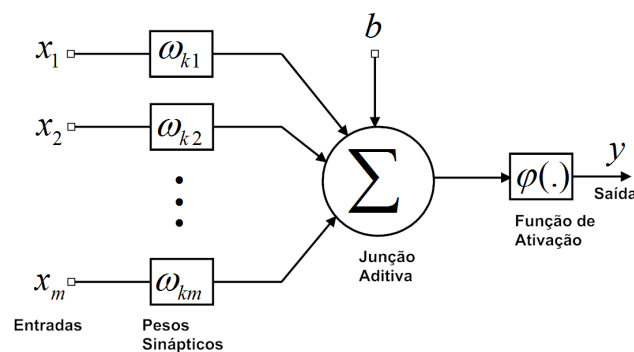


Figura 2.4: Neurônio Artificial

Analisando matematicamente a Fig.2.4, tem-se $X = [x_1 \ \cdots \ x_m]$ como o conjunto de informação que chegam para o perceptron, que são ponderados pelos pesos sinápticos $W_k = [w_{k1} \ \cdots \ w_{km}]$ com b denominado de *bias* sendo todos os sinais unificados pela função aditiva.

A equação que é uma combinação linear dos pesos sinápticos ω_{ki} com a entrada x_i , que é denominado de atividade pré-sináptica, é dada por

$$I_k = \sum_{i=1}^m \omega_{ki} x_i. \quad (2.35)$$

A atividade pós-sináptica é dada pelo funcional $y = \varphi(I_k, \omega_k)$, onde φ pode assumir diversas formas, como funções lineares e não lineares. Este trabalho visa utilizar estas funções, para se representar a planta térmica na forma de uma rede neural artificial.

Existem diversos métodos de treinamento e estruturas para essas redes, e neste caso, o treinamento *backpropagation* é utilizado, bem como a busca por melhorias.

2.7 Arquiteturas de Rede Neurais Artificiais

Uma RNA, como se sabe, tem como elemento básico o neurônio artificial ou perceptron, podendo este ser combinado com outros perceptrons dando origem à vários tipos de rede voltadas para diversas aplicações. Assim, são apresentadas algumas estruturas existentes, que servem de base para este estudo, como redes de camada única, redes de múltiplas camadas (do inglês - MLP) e as recorrentes.

2.7.1 Redes de Camada Única

As RNA's, de forma geral, estão dispostas em camadas, podendo ter em cada camada uma quantidade N de neurônios. A rede de camada única, segundo (HAYKIN, 1999), trata da arquitetura mais simples, onde a camada de entrada (fonte) é projetada para a camada de saída, por apenas um neurônio por saída como descrito na Fig.2.5.

Este tipo de rede tem sido bastante utilizada para resolução de problemas de otimização, como em (GAO; LIAO, 2010) que utiliza uma rede de uma camada para resolver um problema linear quadrático em tempo real, onde a rede é uma função estável de Lyapunov podendo convergir para solução ótima, assim como em (CASTILLO FONTENLA-ROMERO; ALONSO-BETANZOS, 2002) que a RNA é utilizada para minimizar a soma de erros qua-

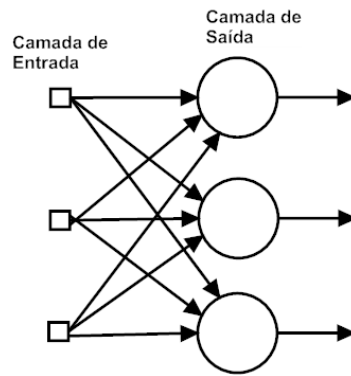


Figura 2.5: RNA de uma camada

dráticos.

Além destes exemplos, podem ser citados outras formas de utilização deste tipo de rede, onde esta pode assumir a forma de um controlador PID, onde os pesos da rede passam a ser o ganhos do controlador e como representação de sistemas MIMO e SISO, como (HAYKIN, 1999) relaciona em seus últimos capítulos.

2.7.2 Redes com Múltiplas Camadas

As redes de múltiplas camadas são as mais utilizadas e se caracterizam por possuírem uma camada de entrada, camadas intermediárias ou camadas ocultas e a camada de saída. As camadas ocultas passam a intermediar a informação entre a entrada e saída da rede, tornando-a mais capaz de extrair informações de ordem elevada, segundo (HAYKIN, 1999). A Fig.2.6 ilustra um exemplo deste tipo de arquitetura para uma camada de entrada, duas ocultas e uma de saída.

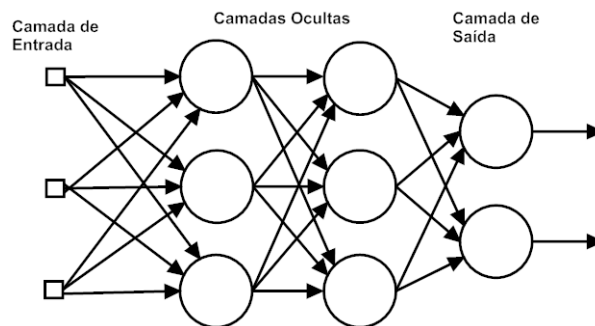


Figura 2.6: RNA com múltiplas camadas

Pode-se observar na figura, que se trata de uma rede com duas camadas ocultas, três variáveis de entrada e duas de saída. Este tipo de rede é bastante utilizada para aproximar funções, classificar dados, representar sistemas, na realização de controle, dentre outros (LI;

SUNDARARAJAN; SARATCHANDRAN, 2000).

2.7.3 Redes Recorrentes

As redes recorrentes (HAYKIN, 1999) tem por característica a existência de ao menos um laço de realimentação, ou seja, a saída anterior da rede é utilizada como entrada, podendo ser constituída de uma ou mais camadas, como exemplifica a Fig.2.7 que apresenta uma rede de uma camada com realimentação de saída, onde tem-se três entradas externas e três saídas. Quando estas não possuem entradas externas, são ditas redes auto-realimentadas.

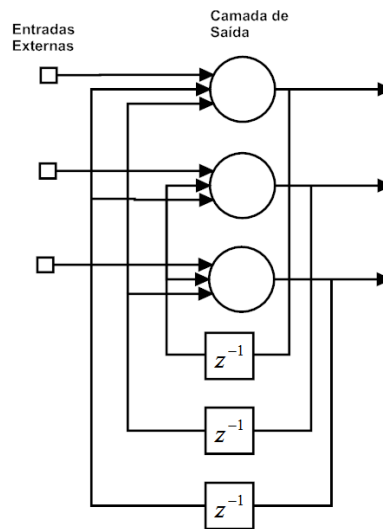


Figura 2.7: RNA realimentada

As redes recorrentes são as mais convenientes para aplicações em identificação de sistemas, já que com estas, podem ser montadas diversas estruturas que podem representar sistemas de diversas ordens (RAOL; MADHURANATH, 1996),(BHARADWAJ; PARLOS; TOLIYAT, 1999). Dessa forma, é apresentada na próxima seção algumas formas de identificação e algumas estruturas de redes com base em (HAYKIN, 2001).

2.8 Redes Neurais e Identificação

As Redes Neurais Artificiais são uma tecnologia com um potencial de aplicações bastante abrangente, englobando áreas, desde humanas e biológicas até as engenharias. Uma dessas aplicações, é na identificação de sistemas, sejam eles lineares ou não lineares.

Na identificação de sistemas, podem ser citadas duas formas básicas da utilização de RNA's. Em uma delas, estas podem servir como estimador dos parâmetros de um modelo

com estrutura já conhecida, como exemplifica a Fig.2.8, sendo θ^* os parâmetros estimados pela rede para uma estrutura linear ARX, $u(t)$ a entrada, $y(t)$ a saída e $y^*(t)$ a saída estimada (HAYKIN, 1999).

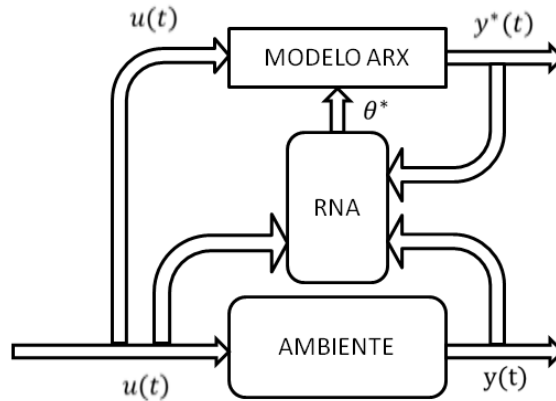


Figura 2.8: Exemplo de uma RNA como estimador paramétrico

Neste caso, o processo de treinamento da rede pode ocorrer tanto durante o funcionamento da planta, caso *online*, quanto pode-se tratar de forma *offline*, bastando que se realize o treinamento da rede separadamente, utilizando um conjunto de dados obtidos previamente.

Uma outra forma de utilização das RNA's no contexto de identificação, é o seu uso como modelo do sistema dinâmico, onde o método de treinamento ajustará os pesos sinápticos de forma que a rede possa representá-lo, como apresenta a Fig.2.9, podendo também ser realizada de forma *online* e *offline*.

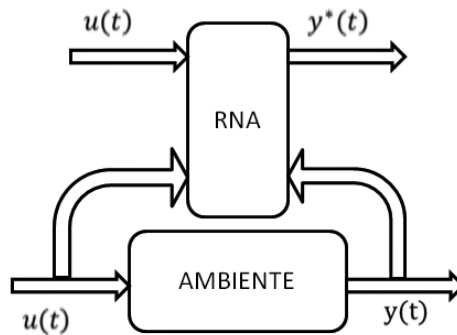


Figura 2.9: Exemplo de uma RNA como modelo

Uma das vantagens da estrutura da Fig.2.9 é que o número de parâmetros a serem estimados são reduzidos, considerando estruturas semelhantes de rede em relação ao caso da Fig.2.8, já que tem-se a redução de um bloco, e por consequência, uma menor quantidade de cálculos e variáveis.

Podem ser citados vários trabalhos científicos em que uma RNA é utilizada como modelo ou como elementos de adaptabilidade paramétrica, como em (PEZZOTTI; LONDONO, 2012) que realizou uma identificação utilizando redes neurais para determinar o modelo da pressão em um tanque de etanol para fins de controle e em (JANCZAK, 2004), onde é realizada a identificação de sistemas não lineares utilizando modelos polinomiais e RNA.

Existem, segundo (HAYKIN, 2001), um conjunto de estruturas de redes sugeridas para processos de identificação de sistemas. Dentre elas, estão as redes realimentadas com sinais atrasados no tempo. Tais redes, buscam seguir a estrutura funcional dos modelos clássicos como ARX e NARX, por exemplo. A Fig.2.10 apresenta uma estrutura generalista de uma rede com base no ARX.

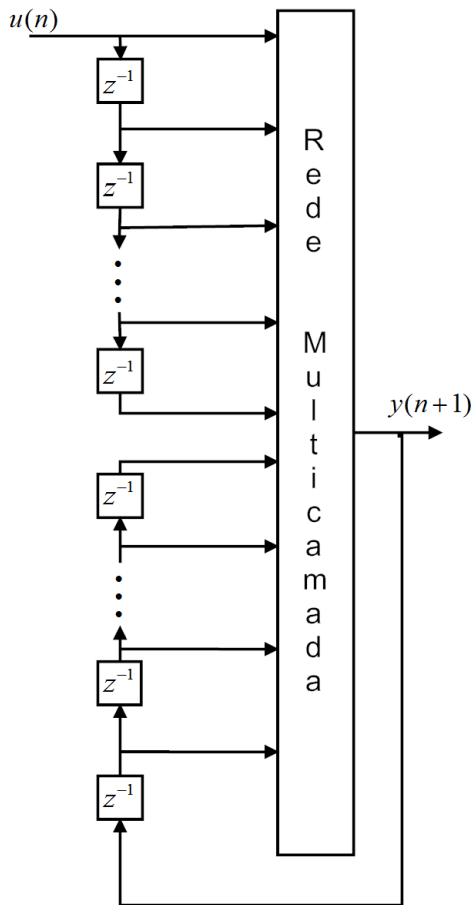


Figura 2.10: Modelo RNA/ARX obtido de (HAYKIN, 2001)

Essa rede representa um modelo entrada/saída com realimentação da saída $y(n+1)$ atrasada no tempo e uma entrada $u(n)$ que também é atrasada no tempo. Se for imposto uma função de ativação na saída da rede com características não lineares, esta pode ser considerada um modelo do tipo NARX na forma da Eq.(2.19), onde φ é a função de ativação. Portanto, seja uma rede recorrente com multicamadas (RMLP) com $I^1(n)$ sendo o sinal de saída da

primeira camada de neurônios e $I^l(n)$ da última camada, dada uma entrada $u(n)$, as equações que relacionam a entrada com a saída da rede são dadas por

$$\begin{aligned} I^1(n+1) &= \varphi(I^1(n), u(n)), \\ I^2(n+1) &= \varphi(I^2(n), I^1(n+1)), \\ &\vdots \\ I^l(n+1) &= \varphi(I^l(n), I^{l-1}(n+1)). \end{aligned} \tag{2.36}$$

A partir dessa relação é possível definir modelos de ordem 1 até n para aplicações em identificação de sistemas.

2.9 Processos de Aprendizagem

A característica mais importante das redes neurais artificiais é a capacidade de aprendizagem. Dentro deste contexto, existem várias metodologias e algoritmos que visam aproximar a resposta da rede ao valor alvo. Como por exemplo, na identificação de sistemas dinâmicos a rede é utilizada para assimilar determinadas características de dada planta física, realizando uma aproximação funcional satisfatória.

O processo de aprendizagem de uma rede se dá pelo ajuste de seus pesos sinápticos, que são os parâmetros livres que se adaptam através de um algoritmo devido a um estímulo aplicado ao sistemas, segundo (HAYKIN, 1999).

Existem diversos métodos de treinamento que envolvem RNA's, como por exemplo, correção de erro, baseado em memória, competitivo, Hebbiana, Boltzmann, dentre outras, mas somente alguns serão abordados neste capítulo, ou seja, os que servem de fundamento para esta pesquisa, como a Hebbiana, aprendizagem com e sem professor e correção de erro.

Vale apenas salientar, neste contexto, a importância dos dados obtidos através de sensores para esses treinamentos, bem como na quantidade de informação sobre um dado sistema de interesse a ser observado. Isso resulta em uma melhor qualidade dos resultados, o que implica na escolha da frequência com que estes dados serão obtidos, seguindo o princípio de Nyquist e a separabilidade entre dados para treinamento e validação, que convencionalmente utiliza-se 70 e 30%, respectivamente.

2.9.1 Aprendizagem com e sem professor

No contexto geral do processo de treinamento de RNA's, pode-se dividir o processo de aprendizagem em duas formas diferentes: com e sem um professor.

A aprendizagem com professor ou supervisionada, baseando-se na existência de um ambiente desconhecido para uma dada rede neural que se deseja treinar. O professor é o elemento que possui o conhecimento prévio sobre o ambiente em questão, que são representados pelos seus dados de entrada e saída, como exemplifica a Fig.2.11, onde a resposta da rede é comparada com a saída desejada gerando um sinal de erro, que deseja-se minimizar através de métodos ótimos.

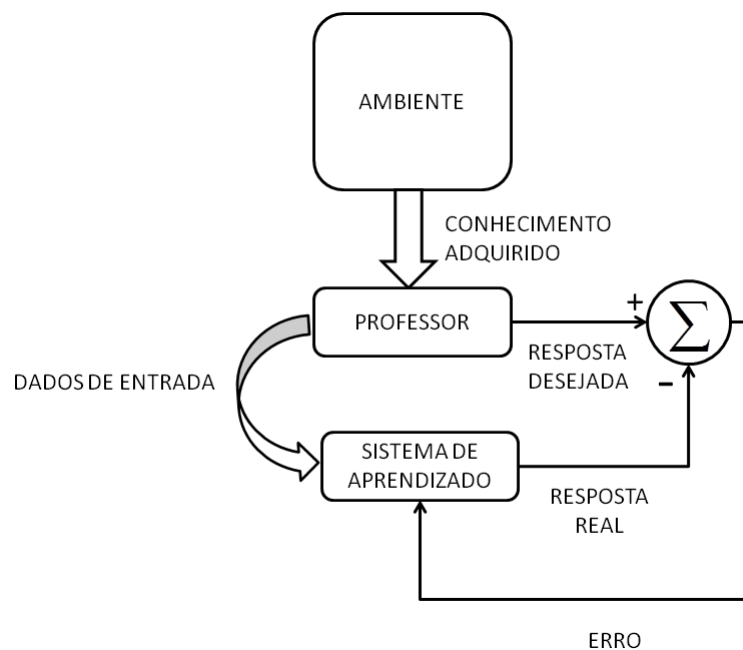


Figura 2.11: Aprendizado com professor (HAYKIN, 2001)

O processo de aprendizagem sem professor, diferentemente do processo anterior, não possui uma supervisão do aprendizado, ou seja, não possui uma referência e portanto não possui um sinal de erro. Nesse contexto, são encontradas duas formas nas bibliografias: uma com aprendizado por reforço ou neurodinâmica e aprendizagem não-supervisionada, segundo (HAYKIN, 2001).

2.9.2 Aprendizado por Correção de Erro

O aprendizado por correção de erro, é um dos métodos mais básicos de treinamento de RNA's. Portanto, supondo uma rede de múltiplas camadas e um nó de saída, ou seja,

com um único neurônio k na camada de saída e seja $x(n)$ o conjunto de sinais de entrada desse neurônio, que são provenientes das camadas anteriores, e que são ponderadas pelos j pesos sinápticos, definidos como ω_{kj} . Esta rede produz uma saída $y_k(n)$ tendo como valor alvo $d_k(n)$, que produzindo um sinal de erro que é dado por

$$e_k(n) = d_k(n) - y_k(n). \quad (2.37)$$

Este método faz uso da *regra delta* ou *regra de Widrow-Holf* criada em 1960. Portanto, dado o erro instantâneo $e_k(n)$, o peso sináptico ω_{kj} varia em um passo de tempo de

$$\Delta\omega_{kj} = \eta e_k(n) x_j(n), \quad (2.38)$$

sendo η definida no intervalo fechado de 0 a 1, como sendo a taxa de aprendizado. Este fator pondera o quanto $\Delta\omega_{kj}(n)$ influenciará em $\omega_{kj}(n+1)$.

Tendo o valor de ajuste dado pela Eq.(2.38), o novo peso é dado por

$$\omega_{kj}(n+1) = \omega_{kj}(n) + \Delta\omega_{kj}, \quad (2.39)$$

até que se encontre a solução ótima local, no caso ω_{kj} , que minimize o erro quadrático $\frac{1}{2}e_k^2$.

É necessário escolher bem o fator de ajuste dos pesos η , obtendo-se o valor que seja mais conveniente à um dado processo de treinamento, para que se possa alcançar estabilidade e convergência no processo de aprendizagem da rede. Muitos autores sugerem um valor padrão entre 0.95 e 0.98.

2.9.3 Aprendizagem Hebbiana

O aprendizado de Hebb, postulada em 1949, é umas das mais antigas metodologias para treinamento de redes neurais artificiais. Este método se baseia no aumento de atividades sinápticas pela proximidade entre duas células, ou seja, quando uma célula A tem seus axônios próximos a uma célula B, com A auxiliando persistentemente no disparo de B, passam a ocorrer alterações metabólicas que melhoram a eficiência com que A dispara(ativa) B.

Esse processo se dá pela sinapse hebbiana, como cita (HAYKIN, 1999), que é definida quando dois neurônios em ambos os lados da atividade sináptica são ativados simultanea-

mente, ou seja, há um sincronismo e portanto dependência do tempo e uma correlação entre as atividade pré e pós sináptica.

Matematicamente, o fator que altera os pesos de uma rede $\Delta\omega_{kj}$ no passo n depende inteiramente da atividade pré, $x_j(n)$ e pós sináptica, $y_k(n)$, expressa por

$$\Delta\omega_{kj}(n) = F(y_k(n), x_j(n)). \quad (2.40)$$

Portanto, a forma mais simples de aprendizado Hebbiano é dada por

$$\Delta\omega_{kj}(n) = \eta y_k(n) x_j(n), \quad (2.41)$$

sendo a atualização dos pesos dada pela Eq.(2.39), sendo η o fator de aprendizado.

Como pode ser observado, a expressão (2.41) é estritamente crescente, levando as conexões da rede a saturação. Assim, surge como alternativa, à hipótese da covariância introduzida por Sejnowski em 1977, onde são considerados os valores médios dos sinais de entrada e saída, ao longo do tempo, mantendo o conceito da sinapse de Hebb e criando um ponto de equilíbrio. Isso faz com que $\Delta\omega_{kj}(n)$ possa assumir tanto valores positivos quanto negativos, evitando a saturação. Dessa forma, este processo de treinamento é dado por

$$\Delta\omega_{kj}(n) = \eta(y_k(n) - \bar{y})(x_j(n) - \bar{x}), \quad (2.42)$$

sendo que a convergência ocorre quando $y_k(n) = \bar{y}$ e $x_j(n) = \bar{x}$, tido a ocorrência da solução não trivial.

Treinamento Online de RNA's

A base deste estudo está imerso nas teorias de otimização voltadas para o treinamento de redes neurais artificiais, que possui inúmeras aplicações e em diversas áreas. Assim como os *Mínimos Quadrados*, apresentado no Cap. 2, que trata de um método que permite a estimação de parâmetros de forma não recursiva, neste capítulo são apresentados outros métodos, que tem por característica a recursividade, como RLS, Gradiente e Gradiente Acelerado, voltados principalmente para estimação inteligente *online* de parâmetros e que podem ser aplicados também em controle adaptativo.

3.1 Princípios de Otimização

Os estudos em otimização ajudaram bastante no desenvolvimento de métodos e algoritmos existentes em diversas áreas de pesquisa, como por exemplo, no treinamento de redes neurais artificiais. Segundo (LUENBERGER; YE, 2008) a otimização é um princípio enraizado e importante para muitas análises de problemas complexos, que envolvem decisão e alocação de recursos. Isto envolve a obtenção de valores para variáveis, objetivando-se quantificar parâmetros de desempenho e a qualidade das decisões.

No contexto histórico, a otimização teve como principal advento as aplicações dos métodos do gradiente de Newton em 1940, época em que a tecnologia não era avançada e métodos como estes não eram utilizados com facilidade. No período de 1947 a 1949, desenvolveu-se os métodos de programação linear e Método Simplex (BAZARAA; SHETTY, 2006) de Dantzig e a solução e modelagem de problemas complexos envolvendo estes métodos. Posteriormente, entre 1951 e 1959 teve-se os estudos sobre as condições de otimalidade com Kuhn-Tucker e estudos em programação não-linear.

Em um contexto mais atual, os diversos estudos de métodos e algoritmos, com a maior

capacidade de processamento e também com a maior complexidade dos problemas, coloca-se em foco os métodos de otimização aplicados a inteligência computacional e algoritmos evolutivos juntamente a ideia de processamento paralelo.

As teorias de otimização podem ser aplicadas de diversas formas e em diversas áreas, como na busca pela melhor solução possível para um dado problema. Como exemplo, suponha-se a determinação do menor custo de combustível para que um ônibus, partindo de um ponto x com destino a y , com diversos trajetos possíveis. A solução deste problema trata-se de determinar a melhor rota para que uma empresa tenha o menor gasto possível.

Assim como os exemplos citados, existem uma diversidade de casos que são apresentadas em (BAZARAA; SHETTY, 2006), explicitando o problema de otimização restrita e irrestrita. De forma geral, o problema de otimização é definido como:

$$\begin{aligned} \min f(x) \\ \text{s.a.} \quad & g_i(x) \leq 0 \quad \text{com } i = 1, \dots, m \\ & h_i(x) = 0 \quad \text{com } i = 1, \dots, n \\ & x \in X, \end{aligned} \tag{3.1}$$

sendo $f(x)$ a função que se deseja minimizar no caso de custo ou maximizar no caso de recompensa e definida como função objetivo. Caso o problema se caracterize pela existência de restrições, tem-se $g_i(x)$ definidas como restrições de desigualdade e $h_i(x)$ restrições de igualdade, para todo $x \in X$, com X sendo um subespaço do \mathbb{R}^n . Caso não sejam consideradas ou não existam restrições o problema ao caso irrestrito que se resume somente a

$$\min f(x). \tag{3.2}$$

O objetivo quando se fala em otimização, é encontrar uma solução $x \in X$ de forma que se obtenha o menor valor possível para função de custo, dentro do conjunto de valores de x que atendam as condições $g_i(x)$ e $h_i(x)$. Esse conjunto de valores de x , é a região factível ou região de possíveis soluções, que minimizam $f(x)$.

Dentre os diversos tipos de problemas e métodos de solução, tem-se a programação linear e não linear. Assim, em caráter de conhecimento, segundo (LUENBERGER; YE, 2008), a programação linear trata da formulação mais natural na caracterização de problemas de otimização que utilizam funções lineares como custo ou recompensa, cujas restrições, também são funções lineares. Já a não linear, como o próprio nome sugere, aplica-se a casos cuja função

objetivo são não lineares, como nos casos estudados neste trabalho que envolve a minimização do erro quadrático, onde tem-se como método para se encontrar a solução do problema, as proposições de Newton que faz uso das expansões em série de Taylor, o que de maneira geral fazem uso do gradiente da função objetivo em função dos parâmetros, método de Lagrange, dentre outros.

Outro conceito bastante utilizado neste trabalho, que envolve problemas de otimização não lineares, é a convexidade das funções objetivo. Segundo (BAZARAA; SHETTY, 2006; LUENBERGER; YE, 2008), para se encontrar a solução global e não local de um problema é necessário que a função objetivo atenda ao princípio da convexidade dado por

$$f(\alpha x_1 + (1 - \alpha)x_2) \leq \alpha f(x_1) + (1 - \alpha)f(x_2), \quad (3.3)$$

sendo f definida no conjunto convexo X , com $x_1, x_2 \in X$ e todo α no intervalo fechado $0 \leq \alpha \leq 1$.

A forma de solução desses problemas está no desenvolvimento de algoritmos, que segundo (LUENBERGER; YE, 2008) é um mapeamento, onde dado um ponto x que pertence a um espaço X , a saída do algoritmo aplicado nesse ponto é um novo ponto. Portanto, seja um algoritmo A que mapeia o espaço X , inicializado em $x_0 \in X$, o novo ponto é dado por

$$A : x_k \rightarrow x_{k+1}. \quad (3.4)$$

A teoria apresentada é a base para construção deste capítulo, bem como dos algoritmos RLS, Gradiente e Gradiente Acelerado. Assim, o primeiro método a ser estudado é o do gradiente, voltado para o treinamento de redes neurais artificiais como modelo, para estimação dos pesos sinápticos.

3.2 Método do Gradiente

O método do gradiente tem sua formulação baseada em derivadas direcionais, como por exemplo, seja uma função $f(x, y)$, a derivada direcional fornece a taxa de variação dessa função, na direção e sentido de um vetor unitário U , dado um ponto $P(x_0, y_0)$ no plano xy (LEITHOLD, 1994). Assim dado o caso generalizado, definindo-se i como vetor unitário na direção do eixo-x e j o vetor unitário na direção do eixo-y e seja $U = i + j$ a composição

desses vetores, o gradiente é definido como a taxa de variação de $f(x, y)$, na direção de U , ou seja, o gradiente de $f(\cdot)$ é dado por

$$\nabla f(x, y) = f_x(x, y)i + f_y(x, y)j, \quad (3.5)$$

sendo $f_x(x, y)$ a derivada de $f(\cdot)$ em relação a x e $f_y(x, y)$ a deriva em relação a y . O vetor gradiente dado pela Eq.(3.5) possui direção e sentido do máximo crescimento da função. Assim, o gradiente pode ser utilizado tanto para solucionar problemas que envolvem maximização, utilizando o sentido de crescimento e de minimização no sentido de decrescimento máximo $-\nabla f(x, y)$, denominados métodos descendentes.

Este método pode ser associado para solucionar vários problemas de otimização bem como para treinamento de redes neurais, como foi apresentado em alguns processos de treinamento presentes no Capítulo 2. Assim, no caso da rede neural (HAYKIN, 1999), tem-se por objetivo minimizar o erro quadrático médio dado por

$$\epsilon = \frac{1}{2} \sum_{n=0}^p (d(n) - y(n))^2, \quad (3.6)$$

sendo $d(n)$ o alvo ou valor desejado no instante n e $y(n) = \varphi(\omega_k(n), x_i(n))$ a saída da rede, sendo n um número inteiro. Assim, o aprendizado, no caso de uma rede neural, baseada no método do gradiente, trata da obtenção da solução ótima considerando a minimização do erro quadrático ϵ , de forma tal que os pesos variem no sentido de diminuição desse erro, ou seja, na direção do gradiente descendente, o que consiste na equação de atualização dos pesos dada por

$$\omega_k(n+1) = \omega_k(n) - \lambda \nabla_{\omega} \epsilon(\omega_k), \quad (3.7)$$

sendo λ a taxa de aprendizado e $g(n) = \nabla_{\omega} \epsilon(\omega_k)$ é dito vetor gradiente na direção de ω_k .

Segundo (HAYKIN, 2001), este caso trata de um problema de otimização irrestrita, sendo $\epsilon(\omega)$ diferenciável, tendo como solução ω^* , satisfazendo a condição $\epsilon(\omega^*) \leq \epsilon(\omega)$, sendo que o vetor gradiente é dado por

$$\nabla\epsilon(\omega) = \left[\frac{\partial\epsilon}{\partial\omega_1}, \frac{\partial\epsilon}{\partial\omega_2}, \dots, \frac{\partial\epsilon}{\partial\omega_n} \right]^T, \quad (3.8)$$

onde a condição de convergência é dada pela condição $\epsilon(\omega(n+1)) < \epsilon(\omega(n))$.

A atualização dos pesos dada pela Eq.(3.7) é dito de descida ingreme, pois se opõe ao gradiente para a obtenção de uma solução ótima local. Assim, este princípio serve de base para os diversos métodos de treinamento em redes neurais, como os métodos de Newton e o próprio *backpropagation* que tem por base o gradiente.

3.3 Método do Gradiente Acelerado

O Método de aceleração que teve origem nos estudos realizados por Nesterov, como em (NESTEROV, 1983), cuja aplicação se restringe à problemas de otimização de funções convexas. Mais tarde, Saeed Ghadimi propõe em (GHADIMI; LAN, 2013), um método de aceleração com base em Nesterov que se aplica tanto a problemas de otimização convexa quanto não convexa. Este método também é aplicado em controle preditivo, como em (PATRINOS; BEMPORAD, 2014), que une a aceleração do gradiente com o princípio da dualidade de Lagrange. Já neste trabalho, faz-se uso do método proposto em (GHADIMI; LAN, 2013), juntamente ao algoritmo *Backpropagation* na tentativa de compensar o maior número de operações com a redução no número de iterações, implicando na redução do tempo de treinamento e esforço computacional.

Portanto, seja uma função de custo genérica, contínua e convexa, tendo a seguinte expressão:

$$f^*(x) = \min_{x \in \mathbb{R}^n} f(x), \quad (3.9)$$

leva-se em consideração a condição de convexidade de Lipschitz, dado por

$$|f(x) - f(x')| \leq L|x - x'|, \quad (3.10)$$

com $L > 0$ e $\forall x, x' \in \mathbb{R}^n$. Assim considerando-se $f(\cdot)$ uma função convexa, diz-se que a função gradiente é contínua e convexa sobre o mesmo critério se

$$\|\nabla f(x) - \nabla f(x')\| \leq L\|x - x'\|. \quad (3.11)$$

Baseado nessa teoria, (NESTEROV, 1983) mostra que o grau de complexidade em um algoritmo utilizando aceleração, dada uma solução \bar{x} e uma tolerância $f(\bar{x}) - f^* \leq \epsilon$, pode ser determinada por um grau de complexidade $O\left(\frac{1}{\sqrt{\epsilon}}\right)$ enquanto que para o gradiente descendente é dada por $O\left(\frac{1}{\epsilon}\right)$. No entanto, o método de Nesterov não se aplica à otimização convexa quando o número de amostras é grande.

Como desenvolvimento, (GHADIMI; LAN, 2013) demonstra a convergência e ampliação do método acelerado para quaisquer caso, seja ele um caso convexo ou não. Neste caso é levado em consideração uma classe de problemas dado por

$$\min_{x \in \mathbb{R}^n} y(x) + h(x), \quad (3.12)$$

com $y(x) := f(x) + j(x)$ e $f(x) \in \mathbb{R}^n$ uma função possivelmente não convexa, $j(x) \in \mathbb{R}^n$ uma função convexa e $h(x) = L_x(x)$ uma função convexa simples com domínio limitado, onde $L_x(x)$ é uma função indicadora de um conjunto convexo compacto $X \subset \mathbb{R}^n$. Desta forma, pode-se definir para cada função um fator limitante, ou seja, uma constante de Lipschitz L_y para $y(x)$, L_f para $f(x)$ e L_j para $j(x)$, onde pela função de energia pode-se definir $L_y = L_j + L_f$ como fator limitante generalizado. Sendo $h(x)$ não diferenciável, segundo (GHADIMI; LAN, 2013), é necessário um novo critério, baseado no mapeamento do gradiente $G(., ., .) = \nabla f(x)$ para analisar a complexidade do GA e mostra que a política, mesmo que agressiva, apresenta uma boa taxa de convergência ao encontrar a solução $\bar{x} \in \mathbb{R}^n$ com grau de complexidade dado por

$$O\left\{\left(\frac{L_f^2}{\epsilon}\right)^{1/3} + \frac{L_y L_f}{\epsilon}\right\}. \quad (3.13)$$

Portanto, apresenta-se o método GA para resolver uma classe de problemas de otimização, tanto convexa quanto não convexa de uma classe de problemas não lineares, onde $y(x)$ é diferenciável e limitada inferiormente, apresentando uma abrangência maior com relação ao método do gradiente decrescente, como apresenta o Algoritmo 1.

Algoritmo 1: Algoritmo do Gradiente Acelerado (AG)

H **Entrada:** $x_o \in \mathbb{R}^n$, $\alpha \in (0, 1)$ para qualquer $k \geq 2$, $\beta > 0$ e $\lambda > 0$

Inicializar $x_o^{ag} = x_o$, $x_k^{md} = x_o$ e $k = 0$;

para $k \leftarrow k + 1$ faça

$$x_k^{md} = (1 - \alpha)x_{k-1}^{ag} + \alpha x_{k-1}$$

$$x_k = x_{k-1} - \lambda \nabla y(x_k^{md})$$

$$x_k^{ag} = x_k^{md} - \beta \nabla y(x_k^{md})$$

fim

Este algoritmo pode ser levado para aplicação com o método de aprendizado *backpropagation* sendo facilmente posto na forma do gradiente descendente bastando fazer $\beta = \lambda$, sendo estes os parâmetros de ajuste do treinamento, x_k^{md} , x_k e x_k^{ag} os parâmetros ponderado, normal e acelerado, respectivamente.

3.4 Treinamento Backpropagation

O *Backpropagation* é um dos tipos de treinamento mais utilizados em redes neurais artificiais. Este tipo de treinamento aplica-se, geralmente, à redes que possuem várias camadas de neurônios, como é apresentada na Fig.3.1 que possui uma camada de entrada, duas camadas ocultas e a camada de saída.

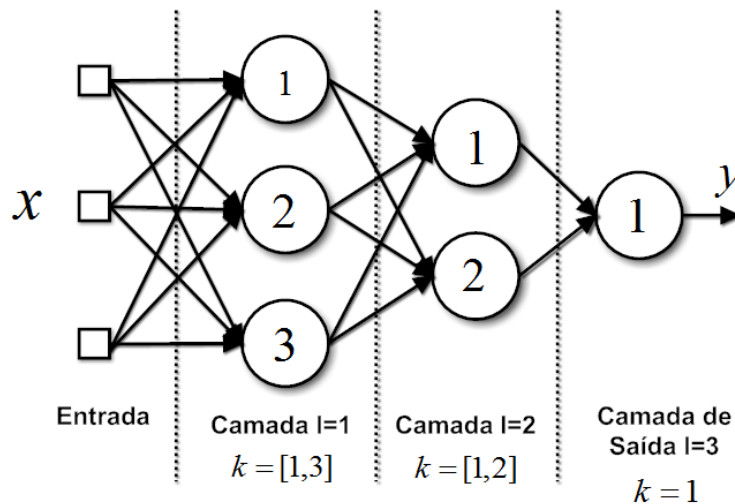


Figura 3.1: Arquitetura de uma rede multicamadas

O método de treinamento é baseado no gradiente do erro entre o sinal de saída da rede e a saída real. Portanto, seja k o neurônio da camada l , com suas i entradas, tendo seus pesos sinápticos definidos por w_{ki}^l . Sendo I_k^l a saída desse neurônio, que tem como expressão

a Eq.(2.35), que representa a atividade pré-sináptica do perceptron, onde para cada camada tem-se:

$$I_k^1 = \sum_{i=1}^{n_x} w_{ki}^1 x_i, \quad (3.14)$$

como sendo a atividade pré-sináptica da primeira camada de neurônios, que é uma combinação linear direta das entradas, é dada por

$$I_k^2 = \sum_{i=1}^{n_{c1}} w_{ki}^2 y_i^1, \quad (3.15)$$

sendo o sinal de saída, ou atividade pré-sináptica da camada oculta representada por

$$I_k^3 = \sum_{i=1}^{n_{c2}} w_{ki}^3 y_i^2, \quad (3.16)$$

onde a atividade pré-sináptica da camada de saída, com n_x sendo o número de entradas, n_{c1} o número de neurônios da primeira camada e n_{c2} o número de neurônios da segunda camada, sendo que para cada I_k^l tem-se $y_k^l = \varphi(I_k^l)$, definida como a saída do perceptron, dada através da função de ativação.

O treinamento é realizado por etapas, onde os pesos são atualizados no sentido camada de saída para camada de entrada. Dessa forma, para camada de saída, a atualização do i -ésimo peso do neurônio k , referente a saída y_k^2 do neurônio k da camada oculta é dada por

$$w_{ki}^3(n+1) = w_{ki}^3(n) + \eta \delta_k^3(n) y_k^2(n), \quad (3.17)$$

sendo η o fator de aprendizagem, $w_{ki}^3(n)$ e $w_{ki}^3(n+1)$ os pesos passado e atual, δ_k^3 o gradiente que é dada por

$$\delta_k^3(n) = (d(n) - y^3(n)) \varphi'(I_k^3(n)), \quad (3.18)$$

com $d(n)$ sendo o valor desejado para a saída da rede e $y^3(n)$ o valor de saída da rede.

Determinada a atualização dos pesos para camada de saída, passa-se para as camadas anteriores. Portanto, fazendo uso da mesma formulação, tem-se para as camadas intermediárias o gradiente referente a atualização dos pesos dado por

$$\delta_k^2(n) = \left(\sum_{j=1}^{n_{e3}} \delta_j^3(n) w_{ji}^3(n) \right) \varphi'(I_k^2(n)), \quad (3.19)$$

que se baseia no gradiente da camada de saída e na atividade pós sináptica da própria camada, determinando-se assim a equação recorrente para camada oculta dada por

$$w_{ki}^2(n+1) = w_{ki}^2(n) + \eta \delta_k^2(n) y_k^1(n). \quad (3.20)$$

Por fim, chega-se a atualização dos pesos sinápticos da camada de entrada, onde o gradiente tem por base o gradiente a segunda camada, sendo representada por

$$\delta_k^1(n) = \left(\sum_{j=1}^{n_{e3}} \delta_j^2(n) w_{ji}^2(n) \right) \varphi'(I_k^1(n)), \quad (3.21)$$

sendo

$$w_{ki}^1(n+1) = w_{ki}^1(n) + \eta \delta_k^1(n) x_i(n). \quad (3.22)$$

Este método de treinamento, possui diversas aplicações, que no caso deste trabalho tem por finalidade a identificação paramétrica *online* aplicado ao sistema térmico e controle adaptativo, tendo como objetivo a busca pela aceleração do processo de treinamento, através da aceleração do gradiente. Dessa forma, propõe-se o backpropagation acelerado.

3.5 Treinamento Backpropagation Acelerado

A adaptação do backpropagation clássico para a sua forma acelerada, trata da tentativa de melhorar a velocidade de adaptação do algoritmo. O processo de aceleração vem

sendo utilizado bastante em controle preditivo, como em (PATRINOS; BEMPORAD, 2014). Dessa forma, neste trabalho, é utilizado o método de aceleração de gradiente para o processo de estimação do tipo *online*, aplicado em um sistema térmico baseado no efeito Peltier, sendo proposto o treinamento *Backpropagation* Acelerado.

Portanto, aplicando-se o aceleração, segue-se a mesma linha de atualização dos pesos sinápticos, ou seja, pela retropropagação do erro, o que para camada de saída é dado por

$$\begin{aligned} w_{k,i}^{md,3}(n+1) &= (1-\alpha)w_{k,i}^{ag,3}(n) + \alpha w_{k,i}^3(n), \\ w_{k,i}^3(n+1) &= w_{k,i}^3(n) - \lambda \delta_k^3(n) y_k^2(n), \\ w_{k,i}^{ag,3}(n+1) &= w_{k,i}^{md,3}(n+1) - \beta \delta_k^3(n) y_k^2(n). \end{aligned} \quad (3.23)$$

Já para a camada oculta, o mesmo princípio é utilizado, obtendo-se

$$\begin{aligned} w_{k,i}^{md,2}(n+1) &= (1-\alpha)w_{k,i}^{ag,2}(n) + \alpha w_{k,i}^2(n), \\ w_{k,i}^2(n+1) &= w_{k,i}^2(n) - \lambda \delta_k^2(n) y_k^1(n), \\ w_{k,i}^{ag,2}(n+1) &= w_{k,i}^{md,2}(n+1) - \beta \delta_k^2(n) y_k^1(n). \end{aligned} \quad (3.24)$$

Por fim, para camada de entrada tem-se

$$\begin{aligned} w_{k,i}^{md,1}(n+1) &= (1-\alpha)w_{k,i}^{ag,1}(n) + \alpha w_{k,i}^1(n), \\ w_{k,i}^1(n+1) &= w_{k,i}^1(n) - \lambda \delta_k^1(n) x_i(n), \\ w_{k,i}^{ag,1}(n+1) &= w_{k,i}^{md,1}(n+1) - \beta \delta_k^1(n) x_i(n). \end{aligned} \quad (3.25)$$

Definido o equacionamento, é possível apresenta o algoritmo de treinamento, como é proposto abaixo.

Outro método a ser utilizado neste trabalho, os mínimos quadrados recursivo, que serve como uma outra base de estudos e para que se possa realizar comparações entres as metodologias aplicadas a identificação *online*.

3.6 Mínimos Quadrados Recursivos (RLS)

O método dos mínimos quadrados recursivo tem sua fundamentação no LS (ASTROM; WITTENMARK, 2008), que tem por objetivo a estimação paramétrica para modelos, mi-

Algoritmo 2: Treinamento Backpropagation Acelerado (BPAG)

Entrada: $w_{k,i}^l(0)$, $u(0)$, $\alpha \in (0, 1)$ para qualquer $n \geq 2$, $\beta > 0$ e $\lambda > 0$

Inicializar $x_{k,i}^{ag,l} = w_{k,i}^l(0)$ e $n = 0$;

para $n \leftarrow n + 1$ **faça**

 Calcular: y_k^3 e $\epsilon(n) = d(n) - y^3(n)$

 Calcule: δ_k^3 e Equação (3.23) - Camada de Saída

 Calcule: δ_k^2 e Equação (3.24) - Camada de Oculta

 Calcule: δ_k^1 e Equação (3.25) - Camada de Entrada

fim

minimizando o erro quadrático entre valor real e estimado. O algoritmo RLS, diferentemente do LS, trabalha com a questão da recursividade com abordagens do tipo adaptativas e com referências temporais em t e $t - 1$. Como exemplo das inúmeras aplicações, (HUYNH; WON, 2009) faz uma aplicação *online* que utiliza o algoritmo RLS para treinar uma RNA de forma *online* e em (YEH; SU, 2012) que aplica o RLS em um sistema neuro-fuzzy. Além desses exemplos, encontra-se aplicações em controle adaptativo, realização de observadores de estado em tempo real, dentre outras.

Baseado nesse contexto, é apresentado a seguir as duas principais variações do RLS, com base em (ASTROM; WITTENMARK, 2008).

A primeira variação é chamada de RLS padrão, que minimiza o erro médio quadrático, onde a atualização dos pesos sinápticos de uma rede neural artificial é dada por

$$\hat{\omega}(n) = \hat{\omega}(n - 1) - P(n)\varphi(n)\varphi^T(n)\hat{\omega}(n - 1) + P(n)\varphi(n)y(n), \quad (3.26)$$

sendo $P(n)$ a matriz de covariância, dada por $\varphi^T(n)\varphi(n)$, com $\varphi(n)$ sendo a matriz de dados e $y(n)$ a saída real ou alvo. Assim, colocando-se em evidência $P(n)\varphi(n)$, tem-se:

$$\hat{\omega}(n) = \hat{\omega}(n - 1) + P(n)\varphi(n) (y(n) - \varphi^T(n)\hat{\omega}(n - 1)), \quad (3.27)$$

sendo $\varepsilon(n) = y(n) - \varphi^T(n)\hat{\theta}(n - 1)$ o erro entre estimativa e valor real e $K(n) = P(n)\varphi(n)$, $P(n) = (\varphi^T(n)\varphi(n))^{-1}$ e $\varphi(n)$ o vetor de regressores ou de dados, chega-se a equação de recursividade dado por

$$\hat{\omega}(n) = \hat{\omega}(n - 1) + K(n)\varepsilon(n). \quad (3.28)$$

A segunda variação do RLS apresentado em (ASTROM; WITTENMARK, 2008), que diferentemente do caso anterior, se aplica a variações abruptas e com pouca frequência devido ao grande passo de $K(n)$, esta se aplica à variações mais frequentes, porém com maior suavidade. Para este caso considera-se como função de custo o erro médio quadrático dado por

$$V(\omega, n) = \frac{1}{2} \sum_{i=1}^n \lambda^{n-i} (y(n) - \varphi^T(n)\omega(n))^2 \quad (3.29)$$

onde λ é dito fator de esquecimento. Dessa forma o algoritmo RLS é formado por três equações dadas por

$$\begin{aligned} \omega(n) &= \omega(n-1) + K(n) (y(n) - \varphi^T(n)\omega(n-1)), \\ K(n) &= P(n)\varphi(n) = P(n-1)\varphi(n)(\lambda + \varphi^T(n)P(n-1)\varphi(n))^{-1}, \\ P(n) &= (I - K(n)\varphi^T(n)) P(n-1)/\lambda, \end{aligned} \quad (3.30)$$

sendo a primeira a de atualização dos pesos sinápticos da rede, a segunda sendo a atualização de $K(n)$, que é constituído pela matriz de covariância $P(n)$, de dados $\varphi(n)$ e fator de esquecimento λ , e a terceira a atualização da matriz de covariância.

Com base neste algoritmo e nos apresentados anteriormente, podem ser feitos estudos sobre a estimação paramétrica *online* tomando como referência a planta térmica em uso para este projeto.

Sistema Térmico e Medição Indireta

Este capítulo tem por finalidade, apresentar o sistema térmico para as aplicações dos algoritmos de estimação *online* utilizando inteligência computacional em face as suas não linearidades introduzidas pela célula Peltier, com intuito de representá-lo de forma funcional. Neste capítulo apresenta-se o levantamento de modelos clássicos não lineares, como o modelo de Hamerstein para que se possa realizar comparações e para ser utilizado nas simulações envolvendo a estimação inteligente de temperatura, bem como o estudo, desenvolvimento e avaliação da influência, no sistema como um todo, de um sistema de medição indireta de temperatura, que possui grande potencial para aplicações industriais.

4.1 O Sistema Térmico

O sistema térmico em questão, tem capacidade tanto de aquecer quanto de resfriar pequenos objetos ou ambientes. No caso deste trabalho, trata-se do levantamento de modelos e controle da temperatura em um objeto cúbico de alumínio interno a uma câmara térmica, conforme apresenta a Fig. 4.1.

De forma geral, tem-se uma caixa térmica de isopor, três sensores de temperatura do tipo LM35 (NATIONAL, 2000), onde um é destinado para medição da temperatura do ambiente externo (Sensor 3), um para medir a temperatura do ar interno a câmara (Sensor 1) e um diretamente em contato com o objeto (Sensor 2).

Já a célula Peltier é tida como atuador térmico que apresenta características não lineares. Esta se baseia no princípio de Peltier, que diz que ao passar uma corrente elétrica por uma junção entre dois materiais, sejam eles condutores ou semicondutores, um diferencial de temperatura irá surgir formando uma face fria e uma quente, onde o que determina qual face resfriará e qual aquecerá são as características químicas e físicas dos materiais.

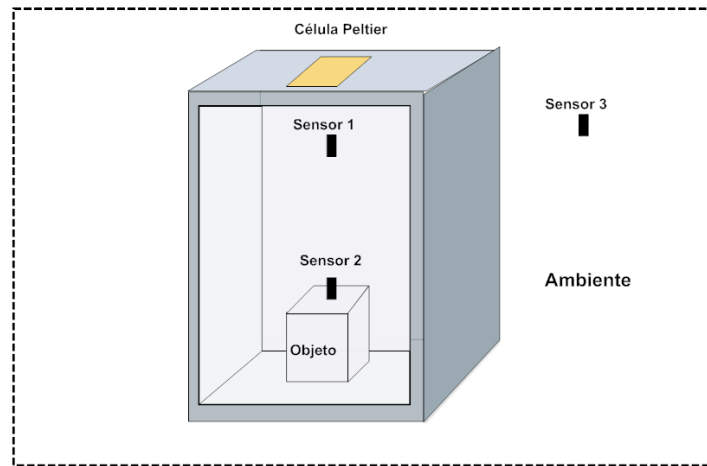


Figura 4.1: Câmara térmica

Com base nessa explicação, o atuador térmico em questão se caracteriza pela composição de vários blocos de materiais semicondutores p e n , formando junções $p - n$ e $n - p$, ligados em série, onde junções do mesmo tipo formam duas faces distintas (LIMA G.S. DEEP; FONTANA, 2001).

Dados da Planta:

- Célula TEC1-1209;
- Câmara térmica de 17cm por 25cm ;
- Objeto cúbico de alumínio de aresta $2,5\text{cm}$.

4.1.1 Modelo da Célula

O modelo para a potência fornecida por uma face da célula, obtidas em (LIMA G.S. DEEP; FONTANA, 2001), é dada por

$$P_h = \frac{I^2 R_m}{2} + \alpha_{p,n} T_h I, \quad (4.1)$$

sendo R_m a resistência interna, $\alpha_{p,n}$ uma constante (Constante de Seebeck), T_h a temperatura do lado quente e I , a corrente elétrica.

Pode-se verificar que a Eq.(4.1) é uma função não linear, já que depende do quadrado da corrente aplicada e da própria temperatura T_h , o que na conjuntura geral se faz necessária a

utilização de técnicas de identificação e modelagem adequadas, sendo proposto neste trabalho a utilização de RNA's para estimação paramétrica *online* para fins de controle, onde são avaliados algoritmos de treinamento, utilizando a rede como modelo para representar a planta.

Além da estimação *online*, trata-se a questão do sistema de medição indireta (SMI) de temperatura, onde pretende-se utilizar as medições através do sensor 1 como referência para se gerar uma estimativa sobre a temperatura no objeto. Para isso, além dos dados do sensor 1, são utilizadas as medições obtidas do sensor 2 para se determinar um modelo que relacione a temperatura do ambiente interno a do objeto, caracterizando a medição indireta de temperatura no objeto, utilizando modelagem e técnicas de estimação paramétrica.

Avaliar a influência e viabilidade do SMI também se faz necessário, o que resulta, além da validação dos modelos, análises no domínio da frequência, ou seja, verificar alterações nas margens de ganho e de fase, comparando o sistema com e sem SMI, permitindo que se obtenha conclusões sobre as vantagens e desvantagens desta aplicação.

4.1.2 Modelo da Planta

O sistema térmico que é apresentado na Fig. 4.1 tem seus princípios baseados nas leis de transferência de calor (OGATA, 2011). Baseado nessas leis e no modelo do atuador térmico, Eq.(4.1), será possível, avaliar as não linearidades, levantar modelos que servirão de referência para as estimações paramétricas dos modelos clássicos para futuras comparações, para determinação e avaliação do SMI, e permite discernir sobre a estruturas das redes que são desenvolvidas.

Seja o sistema térmico da Fig. 4.1, onde a célula Peltier é instalada no centro superior da câmara, obtêm-se as equações que representam a sua dinâmica, relacionando a transferência de energia da célula para o ambiente interno e do ambiente interno para o objeto, que são dadas por

$$m_o c_o \frac{dT_o}{dt} = K_{in,o}(T_{in} - T_o), \quad (4.2)$$

$$m_{in} c_{in} \frac{dT_{in}}{dt} = K_{in,o}(T_o - T_{in}) + K_{in,ext}(T_{ext} - T_{in}) + u_{célula}, \quad (4.3)$$

sendo m_o e m_{in} as massas do objeto e do ar interno a câmara, c_o e c_{in} os calores específicos, $K_{in,o}$ a condutibilidade térmica do ambiente interno para o objeto, $K_{in,ext}$ representa a perda

para o ambiente externo, T_o , T_{in} e T_{ext} as temperaturas no objeto, no ar interno e externo, respectivamente, e $u_{célula}$ a quantidade de calor fornecida pelo atuador.

Conhecida as equações diferenciais é possível definir o sistema de medição indireta, que é dado pela Eq.(4.2), a temperatura do ar com relação a energia fornecida pela célula ($u_{célula}$), Eq.(4.3). Com isso, tem-se em mãos o necessário para se caracterizar o sistema, onde são utilizados métodos de estimação para se determinar os parâmetros destas funções. Assim com base na Eq. (4.2) tem-se que para o sensor virtual uma função de transferência de primeira ordem, dada por

$$\frac{T_o(s)}{T_{in}(s)} = \frac{k_{in,o}}{m_o c_o s + 1}, \quad (4.4)$$

sendo a função de transferência que relaciona a temperatura no objeto com a energia fornecida pela célula dada pela substituição da Eq.(4.2) na Eq.(4.3), resultando na equação dada por

$$\frac{T_o(s)}{U_{célula}(s)} = \frac{k_{in,o}}{(m_o c_o m_{in} c_{in}) s^2 + [(k_{in,o} - k_{in,ext}) m_o c_o + m_{in} c_{in}] s + (k_{in,o} - k_{in,ext})}, \quad (4.5)$$

o que caracteriza a representação do sistema ou planta térmica por um modelo de segunda ordem.

4.2 Modelo para o Núcleo do Sistema de Medição Indireta (SMI)

Visto a grande importância da medição indireta, como exemplifica os vários trabalhos apresentados no Capítulo 1, nesta seção apresenta-se a estimação paramétrica deste sistema, bem como as análises do ponto de vista de estimação paramétrica de modelos matemáticos do sistema térmico que é representado na Fig.4.1. Como pode ser visto na Eq.(4.2), o modelo que referencia a temperatura no objeto (*sensor 2* da Fig.4.1) em função da temperatura do ambiente interno (*sensor 1* da Fig.4.1) é de primeira ordem. Isso facilita o processo de estimação, já que a forma da equação que representa a dinâmica desse sistema é conhecida. Neste caso e no restante deste trabalho, é feita uma abordagem discreta, utilizando um tempo de amostragem de $T_s = 60s$, que foi obtido com base na análise dos dados da planta, medidos através de um sistema de aquisição de dados. Os dados utilizados para estimação do SMI

podem ser vistos na Fig.4.2, onde o período de amostragem escolhido representa um período de tempo de 5 a 7 vezes menor que o tempo de resposta do sistema (LANDAU, 1938).

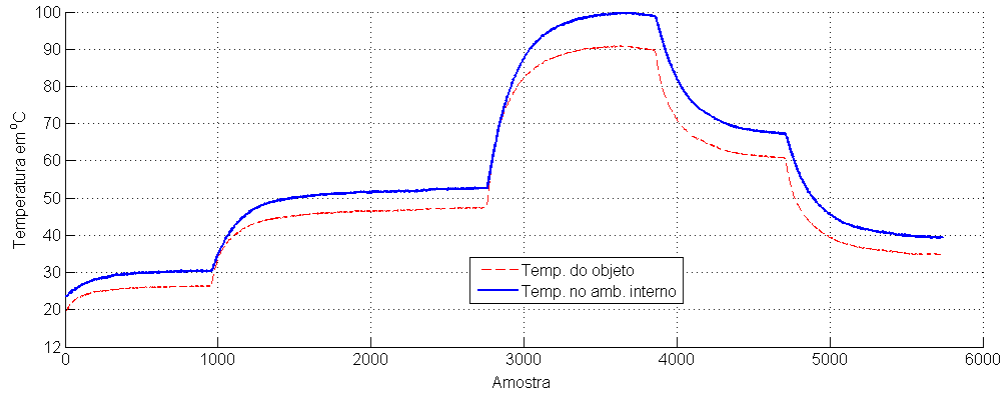


Figura 4.2: Dados para estimação do núcleo do SMI, medidos de 4 em 4 segundos

A planta térmica, como foi visto, é composta por um cubo de alumínio, uma célula Peltier, uma caixa de isopor e dois sensores, um situado no objeto e outro no interior da câmara, com o objeto situado no fundo da câmara e o sensor que mede a temperatura do ar interno a 16cm do mesmo. Na medição das temperaturas foi observado uma diferença de 10 graus, entre a temperatura do objeto e a temperatura do ar, medido na parte superior da câmara, sendo que o objeto está situado na parte inferior a câmara. A diferença de temperatura pode ser explicada pelas perdas existentes no sistema, devido ao mau isolamento térmico da tampa frontal, da face fria com relação a face quente e pela distância entre a fonte de calor e o objeto, sendo a transmissão de calor inversamente proporcional a distância.

Após todo o trabalho desenvolvido, foi realizado uma melhoria na isolação entre as faces da célula e uma aproximação do sensor que mede a temperatura do ar interno ao objeto, fazendo com que o diferencial de temperatura fosse reduzido, como apresenta a Fig.4.3.

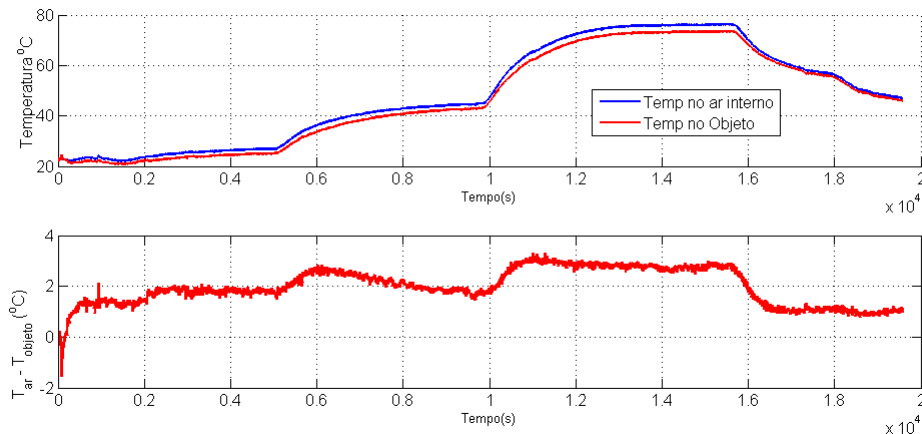


Figura 4.3: Medição para verificação das diferenças de temperatura entre ar interno e objeto

Pode-se observar na Fig. 4.3 que a diferença de temperatura entre ar interno e o objeto foi reduzida devido a aproximação do objeto da fonte de calor, que passou de aproximadamente 21cm para 10cm, porém ainda existe um gradiente de temperatura, já que o sistema não é ideal.

Estes dados são utilizados apenas para verificação da melhoria de rendimento da planta com o objetivo de reduzir a diferença de temperatura entre a fonte e o objeto e não para as estimativas.

Com os dados da Fig.4.2, é realizado uma análise da característica estática do sistema, relação entre temperatura do ambiente interno e temperatura do objeto em regime permanente, para verificar se o sistema se caracteriza, realmente, como linear ou não linear. Com isto, é definido se a estrutura do modelo é do tipo ARX ou NARX. Portanto, a característica estática para o SMI pode ser vista na Fig.4.4.

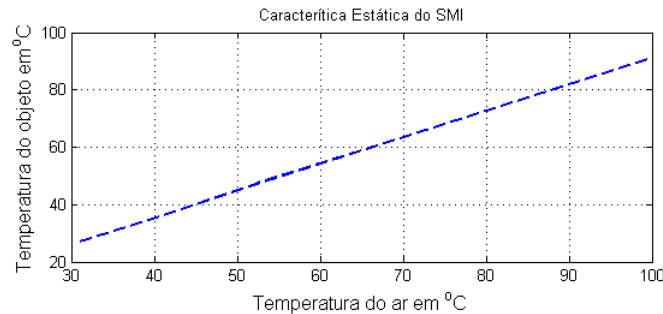


Figura 4.4: Característica estática para o modelo do SMI

É possível observar, através Eq.(4.4) que o sensor virtual se trata de um sistema de primeira ordem com característica linear, como apresenta a Fig.4.4, e portanto uma estrutura ARX pode ser utilizada. Para o modelo a ser estimado, a ordem já é conhecida, portanto utiliza-se uma estrutura ARX de primeira ordem. Além disso, um modelo de ordem 2 também é levantado, para se avaliar a resposta do modelo quanto ao aumento da ordem e realizar comparações com relação ao de primeira ordem.

4.2.1 Estimação do Núcleo do SMI

Como já foi dito anteriormente, o SMI possui, pelas equações físicas do sistema, um modelo de primeira ordem e com isso são feitas estimativas com estruturas de primeira e segunda ordem para as primeiras avaliações e caso não satisfaça, ordens maiores são avaliadas.

Para estimação dos parâmetros, o método dos mínimos quadrados (LS) é utilizado para fornecer os parâmetros θ_i ótimos que minimizam um erro quadrático $\frac{1}{2} \sum e^2$, sendo φ denominada matriz de regressores, e y a saída desejada.

Dados os modelos de primeira e segunda ordem baseados na estrutura ARX, são estimados os parâmetros θ_i , utilizando o método dos mínimos quadrados, sendo a estrutura ARX de primeira ordem dada por

$$\hat{T}_{obj}(k) = \theta_1 \hat{T}_{obj}(k-1) + \theta_2 T_{ar}(k) + \theta_3 T_{ar}(k-1), \quad (4.6)$$

e a estrutura ARX de segunda ordem dada por

$$\begin{aligned} \hat{T}_{obj}(k) = & \theta_1 \hat{T}_{obj}(k-1) + \theta_2 \hat{T}_{obj}(k-2) + \\ & \theta_3 T_{ar}(k) + \theta_4 T_{ar}(k-1) + \theta_5 T_{ar}(k-2), \end{aligned} \quad (4.7)$$

sendo a estimação dos parâmetros da Eq.(4.6) caracterizado como uma modelagem do tipo caixa cinza, visto que se conhece a forma mas não os parâmetros e para a Eq.(4.7), caracteriza-se como sendo do tipo caixa preta, já que esta estrutura não se baseia nas equações físicas do sistema. Assim este modelo de segunda ordem é utilizado caso o de primeira ordem não apresente resultados satisfatórios.

Para estimação dos parâmetros, foi utilizado um conjunto de dados obtido através de um sistema de aquisição de dados de forma direta, através de sensores LM35 com tempo de amostragem de quatro segundos, varrendo o máximo de pontos de operações que os equipamentos permitiram, onde foi observado que o tempo de amostragem necessária era de 60s. Assim, com base nesses dados e utilizando $T_s = 60s$, foram estimados para cada equação, Eq.(4.6) e Eq.(4.7), respectivamente $\theta_{ord1} = [0, 8094 \ 1, 3737 \ -1, 2017]$ e $\theta_{ord2} = [1, 0440 \ -0, 2038 \ 1, 2508 \ -1, 2633 \ 0, 1568]$, tendo suas respostas representadas na Fig.4.5.

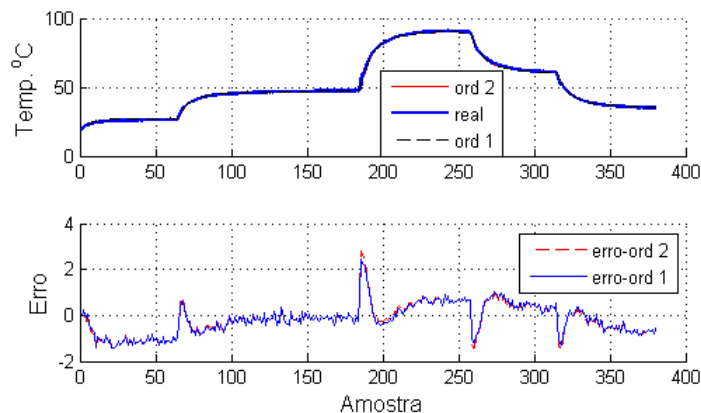


Figura 4.5: Estimação de primeira e segunda ordem para o modelo do SMI

Observa-se que os modelos de ambas as ordens estão sobrepostos, juntamente ao valor

real. Em uma análise prévia, pode-se observar que os erros também estão muito próximos, embora o modelo de ordem 1 apresente erro levemente menor. Isto leva a escolha do modelo mais simples, que em caso de uma implementação, exigirá menor esforço computacional.

4.2.2 Validação do Modelo do SMI

Para a validação do modelo, foram utilizados dados que diferem dos utilizados para estimação paramétrica. Dessa forma será possível avaliar melhor os resultados. Na Fig.4.6 pode ser observado a resposta real e estimada, a duas entradas diferentes.

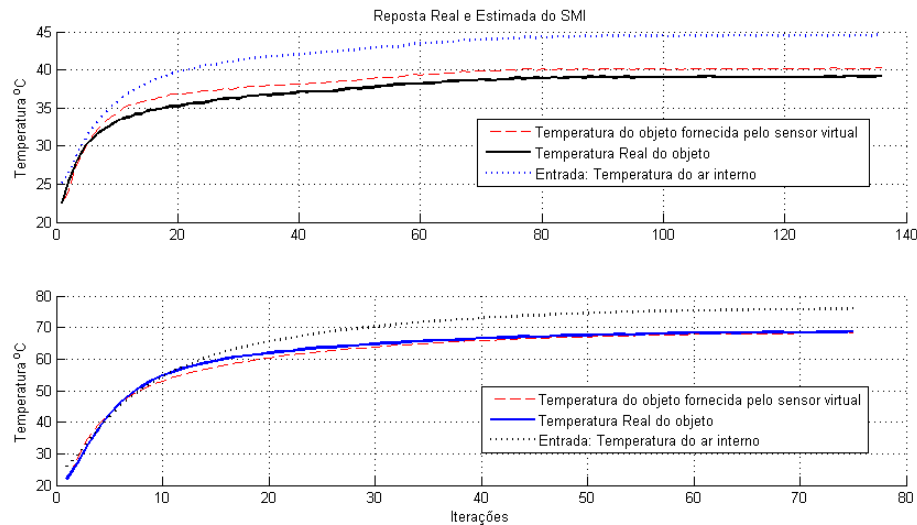


Figura 4.6: Validação para o modelo do SMI

Neste caso, apesar da existência de uma diferença no gráfico superior da Fig.4.6, o erro é de aproximadamente um, ou seja dentro dos 2% dos sensores em geral. Já no gráfico inferior, a resposta se mostrou melhor com relação a primeira, o que leva a aceitar o modelo como uma boa aproximação do sistema real. A sua simplicidade proporciona fácil implementação, a redução de custos e chances menores de falhas devido a retirada de um dos sensores.

4.3 Modelo Não Linear do Sistema Térmico

Nesta seção são abordadas, as estimações da temperatura do ambiente interno e do objeto com relação a tensão aplicada a célula, que possui característica não linear, com o objetivo de avaliar a influência do SMI no sistema como um todo e realizar comparações com os modelos utilizando estimação paramétrica inteligente, que são apresentadas nos capítulos posteriores.

Para isso, modelos não lineares de Hammerstein são utilizados com base na estrutura ARX. Assim, da mesma forma que foi realizada para o SMI, é elevada a ordem do sistema a ser estimado até que se obtenha um modelo satisfatório.

A primeira etapa para se determinar o modelo de Hammerstein, é a obtenção da característica estática do sistema. Dessa forma, na Fig.4.7 são apresentadas as características estáticas da temperatura do ambiente interno, denominado de T_{ar} e do objeto com relação a tensão de entrada sendo seus respectivos polinômios estimados para representá-las, dados por

$$f_{ar}(u) = -0,0651u^4 + 1,664u^3 - 14,0497u^2 + 50,5988u - 19,0829, \quad (4.8)$$

para característica estática da temperatura do ambiente interno e

$$f_{obj}(u) = -0,056u^4 + 1,451u^3 - 12,3016u^2 + 44,8405u - 17,633, \quad (4.9)$$

para característica estática da temperatura no objeto. Estas funções foram obtidas a partir da Fig. 4.7 e as mesmas também estão representadas de forma gráfica na figura.

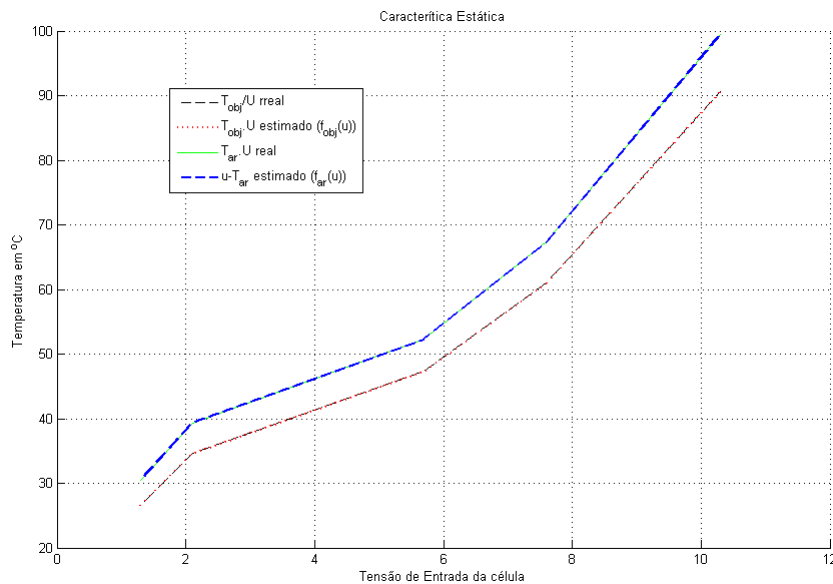


Figura 4.7: Característica estática real e estimada do sistema não linear

Ambas as estimativas para característica estática mostram bons resultados. Com essas representações das características estáticas, pode-se avançar para o processo de determinação

do modelo para planta com base na estrutura de Hammerstein.

4.3.1 Estimação paramétrica da planta

Como já mencionado anteriormente, é utilizada uma estrutura do tipo NARX na forma proposta por Hammerstein, para se encontrar uma representação matemática para o sistema térmico não linear. Da mesma forma, assim como foi realizada para o SMI, é utilizado o método dos mínimos quadrados para determinar a parte linear do modelo de Hammerstein, tendo em vista as Eq. (4.8) e (4.9).

Portanto, primeiramente é realizada a estimação referente a temperatura no ambiente interno a câmara, em relação a tensão de entrada e posteriormente em relação a do objeto, e uma análise da influência do SMI no sistema de forma geral.

No caso da temperatura do ar interno a câmara, duas representações foram encontradas, uma de segunda ordem, caracterizado como estimação caixa cinza e um de terceira, caracterizado como caixa preta para comparação, sendo a estimativa de ordem dois dada por

$$\hat{T}_{ar}(k) = 0,9524\hat{T}_{ar}(k-1) - 0,03\hat{T}_{ar}(k-2) + 0,0631f_{ar}(u(k-1)) + 0,0152f_{ar}(u(k-2)) \quad , \quad (4.10)$$

e de ordem três, dada por

$$\hat{T}_{ar}(k) = 0,97\hat{T}_{ar}(k-1) - 0,0289\hat{T}_{ar}(k-2) - 0,0158\hat{T}_{ar}(k-3) + 0,0632f_{ar}(u(k-1)) + 0,0137f_{ar}(u(k-2)) - 0,0015f_{ar}(u(k-3)) \quad , \quad (4.11)$$

sendo \hat{T}_{ar} a temperatura do ar estimada e u a tensão aplicada a célula.

Entre os dois modelos apresentados, ambos tiveram respostas bem próximas, ou seja, a diferença foi irrelevante entre os modelos de segunda e terceira, como mostra a Fig.4.8.

Neste caso escolhe-se a função mais simples, Eq.(4.10). Outros modelos de primeira, quarta e quinta ordem também foram estimados, porem não houveram mudanças relevantes.

A outra estimação necessária é a que relaciona diretamente a temperatura do objeto com a tensão aplicada. A mesma metodologia para o caso da temperatura do ambiente interno foi utilizada, fazendo-se uso da Eq.(4.9), o modelo de ordem dois é dado por

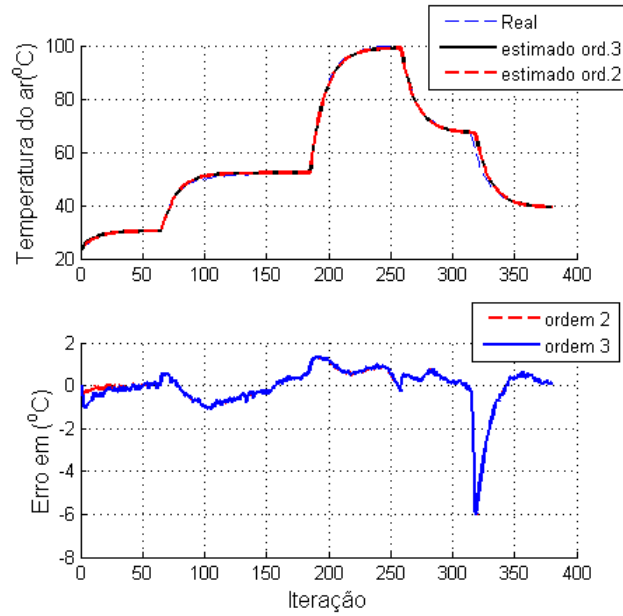


Figura 4.8: Estimação para temperatura no ar em relação a tensão na célula

$$\hat{T}_{obj}(k) = 10,0388\hat{T}_{obj}(k-1) - 00,1074\hat{T}_{obj}(k-2) + 00,096f_{obj}(u(k-1)) - 00,0304f_{obj}(u(k-2)) \quad (4.12)$$

e de ordem três, é dado por

$$\begin{aligned} \hat{T}_{obj}(k) = & 10,0482\hat{T}_{obj}(k-1) - 0,037\hat{T}_{obj}(k-2) - \\ & 0,0683\hat{T}_{obj}(k-3) + 0,0984f_{obj}(u(k-1)) - \\ & 0,0204f_{obj}(u(k-2)) - 0,0234f_{obj}(u(k-3)) \end{aligned} \quad (4.13)$$

Assim como nos casos anteriores, é apresentada uma segunda equação de ordem diferente da escolhida, afim de realizar comparações e mostrar que não há diferenças relativas sobre os dados utilizados na estimação. Para o caso da Fig.4.9 é mostrada essa estimação, apresentando os resultados para segunda e terceira ordens, e para o caso da temperatura no objeto.

Para este caso, o resultado da estimação piorou com relação à feita para a temperatura do ar interno, mas foi satisfatória. O erro apresentou picos maiores devido a uma pequena diferença no regime transitório.

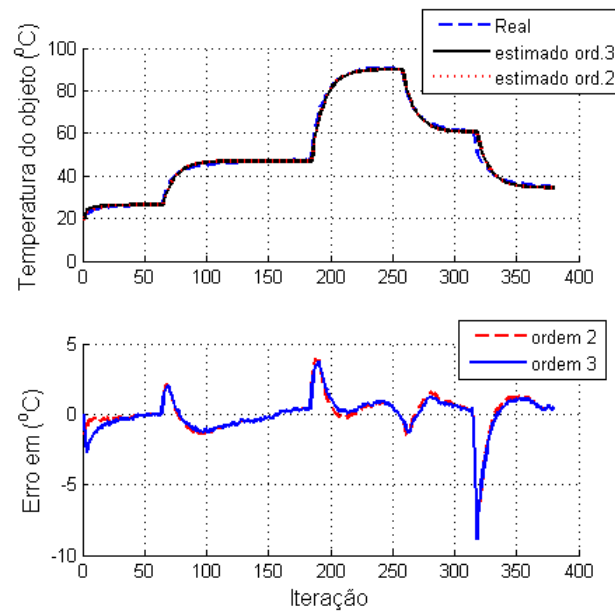


Figura 4.9: Estimação para temperatura no objeto em relação a tensão na célula

4.3.2 Validação

Para validar os modelos escolhidos, Eq.(4.12) e (4.10), assim como do SMI, é feito uso de dados diferentes dos utilizados para estimação, conforme a Fig.4.10, que mostra a resposta dos dois sistemas para entradas de 3V e 12V.

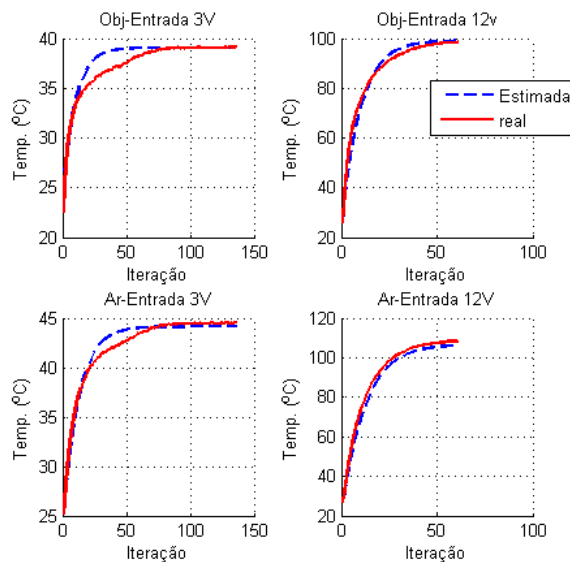


Figura 4.10: Validação para os modelos referentes ao ar e objeto

Ao observar os gráficos, é possível perceber que ambos alcançam o regime com valor muito próximo ao valor real. Para a entrada de 3V houve uma diferença de dinâmica no

transitório, onde neste caso, a realização de pequenos ajustes e a utilização de estimativas de ordens mais elevadas não corrigiram o erro.

De uma maneira geral, considera-se possível a utilização destes modelos, para determinadas aplicações, como controle, por exemplo.

4.4 Análise dos Modelos Levantados

Um das primeiras perguntas que podem ser feitas, é o porque do uso do modelo de Hammerstein. O fator primordial que levou a sua utilização, é a análise em cima de sua parcela linear. Outras estruturas, como a utilização do modelo de Wiener (LJUNG, 1998), é que possuem função inversa, o que dificulta um pouco o processo de estimação, pois limita a utilização somente de funções inversíveis.

Duas análises foram consideradas para avaliar os modelos obtidos e verificar a influência do SMI sobre o sistema, sendo as análises de estabilidade e de frequência.

Como está se trabalhando com sistemas discretos, a condição necessária para garantir a estabilidade é verificar se estes possuem polos dentro do círculo unitário, como pode ser observado na Tabela 4.4.

Tabela 4.1: Polos dos modelos

Modelo/polos	Polo 1	Polo 2
Modelo do SMI	0,809	-
Modelo Temp. do ar interno/Tensão	0,0297	0,92
Modelo Temp. do Obj/Tensão	0,0699	0,902

Os polos do sistema apresentam estabilidade. Um outro fator importante a se observar é que os pólos do sistema para os modelos que referenciam temperatura em função da tensão de entrada são bem próximos. Isso se justifica devido as dinâmicas parecidas.

Mais uma análise se faz necessária, esta é para verificar a influência do sistema de medição indireta sobre o conjunto SMI e $\hat{T}_{ar}(z)/U(z)$. Para isso, a análise em frequência indicará se há mudança nas margens de ganho e de fase.

A metodologia utilizada nesta análise, trata de multiplicar a função de transferência do SMI (transformada z da Eq.(4.6)) a função de transferência dada pela transformada z da Eq.(4.10) e realizar a análise em frequência em comparação a transformada z da Eq.(4.12).

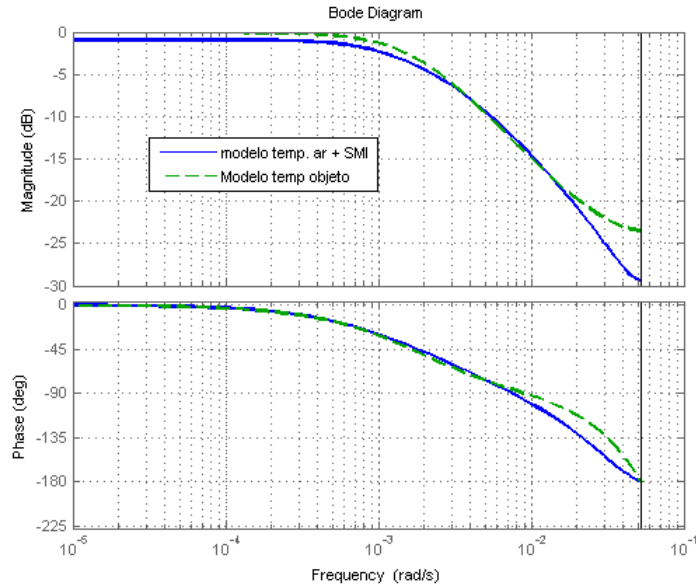


Figura 4.11: Diagrama de Bode para verificação da influência do SMI

Portanto, observando a Fig.4.11, que apresenta a resposta em frequência do sistema com e sem SMI, tem-se que, para o conjunto SMI e $\hat{T}_{ar}(z)/U(z)$ obteve-se uma margem de ganho de $29dB$ e para $\hat{T}_{obj}(z)/U(z)$ obteve-se margem de $23dB$, e em termos de margem de fase, ambos apresentaram resposta infinita. Da mesma forma, observa-se que a largura de banda BW também se mostrou igual para ambos os caso, mostrando que não há diferenças consideradas significativas.

Tais análises indicam que a utilização do sistema da medição indireta não causa uma mudança brusca na resposta em frequência, tornando-o factível de utilização, proporcionando diversas vantagens já mencionadas, onde pode-se afirmar que o balanço benefício/influência possui fator positivo.

Analisando os dados da planta, foi constatado que o atraso puro de tempo do sistema é muito menor que o tempo de resposta da planta, estando o atraso por volta de 8 segundos, enquanto a constante de tempo da planta é de $600s$. Assim, para o tempo de amostragem escolhida, que foi de 60 segundos, a inclusão de atrasos de z^{-n} , a mais, é inviável pois distorceria a resposta estimada do real. Além disso o modelo foi baseado em dados reais onde o ARX obtido já leva em consideração tal atraso puro de tempo. Com isso, pode-se afirmar que os modelos obtidos através de uma estimação com base em dados reais representam de forma satisfatória o sistema.

Estimação Paramétrica Online Inteligente

Este capítulo apresenta a estimação paramétrica *online* que faz uso de redes neurais com estruturas diversificadas e realimentadas, tais como redes na forma espaço de estados e ARX, para que sirvam como modelo representativo do sistema térmico sendo estudados e desenvolvidos três algoritmos distintos, dedicados ao treinamento de RNA's, tais como: Backpropagation, Backpropagation acelerado e o RLS, aplicando-se à dois estudos de caso, cujas estruturas das redes são baseadas na identificação realizada no Capítulo 4.

5.1 Estruturas das RNA's Online

Para o desenvolvimento dos algoritmos de treinamento, são propostas duas estruturas *online* de redes que possibilitam a utilização dos três algoritmos apresentados no Capítulo 3. Assim para cada caso será analisado o Backpropagation normal e com aceleração de gradiente e algoritmos da família RLS para que se possa discernir sobre o desempenho de cada um e compará-los, também, com os modelos não lineares cuja identificação paramétrica foi realizada em batelada, apresentadas no Capítulo 4.

Estruturas mais complexas não foram utilizadas, pois seria difícil se embarcar em microcontroladores de baixo custo e de arquitetura simples. Assim, as estruturas foram escolhidas de forma que se possa utilizar tanto Backpropagation quanto RLS, possibilitando os diversos tipos de controle. Um outro fator que influenciou na escolha das estruturas, foi que através dos cálculos e desenvolvimento algébrico utilizando redes MLP com regressão, onde obteve-se uma combinação múltipla dos pesos que multiplicados pelo vetor de regressores, resulta nas formas escolhidas, mas apresentado, uma quantidade maior de pesos a estimar, tendo por consequência um esforço computacional maior.

5.1.1 Estrutura RNA/ARX

A primeira estrutura de rede artificial   baseada na estrutura de um modelo ARX, compondo um combinador linear do sinal de entrada e a realimentaco da saıda. A utilizaco desta composico visa o projeto de controladores adaptativos que tem como crit rios de projetos as figuras de m rito ξ , t_s , t_r e M_p , que so respectivamente o coeficiente de amortecimento, o tempo de subida, o tempo de acomodaco e o valor de pico m ximo no transit rio. A estrutura RNA/ARX pode ser vista na Fig.5.1.

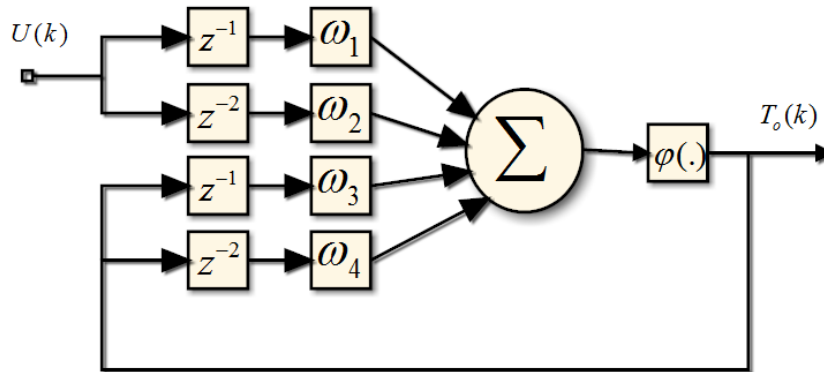


Figura 5.1: Estrutura RNA/ARX para controle

Em particular, esta estrutura   utilizada para o projeto, de forma mais direta, dos controladores cl ssicos, como os da fam lia PID, onde al m das vers es cl ssicas, pode-se abordar uma vers o adaptativa atrav s de v rias t cnicas como chaveamento, infer ncia *fuzzy* e RNA's, como   apresentada em (SA DENIS F.;FONSECA NETO, 2014). Assim analiticamente, sendo ω_i os pesos sinpticos da redes e φ a funcao de ativacao, a saıda da rede da Fig.5.1   dada por

$$T_o(n) = \varphi \left(\begin{bmatrix} U(n-1) & U(n-2) & T_o(n-1) & T_o(n-2) \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ \omega_4 \end{bmatrix} \right), \quad (5.1)$$

sendo $\begin{bmatrix} U(n-1) & U(n-2) & T_o(n-1) & T_o(n-2) \end{bmatrix}$ o vetor de regressores.

Atrav s da observaco da Eq.(5.1) fica f cil observar o vetor de regressores que ser  utilizada no RLS, bem como a ordem, onde se φ for no linear, a estrutura caracteriza-se como sendo um RNA/NARX. Em se tratando de um sistema com aprendizado online, φ pode assumir tanto a forma linear quanto no linear, j  que $\omega(n) = \begin{bmatrix} \omega_1(n) & \omega_2(n) & \omega_3(n) & \omega_4(n) \end{bmatrix}'$

é variável no tempo, onde a rede é dita linear com parâmetros variáveis no tempo, ou seja um sistema LTV.

5.1.2 Estrutura RNA no Espaço de Estados

A segunda estrutura em questo se baseia na representao no espao de estados. Para este caso podem ser estudados diretamente a observabilidade e controlabilidade do sistema em tempo real, onde destaca-se pela realimentao dos estados, como apresenta a Fig.5.2. Como pode ser observado, esta se caracteriza pela realimentao de dois estados, SISO (Sistema de uma entrada e uma saıda), onde as mesmas consideraes para o primeiro caso (RNA/ARX) podem ser consideradas. Dessa forma, a RNA assume a estrutura dada por

$$x(n) = A(\omega(n))x(n - 1) + B(\omega(n))U(n - 1), \tag{5.2}$$

sendo a saıda dada pelo produto de uma matriz constante C e os estados $x(n)$, resultando na equao dada por

$$y(n) = Cx(n). \tag{5.3}$$

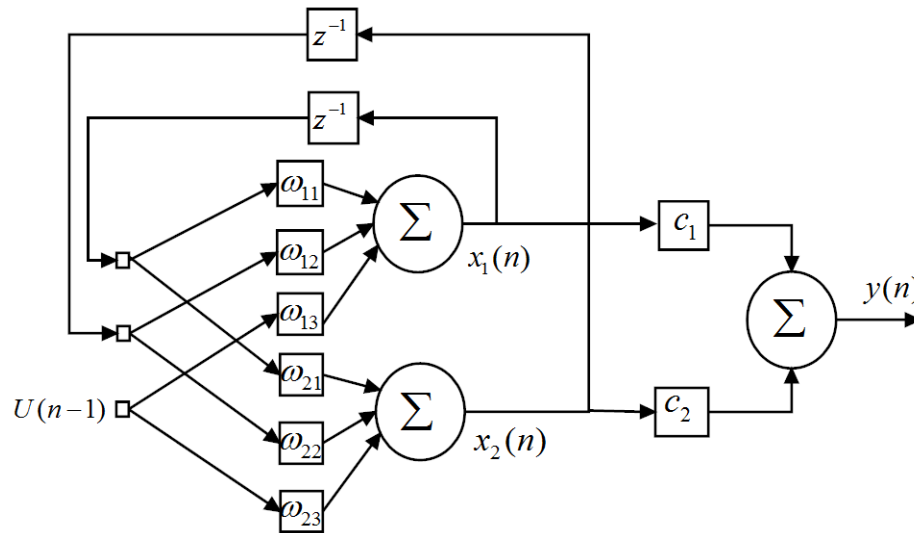


Figura 5.2: Estrutura RNA no Espaço de Estados

Para este estudo, sendo $\omega_{k,1}$ os pesos sinápticos pertencentes ao neurônio que pondera o estado $x_1(n)$ realimentado e $\omega_{k,2}$ os pesos que ponderam o estado $x_2(n)$ realimentado, com cada neurônio possuindo suas respectivas funções de ativação φ , a equações no espaço de estados da rede são dadas por

$$\begin{bmatrix} x_1(n+1) \\ x_2(n+1) \end{bmatrix} = \varphi \left(\begin{bmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{21} & \omega_{22} & \omega_{23} \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \\ U(n) \end{bmatrix} \right). \quad (5.4)$$

O modelo da rede representado pela Eq.(5.4), est na sua forma generalizada no linear. Porm ser considerando que $\varphi(\cdot)$ seja uma funo linear, que conseqentemente a equaco de estado da rede passa a ser dada por

$$\begin{bmatrix} x_1(n+1) \\ x_2(n+1) \end{bmatrix} = \begin{bmatrix} \omega_{11} & \omega_{12} & \omega_{13} \\ \omega_{21} & \omega_{22} & \omega_{23} \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \\ U(n) \end{bmatrix}. \quad (5.5)$$

Com isso  possvel apresent-la na forma $x(n) = A_k x(n-1) + B_k U(n-1)$, onde as matrizes A_k e B_k so compostas pelos pesos sinpticos, que so dados por

$$\begin{bmatrix} x_1(n+1) \\ x_2(n+1) \end{bmatrix} = \begin{bmatrix} \omega_{11} & \omega_{12} \\ \omega_{21} & \omega_{22} \end{bmatrix} \begin{bmatrix} x_1(n) \\ x_2(n) \end{bmatrix} + \begin{bmatrix} \omega_{13} \\ \omega_{23} \end{bmatrix} U(n). \quad (5.6)$$

Determinadas as estruturas e equacoes para os dois estudos de caso,  possvel passar para o processo de estimaco dos pesos sinpticos, bem como as anlises de convergncia e exatido para cada algoritmo de treinamento desenvolvido.

5.2 Estimaco com Algoritmos de Treinamento Backpropagation

Nesta seo  utilizado um mtodo de treinamento de redes neurais considerado clssico. Neste estudo o algoritmo  aplicado para dois casos, sendo o caso 1 a RNA/ARX e caso 2 a RNA no espao de estados, como j mencionado, que diferentemente das metodologias clssicas de estimaco no h um processo de validao mas sim um estudo referente a capacidade de adaptao, convergncia, esforo computacional e exatido. Assim,  apresentado a seguir as aplicao dos algoritmos desenvolvidos para cada caso.

- **Caso 1:** Treinamento Backpropagation clssico para a RNA/ARX:

Neste primeiro tratamento,  utilizado o modelo RNA/ARX, onde  analisado o desempenho do algoritmo em trs situaes, sendo nesta primeira instncia visto o comportamento, tendo Backpropagation apenas uma iterao para aproximar a sada da RNA do sistema real, como na Fig.5.3, que mostra a sada da rede e a sada real, o erro instantneo e os parmetros da rede.

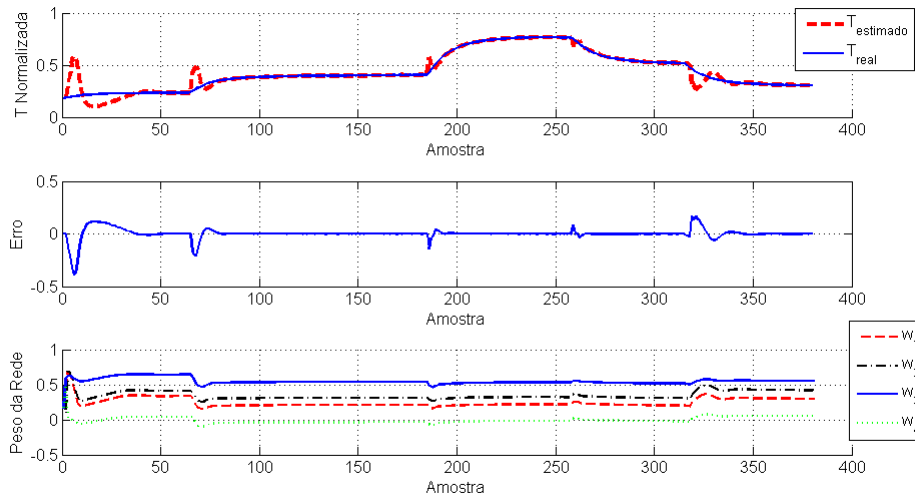


Figura 5.3: Temperatura do objeto, Erro e Pesos com treinamento BP - RNA/ARX para uma iterao por amostra

Com estes grficos, analisa-se o comportamento da rede e percebe-se uma certa dificuldade de adaptao para a mudana de dinmica, concentrando-se picos maiores no incio do treinamento. Aps o sistema entrar em regime permanentes no h problemas, como se pode ver no grfico do erro. Na mesma figura pode ser visto o comportamento dos pesos sinpticos da rede que para este caso sofreram leves mudanas sem extrapolar a zona entre -1 e 1 , implicando em uma no saturao dos pesos. Estes testes foram feitos utilizando valores iniciais aleatrios para rede, buscando verificar a convergncia da mesma seja qual for seu ponto de partida.

Considerando a mesma situao, a Fig.5.4 apresenta os autovalores das funes estimadas para cada amostra e a energia acumulada no treinamento de cada ponto dada pela soma do erro quadrtico, onde se obtm modelos estveis basicamente para todo o processo, com exceo dos pontos iniciais, que se caracteriza tambm possuir o maior pico de energia.

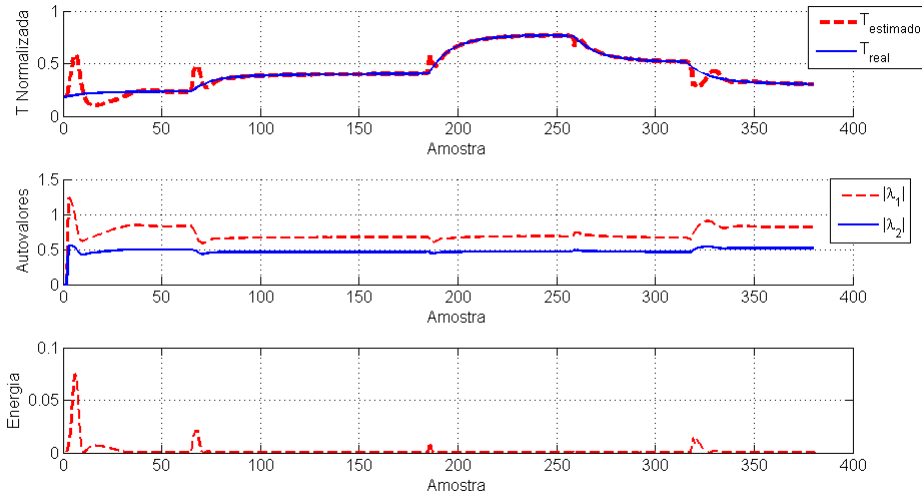


Figura 5.4: Temperatura no objeto, Autovalores e Energia com treinamento BP- RNA/ARX para uma iterao por amostra

No segundo teste realizado, foram consideradas dez iteraes por amostra, como apresenta as Figuras 5.5 e 5.6, apresentando as mesmas informaes que as da situao anterior.

Neste caso o algoritmo apresentou um comportamento mais estável com menor erro e minimizao dos picos na mudana de dinâmica além da existênciade picos no gráfico de erro, que ocorre devido a mudana de dinâmica da planta.

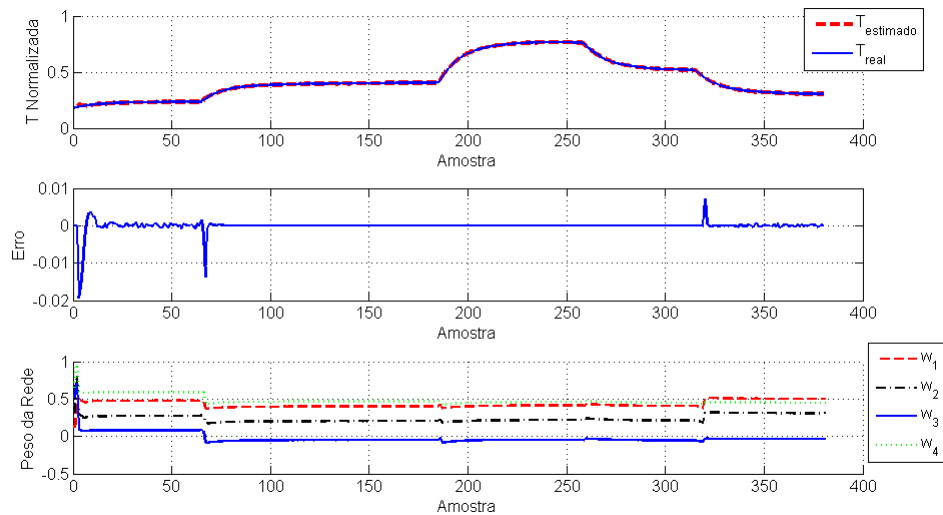


Figura 5.5: Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/ARX para 10 iteraes por amostra

Diferentemente do caso de uma iterao, no foram estimados modelos instáveis, nem mesmos para o caso inicial em que os pesos iniciais se encontram mais distantes do ótimo. Os

valores máximos de energia também foram menores, embora a quantidade de cálculos tenha aumentado.

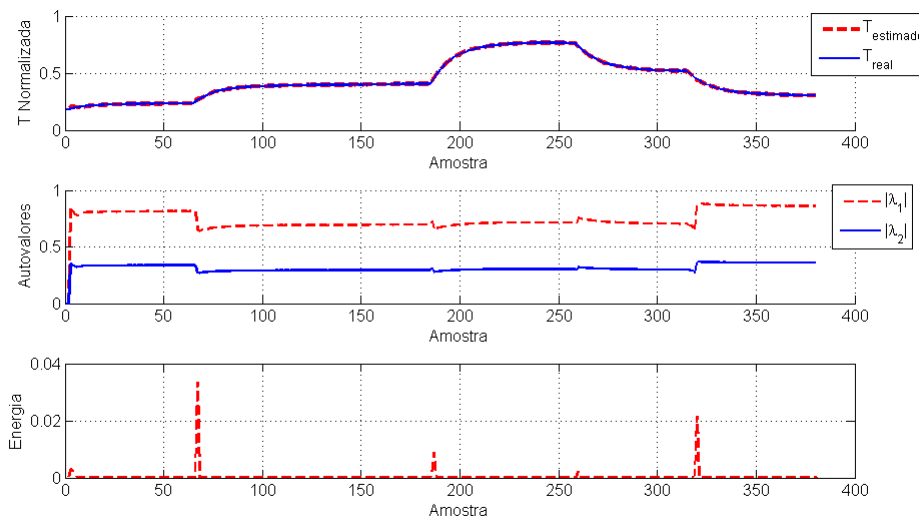


Figura 5.6: Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/ARX para 10 iterações por amostra

A última situação analisada, trata do algoritmos limitado por um critério de erro, possibilitando analisar se este consegue atingir um nível de precisão desejável. Neste caso considerou-se uma margem de erro entre ± 0.01 , representada pelas Figuras 5.7 e 5.8.

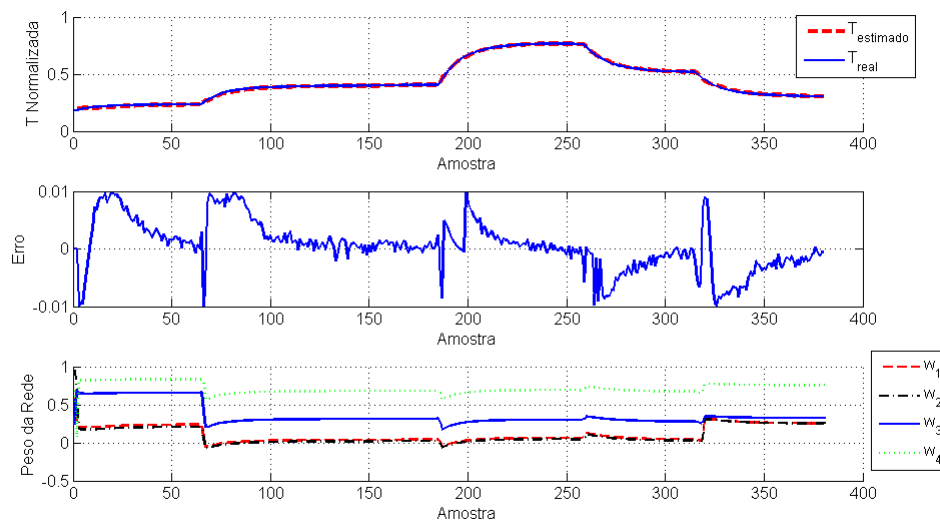


Figura 5.7: Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/ARX com iterações por amostra limitado por condiço de erro

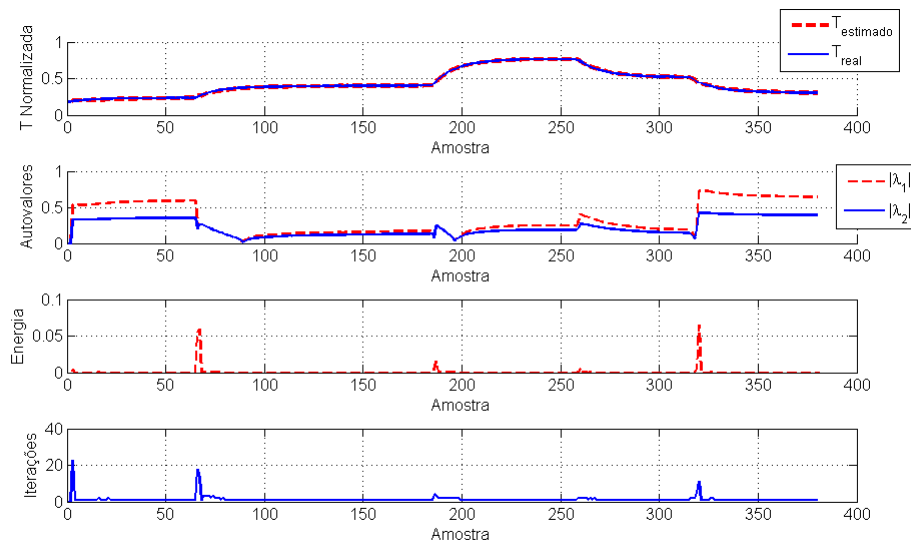


Figura 5.8: Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/ARX com iterações por amostra limitado por condição de erro

Utilizando como critério de parada um valor admissível para o erro, dadas as Figuras 5.7 e 5.8, o algoritmo mostrou boa precisão, com pesos não saturados, autovalores dentro da faixa de estabilidade e mais próximos de zero com relação aos outros casos, bem como baixa energia.

Tabela 5.1: BP caso 1

Testes	Media de Iterações no processo	Maior número de iterações em uma amostra
1	1	43
2	1	39
3	1	52
4	1	48
5	1	50
6	1	58
7	1	64
8	1	50
9	1	48
10	1	56

Foram realizados vários testes, Tab.5.1 com diversos valores iniciais, onde obteve-se para todo o processo de treinamento uma média de 10 iterações para cada ponto de operação e o maior número de iterações em um ponto foi no processo de treinamento foi de 64. Estes valores foram obtidos somando-se todas as quantidades de iterações de todos os pontos de operação e dividindo-os pela quantidade de medições, e o valor de pico é o valor máximo

dado o vetor iterações obtido durante o processo.

- **Caso 2:** Treinamento Backpropagation clássico tendo uma RNA/Espaço de Estados:

De forma similar ao primeiro caso, são realizados os mesmos testes para verificar a eficácia em uma iteração, em 10 e grau de precisão para rede na forma de espaço de estados. Assim, o teste com uma iteração por amostra, Fig.5.9 e Fig.5.10, apresentou picos menores nas mudanças de dinâmica, ausência de auto valores instáveis e apesar de um número maior de parâmetros a serem estimados, mas apresentando picos de energia menores em quantidade e amplitude.

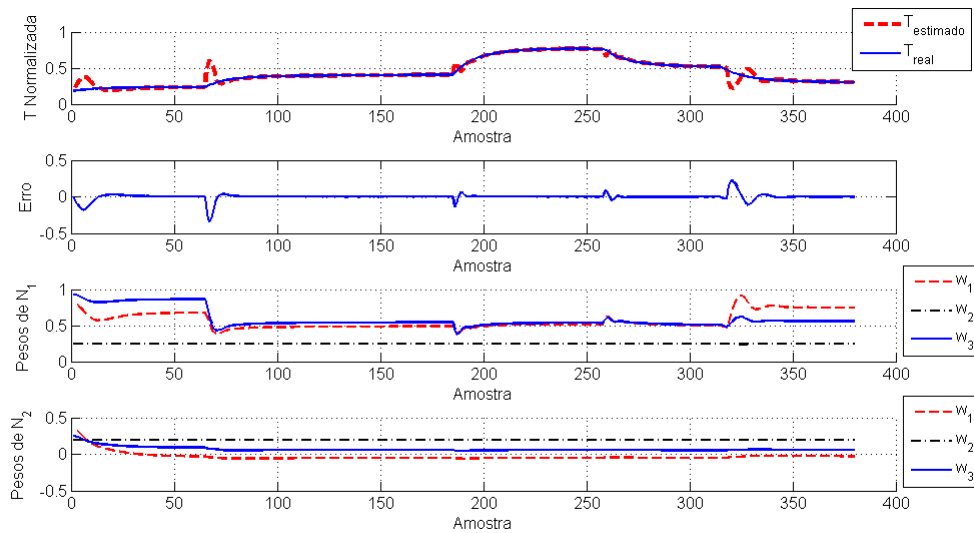


Figura 5.9: Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/Espaço de Estados com uma iteração por amostra

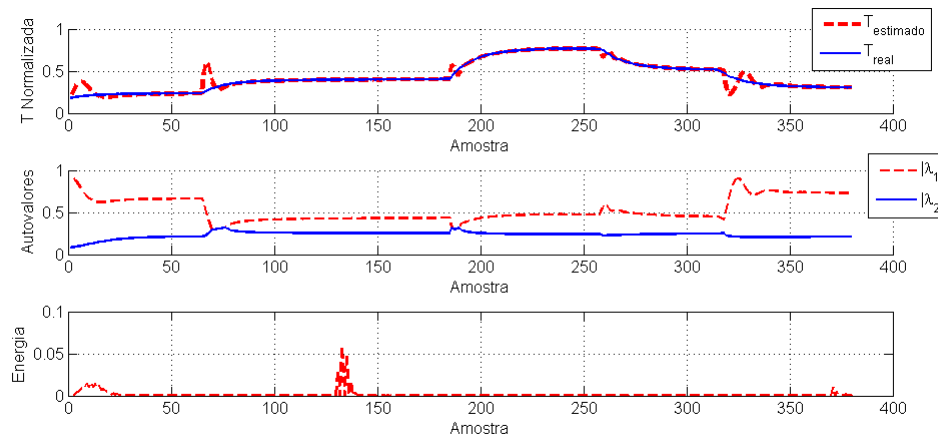


Figura 5.10: Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/Espaço de Estados com uma iteração por amostra

Para o caso em que se avalia o comportamento em 10 iterações por amostra, Figuras 5.11 e 5.12, apresentou melhoras mais significativas ainda, tanto em comparação ao primeiro teste deste caso, quanto em comparação com a mesma situação do primeiro caso, onde pode-se observar oscilações bruscas do erro somente nos instantes iniciais, a não saturação dos pesos, autovalores estáveis e picos de energia somente no início do processo, contrariando as situações anteriores que apresentaram picos a cada mudança de dinâmica.

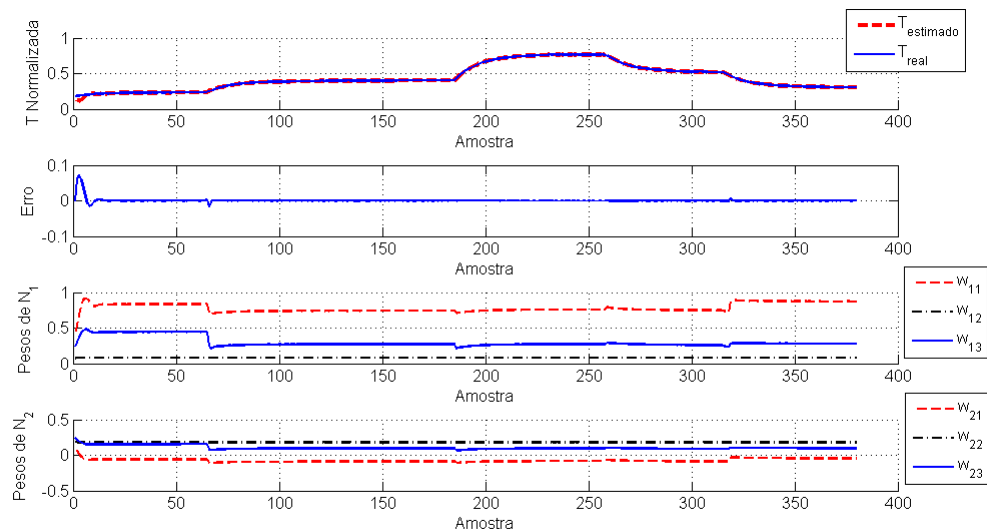


Figura 5.11: Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/Espaço de Estados com 10 iterações por amostra

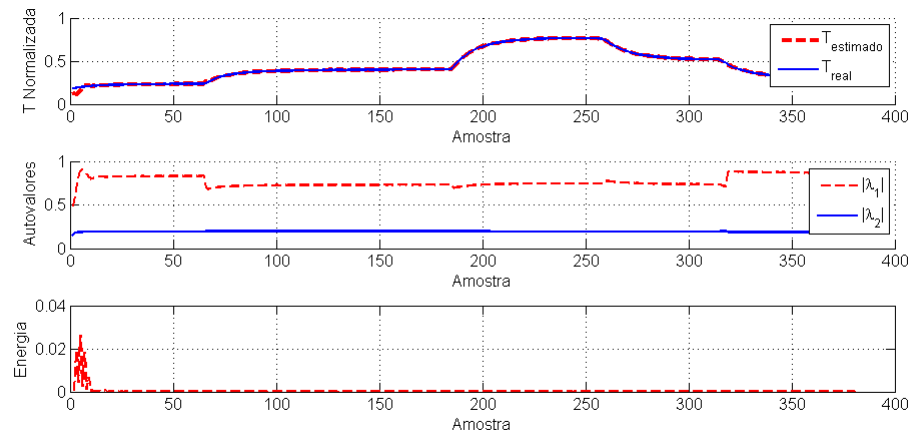


Figura 5.12: Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/Espaço de Estados com 10 iterações por amostra

Por fim, em termos de gráficos para este caso, tem-se a Fig.5.13 e Fig.5.14 para se avaliar o comportamento. Portanto, pode-se observar uma boa resposta, mantendo o

erro dentro da margem determinada, porém apresenta muitas variações nos parâmetros, não somente nas mudanças de dinâmica, mas ao longo de todo o processo, embora não tenham saturado.

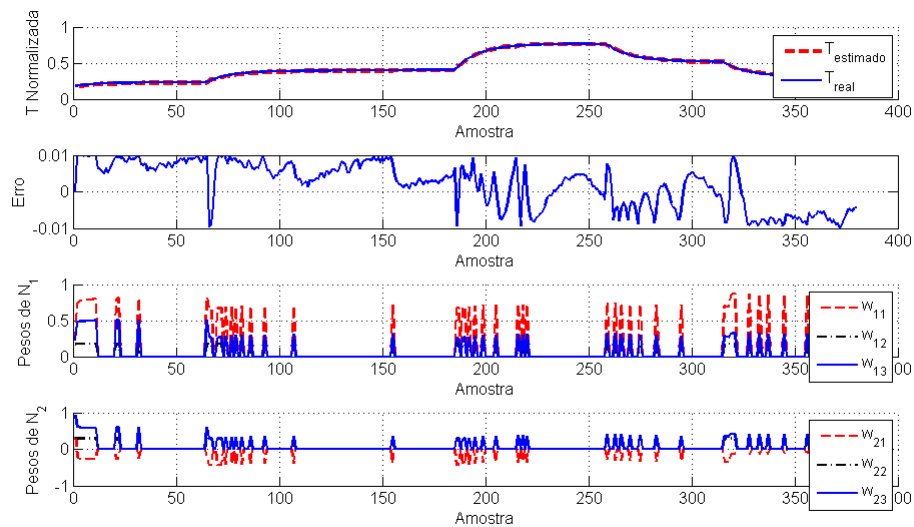


Figura 5.13: Temperatura no objeto, Erro e Pesos com treinamento BP - RNA/Espaço de Estados com número de iterações limitado por critério de erro

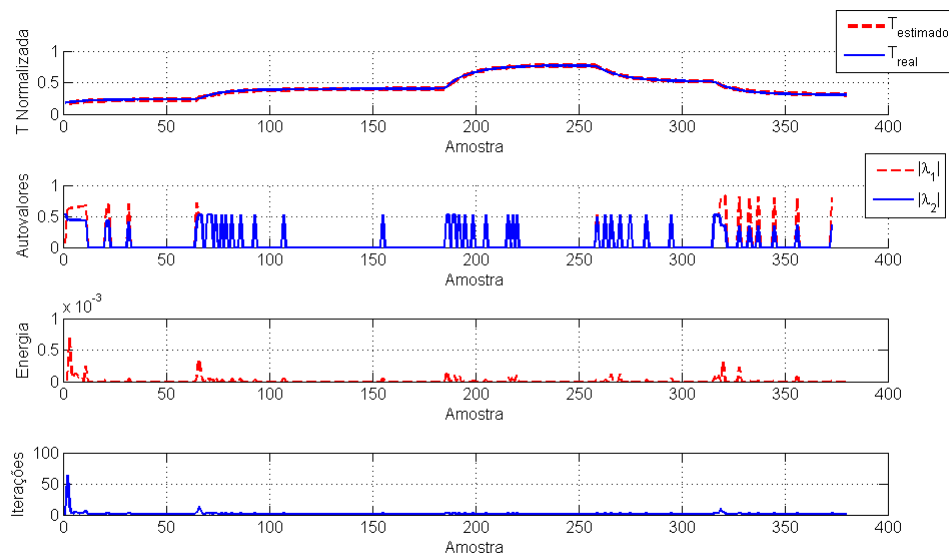


Figura 5.14: Temperatura no objeto, Autovalores e Energia com treinamento BP - RNA/Espaço de Estados com número de iterações limitado por critério de erro

Considerando valores aleatórios para os pesos iniciais, assim como no caso anterior, foram realizados dez testes, verificando-se a convergência, a média de iterações por amostra e

o maior nmero de iteraes necessrios para rede ser treinada em uma amostra. Da mesma forma que no caso 1, os picos de iteraes se deram no comeo do treinamento, onde a mdia total nos dez testes foi de 1 e a mdia dos maiores nmeros de iteraes foi de 51, com base na Tab.5.2, havendo convergncia para estes e outros vrios testes.

Tabela 5.2: BP caso 2

Testes	Mdia das iteraes	Maior nmero de iteraes em uma amostra
1	1	26
2	1	54
3	1	63
4	1	58
5	1	51
6	1	39
7	1	61
8	1	66
9	1	63
10	1	35

5.3 Estimaco Utilizando Backpropagation Acelerado

No caso do backpropagation com acelerao de gradiente, Algoritmo 1, os mesmos testes so realizados para ttulo de comparao e avaliar cada algoritmo quanto a melhor resposta dentre as estruturas de rede utilizadas.

- **Caso 1:** Treinamento Backpropagion Acelerado para a RNA/ARX:

Utilizando a rede na forma de um ARX de ordem 2, os testes para uma iterao por amostra, 10 iteraes por amostra e por critrio de erro so apresentados. Portanto, considerando uma iterao por amostra o algoritmo apresentou boa resposta, porm com picos nas mudanas de dinmica um pouco mais elevados com relao aos outros estudos. Tambm apresentou pesos no saturados e autovalores estveis durante todo o processo, como mostra a Fig.5.15 e Fig. 5.16. Quanto a energia utilizada, mostrou-se mais elevada devido ao processo de acelerao.

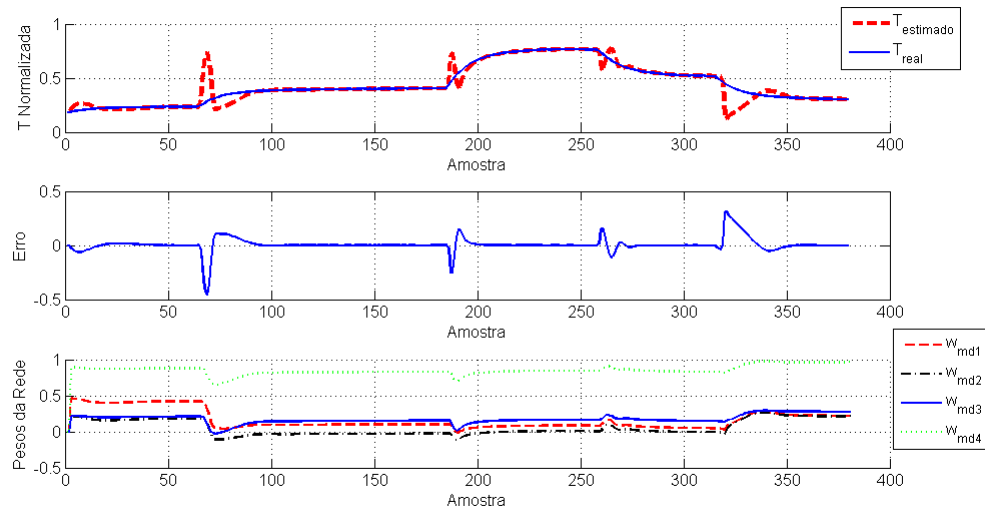


Figura 5.15: Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/ARX com uma iteração por amostra

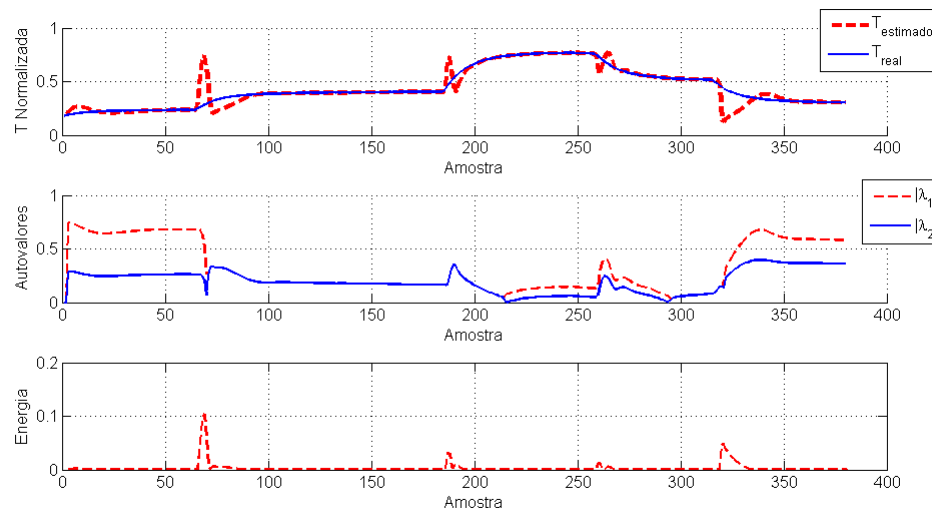


Figura 5.16: Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/ARX com uma iteração por amostra

Para o caso de 10 iterações por amostra, Fig.5.17 e Fig.5.18, a resposta apresentou semelhança com o backpropagation simples, com leve pico na segunda mudança de dinâmica, mas mostrando-se satisfatório no contexto geral, com pesos não saturados e autovalores estáveis. Porém o gráfico de energia apresentou uma alta significativa, chegando a ter um pico de aproximadamente $J = 0,2$, que significa uma mudança mais brusca dos parâmetros da rede.

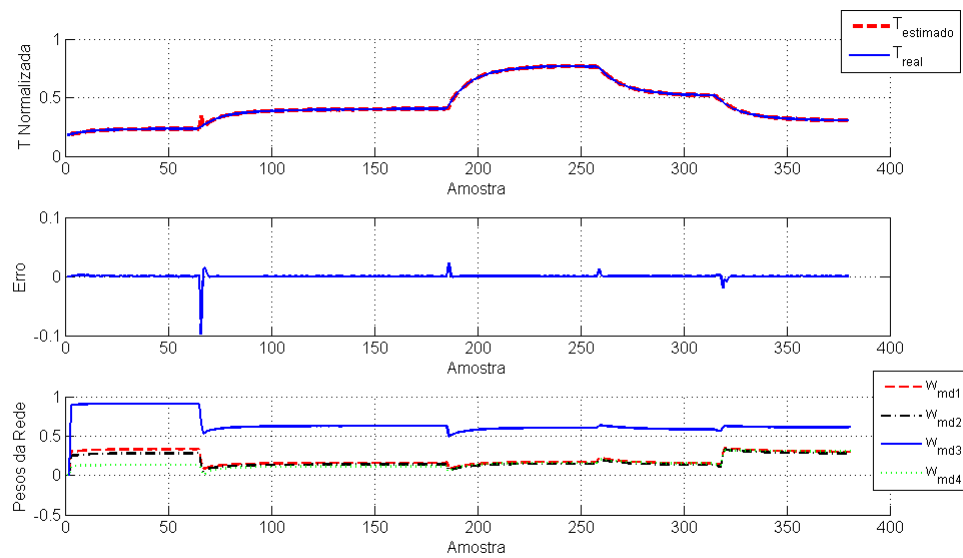


Figura 5.17: Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/ARX com 10 iterações por amostra

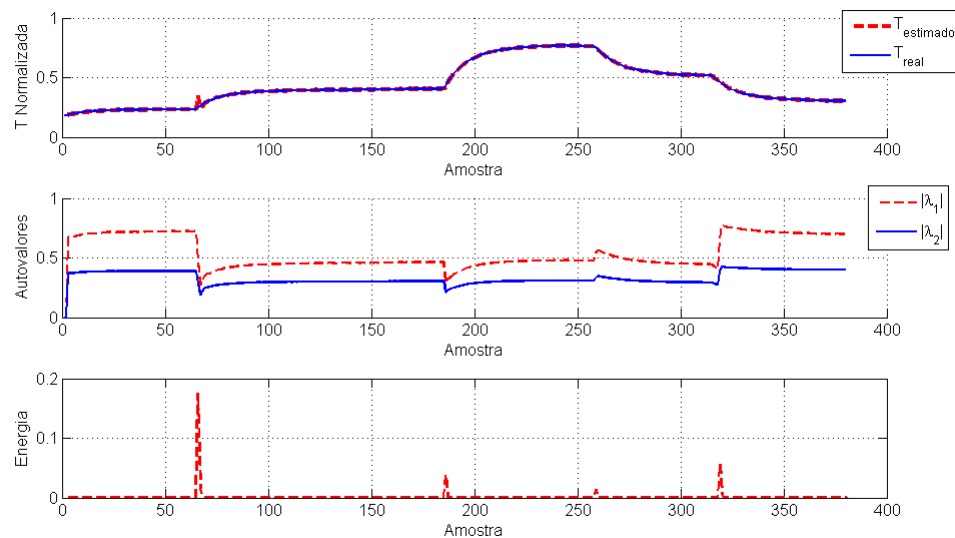


Figura 5.18: Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/ARX com 10 iterações por amostra

Para última situação deste caso, Fig.5.19 e Fig.5.20, que trata o critério de parada do treinamento uma tolerância para o erro, apresentou o melhor comportamento para o BP acelerado onde não houve picos na temperatura estimada, com autovalores estáveis e uma redução dos picos de energia com relação as situações de uma iteração e dez iterações.

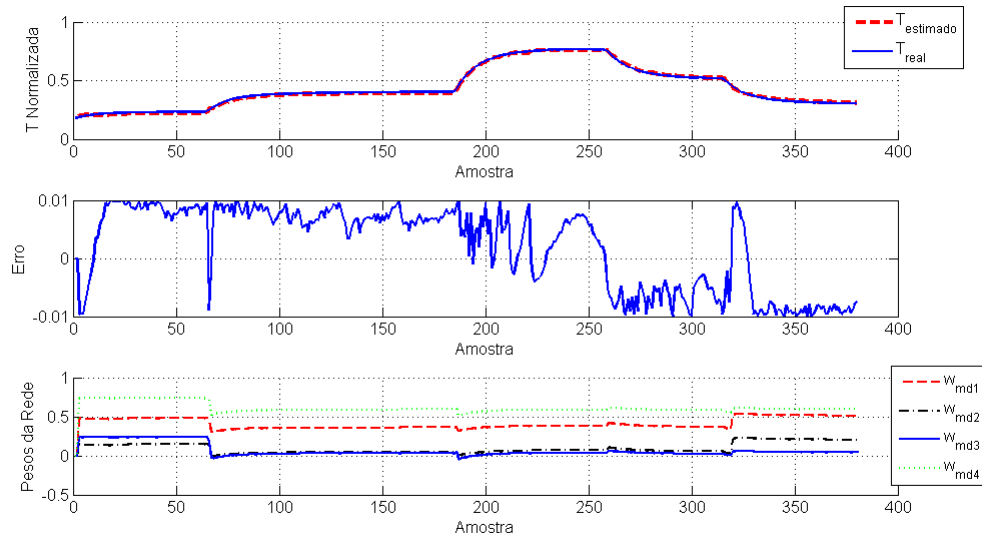


Figura 5.19: Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/ARX com iteraes por amostra limitadas por critério de erro

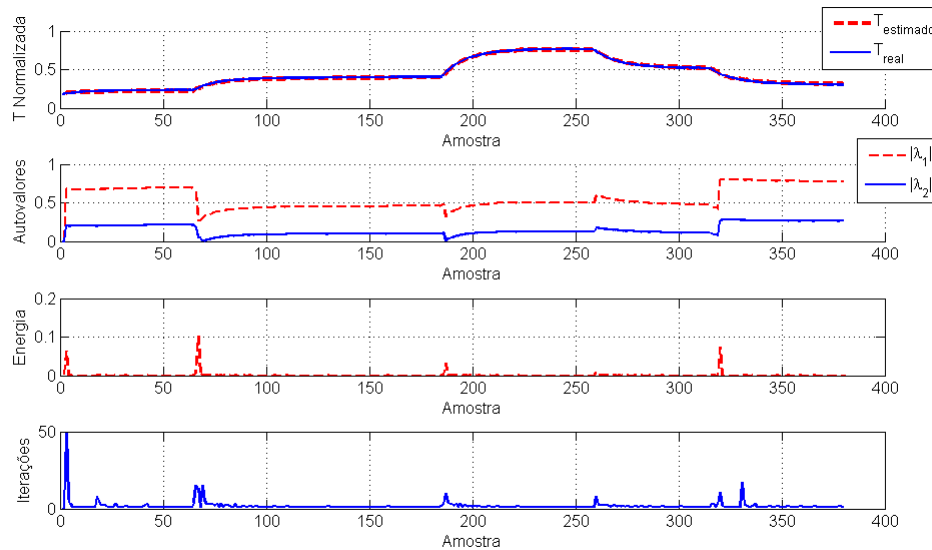


Figura 5.20: Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/ARX com iteraes por amostra limitadas por critério de erro

Por fim, apresenta-se a Tabela 5.3, com dez testes para o algoritmo, assim como realizado para o BP simples, cuja média de iteraes no processo foi de uma iterao, e uma média de iteraes máximas de 49, abaixo do BP simples.

Tabela 5.3: BP-AG caso 1

Testes	Mdia das iteraoes	Maior nmero de iteraoes em uma amostra
1	1	64
2	1	17
3	1	67
4	1	59
5	1	63
6	1	50
7	1	60
8	1	27
9	1	39
10	1	48

- **Caso 2:** Treinamento Backpropagation Acelerado para RNA/Espao de Estados:

Para o caso em que a rede est na forma do espao de estados, no caso de uma iterao por amostra, Fig.5.21 e Fig.5.22, o backpropagation acelerado (BP-AG) apresentou mais oscilaoes na temperatura estimada, apresentando um resultado mesmo que satisfatrio, pior com relao a RNA/ARX.

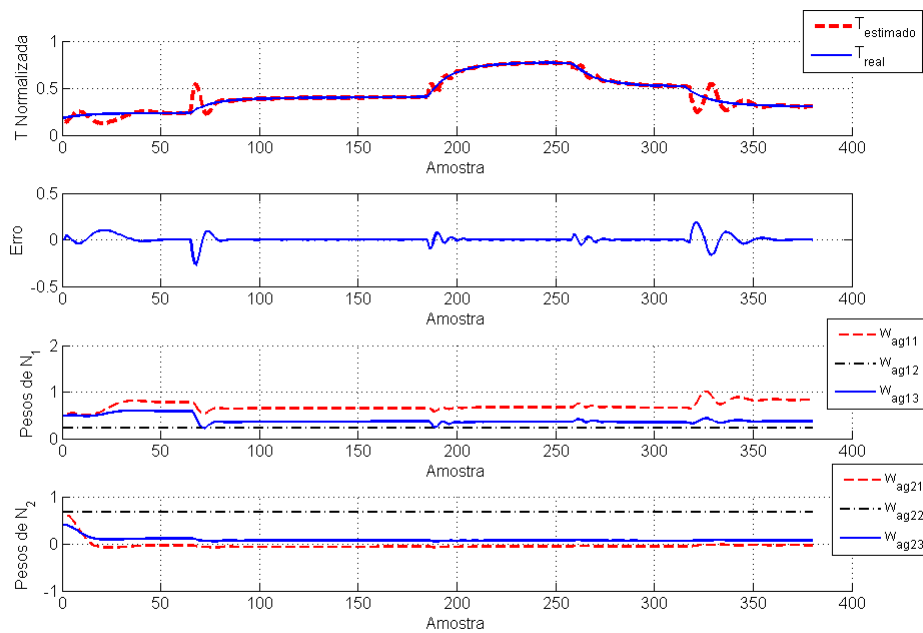


Figura 5.21: Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/Espao de Estados com uma iterao por amostra

Os pesos da rede chegaram, em certos momentos, ao limiar de saturao, assim como os autovalores da rede que chegaram ao limite de estabilidade em dois pontos. Por consequncia

das oscilações o gráfico de energia também apresentou uma maior quantidade de picos.

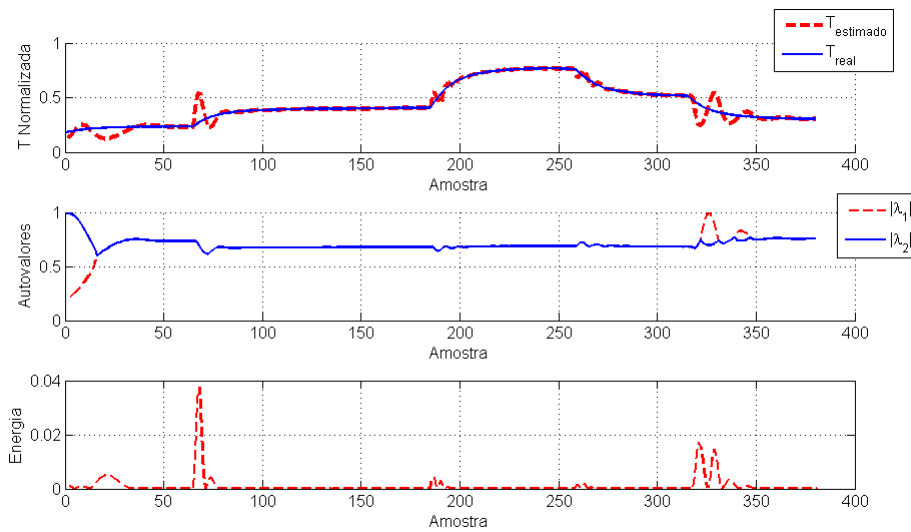


Figura 5.22: Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/Espaço de Estados com uma iteração por amostra

Considerando dez iterações por amostra, Fig.5.21 e Fig.5.22, a pesar da boa resposta, houve um erro maior nas primeiras estimativas, comparado ao RNA/ARX e também ao BP clássico.

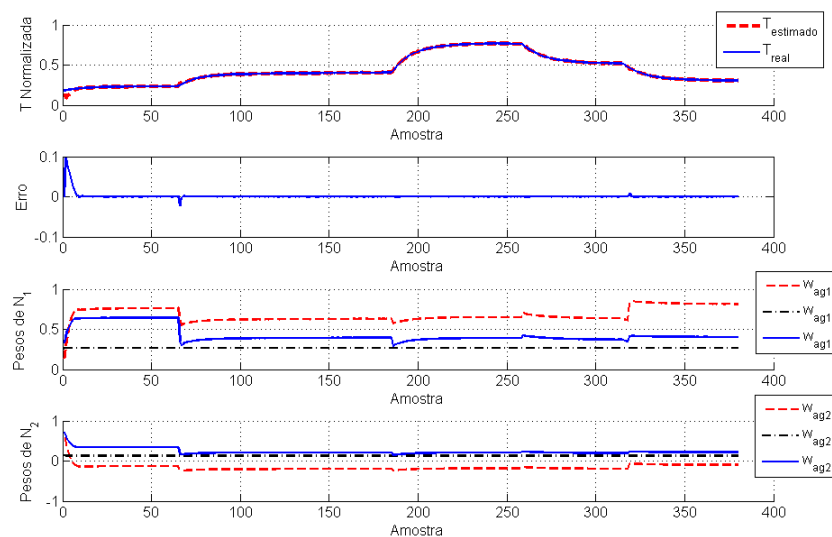


Figura 5.23: Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/Espaço de Estados com 10 iterações por amostra

Diferentemente da primeira situação, os pesos não saturaram, e houve uma redução da quantidade de picos de energia cuja amplitude diminuiu em relação ao caso de uma iteração por amostra.

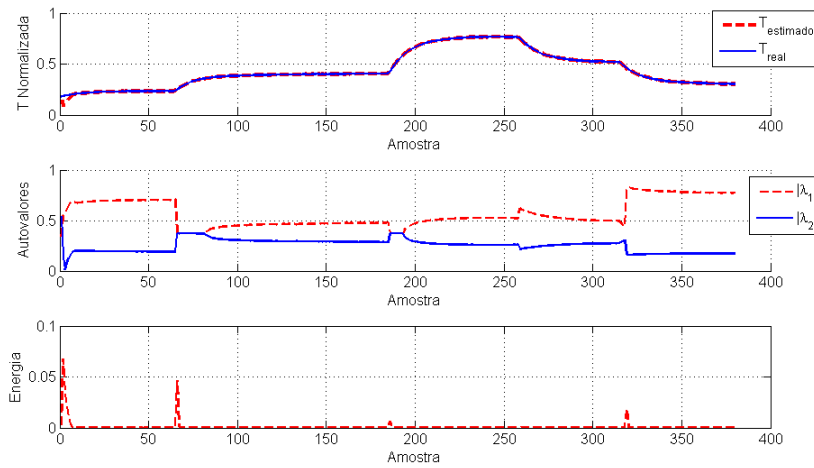


Figura 5.24: Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/Espaço de Estados com 10 iterações por amostra

Já no caso do critério de parada por limite do erro, Fig.5.25 e Fig.5.26, o comportamento se mostrou parecido com os outros algoritmos e com o do caso 1, não havendo saturação dos pesos mas apresentando grande oscilação dos mesmo.

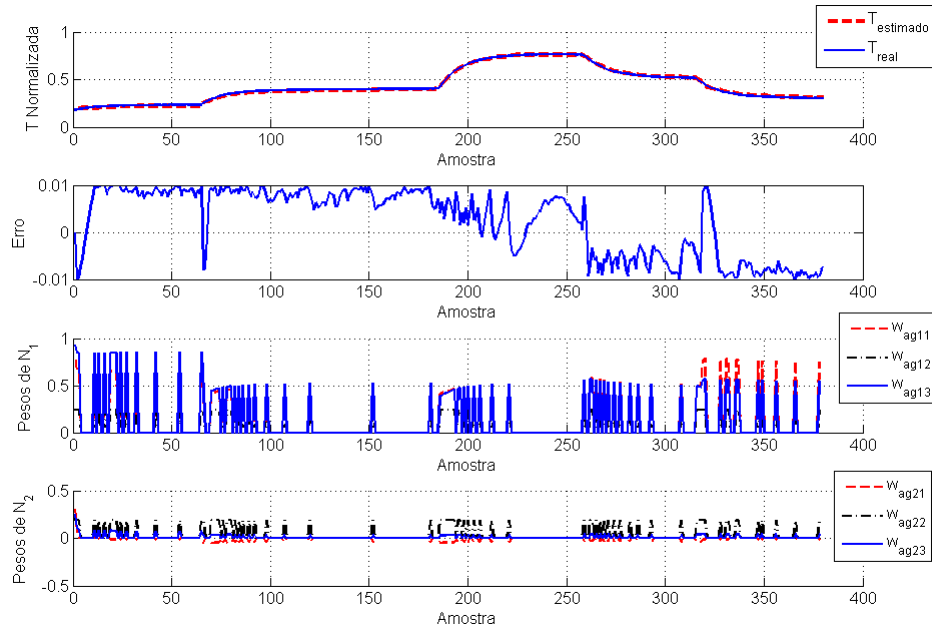


Figura 5.25: Temperatura no objeto, Erro e Pesos com treinamento BP-AG - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro

Os autovalores apresentaram, também, grandes oscilações, embora o gráfico de energia apresente poucos picos com amplitudes parecidas aos casos já estudados.

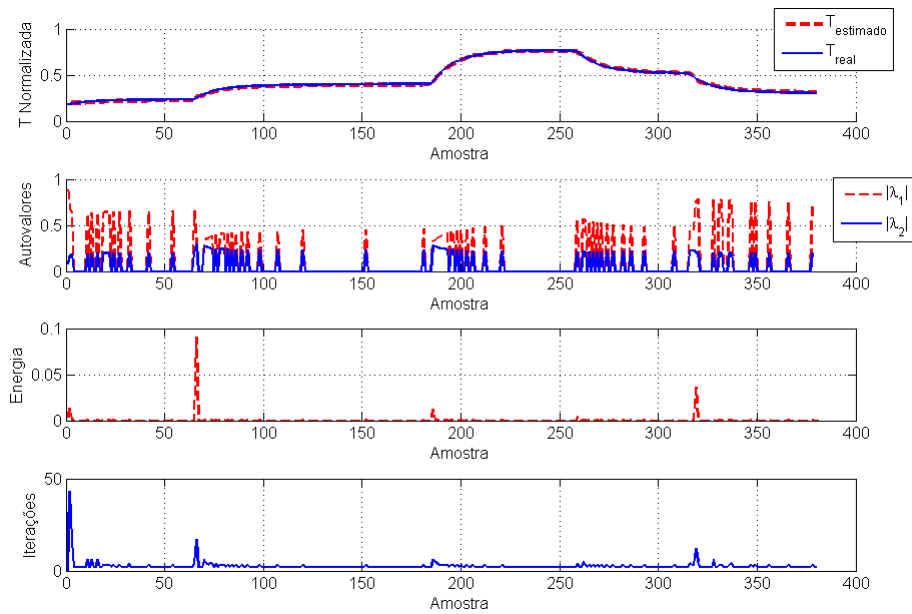


Figura 5.26: Temperatura no objeto, Autovalores e Energia com treinamento BP-AG - RNA/Espao de Estados com iteraoes por amostra limitada por critrio de erro

Na realizao de dez testes utilizando como critrio de parada uma tolerncia para o erro, com base na Tabela 5.4, obteve-se convergncia em todos os caso, com mdia de iteraoes por amostra igual a 2 e mdia dos picos de iteraoes de 42,4 e se caracterizou por ser o caso em que se obteve as menores quantidades de iteraoes para a parte inicial do treinamento.

Tabela 5.4: BP-AG caso 2

Testes	Mdia das iteraoes	Maior nmero de iteraoes em uma amostra
1	2	37
2	2	39
3	2	48
4	2	16
5	2	30
6	2	40
7	2	62
8	2	51
9	2	65
10	2	36

5.4 Estimaco paramétrica utilizando RLS

Nesta seo tm-se os testes para o ltimo dos algoritmos avaliados, que no caso é o RLS, para comparar com o *Backpropation* clssico e acelerado.

- **Caso 1:** Treinamento Backpropagion clssico para a RNA/ARX:

Para o estudo considerando uma iterao por amostra, Fig.5.27 e Fig.5.28, apresentou oscilaes nas mudanas de dinmica, porm apresentam picos pequenos e pesos no saturados. Os autovalores chegaram prximos da instabilidade nos instantes iniciais e proporcionalmente as oscilaes, houveram picos de energia, o que em suma pode-se considerar a resposta satisfatria, j que a RNA conseguiu acompanhar a dinmica do sistema.

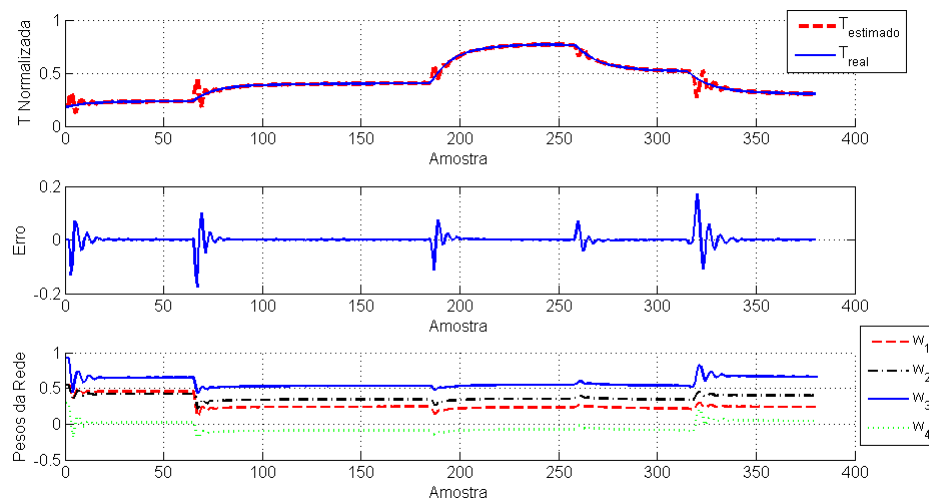


Figura 5.27: Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/ARX com 1 iterao por amostra

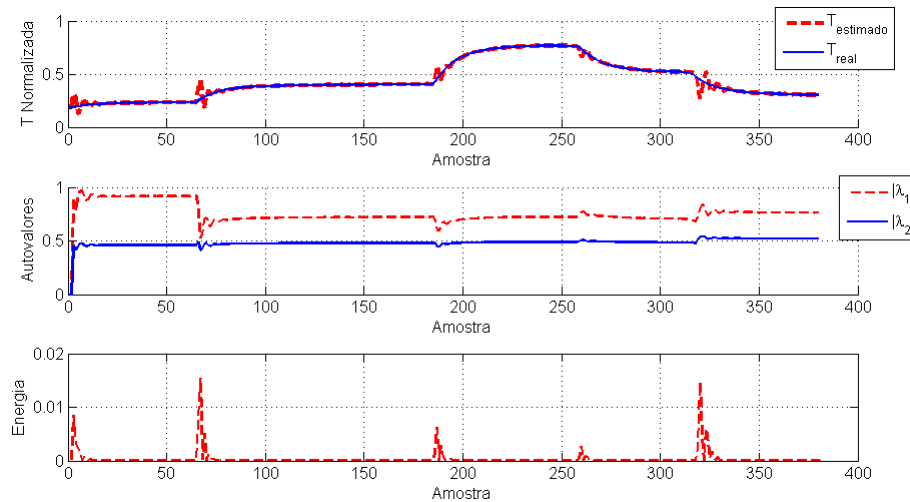


Figura 5.28: Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/ARX com 1 iteração por amostra

Para o caso de dez iterações por amostra, Fig.5.29 e Fig.5.30, foi o que apresentou menor erro, com os pesos não saturados e sem grandes oscilações. No entanto, no início do processo estimou-se modelos instáveis, como se pode ver no gráfico dos autovalores, e energia com comportamento semelhante aos casos já estudados.

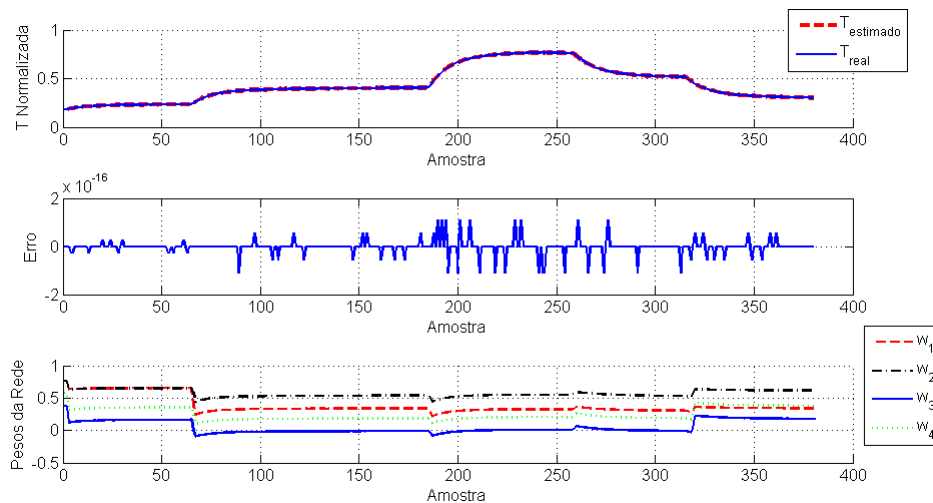


Figura 5.29: Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/ARX com 10 iterações por amostra

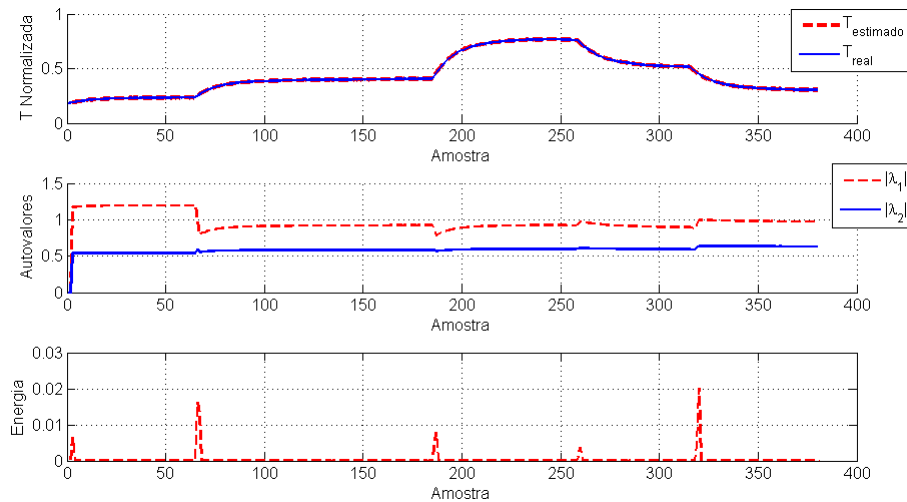


Figura 5.30: Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/ARX com 10 iterações por amostra

Já para o caso com número de iterações limitadas por critério de erro, Fig.5.31 e Fig.5.32, obteve-se boa resposta, com pesos não saturados e autovalores estáveis para todo processo. Os picos de energia se mantiveram dentro do padrão, referentes aos casos anteriores, porém, diferentemente dos algoritmos BP e BP-AG, não houveram grandes picos de iterações por amostra, caracterizando-se como uma rápida estimativa.

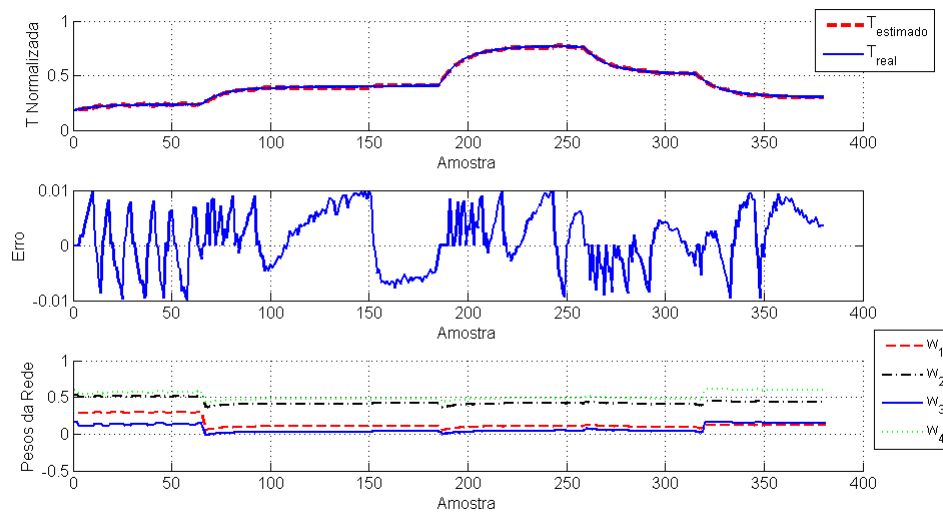


Figura 5.31: Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/ARX com iterações por amostra limitada por critério de erro

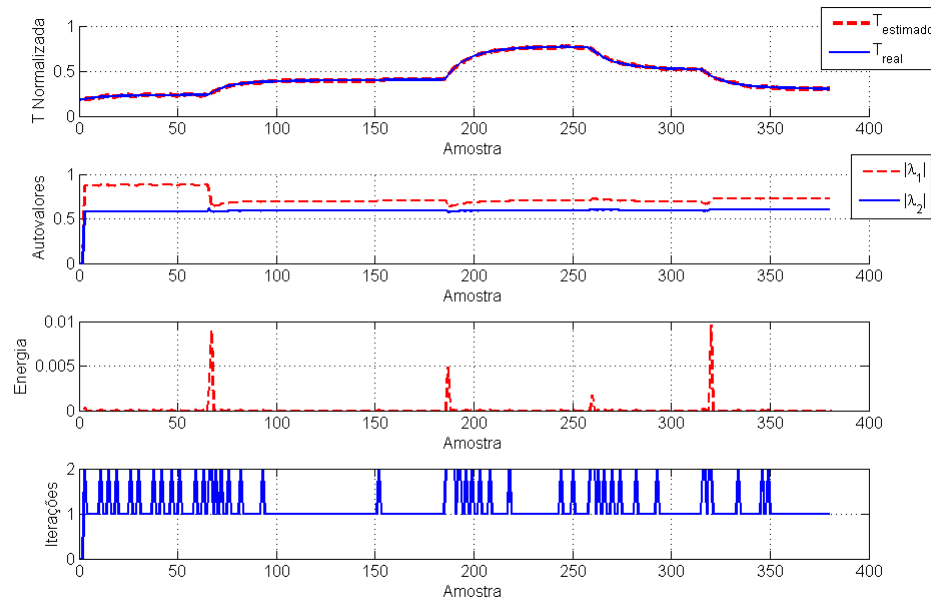


Figura 5.32: Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/ARX com iterações por amostra limitada por critério de erro

Outro fator relevante com relação a este algoritmo nesta aplicação, é a estabilidade com relação a média e maior quantidade de iterações em uma amostra no processo, onde para todos os casos observados obteve-se média 1 de iterações por ponto de operação e máximo de duas iterações por ponto. As oscilações de uma para duas iterações por ponto podem ser consequência dos pequenos ruídos presentes no sinal, fazendo com que o RLS leve duas iterações para aproximar as mudanças.

- **Caso 2:** Treinamento RLS para RNA/Espaço de Estados:

No estudo de caso tendo a rede na forma de espaço de estados, com uma iteração por amostra, Fig.5.33 e Fig.5.34, os resultados mostraram semelhanças com relação ao caso 1, tendo oscilações nas mudanças de dinâmica. Mas, apresentam pesos sinápticos próximos de 1, ou seja, próximos a saturação. Os autovalores chegaram a ser instáveis, como mostra o gráfico dos autovalores na ultima mudança. Outro fator importante a ser observado é que a energia acumulada, pela primeira vez, não se comportou através de picos, mas de forma crescente, tendo o maior acumulo total de energia visto.

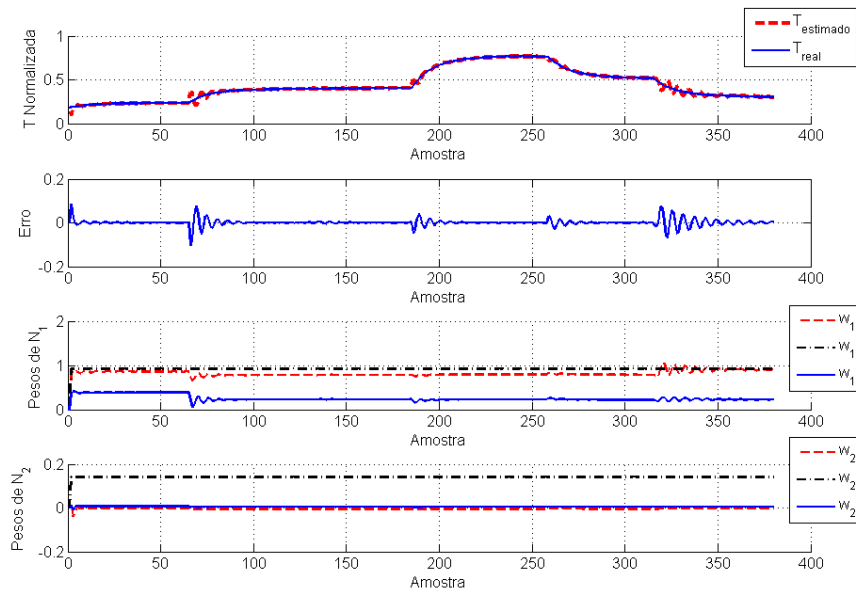


Figura 5.33: Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/Espaço de Estados com 1 iteração por amostra

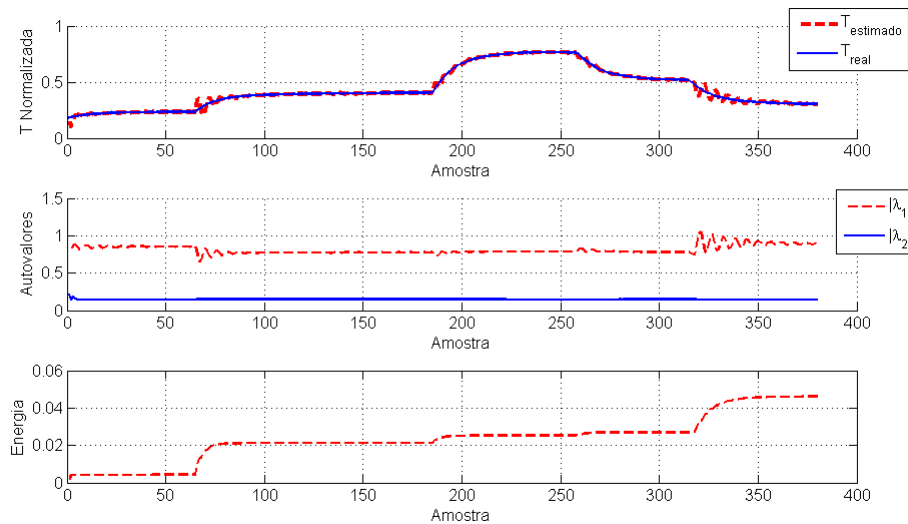


Figura 5.34: Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/Espaço de Estados com 1 iteração por amostra

Já analisando o caso de dez iterações por amostra, Fig.5.35 e Fig.5.36, percebe-se que o resultado aproximou-se do caso um em quase todos os aspectos, como na escala do erro obtido e na não saturação dos pesos sinápticos. Mas para os autovalores, apresentaram instabilidade no início e a energia foi reduzida com relação ao caso de uma iteração por amostra.

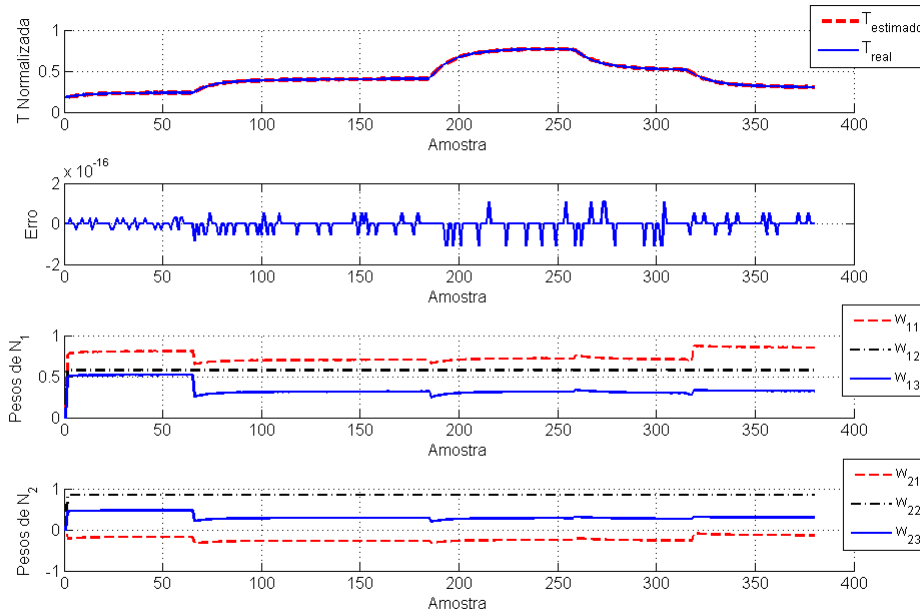


Figura 5.35: Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/Espaço de Estados com 10 iterações por amostra

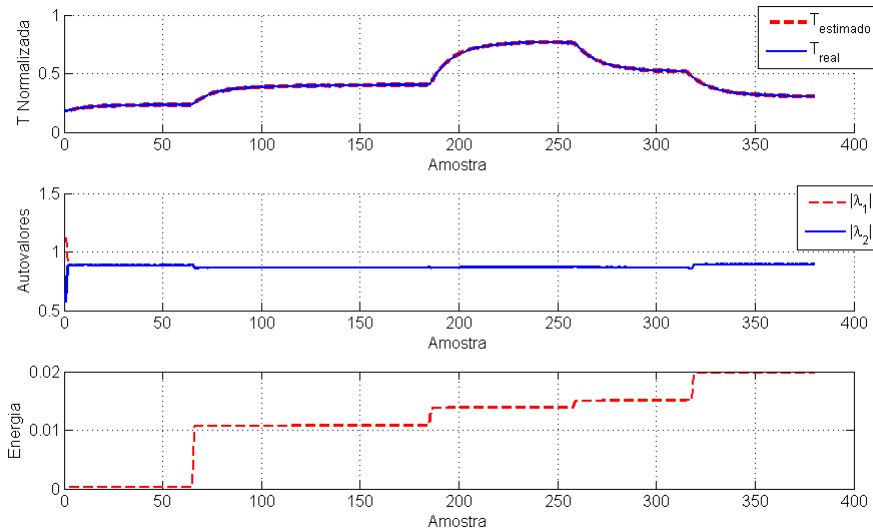


Figura 5.36: Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/Espaço de Estados com 10 iterações por amostra

Utilizando como critério de parada do treinamento o erro, Fig.5.37 e Fig.5.38, o algoritmo apresentou resultado satisfatório, com no máximo duas iterações por amostra, mas assim como nos casos anteriores, apresentou grande oscilação dos pesos sinápticos e dos autovalores e aumento da energia.

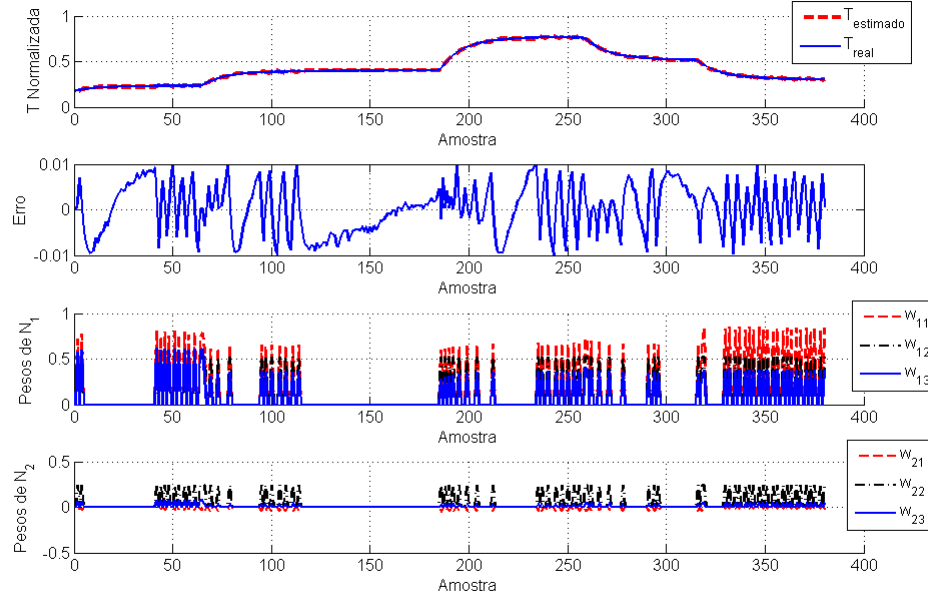


Figura 5.37: Temperatura no objeto, Erro e Pesos com treinamento RLS - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro

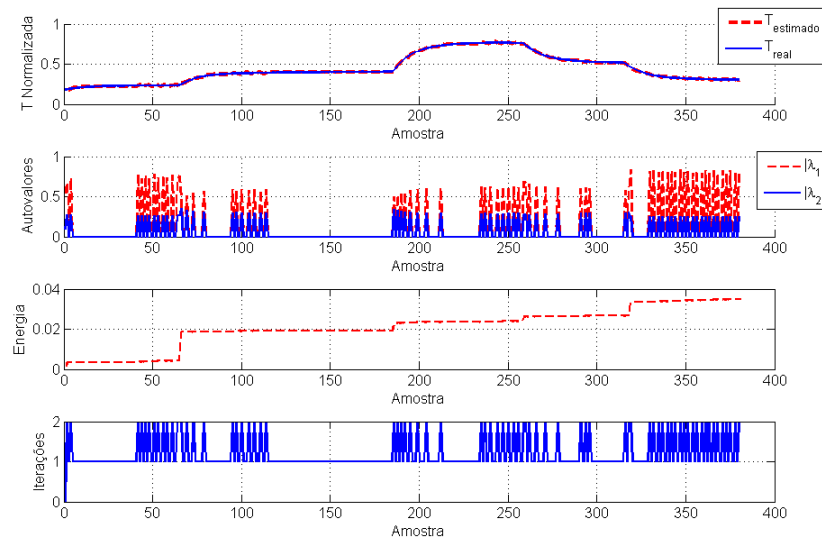


Figura 5.38: Temperatura no objeto, Autovalores e Energia com treinamento RLS - RNA/Espaço de Estados com iterações por amostra limitada por critério de erro

Assim como realizado para os casos anteriores, neste caso obteve-se também média 1 e valor máximo de iterações por ponto de operação igual a dois.

5.5 Comparaco entre os métodos.

Nesta seo é apresentada uma análise sobre os diversos métodos aplicados à estimaco *online* dos parâmetros voltados para o sistema térmico não linear baseado no efeito Peltier. Foram realizados cinco testes em cada algoritmo para os dois casos propostos, analisando a energia total acumulada, ou seja, a soma das energias instantâneas durante todo processo de aprendizagem, a energia média total, os picos máximos de energia, a média dos picos de iteraes por ponto de operao dos testes, o número médio de iteraes para todo processo de aprendizagem, o tempo médio de execuo em cada teste e o tempo máximo de aprendizado em uma amostra por teste. Destes cinco casos, são apresentados nas Tabelas 5.5 e 5.5, o pior e o melhor resultado de cada item analisado e a média de todos os testes para o caso 1 (RNA/ARX) e caso 2 (RNA no espaço de estados). Assim, a segunda coluna mostra a média de iteraes em um teste, a terceira o maior número de iteraes apresentado na aprendizagem de uma amostra, a quarta o tempo médio necessário para o aprendizado de uma amostra, a quinta o maior tempo demandado pelo algoritmo para ajustar a rede para um ponto de operao e as sexta e sétima colunas o pico de energia em um teste e a energia total acumulada.

Tabela 5.5: Análise dos algoritmos para o caso 1

Algoritmo	BP					
	Iteraço média	Iteraço máxima	Tempo médio(s)	Tempo máximo (s)	Energ. máxima	Energ. total
Pior caso	1	21	$6,3 * 10^{-5}$	0,0051	0,2851	0,3528
Melhor caso	1	64	$6,35 * 10^{-5}$	0,0062	0,0757	0,1570
Média	1	48,6	$6,33 * 10^{-5}$	0,0058	0,1397	0,2648
Algoritmo	BP-AG					
Pior caso	1	60	$3,27 * 10^{-5}$	0,0085	0,2096	0,4647
Melhor caso	1	19	$3,22 * 10^{-5}$	0,0029	0,028	0,094
Média	1	45	$3,25 * 10^{-5}$	0,0067	0,1096	0,2922
Algoritmo	RLS					
Pior caso	1	2	$3,37 * 10^{-5}$	0,0048	0,0406	0,1486
Melhor caso	1	2	$3,27 * 10^{-5}$	0,0028	0,002	0,0104
Média	1	2	$3,31 * 10^{-5}$	0,0038	0,0194	0,0684

Observando a Tabela 5.5 verifica-se uma homogeneidade da média de iteraes em cada teste, havendo diferenas somente entre as iteraes máximas, onde o BP clássico apresentou a maior média e maior pico, mostrando que este teve mais dificuldade em encontrar os parâmetros ótimos na parte inicial do treinamento. O BP-AG apresentou picos e média menores que o BP, mas neste quesito que teve melhor comportamento foi o algoritmo RLS, apresentando além do baixo número de picos, uma certa homogeneidade.

Levando-se em consideraco o tempo mdio necessrio para se aprender uma amostra o BP se mostrou em grande desvantagem com relao ao BP-AG e o RLS, que reduziram basicamente a metade o tempo mdio, com o BP-AG apresentando leve vantagem sobre o RLS. J com relao ao pior caso de tempo de convergncia obtido, ou seja, o tempo necessrio para aprender em um ponto de operao, todos se mantiveram na mesma faixa de tempo, com o BP-AG apresentando o pior dos piores resultados. Neste caso pode-se afirmar que os algoritmos esto com desempenho parecido.

Considerando o fator energia ou erro quadrtico, no caso 1, quem obteve maior pico de energia no treinamento foi o BP clssico, tanto inferior quanto superior. J o BP-AG mostrou resultados pouco melhor, onde quem teve melhor desempenho foi o RLS, com menor mdia e menores picos inferiores e superiores.

Tabela 5.6: Anlise dos algoritmos para o caso 2

	Iterao mdia	Iterao mxima	Tempo mdio(s)	Tempo mximo (s)	Energ. mxima	Energ. total
Algoritmo	BP					
Pior caso	1	68	$8,311 * 10^{-6}$	0,0017	0,2073	0,2131
Melhor caso	1	27	$7,82 * 10^{-6}$	0,00069	0,00046	0,0053
Mdia	1	48,4	$8,11 * 10^{-6}$	0,0062	0,0460	0,0054
Algoritmo	BP-AG					
Pior caso	2	124	$9,53 * 10^{-6}$	0,002	0,1162	0,2711
Melhor caso	2	36	$8,311 * 10^{-6}$	0,00068	0,0287	0,0492
Mdia	2	45	$8,75 * 10^{-6}$	0,0013	0,0756	0,1542
Algoritmo	RLS					
Pior caso	1	2	$8,31 * 10^{-6}$	0,00014	0,0462	11,44
Melhor caso	1	2	$7,82 * 10^{-6}$	0,000082	0,007	1,505
Mdia	1	2	$8,16 * 10^{-5}$	0,00052	0,0217	5,684

J para o segundo caso de estudo, Tabela 5.5, em termo de iteraoes os algoritmos RLS e BP apresentaram melhores mdias, enquanto o BP-AG a mdia foi o dobro, J com relao ao maior nmero de iteraoes utilizados para aprender uma amostra o BP-AG apresenta o maior pico superior, mas manteve a mdia do caso 1, mesmo com pico inferior sendo o maior de todos. Bem como no caso 1, o RLS mostrou-se o mais satisfatrio.

Em termos temporais, apresentaram resultados prximos em termos de mdia com o BP-AG levemente pior, mas dependendo do sistema a ser utilizado, pode-se dizer que  insignificante. J em termos do tempo mximo quem melhor se comportou foi o RLS e com o BP-AG melhor que o BP, como pode-se ver atravs da mdia dos tempos mximos nos cinco testes.

Analisando a energia para este caso o BP e o BP-AG mostram-se em grande vantagem

com relação ao RLS principalmente com relação ao acúmulo de energia durante todo o processo, embora em termos de pico máximo de energia tenha apresentado melhores resultados.

Sistema de Controle Adaptativo e Ótimo

Neste capítulo é apresentado um estudo do controle adaptativo do ponto de vista de controle clássico e moderno em uma abordagem discreta. É apresentado o controle PID adaptativo juntamente a estimação *online* do tipo RNA-ARX e controle ótimo adaptativo para a estimação do tipo RNA-Espaço de Estados, além de embarcar, um dos tipos apresentado no microcontrolador, apresentando as dificuldades e resultados mais próximos da realidade.

Quando se fala em adaptabilidade, o que vem a mente é justamente a palavra variação, e assim como os seres alteram com o passar do tempo e as mudanças no meio em que vivem, algumas características que lhes permitem sobreviver e evoluir, o controle adaptativos, de forma semelhante, tem como característica a alteração de seus parâmetros para controlar uma variável ou mais, dado um sistema (meio) cujos parâmetros que determinam seu comportamento se alteram ao longo do tempo.

Este estudo tem como aplicação um sistema térmico já apresentado no Capítulo 4, onde uma RNA é utilizada como modelo e um algoritmo de treinamento *online* estima os pesos sinápticos se ajustando a mudança de característica da planta ao longo do tempo, e o controle adaptativo, com base nos pesos da rede busca o auto ajustar para que se possa impor características ao sistema como princípio do estudo de controladores e sistemas térmicos de alto rendimento, que é o conjunto planta mais controle adaptativo, bem como o desenvolvimento de sensores baseados em modelo.

A primeira estrutura proposta, baseada nas escolas francesas, é o controlador PID digital adaptativo (LANDAU, 1938) e posteriormente o Regulador linear quadrático (LEWIS, 1992).

6.1 Controle PID Digital Adaptativo

O controlador PID é usualmente utilizado na forma polinomial, tendo suas origens nos anos cinquenta e até hoje é um dos mais aplicados em escala industrial. Embora o conhecimento prévio deste tipo de controlador seja bastante consolidado, algumas variações podem ser utilizadas, como a questão da adaptabilidade dos parâmetros do controlador às variações paramétricas de uma planta não linear. Dessa forma a estrutura do sistema com controlador PID Digital Adaptativo (LANDAU, 1938) e estimação inteligente pode ser observada na Fig.6.1 e possui equação dada por

$$H_{PID}(z^{-1}) = K \left[1 + \frac{T_s}{T_i} \frac{1}{1 - z^{-1}} + \frac{NT_s}{T_d + NT_s} \frac{(1 - z^{-1})}{1 - \frac{T_d}{T_d + NT_s} z^{-1}} \right]. \quad (6.1)$$

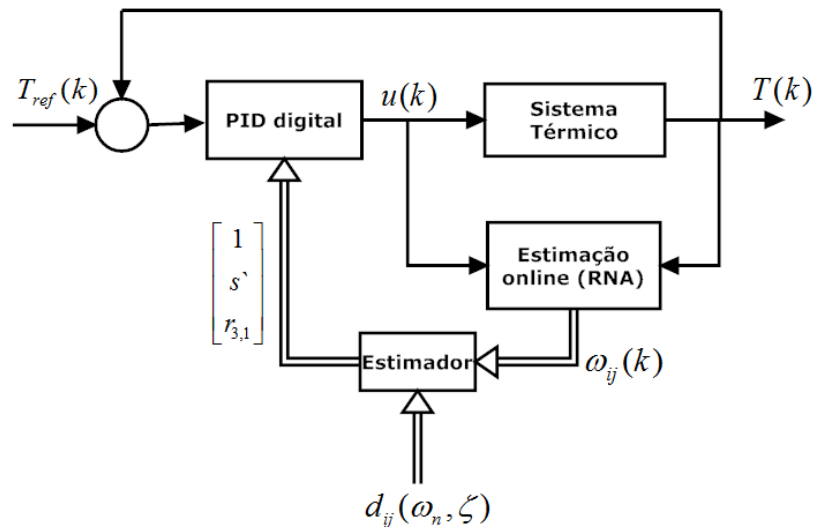


Figura 6.1: Estrutura para aplicação do PID Digital adaptativo

Assim, o controlador PID é dado pela Eq.(6.1), sendo K a constante proporcional, T_i o tempo de integração, T_d a tempo de derivação, N o parâmetro de ajuste do filtro do derivativo responsável por limitar a ação deste, evitando instabilidade e T_s o tempo de amostragem, sendo que estes parâmetros podem ser relacionados à estrutura digital de forma generalizada, em função das variáveis de saída da planta da entrada e da referência.

Neste estudo, o bloco Estimador da Fig.6.1 receberá como parâmetros de entrada os pesos da rede neural que representam os parâmetros do modelo RNA da planta a cada amostra de tempo e também a performance desejada $d(\omega_n, \zeta)$, para que este determine as constantes do controlador para um dado instante kT_s .

Dessa forma, assume-se como forma generalizada para o controlador digital a estru-

tura da Fig.6.2, constituído por um bloco de realimentação do sinal de controle, um de realimentação de saída e um referente a entrada de referência atual e anteriores.

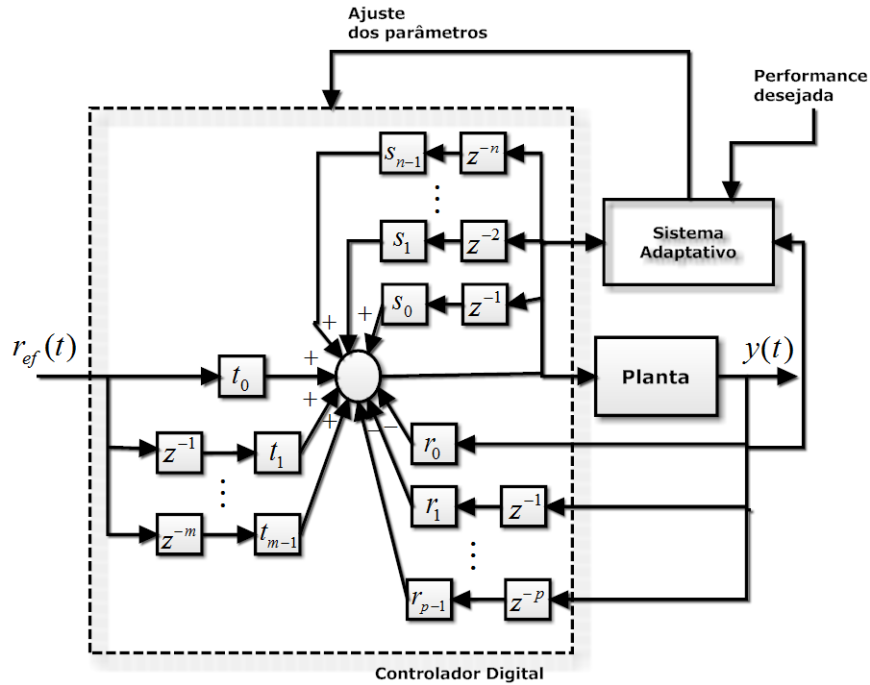


Figura 6.2: Estrutural geral para controle digital, com base em (LANDAU, 1938)

Através de manipulações algébricas e com base no modelo da planta de forma genérica dada pela equação de segunda ordem, onde podem ser feitas as correlações entre o controlador digital genérico e o PID digital é dada por

$$H(z^{-1}) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}}. \quad (6.2)$$

Dessa forma, definindo $R(z^{-1})$ como a função da realimentação da saída, $S(z^{-1})$ a função de realimentação do sinal de entrada da planta e composta de um integrador puro, assume-se que

$$R(z^{-1}) = r_0 + r_1 z^{-1} + r_2 z^{-2}, \quad (6.3)$$

e

$$S(z^{-1}) = (1 - z^{-1})(1 + s_1 z^{-1}), \quad (6.4)$$

sendo os parâmetros de $R(z^{-1})$ e $S(z^{-1})$ em função dos parâmetros do PID, cuja dedução pode ser vista em (LANDAU, 1938), são dados por

$$s_1 = -\frac{T_d}{T_d + NT_s}, \quad (6.5)$$

$$r_0 = K \left(1 + \frac{T_s}{T_i} - N\frac{T_s}{T_d}s_1 \right), \quad (6.6)$$

$$r_1 = K \left[s_1 \left(1 + \frac{T_s}{T_i} + 2N\frac{T_s}{T_d} \right) - 1 \right], \quad (6.7)$$

e

$$r_2 = -Ks_1 \left(1 + N\frac{T_s}{T_d} \right). \quad (6.8)$$

Tendo-se determinado o controlador digital, é necessário relacionar parâmetros desejados para o sistema em malha fechada aos parâmetros da planta e do controlador. Portanto, seja a função de transferência da planta em sua forma discreta dada por

$$H(s) = \frac{b_1z^{-1} + b_2z^{-2}}{1 + a_1z^{-1} + a_2z^{-2}} = \frac{B(z^{-1})}{A(z^{-1})}. \quad (6.9)$$

Dessa forma a equação de malha fechada em função de $R(z^{-1})$ e $S(z^{-1})$ é dada por

$$H_{mf}(z^{-1}) = \frac{B(z^{-1})R(z^{-1})}{A(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1})} = \frac{B_{mf}(z^{-1})}{P(z^{-1})}. \quad (6.10)$$

Com isso é possível dividir o polinômio característico de malha que é dado por

$$P(z^{-1}) = (1 + a_1z^{-1} + a_2z^{-2})(1 - z^{-1})(1 + s_1z^{-1}) + (b_1z^{-1} + b_2z^{-2})(r_0 + r_1z^{-1} + r_2z^{-2}), \quad (6.11)$$

sendo

$$\begin{aligned}
P(z^{-1}) &= 1 + p_1 z^{-1} + p_2 z^{-2} + p_3 z^{-3} + p_4 z^{-4} \\
&= (1 + p_1' z^{-1} + p_2' z^{-2}) (1 + \alpha_1 z^{-1}) (1 + \alpha_2 z^{-1}),
\end{aligned} \tag{6.12}$$

sendo p_1' e p_2' o resultado de uma discretização de uma função de transferência de segunda ordem com ω_n e ζ como especificação de desempenho e os parâmetros α_1 e α_2 que correspondem a polos auxiliares. Estes, para não influenciar muito na dinâmica, devem ser rápidos e reais. Para isso, isso (LANDAU, 1938) sugere valores, com $\alpha_1 = \alpha_2$ ou $\alpha_2 = 0$, no intervalo $-0,05 \leq \alpha_1, \alpha_2 \leq -0,5$. Assim é possível determinar o conjunto de equações para se determinar os parâmetros para uma alocação de polos, que são dadas por

$$\begin{aligned}
p_1 &= b_1 r_0 + s_1 + a_1', \\
p_2 &= b_2 r_0 + b_1 r_1 + s_1 a_1' + a_2', \\
p_3 &= b_2 r_1 + b_1 r_2 + s_1 a_2' + a_3', \\
p_4 &= b_2 r_2 + s_1 a_3',
\end{aligned} \tag{6.13}$$

sendo $a_1' = a_1 - 1$, $a_2' = a_2 - a_1$ e $a_3' = -a_2$. Assim pode-se passar o problema para forma matricial, cuja forma equivalente é dada por $Mx = p$, sendo:

$$x^T = [1, s_1, r_0, r_1, r_2], \tag{6.14}$$

$$p^T = [1, p_1, p_2, p_3, p_4], \tag{6.15}$$

sendo a matriz M dada por

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ a_1' & 1 & b_1 & 0 & 0 \\ a_2' & a_1' & b_2 & b_1 & 0 \\ a_3' & a_2' & 0 & b_2 & b_1 \\ 0 & a_3' & 0 & 0 & b_2 \end{bmatrix}, \tag{6.16}$$

tendo como solução $x = M^{-1}p$.

Determinado o controlador digital e a equação solução, determina-se os parâmetros do PID em função de uma performance desejada e dos parâmetros da planta. Mostra-se a estrutura de controle adaptativo, tendo como referência a Fig.6.1 que apresenta o bloco de controle PID digital, o sistema térmico, a identificação *online* e a determinação dos parâmetros do controlador dada a cada variação paramétrica da planta.

Neste primeiro momento será utilizado como estimador a forma matricial $x = M^{-1}p$, para realização de testes e servindo como aplicação, dada a utilização da estimação inteligente. Assim para realizar os testes em simulação e montar a estrutura dada na Fig.6.1 foi utilizado o modelo NARX/Hammertein da planta, como mostra a teoria do Capítulo 2 com estimação apresentada no Capítulo 4. A estimação *online* inteligente é realizada com base nos dados fornecidos pelo modelo, obtendo-se os parâmetros da planta para o estimador do controlador, dados os parâmetros de desejados.

Assim são apresentados os resultados na Fig.6.3, mostrando o esforço de controle, resposta dada pela RNA da identificação *online* e a dinâmica desejada nas formas gráficas. Como exemplo, segue a resposta temporal para um valor de referência de $70^{\circ}C$, onde configurou-se como especificações desejadas de um $\tau = 300s$, visto que a constante de tempo da planta sem controle é de aproximadamente $540s$, e um fator de amortecimento $\zeta = 0.8$.

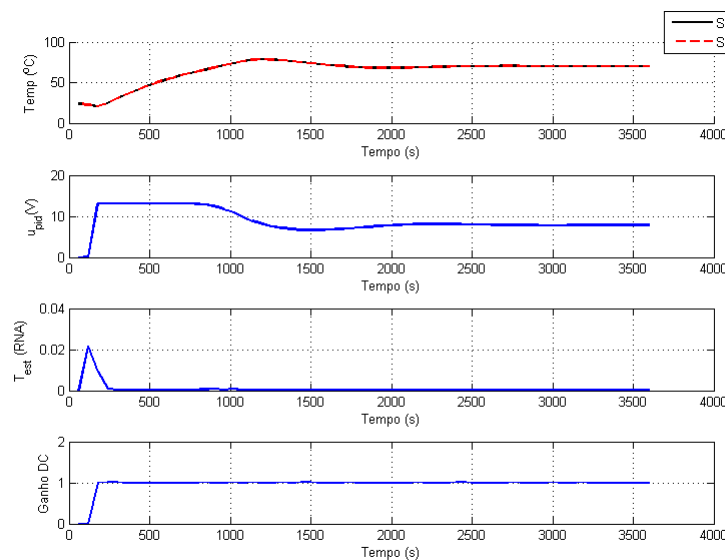


Figura 6.3: Planta com PID digital para $\tau = 300$, $\zeta = 0.8$ e $T_{ref} = 70$

Dada a constante de tempo desejada para o caso da Fig.6.3 de $300s$, e sabendo que o controlador começa a atuar somente no instante $T = 180s$, o ponto que marca os 63% em cima da variação de $23^{\circ}C$ a $70^{\circ}C$, cujo valor é de $46^{\circ}C$ está no instante $T_2 = 482s$, aproximadamente, que implica em uma contante de tempo, $\tau_{obtido} = 360s$, próximo ao desejado mostrando que para este caso o sistema de controle apresentou resultados satisfatórios.

Assim, verifica-se o desempenho do sistema controlado na Tabela 6.1.

Tabela 6.1: Tabela de desempenho do controle adaptativo digital

Desempenho desejado			Desempenho obtido		
τ desejado	ζ	T_{ref}	τ obtido	M_p	T_{saida}
300	0.8	70	360	78	70,38
400	0,8	70	390	78	70,29
500 a 700	0,8	70	390	76	70,01
1000	0,8	70	420	73	69,6 a 71,6
300 a 1200	0,8	95	720	99	94,7
1300	0,8	95	1110	98	95,21

É possível analisar, através dos resultados apresentados na Tabela 6.1, o desempenho do sistema tomando dois valores de referência e variando-se a constante de tempo desejada. Verificam-se faixas em que o sistema não sofreu qualquer alteração, como para referência $70^\circ C$. Para constantes de tempo entre 500 e 700 segundos, onde apenas a precisão com que a saída de aproximou do valor de referência foi alterada com relação a $\tau = 400s$. Já quando se aproxima o valor de referência do limite ao qual a célula pode fornecer, a forma com que o sinal de controle se comportou foi sempre próximo aos limites máximos fazendo com o sistema se comporte temporariamente em torno de sua constante de tempo sem controle que está entre 500 e 700 segundos, dependendo do ponto de operação.

Ainda observando a Tabela 6.1, os valores de constantes de tempo desejados bem mais elevados o sistema volta a se aproximar da característica desejada. A nível de conhecimento, (LANDAU, 1938) recomenda uma faixa de valores para ω_n dada pelo intervalo $0,25 \leq \omega_n T_s \leq 1,5$, onde ω_n é a frequência natural e T_s o tempo de amostragem, que gera para este sistema constantes de tempo desejadas entre 250 e 1500s.

6.2 Regulador Linear Quadrático Adaptativo

O controle ótimo teve, entre os anos de 1960 e 2010, um grande crescimento, devido o desenvolvimento da computação digital e do avanço da eletrônica, permitindo o tratamento mais rápido de dados e maior capacidade de cálculos. Os métodos de otimização aplicados sistemas de controle permitiu obter-se índices de desempenhos com custos reduzidos, dentro dos limites impostos pelo sistema a ser controlado.

Para resolver problemas de controle ótimo, deve-se ter como objetivo, a minimização, ou maximização de um índice de desempenho, que no caso do Regulador Linear Quadrático (LQR), trata de encontrar uma política de decisão que gere um custo mínimo, sujeita a algumas restrições, onde o índice de desempenho é dado pelo quadrado do erro.

Segundo (LEWIS, 1992), o processo de otimização deve prover não somente as leis de controle e os parâmetros ótimas, mas também uma análise de desempenho pelo afastamento da função de índice de desempenho do seu valor mínimo ou máximo que resultam em leis de controle não ótimas.

Com base nesse contexto, é feita uma abordagem discreta para o LQR, denominado de DLQR, visto que o modelo do processo é discreto. Assim da mesma forma como foi apresentado para o controlador PID Digital adaptativo, a Fig.6.4 apresenta o diagrama esquemático para o DLQR adaptativo, como uma segunda aplicação a utilização de identificação inteligente *online*.

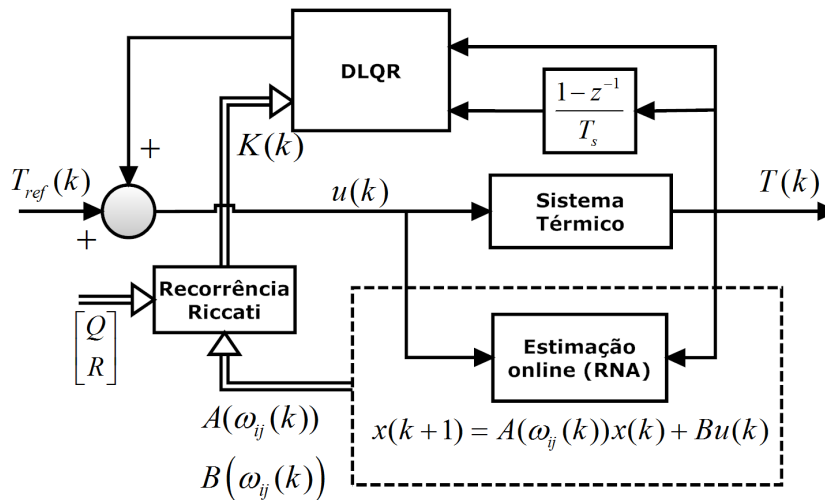


Figura 6.4: Estrutural geral para DLQR adaptativo

Neste caso, a identificação *online* é responsável por fornecer as matrizes $A(k)$ e $B(k)$ estimadas que são compostas dos próprios pesos da rede. Fornecido tal dado, com base nas matrizes Q e R fornecida pelo usuário, o bloco Estimador Riccati, utilizando um método iterativo deve fornecer os ganhos $K(k)$ do bloco DLQR para que componha o combinador linear dos dois estados de entrada. Essa estrutura permite que se determine um regulador ótimo para cada ponto de operação da planta, caracterizando assim sua adaptabilidade.

Dessa forma, é explorado aqui um pouco da teoria que envolve o DLQR, bem como o método para se determinar $K(k)$. Portanto, seja o sistema estimado dado por:

$$x(k+1) = A(k)x(k) + B(k)u(k), \quad (6.17)$$

sendo $x(k+1)$ composta pelos estados futuros do sistema dados pela temperatura no instante $k+1$ e sua derivada $\frac{T(k+1)-T(k)}{T_s}$, $x(k)$ os estados anteriores, a matriz $A(k)$ fornecida pela rede

neural e $B(k)$ uma matriz constante igual a $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$ e dada a lei de controle por realimentação de estados na forma

$$u(k) = T_{ref}(k) - kx(k), \quad (6.18)$$

o problema de otimização para formulação do DLQR por meio da Eq.(6.17), consiste na minimização da função de custo dada por

$$J = \frac{1}{2} \sum_{k=1}^N x_k^T Q_k x_k + u_k^T R_k u_k, \quad (6.19)$$

dada as matrizes de ponderação $Q \geq 0$ e $R > 0$.

O objetivo é encontrar $K(k)$ que minimize a Eq.(6.19), e para isso, trata-se como um problema de otimização restrita, onde o método de Lagrange é o mais apropriado para se encontrar a solução. Dessa forma, a função lagrangiana composta pela função objetivo é dada por

$$L(x_k, u_k) = -\frac{1}{2} (x_k^T Q_k x_k + u_k^T R_k u_k), \quad (6.20)$$

sendo a função Hamiltoniana dada por

$$H_k = \lambda_{k+1}^T f(x_k, u_k) + L(x_k, u_k), \quad (6.21)$$

onde defini-se

$$x_{k+1} = \frac{\partial H_k}{\partial \lambda_{k+1}} = f_k(x_k, u_k), \quad (6.22)$$

$$\frac{\partial H_k}{\partial u_k} = 0, \quad (6.23)$$

$$\frac{\partial H_k}{\partial x_k} = -\lambda_k. \quad (6.24)$$

Realizando as devidas substituições obtém-se a Hamiltoniana dada por

$$H_k(\lambda_{k+1}, x_k, u_k) = \lambda_{k+1}^T A_k x_k + \lambda_{k+1}^T B_k u_k - \frac{1}{2} x_k^T Q_k x_k - \frac{1}{2} u_k^T R_k u_k, \quad (6.25)$$

onde aplicando as derivadas, obtém-se

$$\frac{\partial H_k}{\partial u_k} = (\lambda_{k+1}^T B_k) - (u_k^T R_k) = 0. \quad (6.26)$$

Dessa forma a lei de controle ótimo será dado por:

$$u_k = R_k^{-1} B_k^T \lambda_{k+1}^T. \quad (6.27)$$

Portanto, admitindo-se como solução $\lambda_k = P_k x_k$, ou seja, $\lambda_{k+1} = P_{k+1} x_{k+1}$ e utilizando a Eq.(6.17), temos que a lei de controle que minimiza a função de custo é dada por

$$u_k = -((R_k + B_k^T P_{k+1} B_k)^{-1} B_k^T P_{k+1} A_k) x_k, \quad (6.28)$$

onde substituindo a Eq.(6.28) na Eq.(6.17) tem-se

$$x_{k+1} = (A_k - B_k (R_k + B_k^T P_{k+1} B_k)^{-1} B_k^T P_{k+1} A_k) x_k, \quad (6.29)$$

onde a partir da derivada da Hamiltoniana em relação ao estado $x(k)$, tem-se

$$\frac{\partial H_k}{\partial x_k} = \lambda_k = A_k^T \lambda_{k+1} + Q_k x_k, \quad (6.30)$$

e substituindo λ_k e λ_{k+1} obtêm-se

$$P_k x_k = Q_k x_k + A_k^T P_{k+1} x_{k+1}. \quad (6.31)$$

E por fim, substituindo (6.29) em (6.31), obtêm-se a equação do DLQR (Regulador Linear Quadrático Discreto), dita equação de Riccati discreta que pode ser utilizada de forma iterativa:

$$\begin{aligned} -A_k^T P_{k+1} B_k (R_k + B_k^T P_{k+1} B_k)^{-1} B_k^T P_{k+1} A_k \\ + A_k^T P_{k+1} A_k - P_k + Q_k = 0. \end{aligned} \quad (6.32)$$

Assim os ganhos de realimentação dos estados, $-Kx_k$, pode ser dada pela expressão

$$K(k) = (R_k + B_k^T P_{k+1} B_k)^{-1} B_k^T P_{k+1} A_k. \quad (6.33)$$

Visto o equacionamento para o regulador LQR, é possível realizar as aplicações na planta dado o sistema de identificação *online* com RNA na forma de espaço de estados. Assim dado o bloco Estimador de Riccati na Fig.6.4 representado pelas Equações (6.32) e (6.33), dito método de Shur, o vetor $K(k)$ será estimado para cada novo conjunto de pesos fornecidos pela rede neural. Assim, para matriz $R = 1$ e $Q = I_{2,2}$ apresenta-se o sistema com o LQR somente, sendo $T_{ref} = 0^\circ C$:

Este tipo de regulador pode ser associado a ganhos proporcionais e integradores para que possa seguir uma dada referência, para isso foram feitos ajuste de forma empírica para estes ganhos, proporcional e integral, formando um sistema de controle acoplado do LQR com um PI clássico. Portanto os valores utilizados para o PI foram $k_p = 1$ e $k_i = 0,15$. Dessa forma é possível analisar os efeitos de Q e R na dinâmica do sistema, onde como exemplo tem-se a resposta para um $T_{ref} = 80^\circ C$ na Figura 6.6, onde obteve-se uma constante de tempo $\tau = 545s$ e $T_{saida} = 77,2^\circ C$. Pode ser observado também um custo J bem mais elevado, com relação ao caso LQR puro. Mas o custo baixo para aquele caso não se aproxima da realidade da planta, já que o modelo ARX/Hammerstein utilizado na simulação, não foi estimado levando-se em consideração valores negativos de tensão aplicado a planta. Mas pode-se dizer que o resultado foi satisfatório considerando, tanto a regulação quanto a capacidade de seguir uma referência.

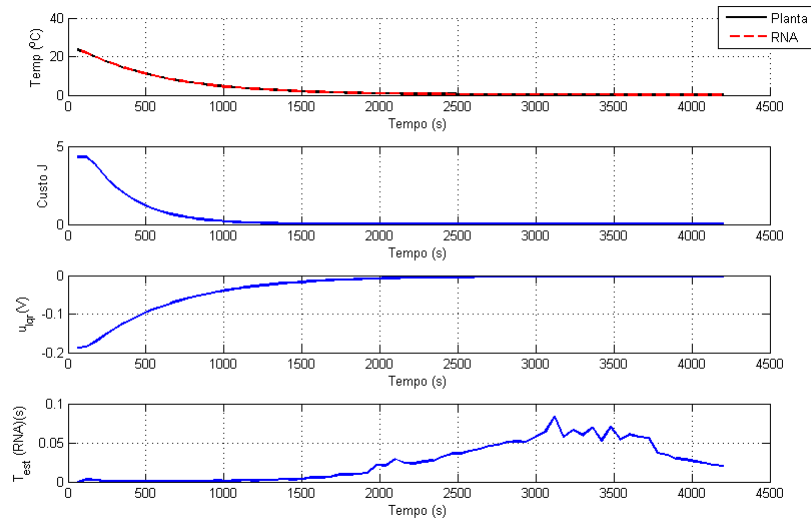


Figura 6.5: Regulador DLQR adaptativo aplicado a planta

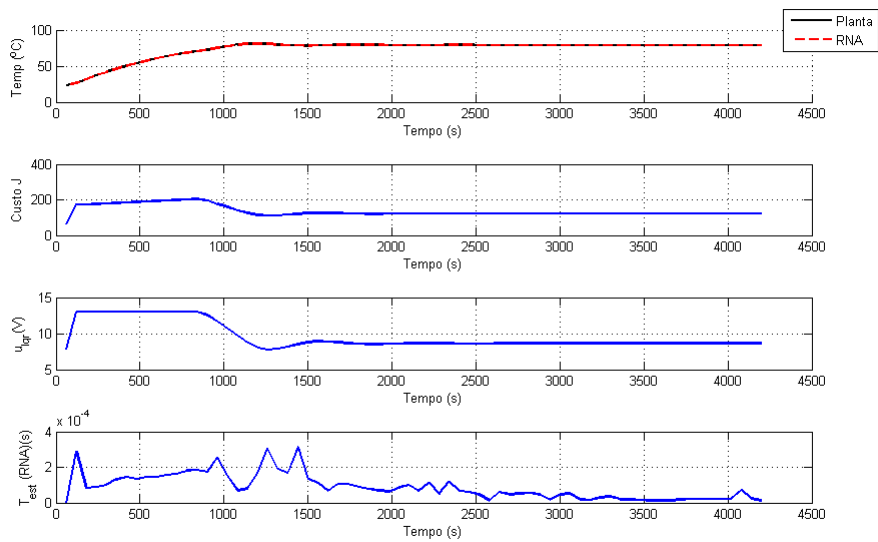


Figura 6.6: Regulador DLQR adaptativo acoplado com controlador PI

Para fins de avaliar o regulador linear quadrático nas duas situações apresentadas é montada uma tabela de desempenho similar ao que foi feito no PID digital. Assim, considerando como convergência para o LQR quando a temperatura atinge $0,3^{\circ}C$, tem-se:

Tabela 6.2: Tabela de desempenho do LQR com $T_{ref} = 0^{\circ}C$ para o LQR puro e $T_{ref} = 80^{\circ}C$ para o LQR/PI

Entradas para LQR e PI/LQR		Desempenho do LQR		Desempenho do LQR/PI			
$Q = a * I$ a	$R = b * 1$ b	Custo Médio J	Tempo de Convergência(s)	T_{regime} $^{\circ}C$	M_p $^{\circ}C$	τ s	Custo Médio J
3	3	0,835	3060	77,78	79,83	560	382
1	3	0,360	3660	76,53	78,55	560	292
0,1	3	0,035	2940	77,31	79,28	560	259
3	1	0,823	3060	77,75	79,79	560	213
1	1	0,304	3900	77,78	79,82	560	128
0,1	1	0,033	4500	76,43	78,56	560	87
3	0,1	0,933	4080	76,63	78,39	560	132
1	0,1	0,306	3960	75,35	76,99	560	48
0,1	0,1	0,029	3360	77,75	79,78	560	12

Em análise da Tabela 6.2, é possível verificar a variação no tempo de convergência para o LQR, variando-se Q e R . Assim pode-se ver que quando se tem uma relação de custo/entrada alta e custo/estado variado de forma decrescente há uma redução do custo e do tempo de convergência, o que ocorre de maneira oposta para relação custo/entrada média e quando esta se torna pequena com relação custo/estado decrescente volta-se a ter um padrão de queda no custo e tempo, onde para este estudo de caso pode-se melhor aproveitar do LQR com ponderações nessas faixas de decrescimento. Já com relação ao LQR acoplado um ganho proporcional e integrador, obteve-se erros de regime muito próximos, bem como valores de pico e constante de tempo. A única mudança significativa foi no custo médio que caiu bruscamente com a diminuição dos valores de a e b e com isso escolher o Q e R que gera um melhor rendimento, que para este caso é de $Q = 0,1 * I$ e $R = 0,1$, oitava linha da tabela.

6.3 Implementação do PID Adaptativo mais Estimação Paramétrica Inteligente em Microcontrolador

Nesta seção é apresentado alguns aspectos e algoritmos embarcados em microcontrolador, como perspectiva de aplicação real, visto a baixa quantidade de trabalhos que desenvolvem a implementação. Em um primeiro momento, tudo é desenvolvido em software de simulação de circuitos, buscando uma aproximação mais real. O diagrama esquemático a seguir ilustra, através de blocos os elementos que compõem o sistema.

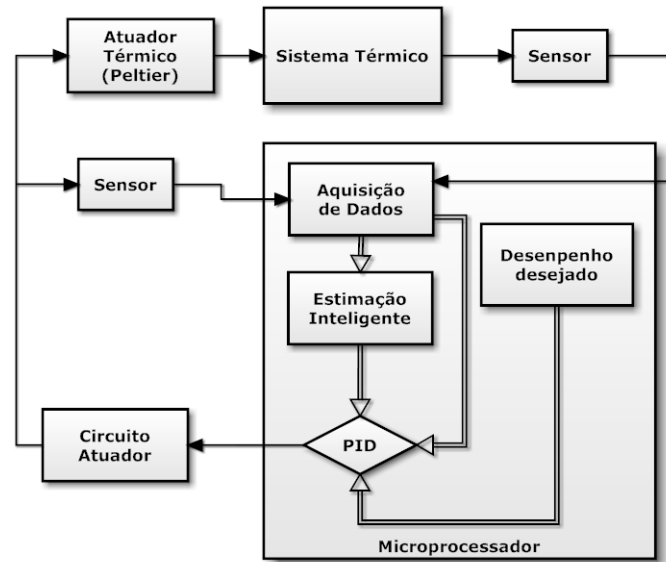


Figura 6.7: Diagrama para implementação do sistema de controle

O diagrama da Fig. 6.7 apresenta o sistema térmico, cujo atuador térmico é a célula Peltier, um sensor para medir a temperatura do ar interno a câmara, já que a do objeto será estimada pelo sensor virtual, um sensor para verificar a tensão aplicada a planta, o circuito atuador que converte o sinal de tomada de decisão do PID, dos padrões de potência do circuito de controle para os do sistema. A aquisição de dados é a função que utiliza os conversores AD (Analógicos/Digitais) para mensurar as variáveis de interesse.

Já de forma embarcada a função de estimação inteligente utilizando os dados provenientes da medição estima os parâmetros da rede neural que é o modelo representativo do sistema. Assim dado um desempenho desejado, juntamente com valores de referência, o bloco PID auto ajustará seus parâmetros na tentativa de impor a dinâmica desejada.

Definidos os blocos funcionais do sistema, é possível apresentar os algoritmos de cada bloco. Os referentes as estimação inteligente, já foram apresentados, porém aqui foi implementado o BP simples, sem aceleração. Mas antes disso, será apresentado um diagrama esquemático aproximado de como o sistema inteligente e controle adaptativos são implementado em um microcontrolador de baixo custo, como o PIC16f877, como é mostrado na Fig.6.8.

No esquemático da Fig.6.7, no *Início* são configuradas as bibliotecas, o tipo de oscilador utilizado, a resolução do conversor A/D e configurações para comunicação com outras interfaces. Posteriormente são alocadas as variáveis globais e da função principal que executa todo o processo e aguarda o comando para inicializar o processo. Um primeiro estímulo é dado a planta, onde é aguardado o tempo de resposta a esse estímulo. Após isso, o primeiro processo de aquisição de dados é realizada e paralelamente a isso um temporizador interno

inicia a contagem de tempo, e assim a função dita *Est_RNA* realiza a estimação inteligente, e assim com base nos valores dos pesos da rede dado o processo de treinamento *online* a função *Est_PID* se auto-ajusta para obter o desempenho desejado. Com isso o temporizador finaliza sua contagem, fornecendo o tempo em microsegundos gastos pelo processo até então. Assim os dados são enviados pela interface de comunicação e o microcontrolador atua sobre a planta através de PWM que se trata de uma modulação de um sinal quadrado por largura de pulso.

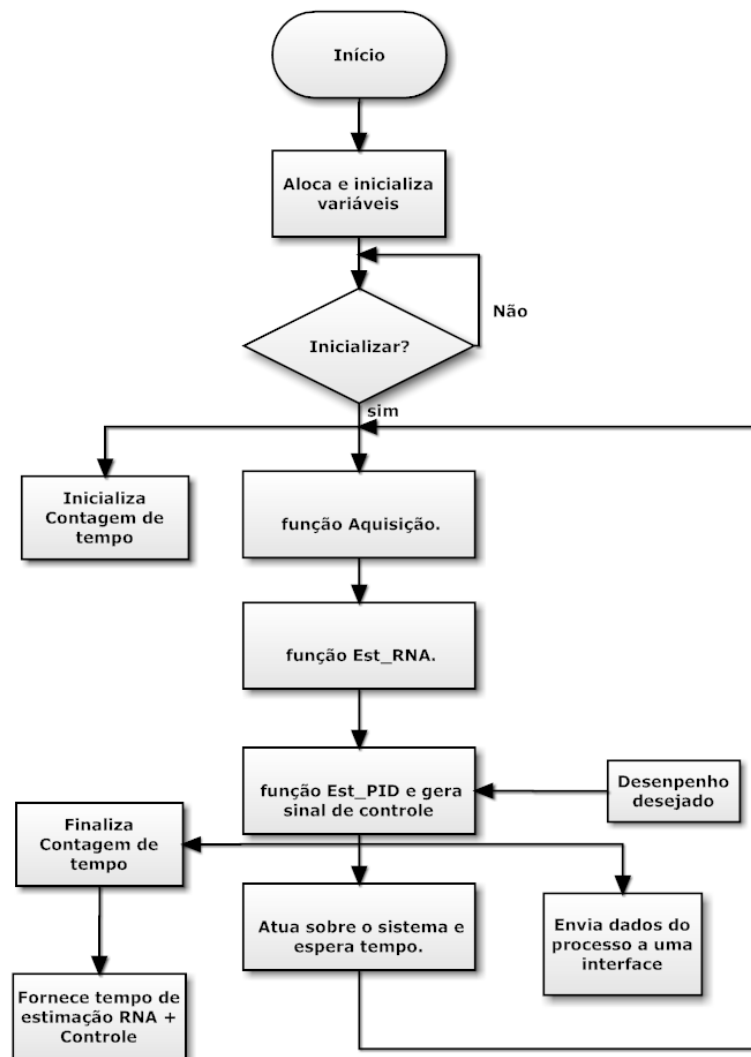


Figura 6.8: Diagrama de tarefas do sistema de controle associado a Fig.6.7

Visto o diagrama geral de execução das funções de estimação inteligente e controle adaptativo, vale salientar algumas características do *hardware* utilizado. Assim, dado o microcontrolador já mencionado, este possui 16 bits de barramento, proporcionando boa resolução no tratamento numérico, conversor A/D de 10 bit, realizando medidas com boa precisão na escala de milivolts, o que é suficiente para o sistema térmico. Possui arquitetura

RISC (Reduced Instruction Set Computer), ou seja um conjunto simplificado de instruções. Portanto, utilizando um oscilador de $8MHz$ como *clock*, foi realizado o desenvolvimento e implementação dos algoritmos em linguagem C para embarca-los.

Dados as características e a forma generalizada de execução, apresenta-se então o Algoritmo 3 da função *Estimador PID* e posteriormente o circuito de atuação para controle do sistema térmico. Portanto, para desenvolver o controle adaptativo, foi utilizada a expressão $x = M^{-1}p$, visto na seção 6.1 deste capítulo. Como estratégia para reduzir os custos computacionais e visto que os métodos de resolução de sistemas lineares iterativos presentes nas bibliografias como Jacobi-Richardson e Gauss-Seidel, por exemplo, nem sempre convergem, utiliza-se o método direto por inversão de matriz.

A estratégia para reduzir os cálculos neste processo foi a utilização da inversão da matriz M somente para achar a solução de um parâmetros, no caso o s_1 e posteriormente, encontra-se os demais por substituição direta.

Algoritmo 3: Estimador PID

Entrada: pesos da rede w_k , parâmetros desejados $gd(n)$, para $k = [0, 3]$ e $n = [0, 4]$ com $gd(0) = 1$.

$$a_1 = -w(0) - 1;$$

$$a_2 = w(0) - w(1);$$

$$\text{Determinante de } M: \det = w^3(4) - (a_1w^2(3) + w(1)w^2(2) - a_2w(3)w(2))w(2);$$

Calculo do primeiro elemento dado a inversa:

$$s_1 = (gd(1)w^3(3) - w(2)gd(2)w^2(3) + w(3 * gd(3)w^2(2) - gd(4)w^3(2))/\det;$$

$$r_2 = (-w(1)s_1 + gd(4))/w(3);$$

$$r_1 = (-a_2s_1 - w(2)r_2 + gd(3))/w(3);$$

$$r_0 = (-a_1s_1 - w(2)r_1 + gd(2))/w(3);$$

Calculado os parâmetros do controlador é possível geral o sinal de controle através da equação a diferença do PID, que é originada das Equações (6.3) e (6.4), é dada por

$$u_{PID}(n) = (1 - s_1)u_{PID}(n - 1) + s_1u_{PID}(n - 2) + r_0e(n) + r_1e(n - 1) + r_2e(n - 2). \quad (6.34)$$

Apresentada a forma de execução do conjunto estimação inteligente com controle PID digital adaptativo, pode-se descrever a forma como este processo foi avaliado, quanto a implementação. Portanto, para o sistema embarcado é utilizado um software de simulação de circuito que permitiu a avaliação do código desenvolvido em linguagem C bem como simular o conjunto hardware e planta. Para que fosse possível a implementação dos algoritmos, foi necessário um estudo dos códigos para se reduzir a quantidade de variáveis envolvendo as

funções desenvolvidas.

O resultado desse teste é apresentado em forma de gráfico, com envio de dados de forma física (real) utilizando a interface de comunicação serial RS-232, onde mostra-se dados de temperatura, sinal de controle e tempos de treinamento da RNA somado ao de determinação dos parâmetros do controlador. Portanto, seja a dinâmica desejada representada pela constante de tempo $\tau = 600s$ e $\zeta = 0.8$, a Fig. 6.9 apresenta a resposta do sistema térmico para um valor de referência de $78^{\circ}C$.

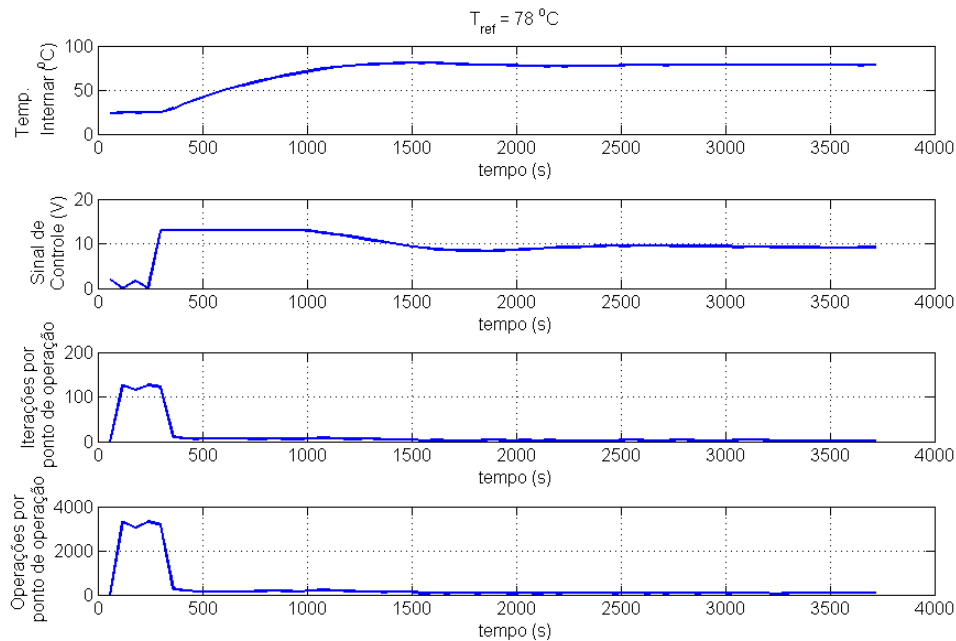


Figura 6.9: Resposta em malha fechada com estimação inteligente e controle adaptativo embarcados utilizando BP

Sabendo-se que o processo de atuação e estimação começa na segunda amostra, aproximadamente $120s$, e sabendo que o valor da constante de tempo se encontra no ponto onde a temperatura é de $57.6^{\circ}C$, no gráfico este ponto se encontra aproximadamente aos $726s$, o que implica que a constante de tempo obtida foi de aproximadamente $606s$, valor considerável aceitável, em face ao desejado e o tempo de amostragem. Observa-se também a presença de um pequeno pico durante o transitório, o que era esperado dado o $\zeta = 0.8$.

Utilizando os mesmos critérios de desempenho, tem-se na Fig.6.10 a resposta em malha fechada utilizando o BP na forma acelerada, onde obteve-se uma resposta semelhante ao caso em que se utiliza o BP clássico.

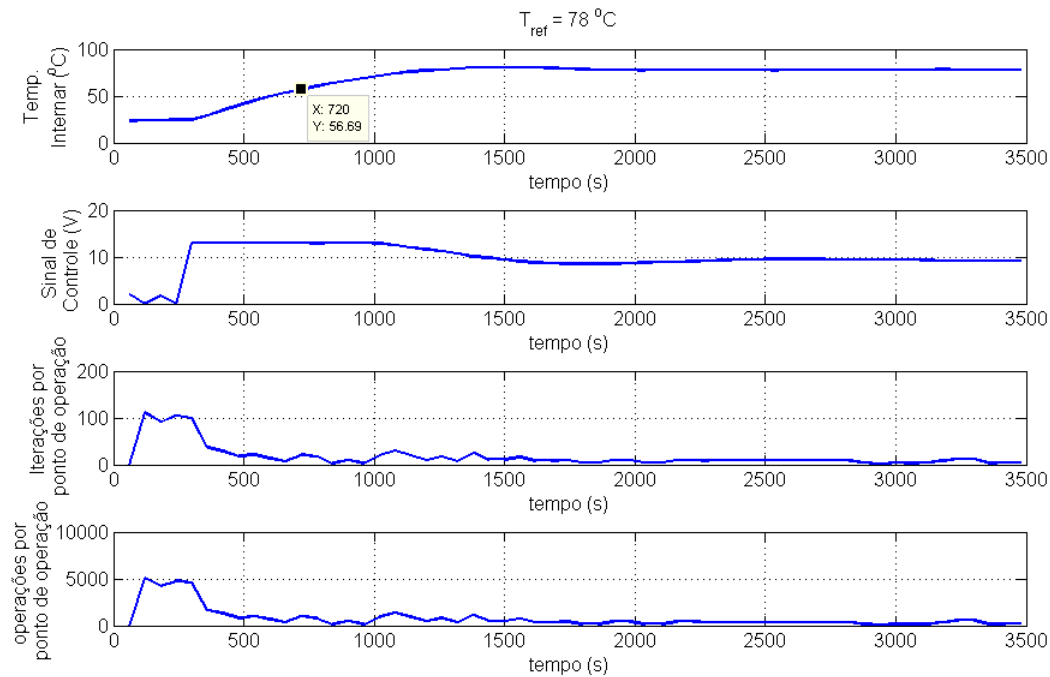


Figura 6.10: Resposta em malha fechada com estimação inteligente e controle adaptativo embarcados utilizando BP-AG

Em análise comparando ambos os resultados, não houve diferenças em termos de resposta dinâmica do sistema em malha aberta. Porém, para o caso deste sistema térmico, o BP clássico teve em média 11 iterações por amostra, onde cada iteração realiza 26 operações levando em consideração atribuições em espaço de memória, adições e multiplicações. Já o BP-AG apresentou certa desvantagem, pois teve uma média por ponto de operação de 16 iteração e dado que cada iteração possui 46 operações, a média foi de 766 operações por ponto de operação entre os dois algoritmos, porém o BP-AG executa três vezes mais cálculos que o BP clássico.

A resposta do sistema em malha fechada utilizando RLS como estimador dos pesos da rede é mostrada na Fig.6.11 também foi obtida. A dinâmica apresentou resultados iguais aos caso BP e BP-AG, mas em termos de esforço computacional apresentou resultado considerado intermediário entre eles, apresentando melhor média de 2 iterações por amostra, porém realizando 32 operações por iteração teve média de 62 operações por ponto de operação, maior que o do BP clássico. Apresentou em um breve momento uma queda do numero de iterações para um devido a um menor ajuste dos pesos da rede, reduzindo o custo computacional nesse instante.

Se for levado em consideração somente o números de operações por iteração a ordem de melhor desempenho é dada respectivamente pelo BP, RLS e BP-AG. Em termos de média

também segue-se a mesma sequencia, o que implica que o BP clássico teve melhor resposta.

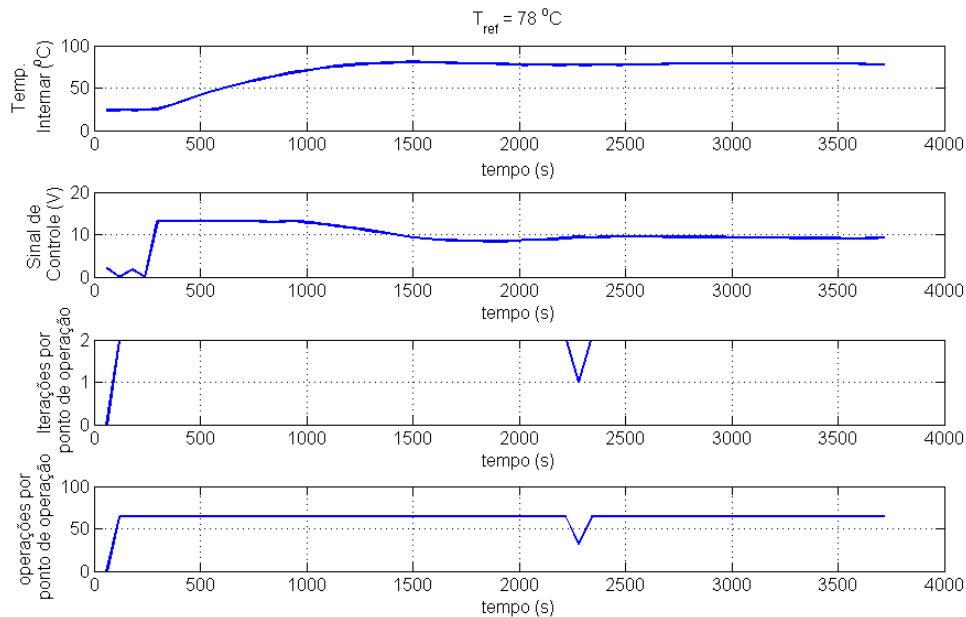


Figura 6.11: Resposta em malha fechada com estimação inteligente e controle adaptativo embarcados utilizando RLS

Ao realizar variações no ponto de operação é possível observar o ajuste dos parâmetros de controle, como mostra a Fig. 6.12. Embora bem sutis, é possível ver a variação dos parâmetros de controle, sendo mais significativa nos instantes iniciais do processo e uma variação de aproximadamente 1, embora não muito perceptível graficamente, em cada parâmetro na segunda mudança de dinâmica, apresentando a adaptabilidade do sistema de controle.

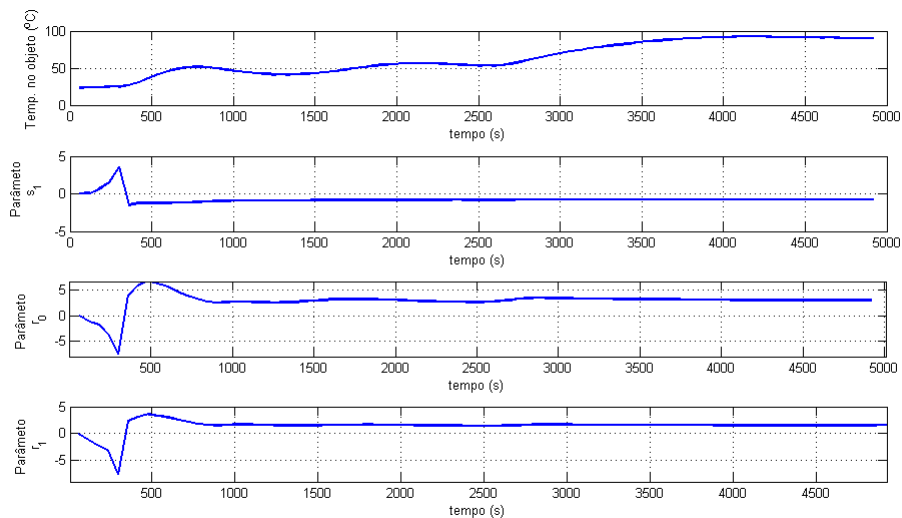


Figura 6.12: Variação dos parâmetros de controle

Considerações Finais

7.1 Conclusão

Neste trabalho foi apresentado o estudo e desenvolvimento de algoritmos para modelagem e estimação *online* de um sistema térmico com características não lineares utilizando inteligência computacional. Apresentou-se também o projeto de um sistema de medição indireta, bem como aplicações em controle utilizando os algoritmos desenvolvidos, onde foram realizadas as devidas análises.

Nesse contexto foi realizado um estudo sobre os modelos clássicos lineares como o ARX e ARMAX e não lineares como NARX e modelos de Hamerstein e Wiener. Dessa forma um estudo sobre redes neurais também foi realizado, onde foram apresentados alguns métodos de treinamento e estruturas voltadas para identificação e modelagem de sistemas físicos presentes nas bibliografias e juntamente foi apresentado duas metodologias de identificação de sistemas utilizando inteligência computacional, onde em um a rede representava o próprio modelo e em outro fazia o papel de estimador paramétrico.

Posteriormente foi apresentado um estudo dos métodos ótimos para treinamento de rede neurais onde foram destacados o método do gradiente que é amplamente utilizado em algoritmos como Backpropagation e métodos de solução de sistemas não lineares. Também foi apresentado o Backpropagation clássico e a proposta de aceleração de gradiente que combinado com o BP clássico obteve-se o BP com aceleração de gradiente denominado de BP-AG. Outro método presente é o RLS padrão, onde foram utilizados na estimação inteligente.

Obtidos os algoritmos para estimação *online*, utilizou-se modelos clássicos como modelo de Hamerstein para representar o sistema térmico para futuras comparações e utilização em simulações. Foi caracterizado um modelo para o sistema de medição indireta, baseados nas análises de linearidade com base em medições obtidas de forma real, onde foi constatado

que o SMI tratava-se neste caso de um processo linear e com isso um sistema ARX de primeira ordem foi estimado utilizando o método dos mínimos quadrados. Assim, uma análise de sua influência sobre o sistema como um todo foi realizado onde foi verificado, através de resposta em frequência que este pouco influência em termos de margem de fase e ganho, viabilizando sua utilização.

Dados os algoritmos ótimos a estimação online inteligente foi realizada sobre duas estruturas de redes proposta, uma na forma ARX e outra no espaço de estado, onde os algoritmos foram avaliados utilizando o modelo de Hammerstain obtido para gerar os dados. Vários teste fora realizados avaliando o tempo de convergência e iteração por ponto de operação, onde o BP-AG não apresentou vantagens relativas ao BP clássico e RLS como era esperado, ao menos para o caso térmico estudado.

Para avaliar os algoritmos de forma mais clara, estes foram implementados em linguagem C e embarcados em um microcontrolador e juntamente a isso foram apresentados duas abordagens de controle adaptativo. Uma utilizando um PID adaptativo digital generalizada e com filtragem no fator derivativo e outra utilizando LQR com recorrência de Riccati. Embarcou-se o PID digital adaptativo no microcontrolador e foi avaliada a performance obtida em malha fechada e comparada com a desejada, obtendo-se bons resultados. Neste contexto, os três algoritmos foram avaliados quanto a quantidade de operações dispendida onde o BP clássico apresentou maior vantagem. Foi apresentado também a estrutura de programação implementada no chip onde todo processo foi simulado utilizando um simulador de circuito que integra dispositivos eletrônicos e modelos de sistemas.

Portanto, os resultados foram satisfatórios, onde o trabalho apresentou todas as etapas propostas, apresentado desde os métodos clássicos aos métodos com inteligência computacional, avaliando o sistema de medição indireta e aplicando os métodos em controle clássico adaptativo e ótimo.

7.2 Publicações

- Identificação e Métodos de Controle Utilizando Fuzzy e Chaveamento para Sistemas Térmicos, Denis Sá, Mailson Barro e João Viana F. Neto. CBA 2014.
- Non-invasive PID Control of Peltier Thermal Actuator based on Artificial Neural Network, Denis Sá, João Viana F. Neto e Marcio Eduardo G, Silva. I2MTC, IEEE, 2012.
- State Space Modeling of Thermal Actuator Based on Peltier Cells for Indirect Measurements and Optmal Control, João Viana F. Neto, Denis Sá e Leandro R. Lopes. UKSIM,

IEEE, 2010

7.3 Trabalhos Futuros

Nesta seção são propostos alguns trabalhos futuros como os apresentados abaixo:

- Busca por melhorias dos Métodos de aceleração de gradiente para treinamento de redes neurais;
- Estudo de sistemas embarcados para implementação de estimação online e controle adaptativo;
- Estudo de estruturas generalizadas de redes neurais para identificação de sistemas.

Bibliografia

- AGUIRRE, L. *Introdução à Identificação de Sistemas – Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais*. [S.l.]: Editora UFMG, 2006. ISBN 9788570415844.
- ALANIS, A. Y. et al. Real-time recurrent neural state estimation. *Neural Networks, IEEE Transactions on*, v. 22, n. 3, p. 497–505, March 2011. ISSN 1045-9227.
- ALFARO-PONCE, M.; CRUZ, A. A.; CHAIREZ, I. Adaptive identifier for uncertain complex nonlinear systems based on continuous neural networks. *Neural Networks and Learning Systems, IEEE Transactions on*, v. 25, n. 3, p. 483–494, March 2014. ISSN 2162-237X.
- ASTROM, K.; WITTENMARK, B. *Adaptive Control: Second Edition*. [S.l.]: Dover Publications, 2008. (Dover Books on Electrical Engineering). ISBN 9780486319148.
- BARUCH, I. et al. Decentralized direct and indirect i-term adaptive fuzzy-neural control of a bioprocess plant. In: *Fuzzy Information Processing Society (NAFIPS), 2012 Annual Meeting of the North American*. [S.l.: s.n.], 2012. p. 1–8. ISSN pending.
- BASHEER, I.; HAJMEER, M. Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods*, v. 43, n. 1, p. 3 – 31, 2000. ISSN 0167-7012.
- BASRI, Y. E. et al. Rotational adaptative-multilevel converter control strategy based on fpga. In: *IECON 2011 - 37th Annual Conference on IEEE Industrial Electronics Society*. [S.l.: s.n.], 2011. p. 2826–2832. ISSN 1553-572X.
- BAZAEI, A.; MOALLEM, M. Online neural identification of multi-input multi-output systems. *Control Theory Applications, IET*, v. 1, n. 1, p. 44–50, January 2007. ISSN 1751-8644.
- BAZARAA, D. S. M. S.; SHETTY, M. *Nonlinear Programming, Theory and Algorithms*. 3. ed. [S.l.]: Wiley Interscience, 2006.
- BELLMAN, R.; KALABA, R. On adaptive control processes. *Automatic Control, IRE Transactions on*, v. 4, n. 2, p. 1–9, Nov 1959. ISSN 0096-199X.

BHARADWAJ, R.; PARLOS, A.; TOLIYAT, H. Adaptive neural network-based state filter for induction motor speed estimation. In: *Industrial Electronics Society, 1999. IECON '99 Proceedings. The 25th Annual Conference of the IEEE*. [S.l.: s.n.], 1999. v. 3, p. 1283–1288 vol.3.

BHASIN, S. et al. Robust identification-based state derivative estimation for nonlinear systems. *Automatic Control, IEEE Transactions on*, v. 58, n. 1, p. 187–192, 2013. ISSN 0018-9286.

BILLINGS, S.; FAKHOURI, S. Identification of nonlinear systems using the wiener model. *Electronics Letters*, v. 13, n. 17, p. 502–504, August 1977. ISSN 0013-5194.

BISHOP, R.; PAYNTER, S.; SUNKEL, J. W. Adaptive control of space station with control moment gyros. *Control Systems, IEEE*, v. 12, n. 5, p. 23–28, Oct 1992. ISSN 1066-033X.

BOILLEREAUX, L.; FLAUS, J. M. A new approach for designing model-based indirect sensors. *Control Systems Technology, IEEE Transactions on*, v. 8, n. 4, p. 601–608, Jul 2000. ISSN 1063-6536.

BOLSTAD, A. et al. Identification and compensation of wiener-hammerstein systems with feedback. In: *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. [S.l.: s.n.], 2011. p. 4056–4059. ISSN 1520-6149.

BUBNICKI, Z. Automatic identification of nonlinear static plant by degressive feedback and digital computer. *Electronics Letters*, v. 4, n. 11, p. 235–236, May 1968. ISSN 0013-5194.

CASTILLO FONTENLA-ROMERO, G.-B.; ALONSO-BETANZOS. A global optimum approach for one-layer neural networks. *Neural Computation*, v. 14, n. 6, p. 1429–1449, June 2002. ISSN 0899-7667.

CHAOUI, H.; SICARD, P. Adaptive fuzzy logic control of permanent magnet synchronous machines with nonlinear friction. *Industrial Electronics, IEEE Transactions on*, v. 59, n. 2, p. 1123–1133, Feb 2012. ISSN 0278-0046.

CHEN, B. et al. Approximation-based adaptive neural control design for a class of nonlinear systems. *Cybernetics, IEEE Transactions on*, v. 44, n. 5, p. 610–619, May 2014. ISSN 2168-2267.

CHEN, C. *Linear System Theory and Design*. [S.l.]: Oxford University Press, Incorporated, 2012. (The Oxford Series in Electrical and Computer Engineering Series). ISBN 9780199959570.

CHEN, S.; WANG, X.; HARRIS, C. Narx-based nonlinear system identification using orthogonal least squares basis hunting. *Control Systems Technology, IEEE Transactions on*, v. 16, n. 1, p. 78–84, Jan 2008. ISSN 1063-6536.

CHIANG, J.-H.; HO, S.-H. A combination of rough-based feature selection and rbf neural network for classification using gene expression data. *NanoBioscience, IEEE Transactions on*, v. 7, n. 1, p. 91–99, March 2008. ISSN 1536-1241.

- CHON, K.; COHEN, R. Linear and nonlinear arma model parameter estimation using an artificial neural network. *Biomedical Engineering, IEEE Transactions on*, v. 44, n. 3, p. 168–174, March 1997. ISSN 0018-9294.
- CHOW, M.-Y.; YEE, S. Methodology for on-line incipient fault detection in single-phase squirrel-cage induction motors using artificial neural networks. *Energy Conversion, IEEE Transactions on*, v. 6, n. 3, p. 536–545, Sep 1991. ISSN 0885-8969.
- CINCOTTI, S.; DANERI, I. Neural network identification of a nonlinear circuit model of hysteresis. *Electronics Letters*, v. 33, n. 13, p. 1154–1156, Jun 1997. ISSN 0013-5194.
- CLYMER, A. Indirect measurement of process variables by minicomputer. *Industrial Electronics and Control Instrumentation, IEEE Transactions on*, IECI-17, n. 4, p. 358–362, June 1970. ISSN 0018-9421.
- COOPER, B.; FALKNER, A. Identification of simple nonlinear systems. *Electrical Engineers, Proceedings of the Institution of*, v. 122, n. 7, p. 753–755, July 1975. ISSN 0020-3270.
- DJUKANOVIC, M. et al. Neuro-fuzzy controller of low head hydropower plants using adaptive-network based fuzzy inference system. *Energy Conversion, IEEE Transactions on*, v. 12, n. 4, p. 375–381, Dec 1997. ISSN 0885-8969.
- DURDAN, M.; KACUR, J. Indirect temperatures measurement in the ucg process. In: *Carpathian Control Conference (ICCC), 2013 14th International*. [S.l.: s.n.], 2013. p. 73–78.
- ERTUNC, H. Prediction of the pool boiling critical heat flux using artificial neural network. *Components and Packaging Technologies, IEEE Transactions on*, v. 29, n. 4, p. 770–777, Dec 2006. ISSN 1521-3331.
- FRANKLIN, G. F.; POWELL, J. D.; WORKMAN, M. C. *Digital Control of Dynamic Systems*. [S.l.]: Adison - Wesley, 1990.
- FU, Z.-J. et al. Nonlinear systems identification and control via dynamic multitime scales neural networks. *Neural Networks and Learning Systems, IEEE Transactions on*, v. 24, n. 11, p. 1814–1823, Nov 2013. ISSN 2162-237X.
- FURUTA, K.; PAQUET, J. An iterative procedure for the state estimation of a nonlinear discrete process. *Automatic Control, IEEE Transactions on*, v. 15, n. 3, p. 400–403, Jun 1970. ISSN 0018-9286.
- GALLMAN, P. A comparison of two hammerstein model identification algorithms. *Automatic Control, IEEE Transactions on*, v. 21, n. 1, p. 124–126, Feb 1976. ISSN 0018-9286.
- GAO, X.; LIAO, L.-Z. A new one-layer neural network for linear and quadratic programming. *Neural Networks, IEEE Transactions on*, v. 21, n. 6, p. 918–929, June 2010. ISSN 1045-9227.
- GE, S.; HANG, C.-C.; ZHANG, T. Adaptive neural network control of nonlinear systems by state and output feedback. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, v. 29, n. 6, p. 818–828, Dec 1999. ISSN 1083-4419.

- GE, S.; ZHANG, J.; LEE, T.-H. Adaptive neural network control for a class of mimo nonlinear systems with disturbances in discrete-time. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, v. 34, n. 4, p. 1630–1645, Aug 2004. ISSN 1083-4419.
- GE, S. S.; WANG, J. Robust adaptive neural control for a class of perturbed strict feedback nonlinear systems. *Neural Networks, IEEE Transactions on*, v. 13, n. 6, p. 1409–1419, Nov 2002. ISSN 1045-9227.
- GEVINS, A.; MORGAN, N. Applications of neural-network (nn) signal processing in brain research. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, v. 36, n. 7, p. 1152–1161, Jul 1988. ISSN 0096-3518.
- GHADIMI, S.; LAN, G. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *arXiv preprint arXiv:1310.3787*, 2013.
- GIOVANINI, L. Robust adaptive control using multiple models, switching and tuning. *Control Theory Applications, IET*, v. 5, n. 18, p. 2168–2178, December 2011. ISSN 1751-8644.
- GOVINDAN, G. N.; HIGGINS, R. A. Convergence of an identification technique for nonlinear systems. *Computers, IEEE Transactions on*, C-21, n. 11, p. 1216–1219, Nov 1972. ISSN 0018-9340.
- HAMETNER, C.; JAKUBEK, S. Nonlinear identification with local model networks using gtl techniques and equality constraints. *Neural Networks, IEEE Transactions on*, v. 22, n. 9, p. 1406–1418, Sept 2011. ISSN 1045-9227.
- HAYKIN, S. *Neural Networks, A Comprehensive Foundation*. 2. ed. [S.l.]: Prentice Hall International, 1999.
- HAYKIN, S. *Redes Neurais - 2ed*. [S.l.]: BOOKMAN COMPANHIA ED, 2001. ISBN 9788573077186.
- HELFERTY, J. et al. On-line parameter identification and model validation for distributed parameter models with applications to solar thermal systems. In: *System Theory, 1988., Proceedings of the Twentieth Southeastern Symposium on*. [S.l.: s.n.], 1988. p. 135–140. ISSN 0094-2898.
- HESSELROTH, T. et al. Neural network control of a pneumatic robot arm. *Systems, Man and Cybernetics, IEEE Transactions on*, v. 24, n. 1, p. 28–38, Jan 1994. ISSN 0018-9472.
- HILTZ, F. F. Analog computer simulation of a neural element. *Bio-Medical Electronics, IRE Transactions on*, v. 9, n. 1, p. 12–20, Jan 1962. ISSN 0096-1884.
- HUYNH, H. T.; WON, Y. Online training for single hidden-layer feedforward neural networks using rls-elm. In: *Computational Intelligence in Robotics and Automation (CIRA), 2009 IEEE International Symposium on*. [S.l.: s.n.], 2009. p. 469–473.
- IBTISSEM, C.; NOUREDINE, L. A hybrid method based on conjugate gradient trained neural network and differential evolution for non linear systems identification. In: *Electrical Engineering and Software Applications (ICEESA), 2013 International Conference on*. [S.l.: s.n.], 2013. p. 1–5.

- JANCZAK, A. *Identification of Nonlinear Systems Using Neural Networks and Polynomial Models: A Block-Oriented Approach*. [S.l.]: Springer, 2004. (Lecture Notes in Control and Information Sciences). ISBN 9783540231851.
- JURADO, F. et al. Continuous-time neural identification for a 2 dof vertical robot manipulator. In: *Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2011 IEEE*. [S.l.: s.n.], 2011. p. 77–82.
- JURADO, F. et al. Continuous-time neural identification for a 2 dof vertical robot manipulator. In: *Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2011 IEEE*. [S.l.: s.n.], 2011. p. 77–82.
- KAHVECI, N.; IOANNOU, P.; MIRMIRANI, M. Adaptive lq control with anti-windup augmentation to optimize uav performance in autonomous soaring applications. *Control Systems Technology, IEEE Transactions on*, v. 16, n. 4, p. 691–707, July 2008. ISSN 1063-6536.
- KANMACHI, T.; TAKAHASAH, I. Sensor-less speed control of an induction motor. *Industry Applications Magazine, IEEE*, v. 1, n. 1, p. 22–27, Jan 1995. ISSN 1077-2618.
- KAWABATA, S.; FUKUSHIMA, T. Indirect measurement of ac current distributions in a bi-2223 solenoidal coil by a pickup coil method. *Applied Superconductivity, IEEE Transactions on*, v. 15, n. 2, p. 3628–3631, June 2005. ISSN 1051-8223.
- KAWABATA, S.; MIYATA, K.; ARAKAWA, K. Indirect measurement of current distributions in bi-2223/ag tapes by pickup coil method. *Applied Superconductivity, IEEE Transactions on*, v. 11, n. 1, p. 3094–3097, Mar 2001. ISSN 1051-8223.
- KHALIL, H. *Nonlinear Systems*. [S.l.]: Prentice Hall, 2002. ISBN 9780130673893.
- KHOSROWABADI, R. et al. Ernn: A biologically inspired feedforward neural network to discriminate emotion from eeg signal. *Neural Networks and Learning Systems, IEEE Transactions on*, v. 25, n. 3, p. 609–620, March 2014. ISSN 2162-237X.
- KING, S.-Y.; HWANG, J.-N. Neural network architectures for robotic applications. *Robotics and Automation, IEEE Transactions on*, v. 5, n. 5, p. 641–657, Oct 1989. ISSN 1042-296X.
- KO, C.-N. Identification of non-linear systems using radial basis function neural networks with time-varying learning algorithm. *Signal Processing, IET*, v. 6, n. 2, p. 91–98, 2012. ISSN 1751-9675.
- KORENBERG, M. Statistical identification of difference equation representations for nonlinear systems. *Electronics Letters*, v. 19, n. 5, p. 175–176, March 1983. ISSN 0013-5194.
- KOSTUR, K. Principles of indirect measurement temperature. In: *Carpathian Control Conference (ICCC), 2012 13th International*. [S.l.: s.n.], 2012. p. 336–340.
- KRAFT, L.; CAMPAGNA, D. A comparison between cmac neural network control and two traditional adaptive control systems. *Control Systems Magazine, IEEE*, v. 10, n. 3, p. 36–43, April 1990. ISSN 0272-1708.

- KRANJCIC, D.; STUMBERGER, G. Differential evolution-based identification of the nonlinear kaplan turbine model. *Energy Conversion, IEEE Transactions on*, v. 29, n. 1, p. 178–187, March 2014. ISSN 0885-8969.
- KUIPERS, M.; IOANNOU, P. Multiple model adaptive control with mixing. *Automatic Control, IEEE Transactions on*, v. 55, n. 8, p. 1822–1836, Aug 2010. ISSN 0018-9286.
- LANDAU, I. D. *Digital Control Systems*. [S.l.]: Springer, 1938.
- LEITHOLD, L. *O cálculo com geometria analítica*. 3. ed. [S.l.]: Harbra, 1994. ISBN 9788529402062.
- LEWIS, F.; VRABIE, D.; VAMVOUDAKIS, K. Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers. *Control Systems, IEEE*, v. 32, n. 6, p. 76–105, Dec 2012. ISSN 1066-033X.
- LEWIS, F. L. *Applied Optimal Control & Estimation - Digital Design & Implementation*. 1. ed. New Jersey: Prentice Hall, 1992. (Digital signal Processing, v. 17).
- LI, G.; WEN, C. Identification of wiener systems with clipped observations. *Signal Processing, IEEE Transactions on*, v. 60, n. 7, p. 3845–3852, July 2012. ISSN 1053-587X.
- LI, Y.; SUNDARARAJAN, N.; SARATCHANDRAN, P. Analysis of minimal radial basis function network algorithm for real-time identification of nonlinear dynamic systems. *Control Theory and Applications, IEE Proceedings -*, v. 147, n. 4, p. 476–484, Jul 2000. ISSN 1350-2379.
- LIMA G.S. DEEP, L. d. A. H. N. A.; FONTANA, M. A gain-scheduling pid-like controller for peltier-based thermal hysteresis characterization platform. *IEEE Instrumentation and Measurement Technology Conference*, may, 2001.
- LIN, H.-T.; LIANG, T.-J.; CHEN, S.-M. Estimation of battery state of health using probabilistic neural network. *Industrial Informatics, IEEE Transactions on*, v. 9, n. 2, p. 679–685, May 2013. ISSN 1551-3203.
- LIU, D.; WANG, D.; LI, H. Decentralized stabilization for a class of continuous-time nonlinear interconnected systems using online learning optimal control approach. *Neural Networks and Learning Systems, IEEE Transactions on*, v. 25, n. 2, p. 418–428, Feb 2014. ISSN 2162-237X.
- LIU, L.-Z. et al. Structure identification and parameter estimation of biological s-systems. In: *Bioinformatics and Biomedicine (BIBM), 2010 IEEE International Conference on*. [S.l.: s.n.], 2010. p. 329–334.
- LJUNG, L. *System Identification: Theory for the User*. [S.l.]: Pearson Education, 1998. ISBN 9780132440530.
- LORY, H.; LAI, D.; HUGGINS, W. On the use of growing harmonic exponentials to identify static nonlinear operators. *Automatic Control, IRE Transactions on*, v. 4, n. 2, p. 91–99, Nov 1959. ISSN 0096-199X.

- LUENBERGER, D.; YE, Y. *Linear and Nonlinear Programming*. [S.l.]: Springer, 2008. (International Series in Operations Research & Management Science). ISBN 9780387745022.
- MAITI, S. et al. An adaptive speed sensorless induction motor drive with artificial neural network for stability enhancement. *Industrial Informatics, IEEE Transactions on*, v. 8, n. 4, p. 757–766, Nov 2012. ISSN 1551-3203.
- MANITSAS, E. et al. Distribution system state estimation using an artificial neural network approach for pseudo measurement modeling. *Power Systems, IEEE Transactions on*, v. 27, n. 4, p. 1888–1896, Nov 2012. ISSN 0885-8950.
- MAO, X.; SHANG, S.; LIU, X. Groundwater level predictions using artificial neural networks. *Tsinghua Science and Technology*, v. 7, n. 6, p. 574–579, Dec 2002.
- MARMARELIS, P. Z.; NAKA, K.-i. Identification of multi-input biological systems. *Biomedical Engineering, IEEE Transactions on*, BME-21, n. 2, p. 88–101, 1974. ISSN 0018-9294.
- MAYHAN, R.; BAILEY, R. An indirect measurement of the effect dielectric constant and loss tangent of typical concrete roadways. *Antennas and Propagation, IEEE Transactions on*, v. 23, n. 4, p. 565–569, Jul 1975. ISSN 0018-926X.
- MEZGHANI-MARRAKCHI, I. et al. Nonlinear audio systems identification through audio input gaussianization. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, v. 22, n. 1, p. 41–53, Jan 2014. ISSN 2329-9290.
- MILANESE, R. T. M.; VICINO., A. *Robustness in identification and control*. [S.l.]: Plenum Press, 1989. (Applied information technology). ISBN 9780306432514.
- MINAMITANI, H.; HAGITA, N. A neural network model of pain mechanisms: Computer simulation of the central neural activities essential for the pain and touch sensations. *Systems, Man and Cybernetics, IEEE Transactions on*, v. 11, n. 7, p. 481–493, July 1981. ISSN 0018-9472.
- MINGXUAN, S.; HAIGANG, H.; YING, K. Identification of nonlinear time-varying systems using time-varying dynamic neural networks. In: *Control Conference (CCC), 2013 32nd Chinese*. [S.l.: s.n.], 2013. p. 1911–1916.
- MORRIS, R.; NEUMAN, C. Model reference adaptive control with multiple samples between parameter adjustments. *Automatic Control, IEEE Transactions on*, v. 26, n. 2, p. 534–537, Apr 1981. ISSN 0018-9286.
- MOUZINHO, L. F. et al. Indirect measurement of the temperature in a resistive furnace. In: *Instrumentation and Measurement Technology Conference, 2006. IMTC 2006. Proceedings of the IEEE*. [S.l.: s.n.], 2006. p. 666–671. ISSN 1091-5281.
- NAKAMORI, S. On-line identification of time variant parameters using covariance information. In: *Acoustics, Speech, and Signal Processing, IEEE International Conference on ICASSP '86*. [S.l.: s.n.], 1986. v. 11, p. 2723–2726.

- NARENDRA, K.; GALLMAN, P. An iterative method for the identification of nonlinear systems using a hammerstein model. *Automatic Control, IEEE Transactions on*, v. 11, n. 3, p. 546–550, Jul 1966. ISSN 0018-9286.
- NATIONAL, S. *LM35, Precision Centigrade Temperature Sensors*. [S.l.], 2000. Disponível em: <<http://pdf1.alldatasheet.com/datasheet-pdf/view/8875/NSC/LM35DZ.html>>.
- NAWROCKI, M.; NAWROCKA, A. Artificial neural networks for identification in real time of the robot manipulator model parameters. In: *Control Conference (ICCC), 2014 15th International Carpathian*. [S.l.: s.n.], 2014. p. 383–386.
- NESTEROV, Y. A method of solving a convex programming problem with convergence rate $O(1/k^2)$. *Soviet Mathematics Doklady*, v. 27, n. 2, p. 372–376, 1983.
- NETO J.V.F., L. L.; SA, D. State space modelling of thermal actuators based on peltier cells for indirect measurements and optimal control. p. 392–397, 2010.
- NING, H.; JING, X.; CHENG, L. Online identification of nonlinear spatiotemporal systems using kernel learning approach. *Neural Networks, IEEE Transactions on*, v. 22, n. 9, p. 1381–1394, Sept 2011. ISSN 1045-9227.
- NIRKHI, S. Potential use of artificial neural network in data mining. In: *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on*. [S.l.: s.n.], 2010. v. 2, p. 339–343.
- NODLAND, D.; ZARGARZADEH, H.; JAGANNATHAN, S. Neural network-based optimal adaptive output feedback control of a helicopter uav. *Neural Networks and Learning Systems, IEEE Transactions on*, v. 24, n. 7, p. 1061–1073, July 2013. ISSN 2162-237X.
- NORGAARD O. RAVN, N. K. P. M.; HANSEN, L. K. *Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner's Handbook*. [S.l.]: Springer London, 2000. (Advanced Textbooks in Control and Signal Processing). ISBN 9781852332273.
- OGATA, K. *Engenharia de controle moderno*. [S.l.]: Pearson Prentice Hall, 2011. ISBN 9788576058106.
- OGUNFUNMI, T. *Adaptive Nonlinear System Identification, The Volterra and Wiener Model Approaches*. [S.l.]: Springer, 2007.
- ORLOWSKA-KOWALSKA, T.; KOWALSKI, C. Neural network application for flux and speed estimation in the sensorless induction motor drive. In: *Industrial Electronics, 1997. ISIE '97., Proceedings of the IEEE International Symposium on*. [S.l.: s.n.], 1997. p. 1253–1258 vol.3.
- PARKER, S.; PERRY, F. A discrete arma model for nonlinear system identification. *Circuits and Systems, IEEE Transactions on*, v. 28, n. 3, p. 224–233, Mar 1981. ISSN 0098-4094.
- PASQUALETTI, F.; DORFLER, F.; BULLO, F. Attack detection and identification in cyber-physical systems. *Automatic Control, IEEE Transactions on*, v. 58, n. 11, p. 2715–2729, 2013. ISSN 0018-9286.

- PATRINOS, P.; BEMPORAD, A. An accelerated dual gradient-projection algorithm for embedded linear model predictive control. *Automatic Control, IEEE Transactions on*, v. 59, n. 1, p. 18–33, Jan 2014. ISSN 0018-9286.
- PENNELLI, G.; NANNINI, A.; MACUCCI, M. Indirect measurement of thermal conductivity in silicon nanowires. *Journal of Applied Physics*, v. 115, n. 8, p. 084507–084507–5, Feb 2014. ISSN 0021-8979.
- PEREZ, R.; MATOS, E.; FERNANDEZ, S. Identification of the nonlinear model proposed by the mit for power transformer applying genetic algorithms. *Latin America Transactions, IEEE (Revista IEEE America Latina)*, v. 7, n. 6, p. 636–642, Dec 2009. ISSN 1548-0992.
- PEZZOTTI, J. V. G.; LONDONO. Discrete deadbeat control of a plant pressure through identification by neural networks. *IEEE Laton America Transactions*, 2012.
- PRAMANIK, S.; SATISH, L. Estimation of series capacitance of a transformer winding based on frequency-response data: An indirect measurement approach. *Power Delivery, IEEE Transactions on*, v. 26, n. 4, p. 2870–2878, Oct 2011. ISSN 0885-8977.
- RAOL, J. R.; MADHURANATH, H. Neural network architectures for parameter estimation of dynamical systems. *Control Theory and Applications, IEE Proceedings -*, v. 143, n. 4, p. 387–394, Jul 1996. ISSN 1350-2379.
- REN, X.; LV, X. Identification of extended hamerstein systems using dynamic self-optimizing neural networks. *Neural Networks, IEEE Transactions on*, v. 22, n. 8, p. 1169–1179, Aug 2011. ISSN 1045-9227.
- ROY, R.; DERUSSO, P. A digital orthogonal model for nonlinear processes with two-level inputs. *Automatic Control, IRE Transactions on*, v. 7, n. 5, p. 93–101, Oct 1962. ISSN 0096-199X.
- SA DENIS F.;FONSECA NETO, J. V. S. B. M. Sousa de. IdentificaÃ§Ã£o e mÃ©todos de controle utilizando fuzzy e chaveamento para sistemas tÃ©rmicos. In: UFMG. [S.l.]: Congresso Brasileiro de AutomÃ¡tica CBA, 2014.
- SANG, Q.; TAO, G. Gain margins of adaptive control systems. *Automatic Control, IEEE Transactions on*, v. 55, n. 1, p. 104–115, Jan 2010. ISSN 0018-9286.
- SANNER, R.; SLOTINE, J. Stable adaptive control of robot manipulators using neural networks. *Neural Computation*, v. 7, n. 4, p. 753–790, July 1995. ISSN 0899-7667.
- SCHOUKENS, M.; PINTELON, R.; ROLAIN, Y. Parametric identification of parallel hamerstein systems. *Instrumentation and Measurement, IEEE Transactions on*, v. 60, n. 12, p. 3931–3938, Dec 2011. ISSN 0018-9456.
- SHANMUGAM, K.; JONG, M. Identification of nonlinear systems in frequency domain. *Aerospace and Electronic Systems, IEEE Transactions on*, AES-11, n. 6, p. 1218–1225, Nov 1975. ISSN 0018-9251.
- SHEN, Y.; LIU, L.; DOWELL, E. Adaptive fault-tolerant robust control for a linear system with adaptive fault identification. *Control Theory Applications, IET*, v. 7, n. 2, p. 246–252, Jan 2013. ISSN 1751-8644.

- SHI, D.; GAO, Y. A new method for identifying electromagnetic radiation sources using backpropagation neural network. *Electromagnetic Compatibility, IEEE Transactions on*, v. 55, n. 5, p. 842–848, Oct 2013. ISSN 0018-9375.
- SILVA, I. D.; SPATTI, D.; FLAUZINO, R. *REDES NEURAIAS ARTIFICIAIS PARA ENGENHARIA E: CIENCIAS APLICADAS - CURSO PRATICO*. [S.l.]: ARTLIBER, 2010. ISBN 9788588098534.
- SISHUBA, S.; REDFERN, M. Adaptive control system for continuity of supply using dispersed generators. *Generation, Transmission and Distribution, IEE Proceedings-*, v. 152, n. 1, p. 23–30, Jan 2005. ISSN 1350-2360.
- STANTON, S. C. et al. Nonlinear piezoelectricity in electroelastic energy harvesters: Modeling and experimental identification. *Journal of Applied Physics*, v. 108, n. 7, p. 074903–074903–9, Oct 2010. ISSN 0021-8979.
- STIBITZ, G. R.; MCCANN, F. V. Indirect measurement of intracellular conductivity. *Biomedical Engineering, IEEE Transactions on*, BME-24, n. 3, p. 300–302, May 1977. ISSN 0018-9294.
- TAHERI, A.; MOHAMMADBEIGI, M. Speed sensor-less estimation and predictive control of six-phase induction motor using extended kalman filter. In: *Power Electronics, Drive Systems and Technologies Conference (PEDSTC), 2014 5th*. [S.l.: s.n.], 2014. p. 13–18.
- TITTI, A.; SQUARTINI, S.; PIAZZA, F. A new time-variant neural based approach for nonstationary and non-linear system identification. In: *Circuits and Systems, 2005. ISCAS 2005. IEEE International Symposium on*. [S.l.: s.n.], 2005. p. 5134–5137 Vol. 5.
- TURCHETTI, C. et al. A computational intelligence technique for the identification of non-linear non-stationary systems. In: *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on*. [S.l.: s.n.], 2008. p. 3034–3038. ISSN 1098-7576.
- VAZQUEZ, E.; FLEURY, G.; WALTER, E. Kriging for indirect measurement with application to flow measurement. *Instrumentation and Measurement, IEEE Transactions on*, v. 55, n. 1, p. 343–349, Feb 2006. ISSN 0018-9456.
- WANG, L.-X. Design and analysis of fuzzy identifiers of nonlinear dynamic systems. *Automatic Control, IEEE Transactions on*, v. 40, n. 1, p. 11–23, Jan 1995. ISSN 0018-9286.
- WEYMAERE, N.; MARTENS, J.-P. Neural network classifiers under changing a priori conditions. *Electronics Letters*, v. 29, n. 6, p. 527–529, March 1993. ISSN 0013-5194.
- WON, S. H. et al. Identification of finite state automata with a class of recurrent neural networks. *Neural Networks, IEEE Transactions on*, v. 21, n. 9, p. 1408–1421, Sept 2010. ISSN 1045-9227.
- XIAO, Y.; LONG, F.; ZHAO, Y. Based on elm forged neural control for a class of strict feedback stochastic nonlinear switched system with time varying delay. In: *Control Conference (CCC), 2013 32nd Chinese*. [S.l.: s.n.], 2013. p. 2168–2173.

- XU, Y.; LI, F. Adaptive pi control of statcom for voltage regulation. *Power Delivery, IEEE Transactions on*, v. 29, n. 3, p. 1002–1011, June 2014. ISSN 0885-8977.
- YAMADA, T.; YABUTA, T. Dynamic system identification using neural networks. *Systems, Man and Cybernetics, IEEE Transactions on*, v. 23, n. 1, p. 204–211, Jan 1993. ISSN 0018-9472.
- YANG, Y.; REN, J. Adaptive fuzzy robust tracking controller design via small gain approach and its application. *Fuzzy Systems, IEEE Transactions on*, v. 11, n. 6, p. 783–795, Dec 2003. ISSN 1063-6706.
- YEH, J.-W.; SU, S.-F. Learning analysis for correlation of fuzzy rules in applying rls for neural fuzzy systems. In: *Granular Computing (GrC), 2012 IEEE International Conference on*. [S.l.: s.n.], 2012. p. 609–613.
- YU, C.; ZHANG, C.; XIE, L. A new deterministic identification approach to hammerstein systems. *Signal Processing, IEEE Transactions on*, v. 62, n. 1, p. 131–140, Jan 2014. ISSN 1053-587X.
- ZENG, N. et al. Identification of nonlinear lateral flow immunoassay state-space models via particle filter approach. *Nanotechnology, IEEE Transactions on*, v. 11, n. 2, p. 321–327, March 2012. ISSN 1536-125X.
- ZHANG, Y. et al. An artificial neural network approach to transformer fault diagnosis. *Power Delivery, IEEE Transactions on*, v. 11, n. 4, p. 1836–1841, Oct 1996. ISSN 0885-8977.
- ZHAO, H. et al. Identification of nonlinear dynamic system using a novel recurrent wavelet neural network based on the pipelined architecture. *Industrial Electronics, IEEE Transactions on*, v. 61, n. 8, p. 4171–4182, Aug 2014. ISSN 0278-0046.
- ZHAO, W.-X.; CHEN, H.-F. Recursive identification for hammerstein system with arx subsystem. *Automatic Control, IEEE Transactions on*, v. 51, n. 12, p. 1966–1974, Dec 2006. ISSN 0018-9286.
- ZHOU, D.; ANNAKAGE, U.; RAJAPAKSE, A. Online monitoring of voltage stability margin using an artificial neural network. *Power Systems, IEEE Transactions on*, v. 25, n. 3, p. 1566–1574, Aug 2010. ISSN 0885-8950.