



UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA QUÍMICA

DEPARTAMENTO DE PROCESSOS QUÍMICOS

**REDES NEURAIS APLICADAS EM
ESTRATÉGIAS DE CONTROLE NÃO LINEAR**

Autor: Laércio Ender

Orientador: Prof. Dr. Rubens Maciel Filho

Tese de Doutorado apresentada à Faculdade de Engenharia Química
como parte dos requisitos exigidos para a obtenção do título de Doutor
em Engenharia Química.

Campinas – São Paulo

Março – 2002

UNICAMP
BIBLIOTECA CENTRAL

UNICAMP
BIBLIOTECA CENTRAL
SEÇÃO CIRCULANTE

UNIDADE	80
Nº CHAMADA	UNICAMP
	En21r
V	EX
TOMBO BC/	53427
PROC.	124/03
C	<input type="checkbox"/>
D	<input checked="" type="checkbox"/>
PREÇO	R\$11,00
DATA	20/07/03
Nº CPD	

CM001B3359--4

-Subid. 294040

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

En21r Ender, Laércio
Redes neurais aplicadas em estratégias de controle
não linear / Ender, Laércio. --Campinas, SP: [s.n.], 2002.

Orientador: Rubens Maciel Filho.
Tese (doutorado) - Universidade Estadual de
Campinas, Faculdade de Engenharia Química.

1. Redes neurais (Computação). 2. Controle de
processos químicos. I. Maciel Filho, Rubens. II.
Universidade Estadual de Campinas. Faculdade de
Engenharia Química. III. Título.

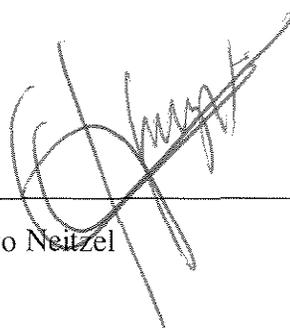
Tese de Doutorado defendida por LAÉRCIO ENDER e aprovada em 21 de março de 2002 pela banca examinadora constituída pelos doutores:



Prof. Dr. Rubens Maciel Filho (Orientador)



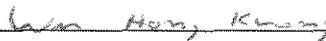
Prof. Dr. Cláudio Odler do Nascimento



Prof. Dr. Ivo Neitzel



Prof. Dr. Wagner Caradori do Amaral



Prof. Dr. Wu Hong Kwong

26/03/02

Este exemplar corresponde à versão final da Tese de Doutorado em Engenharia Química.

A handwritten signature in black ink, reading "Rubens Maciel Filho". The signature is written in a cursive style with a large initial 'R' and 'F'.

Prof. Dr. Rubens Maciel Filho

À minha família, em especial à
minha filhinha *Júlia*, **dedico** este
trabalho.

AGRADECIMENTOS

Devo meus agradecimentos aos:

Ao amigo e professor Doutor Rubens Maciel Filho, pela sua dedicada e valiosa orientação no desenvolvimento deste trabalho.

A professora Doutora Maria Regina Wolf Maciel pelo incentivo, compreensão e amizade.

Aos professores e funcionários do Departamento de Engenharia Química da Universidade Regional de Blumenau pelo apoio.

Aos colegas do LOPCA, em especial a Ramón Scheffer e Luiz Augusto C. Meleiro pelas valiosas discussões e pela disposição em auxiliar.

A Eduardo Coselli Vasco de Toledo e José Alberto Rodrigues pela generosidade em autorizar o uso de suas rotinas computacionais (modelos dinâmicos de processos) para o desenvolvimento deste trabalho.

A minha família, em especial a minha querida esposa *Marli*, pela compreensão e apoio no desenvolvimento deste trabalho.

A CAPES pelo apoio financeiro.

RESUMO

As Redes Neurais Artificiais (RNA) são ferramentas computacionais, com um grande número de aplicações em técnicas de modelagem e controle de processos. Tal fato deve-se à sua capacidade em aprender com suficiente exatidão o comportamento do sistema, gerando modelos genéricos com potencial para projeto de controle não linear, quando equações do modelo são desconhecidas ou somente informações parciais de estados do processo estão disponíveis. Os modelos obtidos através de redes neurais permitem levar em consideração as não linearidades do processo, bem como as interações entre as suas variáveis.

O trabalho desenvolvido explora o uso de redes neurais em estratégias de controle multivariável, enfatizando situações em que estas são utilizadas como modelos dinâmicos na geração de predições, bem como na definição de estratégias de controle totalmente baseadas em redes neurais com aprendizagem *on-line*.

A aprendizagem *on-line* das redes utilizadas permite a sua adaptação continuamente ao longo do tempo, treinando-as em situações não abordadas na etapa de aprendizagem *off-line*. A aprendizagem *off-line* das redes neurais é realizada a partir de um conjunto consistente de dados históricos de perturbações e respostas do processo, a qual deverá garantir, no mínimo, um desempenho satisfatória da rede neural, para a partida do sistema de controle.

Explora-se também, a utilização de modelos estáticos do processo, baseados em redes neurais, acoplados à rotina de otimização *on-line*, objetivando identificar as melhores condições operacionais deste processo para atender especificações de referências preestabelecidas.

Dentro deste contexto, foram desenvolvidas estratégias de controle multivariável explorando as potencialidades das redes neurais, seja como modelo de processos e/ou como controladores, enfatizando a aprendizagem *on-line*.

Para o desenvolvimento deste trabalho foram implementados diversos programas computacionais, em linguagem de programação Fortran 90, relativos aos algoritmos de controle propostos/avaliados

Os resultados alcançados mostram a eficiência das técnicas abordadas, comprovando o potencial do uso das redes neurais em estratégias de controle.

Palavras-chave: redes neurais, controle não linear, aprendizagem *on-line*

ABSTRACT

The artificial neural networks are computational tools with a great number of applications in modeling techniques and process control. Such fact is due its capacity to learn sufficiently accurate models and give good nonlinear control when model equations are not known or only partial state information is available. Neural network approach allows taking into account in an elegant and adequate way process non-linearities as well as variable interactions.

The developed work explores the use of the neural networks in multivariable control strategies as dynamic models for predictions as well as in the definition of the control strategy based on neural networks with on-line learning. The off-line learning of the neural networks is accomplished with a consistent group of historical data of perturbations and responses of the process, which should guarantee at least a satisfactory performance of the neural network for starting of the control system.

It is also explored the use of static models of the process, based on neural networks, coupled with a on-line optimization routine, objectifying to identify the best operational conditions to attend specifications of the process.

In this context, multivariable control strategies were developed exploring the potentialities of the neural networks as process model and/or as controllers, emphasizing the on-line learning.

Several computational programs were implemented for the development of this work in Fortran 90 language program relative to the control algorithms proposed and evaluated.

The obtained results show the efficiency of the approached techniques, checking the potential of the use of the neural networks in control strategies.

Keywords: neural network, non linear control, on-line learning.

SUMÁRIO

RESUMO	vii
ABSTRACT	ix
NOMENCLATURA	xiii
1 INTRODUÇÃO	1
2 REDES NEURAIS	5
2.1 Introdução	5
2.2 Arquitetura de Rede	6
2.2.1 Neurônio Artificial	8
2.3 Treinamento da Rede Neural usando Backpropagation	11
2.4 Aprendizagem on-line da ANN	21
2.5 Redes Neurais aplicadas a Controle de Processos	22
3 REDES NEURAIS APLICADAS A ESTRATÉGIAS DE CONTROLE PREDITIVO	25
3.1 Introdução	25
3.2 Estratégias de identificação de processos utilizando Redes Neurais	26
3.3 O comportamento dinâmico da Rede Neural	29
3.4 Estratégias de controle preditivas	33
3.4.1 O algoritmo Modificado	33
3.4.2 Algoritmo de Controle Preditivo com Restrições	55

	xi
3.5 Conclusões	66
4 REDES NEURAIAS ATUANDO COMO CONTROLADORES	67
4.1 Introdução	67
4.2 A Rede Neural	69
4.3 Estratégia de controle usando Modelo Inverso	70
4.3.1 Fundamentos	70
4.3.2 Resultados	74
4.4 Controle adaptativo baseado em redes neurais	82
4.4.1 Fundamentos	82
4.4.2 Resultados aplicados ao reator catalítico	87
4.4.3 Resultados aplicados ao reator de penicilina	95
4.5 Estratégia baseada na estrutura de controle <i>Feedback</i>	98
4.5.1 Fundamentos	98
4.5.2 Resultados	102
4.6 Conclusões	109
5 ESTRATÉGIAS DE CONTROLE DE CONCENTRAÇÃO USANDO <i>SOFT SENSOR</i>	111
5.1 Introdução	111
5.2 Estratégias de controle da concentração	112
5.2.1 Estratégia de controle baseada em redes neurais	112
5.2.2 Estratégia baseada no algoritmo de controle preditivo com restrições	114
5.2.3 A otimização das referências de temperatura	115
5.3 Resultados	120
5.4 Conclusões	154

	xii
6 CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS	157
6.1 Conclusões	157
6.2 Sugestões para Trabalhos Futuros	159
7 REFERÊNCIAS BIBLIOGRÁFICAS	161
8 ANEXO 1 - MODELO DO REATOR CATALÍTICO DE LEITO FIXO	177
9 ANEXO 2 - MODELO DO REATOR DE PRODUÇÃO DE PENICILINA	185
9.1 Desenvolvimento do Modelo Matemático	186
9.1.1 Fase de Crescimento	186
9.1.2 Fase de Produção	188
9.1.3 Fatores comuns às duas fases	191
10 ANEXO 3 - DMC - MIMO	195
11 ANEXO 4 - ANÁLISE DE SENSIBILIDADE DOS PARÂMETROS DO PROCESSO	199
12 ANEXO 5 - DESCRIÇÃO DOS PROGRAMAS COMPUTACIONAIS DESENVOLVIDOS	203
13 ANEXO 6 - PARÂMETROS UTILIZADOS	207

NOMENCLATURA

1. Redes neurais

Letras Latinas

- d_j : resposta alvo de um neurônio j da camada de saída;
- f : quando referida a um neurônio denota função de ativação ;
- N_p : número de padrões;
- n_k : número de neurônios da camada k ;
- n : número de camadas ocultas;
- $S_i^{(k)}$: função que indica a soma ponderada de todas as entradas de um neurônio i da camada k ;
- $X_i^{(k)}$: Saída (ou ativação) de um neurônio i da camada k ;
- $w_{i,j}^{(k)}$: peso da conexão entre o neurônio j da camada $(k-1)$ e o neurônio i da camada k .

Letras gregas

- ε : função objetivo para treinamento;
- $\delta_i^{(k)}$: erro equivalente na aprendizagem;
- η : taxa de aprendizagem.

2. Estratégias de controle

Letras latinas

- \tilde{A} : matriz dos coeficientes da resposta a um degrau;
- $\hat{e}(k)$: vetor de erro global estimado da malha fechada, para o instante de tempo k ;
- $E(k)$: vetor de erro global da malha fechada, para o instante de tempo k ;
- $e(i, k)$: erro global da malha fechada i , para o instante de tempo k ;

- $h_{ik}(j)$: resposta a um degrau para processos MIMO no tempo j ;
 $J(k)$: função objetiva da estratégia de controle preditivo
 N : horizonte de predição, para a estratégia de controle preditivo;
 N_u : horizonte de controle, para a estratégia de controle preditivo;
 N_m : número de malhas de controle;
 N_c : número de termos de convolução usados na predição;
 $u(i, j)$: ação de controle da malha i no tempo j ;
 \tilde{u} : vetor de incrementos na ação de controle;
 $U(k)$: vetor de ações de controle;
 $y(i, j)$: resposta da malha fechada da malha i no tempo j ;
 $Y(k)$: vetor de respostas da malha fechada;
 $\hat{y}(i, j)$: predição para a resposta da malha fechada i no tempo j ;
 $\hat{Y}(k)$: vetor de predições da malha fechada;
 $y_r(i)$: referência desejada para a malha de controle i ;
 $Y_r(k)$: vetor de referências da malha fechada;
 $y_w(i, j)$: trajetória de referência da malha i no tempo j ;
 $Y_w(k)$: vetor de trajetória de referência da malha fechada.

Letras gregas

- α_i : parâmetro de ajuste da trajetória de referência para a malha i ;
 β^2 : fator de supressão;
 $\Delta u(i, j)$: incremento na ação de controle i no tempo j ;
 δu_j : erro estimado na ação de controle da malha fechada j ;
 δy_j : erro estimado na resposta da malha fechada j ;
 δU : vetor de erros estimados nas ações de controle da malha fechada;
 δY : vetor de erros estimados nas respostas da malha fechada;
 λ : número de informações passadas usadas na representação do processo através de rede neurais;
 $\lambda(i)$: penalização na ação de controle da malha i .

3. Otimização da referência da malha fechada

Letras latinas

$\bar{C}_r(i)$: referência de concentração, na forma de desvio do estado estacionário de referência;

$\bar{C}(i,k)$: concentração da malha i na iteração k , na forma de desvio do estado estacionário de referência;

\bar{C}_{\min} : restrição inferior na concentração, na forma de desvio do estado estacionário de referência;

\bar{C}_{\max} : restrição superior na concentração, na forma de desvio do estado estacionário de referência;

$\bar{T}(i,k)$: temperatura da malha i na iteração k , na forma de desvio do estado estacionário de referência;

\bar{T}_{\min} : restrição inferior na temperatura, na forma de desvio do estado estacionário de referência;

\bar{T}_{\max} : restrição superior na temperatura, na forma de desvio do estado estacionário de referência;

$\bar{T}_{\text{estimado}}$: temperatura estimada, na forma de desvio do estado estacionário de referência.

4. Modelo do reator catalítico de leito fixo

Letras latinas

Bih : número de Biot;

C_{pf} , C_{ps} e C_{PR} : capacidade calorífica para o gás, catalisador e fluido refrigerante (Kcal/KgK);

Def : difusividade radial efetiva (m/h);

f : fator de fricção ;

GM_o : fluxo mássico de alimentação (Kg/m²h);

h_w : coeficiente de filme das paredes internas (Kcal/hm²K);

- L : comprimento do reator (m);
 P : pressão Total do reator;
 P_o : pressão de entrada no reator, adimensional;
 P_{ref} : pressão de referência (atm);
 PM : peso molecular médio (Kg/Kmol);
 r : comprimento radial do reator (m);
 R : relação ar/etanol;
 T : temperatura do reator, adimensional;
 T_{fo} : temperatura de alimentação, adimensional;
 T_W : temperatura da parede do reator, adimensional;
 T_R : temperatura do fluido refrigerante, adimensional;
 T_{RO} : temperatura de entrada do fluido refrigerante;
 T_{ref} : temperatura de referência;
 u_R : velocidade do fluido refrigerante (m/h);
 V_f : velocidade superficial (m/h);
 x : conversão;
 z : comprimento axial do reator (m);

Letras gregas

- ε : porosidade;
 λ_{ef} : condutividade radial efetiva (Kcal/m h K);
 ΔH_R : entalpia molar (Kcal/Kmol);
 ρ_f, ρ_s, ρ_R : massa específica do fluido reagente, catalisador e fluido refrigerante (Kg/m³);
 ρ_B : massa específica aparente do leito (Kgcat/m³);

5. Modelo do Reator para obtenção de penicilina

Letras latinas

- C_{O_2} : concentração relativa de oxigênio dissolvido (%);
 $C_{O_2}^*$: concentração de saturação de oxigênio dissolvido no meio(g/l);

- C_s : concentração de substrato (g/l);
 C_p : concentração do produto (g/l);
 C_x : concentração de células (g/l);
 D_i : diâmetro interno do agitador (m);
 E_m : constante cinética da fração endógena do metabolismo devido a manutenção (g/l);
 E_p : constante cinética da fração endógena do metabolismo devido a produção (g/l);
 F : vazão de alimentação do fermentador (l/h);
 F_m : fração endógena do metabolismo devido a manutenção (adimensional);
 F_p : fração endógena do metabolismo devido a produção (adimensional);
 g : aceleração da gravidade (m/s^2);
 K' : constante da equação de Metzner (adimensional);
 K_h : constante cinética de hidrólise de penicilina a ácido penicilóico (h^{-1});
 K_i : constante cinética de inibição de formação de penicilina pelo substrato (g/l);
 K_{La} : coeficiente volumétrico de transferência de oxigênio (h^{-1});
 K_p : constante cinética de inibição de formação de penicilina pelo produto (g/l);
 K_x : parâmetro cinético (adimensional);
 M_s : concentração de manutenção (g/g.h);
 NRe_m : número de Reynolds;
 N_s : velocidade de agitação (rps);
 NPo : número de potência;
 P : quantidade de produto (g);
 PO : potência requerida para fluido não aerado (W);
 Pg : potência requerida para fluido aerado (W);
 Q : vazão de ar (m^3/s);
 q_{O_2} : demanda volumétrica de oxigênio ($gO_2/h.l$);
 Q_{O_2} : demanda de oxigênio (gO_2/h);
 S : quantidade de substrato (g);

- S_F : concentração de substrato na vazão de alimentação do fermentador (g/l);
 V : volume do reator (l);
 X : quantidade de células (g);
 $Y_{P/S}$: coeficiente de rendimento entre formação de produto e substrato assimilado (g/g)
 $Y_{X/O}$: fator de rendimento entre crescimento celular e demanda de oxigênio (g/g);
 $Y_{X/S}$: fator de rendimento entre crescimento celular e substrato assimilado (g/g).

Legenda gregas

- μ : velocidade específica de crescimento celular (h^{-1});
 σ : velocidade específica de consumo de substrato (h^{-1});
 ϕ : velocidade específica de consumo de oxigênio (h^{-1});
 μ_x : velocidade específica máxima de crescimento celular (h^{-1});
 π : velocidade específica de produção (h^{-1});
 γ : taxa de deformação (s^{-1});
 μ_{ap} : viscosidade aparente do fluido reagente (Pa.s);
 ρ : massa específica do fluido reagente (Kg/m^3);
 ν_{ap} : viscosidade cinemática aparente do fluido reagente (m^2/s);

6. ABREVIATURAS

ANN/RNA: *Artificial Neural Networks* /Redes Neurais Artificiais;

DMC: *Dynamic Matrix Control*;

GPC: *Generalized Predictive Control* ou Controle Preditivo Generalizado;

IAE: integral do erro absoluto;

SQP: *successive quadratic programming* ou programação quadrática sucessiva.

CAPÍTULO 1

INTRODUÇÃO

Durante a última década, a aplicação de redes neurais objetivando a identificação e o controle de processos têm crescido exponencialmente, com um grande número de aplicações em técnicas de modelagem e controle de processos. Tal fato deve-se à sua capacidade em aprender com suficiente exatidão o comportamento do sistema, gerando modelos com potencial para projeto de controle não linear, quando equações do modelo são desconhecidas ou somente informações parciais de estados do processo estão disponíveis. Os modelos obtidos através de redes neurais permitem levar em consideração as não linearidades. As redes neurais são ferramentas computacionais, com implementação possível em “hardware” e/ou “software”, que imitam as habilidades de processamento e aprendizagem do cérebro humano.

Dentro deste contexto, emergiram as estratégias de controle usando otimizações em tempo real, onde diversas ferramentas são empregadas objetivando desenvolver técnicas para o controle de processos considerados com dinâmica complexa. Nesta tese, procura-se desenvolver estratégias de controle usando as redes neurais, abordando diversas formas de

aplicação, seja na forma de modelo dinâmico, como controlador propriamente dito ou como sensor virtual em estratégias de otimização de referências da malha fechada.

Em todos os casos avaliados, utilizou-se aprendizagem *on-line* das redes neurais, de acordo com procedimento proposto, para não gerar problemas de execução em tempo real. A aprendizagem *on-line* das redes permite a sua adaptação continuamente ao longo do tempo, treinando-as em situações não abordadas na etapa de aprendizagem *off-line*. Esta aprendizagem *off-line* das redes neurais é realizada a partir de um conjunto de dados históricos de perturbações e respostas do processo, a qual irá ajustar um conjunto de pesos para a rede neural, a ser usado para a partida do sistema de controle.

Dentro deste contexto, esta tese foi elaborada de tal forma a apresentar aplicações de redes neurais em estratégias de controle; algumas já exploradas na literatura, mas importantes para o entendimento global do tema proposto e propõem-se algumas estratégias de abordar o tema, empregando, em todos os casos, a aprendizagem *on-line* das redes neurais.

O Capítulo 2 apresenta a arquitetura de redes neurais *feedforward*, arquitetura usada nesta tese, enfatizando as principais características e a técnica de aprendizagem *backpropagation*. Utilizou-se esta arquitetura de rede com a aprendizagem *backpropagation*, pelo fato de serem as mais difundidas em aplicações de controle de processos.

O Capítulo 3 explora as redes neurais como modelos dinâmicos do processo com aprendizagem *on-line*, aplicados a estratégias de controle preditivo. Diferentes formas de apresentação dos dados à rede neural são propostas, gerando características distintas no comportamento da malha fechada. São avaliadas duas formas de inserção de modelo dinâmico baseado em redes neurais em estratégias de controle.

No Capítulo 4, diferentes formas de utilização de redes neurais atuando como controladores de processos são avaliadas e algumas estratégias de controle são propostas. Neste capítulo, pode-se ter uma idéia clara do potencial das redes neurais atuando como controladores ou, simultaneamente, atuando como controladores e modelos dinâmicos em estratégias de controle multivariáveis.

Outra importante aplicação das redes neurais é a sua utilização como sensores virtuais para a estimativa de variáveis do processo de difícil mensuração. O Capítulo 5 faz uso deste potencial da rede para otimizar as referências da malha fechada. Este procedimento proposto é executado *on-line*, acoplado à estratégia de controle.

A qualidade dos resultados obtidos ilustram a eficiência das técnicas abordadas, comprovando o potencial de uso das redes neurais na modelagem e controle de processos químicos. As estratégias de controle implementadas e propostas apresentaram excelentes resultados, mesmo atuando em faixas de operação dos processos, com presença de fortes não linearidades e situações onde perturbações enérgicas foram aplicadas ao processo. Estas conclusões estão condensadas no Capítulo 6, onde também são apresentadas as propostas de trabalhos futuros.

O Capítulo 7 apresenta as referências bibliográficas usadas no desenvolvimento deste trabalho

As estratégias de controle foram testadas usando dois processos com dinâmicas distintas. O primeiro é um reator catalítico de leito fixo para obtenção de acetaldeído. Os reatores catalíticos de leito fixo exibem complicados comportamentos estacionário e dinâmico resultantes das não linearidades do sistema e à existência de resposta inversa nas variáveis controladas, gerando, desta forma, um ambiente propício para avaliação de estratégias de controle. O segundo é um reator para obtenção de penicilina. Este processo apresenta uma dinâmica mais lenta, mais não menos complexa. As descrições destes processos encontram-se nos Anexos 1 e 2, desta tese.

Os resultados obtidos com as estratégias de controle preditivo implementadas foram comparados ao algoritmo de controle preditivo DMC auto-ajustável, por ser bastante difundido no meio acadêmico e com implementações em casos reais. O Anexo 3 apresenta as principais equações para a implementação do algoritmo DMC-MIMO.

A definição das variáveis manipuladas e controladas para o reator catalítico de leito fixo foi realizada por análise de sensibilidade usando planejamento fatorial. Este procedimento está descrito no Anexo 4.

O Anexo 5 apresenta uma breve descrição das rotinas computacionais implementadas. Estes programas foram desenvolvidos em linguagem de programação Fortran 90.

Os principais parâmetros para a definição das arquiteturas das redes neurais, bem como parâmetros usados nas estratégias de controle, encontram-se no Anexo 6.

CAPÍTULO 2

REDES NEURAIS

2.1 Introdução

O conceito fundamental contido nos sistemas classificados atualmente como RNA's (Redes Neurais Artificiais) foi introduzido através do famoso trabalho de McCULLOCH e PITTS (1943) que, partindo do princípio de que o cérebro é constituído de unidades básicas, os neurônios, propuseram um modelo matemático para representar essas unidades: o neurônio artificial. No entanto, a classe denominada de RNA's reúne atualmente diversos modelos matemáticos distintos e com domínios de aplicação variados.

Este capítulo dedica-se a apresentar as redes neurais de uma forma clara e objetiva de tal forma a facilitar o entendimento das estratégias de controle desenvolvidas e/ou implementadas nesta tese.

As redes neurais artificiais são ferramentas computacionais que têm sua origem na inteligência artificial. São formadas por elementos de processamento, análogos aos neurônios biológicos, chamados neurônios artificiais.

Os neurônios artificiais são emulações dos neurônios biológicos, os quais recebem informações de sensores ou de outros neurônios artificiais, produzindo operações sobre

estes dados, e passam o resultado para outros neurônios artificiais. As redes neurais processam seus dados usando paralelismo lógico para todos os neurônios da mesma camada, combinado com operações seriais, quando a informação de uma camada é transferida para neurônios de outra camada.

As redes neurais possuem a habilidade de aproximar funções complexas. O conhecimento inerente da rede é dado por pesos nas conexões entre os neurônios artificiais. A computação de uma rede neural é executada, numa primeira etapa, em paralelo por todos os neurônios de uma determinada camada e, posteriormente, em série quando as informações são propagadas de uma camada para outra. Esta coletividade e conectividade de operações resulta em um alto grau de processamento paralelo, que habilita a rede a resolver problemas complexos.

2.2 Arquitetura de Rede

A Figura 2.1 mostra uma típica rede neural formada pelas interconexões entre os neurônios. Estes neurônios são dispostos em camadas altamente interconectadas e elaboram somatórios ponderados de determinados sinais de entrada para gerar uma saída. As camadas da rede podem ser categorizadas como camada de entrada, onde informações disponíveis são apresentadas à rede; camadas intermediárias ou ocultas (no mínimo uma, porém normalmente de uma a três), onde os neurônios interagem entre si; e camada de saída, a qual contém a resposta a uma determinada entrada.

Quanto ao fluxo de propagação das informações, as redes neurais podem ser do tipo *feedback* ou *feedforward* (BAUGHMAN & LIU, 1995). As conexões do tipo *feedback* são usualmente utilizadas em rede neurais recorrentes, a exemplo da rede de Hopfield (NARENDRA, 1990). Já nas redes com conexões do tipo *feedforward*, todos os sinais propagam através das camadas para “frente”. Não há conexões laterais ou como o sinal retroceder. Este tipo de rede possui a capacidade de generalização, ou seja, é capaz de classificar corretamente um padrão complexo mesmo quando este não pertencer ao conjunto de treinamento da rede. As conexões *feedforward* são apropriadas para a

modelagem dinâmica, onde se deseja mapear uma resposta de saída baseada em um sinal de entrada (BAUGHMAN & LIU, 1995). No contexto desta tese foram usadas redes neurais com conexões *feedforward* e *feedback*. Na sua maioria foram utilizadas redes *feedforward*, porém conexões do tipo *feedback* foram utilizadas em previsões ao longo de um horizonte de tempos de amostragem em estratégias de controle.

Os pesos das conexões entre os neurônios são os detentores da maior parte das informações que uma rede neural pode acumular. Isto implica que a fase de ajuste destes pesos tem fundamental importância no sucesso da rede.

A operação de uma rede neural é dividida em 3 fases: a aprendizagem, a representação e a generalização. Na aprendizagem, padrões de entrada e saída são apresentados à rede neural para fins de ajustes em todas as interconexões entre os elementos de processamento (neurônios artificiais). A representação é a fase onde somente padrões de entrada, utilizados na aprendizagem, são apresentados e cabe a rede neural estimar a(s) saída(s) desejada(s). Na fase de generalização, a rede gera a(s) resposta(s) para padrões novos e similares às entrada.

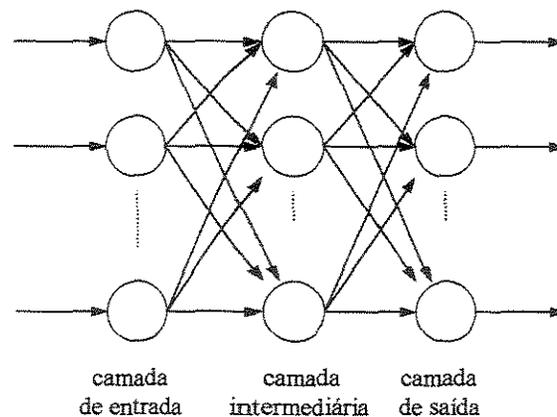


Figura 2.1. - Arquitetura típica de uma rede neural *feedforward*.

2.2.1 Neurônio Artificial

O fundamento das redes neurais artificiais são os neurônios artificiais ou elementos de processamento. São nos neurônios artificiais que os cálculos inerentes são processados. O neurônio artificial recebe sinais de entrada, processa estes sinais e os transmite para as próximas camadas, o qual pode ser excitatório ou inibitório.

Os neurônios da primeira camada têm a única função de distribuir as entradas para as camadas subseqüentes. A partir da segunda camada as informações são processadas e propagadas até a camada de saída, de acordo com as equações apresentadas a seguir.

Considerando que um neurônio “ i ” qualquer da camada “ k ”, recebe um conjunto de informações $X_j^{(k-1)}$ ($j=1, \dots, n_{k-1}$), correspondentes as saídas dos n_{k-1} neurônios da camada anterior, ponderadas, cada uma, pelo correspondente peso da conexão $w_{i,j}^{(k)}$. O neurônio soma essas entradas ponderadas e o valor resultante é por sua vez somado a um limite interno de ativação, um “bias” que pode ser descrito por $w_{i,0}^{(k)}$. A este sinal resultante, o neurônio “ i ”, produz uma resposta $X_i^{(k)}$, de acordo com uma função limite ou função de ativação $f(\bullet)$ (DE SOUZA JR., 1993), conforme mostrado na Figura 2.2. Matematicamente pode-se expressar por:

$$S_i^{(k)} = w_{i,0}^{(k)} + \sum_{j=1}^{(n_{k-1})} w_{i,j}^{(k)} X_j^{(k-1)} \quad (2.1)$$

$$X_i^{(k)} = f(S_i^{(k)}) \quad (2.2)$$

Segundo DE SOUZA JR. (1993), dois tipos de função de ativação são comumente usados para a ativação dos neurônios: global e local.

A função de ativação global produz uma saída ativa para um largo intervalo de entradas (BAKSHI & STEPHANOPOULOS, 1993). São exemplos de funções globais:

- linear: $f(S_i^{(k)}) = S_i^{(k)}$ (2.3)

- degrau limiar: $f(S_i^{(k)}) = \begin{cases} +\xi & \text{se } S_i^{(k)} > 0 \\ -\varsigma & \text{se } S_i^{(k)} \leq 0 \end{cases}$ (2.4)

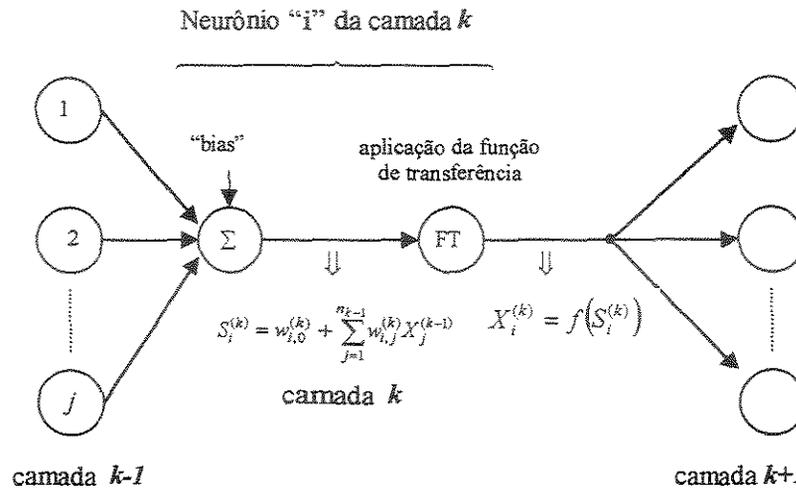


Fig. 2.2 - Anatomia do i-ésimo neurônio artificial da k-ésima camada

Freqüentemente $f(S_i^{(k)})$ é de natureza binária, emitindo 1 se $S_i^{(k)} > 0$ e 0, para os demais casos (Simpson, 1990);

- função sigmoideal: $f(S_i^{(k)}) = \frac{1}{1 + e^{-S_i^{(k)}}}$ (2.5)

- função tangente hiperbólica: $f(S_i^{(k)}) = \tanh(S_i^{(k)})$ (2.6)

As funções locais são aquelas que só apresentam uma resposta ativa para valores de entrada na vizinhança imediata do seu centro de ativação (BAKSHI &

STEPHANOPOULOS, 1993). As funções de base radial (RBF) têm esta natureza. Um exemplo é a função gaussiana:

$$\text{- função gaussiana: } f_i = \prod_{j=1}^{n_{k-1}} \exp \left[\frac{(x_j^{(k-1)} - \mu_{i,j})^2}{-2\sigma_i^2} \right] \quad (2.7)$$

com σ_i indicando a largura (ou desvio-padrão) da gaussiana e $\mu_{i,j}$, os centros adaptativos da mesma, os quais podem ser fixado em μ_i .

No desenvolvimento desta tese utilizou-se as funções de ativação sigmoidal e tangente hiperbólica, cujo comportamento está mostrado nas figuras (2.3.a) e (2.3.b)

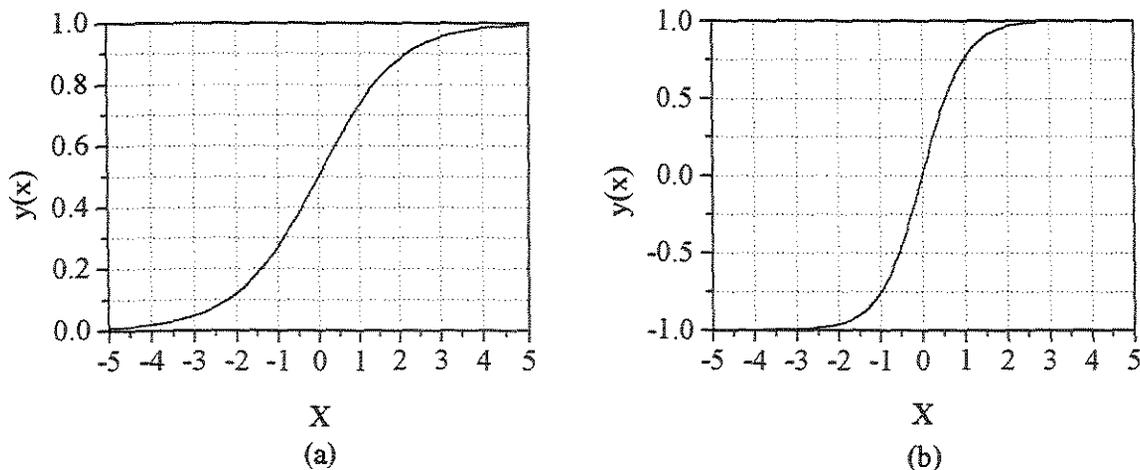


Figura 2.3 – Funções de ativação: (a) Função Sigmoidal; (b) Função Tangente Hiperbólica

A diferença básica entre a função sigmoidal e a função tangente hiperbólica, é que a primeira é “positiva” (saída total variando de 0 a 1) e a segunda é “média-zero” (saída total variando de -1 a 1) (HUNT *et al.*, 1992). São, no entanto, equivalentes por atuarem

em um intervalo dinâmico limitado, o que é realista para todos os sistemas físicos estáveis (RUMELHART & McCLELLAND, 1986).

2.3 Treinamento da Rede Neural usando *Backpropagation*

A aprendizagem ou treinamento é o processo onde os pesos e *biases* da rede são modificados (SIMPSON, 1990). Os métodos de treinamento mais empregados são os chamados supervisionados, onde a rede aprende por experiência, ou seja, exemplos de entradas e de saídas esperadas são apresentados à rede, que ajusta os seus pesos. Um modelo interno do processo que rege os dados fornecidos à rede é criado pela mesma, de modo que ela pode fazer predições para novas entradas e reconhecer padrões, mesmo com ruídos (RUMELHART & McCLELLAND, 1986; GIARRATANO et al., 1990; QUANTRILLE & LIU, 1991; DE SOUZA Jr, 1993).

As redes treinadas pelo método *backpropagation* são multicamadas, não apresentam conexões laterais, entre neurônios da mesma camada, e os seus sinais são *feedforward*; ou seja, dado um vetor de entrada, o vetor de saída é calculado em um passo do tipo *forward* (para frente) que calcula os níveis de atividade de cada camada por vez, usando os níveis de atividade já computados nas camadas anteriores (RUMELHART & McCLELLAND, 1986).

As funções de ativação a serem utilizadas no método *backpropagation* devem ser diferenciáveis. Dentre as funções de ativação anteriormente expostas a função degrau limiar não poderá ser utilizada neste método de aprendizagem, por ser uma função descontínua.

Funções de ativação lineares não são utilizadas, primeiramente porque as redes neurais são basicamente uma ferramenta para modelagem de comportamentos fora dos domínios dos modelos lineares (DE SOUZA JR., 1993). Ademais, o uso de camadas internas de neurônios lineares não aumenta a poder computacional das redes multicamadas comparativamente a redes sem camadas internas (RUMELHART & McCLELLAND, 1986).

Um aspecto a ser considerado é a escolha do número de camadas da rede neural artificial. Segundo HECHT-NIELSEN (1989), uma função contínua pode ser aproximada para qualquer grau de precisão usando uma rede neural *backpropagation* com três camadas, desde que haja um número suficiente de neurônios ativos na camada oculta. Uma outra discussão fundamental para o sucesso de uma rede neural diz respeito ao número de neurônios na camada oculta.

CYBENKO (1989) mostrou que qualquer representação de um conjunto de dados de \mathfrak{R}^p para \mathfrak{R}^q pode ser obtido com duas camadas internas. HORNIK et al. (1990) também mostraram que qualquer mapeamento pode ser obtido, para um grau de precisão, usando uma rede neural com uma camada interna com um número suficiente de neurônios. Entretanto, para incrementar a precisão dos resultados, normalmente aumenta-se o número de camadas internas, aumentando assim a complexidade da rede neural. O principal problema decorrente deste procedimento é que a rede neural com grande número de pesos tem uma forte tendência em representar bem os padrões da aprendizagem e terem uma performance não apropriada com dados não usados na aprendizagem. Além disto o procedimento de estimação dos parâmetros torna-se difícil, devido a quantidade de mínimos locais existentes (HENRIQUE et al., 2000).

A habilidade de redes multicamada em aproximar funções contínuas e limitadas arbitrárias tem sido objeto de análise por vários autores. Segundo HECHT-NIELSEN (1990), o matemático russo KOLMOGOROV (1957), publicou um teorema relativo a representação de funções contínuas que pode ser diretamente adaptado às redes neurais multicamadas, reformulando este teorema da seguinte maneira (KOVÁCS, 1996):

Teorema – Kolmogorov-Nielsen: Dada uma função contínua arbitrária $f : [0,1]^n \rightarrow \mathfrak{R}^m$, $f(x) = y$, existe sempre para f , uma implementação exata com uma rede neural de três camadas, sendo a camada de entrada um vetor de dimensão n , a camada oculta composta por $(2n+1)$ neurônios, e a camada de saída com m neurônios representando as m componentes do vetor y .

De acordo com LOESCH (1996), vários autores sugerem que em redes de 3 camadas com mais entradas que saídas o número ideal de elementos na camada oculta seja algo em torno da média geométrica entre as camadas de entrada e de saída. Se a camada oculta possuir poucos neurônios, a rede neural será incapaz de criar limites de decisão muito complexos, enquanto que um número excessivo de neurônios na camada oculta acarretará em perda da capacidade de generalização da rede neural.

O método de treinamento empregado, regra delta generalizada, por ser de base matemática complexa, apresenta a desvantagem de ser demorado, devido ao conjunto de dados ser reapresentado várias vezes, para que os pesos entre os neurônios estabeleçam-se de modo a classificar corretamente os padrões de entrada.

O objetivo de se treinar uma rede é ajustar seus pesos tal que a aplicação de um vetor de entrada, produza um vetor desejado de saída. O treinamento assume que cada vetor de entrada esteja emparelhado com um vetor de saída, chamados também de par de treinamento. Usualmente, a rede é treinada sobre um conjunto de pares de treinamento.

O algoritmo de aprendizagem possui dois momentos perfeitamente distintos: em primeiro lugar, quando um padrão de entrada é apresentado à rede, o fluxo é alimentado para a frente, isto é, propagado adiante, até a camada de saída. Após, a saída obtida é comparada com a saída desejada e, se a saída desejada não corresponder à obtida dentro de uma determinada precisão desejada, é feita uma correção nos pesos das conexões sinápticas, ajustando-se os pesos na direção oposta do gradiente do erro instantâneo. O ajuste é proporcional ao gradiente, segundo um fator de proporcionalidade denominado de taxa de aprendizagem. O ajuste dos pesos é realizado da última camada em direção à camada de entrada.

A regra delta generalizada usa em essência a regra da cadeia do cálculo diferencial. Seu objetivo é reduzir o erro quadrático da rede, e para tanto emprega o método do passo na direção oposta do gradiente, sobre a superfície de erros, onde deseja-se minimizar um erro que é função de todos os pesos da rede (GOZDZIEJEWSKI Jr. & ENDER, 1998). O erro quadrático calculado na rede é dado por:

$$\varepsilon = \sum_{l=1}^{n_k} (d_l - X_l^{(k)})^2 \quad (2.8)$$

sendo que n_k o número de neurônios da camada de saída, $X_l^{(k)}$ a l -ésima saída da rede e d_l a saída desejada.

Considerando um neurônio “ i ” qualquer da camada de saída “ k ”, em uma determinada iteração. O gradiente do erro para este elemento de processamento é:

$$\vec{\nabla}_{i,j}^{(k)} \varepsilon = \frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} \quad (2.9)$$

onde $\vec{\nabla}_{i,j}^{(k)} \varepsilon$ denota o gradiente do erro quadrático do i -ésimo neurônio da k -ésima camada gerado pela j -ésima conexão sináptica.

Sabe-se que:

$$X_i^{(k)} = f(S_i^{(k)}) \quad (2.10)$$

e

$$S_i^{(k)} = \sum_{j=0}^{n_{k-1}} w_{i,j}^{(k)} X_j^{(k-1)} \quad (2.11)$$

sendo que para $j = 0$, $X_j^{(k-1)} = +1$ que representa a conexão extra (*bias*) e n_{k-1} é o número de neurônios da camada $k - 1$. Então:

$$\varepsilon = \sum_{l=1}^{n_k} \left(d_l - f(S_l^{(k)}) \right)^2 \quad (2.12)$$

portanto:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = \frac{\partial \left(\sum_{l=1}^{n_k} \left(d_l - f(S_l^{(k)}) \right)^2 \right)}{\partial w_{i,j}^{(k)}} \quad (2.13)$$

Sabe-se que quando $i \neq l$ as derivadas serão nulas, portanto:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = -2 \left(d_i - f(S_i^{(k)}) \right) \frac{\partial f(S_i^{(k)})}{\partial w_{i,j}^{(k)}} \quad (2.14)$$

Aplicando a regra da cadeia, chega-se a:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = -2 \left(d_i - f(S_i^{(k)}) \right) \frac{\partial f(S_i^{(k)})}{\partial S_i^{(k)}} \frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}} \quad (2.15)$$

Pela equação (2.11) pode-se determinar:

$$\frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}} = \frac{\partial \left(\sum_{l=0}^{n_{k-1}} w_{i,l}^{(k)} X_l^{(k-1)} \right)}{\partial w_{i,j}^{(k)}} \quad (2.16)$$

A derivada em questão na equação anterior é nula quando $l \neq j$, desta forma:

$$\frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}} = X_j^{(k-1)} \quad (2.17)$$

A equação (2.15) pode ser reescrita:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = -2(d_i - X_i^{(k)})f'(S_i^{(k)})X_j^{(k-1)} \quad (2.18)$$

Definindo:

$$\delta_i^{(k)} = (d_i - X_i^{(k)})f'(S_i^{(k)}) \quad (2.19)$$

onde $\delta_i^{(k)}$ pode ser interpretado como *erro equivalente*, o qual será igual ao erro $d_i - X_i^{(k)}$ se $f(S_i^{(k)})$ for a função identidade (NGUYEN & WIDROW, 1990; CHEN, 1990).

Substituindo tem-se:

$$\vec{\nabla}_{i,j}^{(k)} \varepsilon = \frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = -2\delta_i^{(k)}X_j^{(k-1)} \quad (2.20)$$

Desta forma, os pesos podem ser reajustados na direção oposta ao gradiente do erro, da seguinte forma:

$$w_{i,j}^{(k) \text{ new}} = w_{i,j}^{(k) \text{ old}} + \eta \left(-\vec{\nabla}_{i,j}^{(k)} \varepsilon \right) \quad (2.21)$$

ou substituindo a equação (2.20) na equação (2.21), tem-se:

$$w_{i,j}^{(k) \text{ new}} = w_{i,j}^{(k) \text{ old}} + 2\eta \delta_i^{(k)} X_j^{(k-1)} \quad (2.22)$$

onde $0 < \eta \ll 1$ é o termo de taxa de aprendizagem.

A forma de determinação de $\delta_i^{(k)}$, apresentada na equação (2.19), é válida para o ajuste dos pesos dos neurônios da camada de saída. Para ajustar os pesos das camadas ocultas, o erro ε deverá ser retropropagado (daí o nome *backpropagation*) até a camada oculta em análise, conforme exposto a seguir.

Considerando um neurônio “ i ” de uma camada oculta “ k ”, em uma rede neural de 3 camadas, ou seja, uma camada de entrada ($k-1$), uma camada oculta (k) e uma camada de saída ($k+1$). Pela regra da cadeia, pode-se escrever:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = \frac{\partial \varepsilon}{\partial S_i^{(k)}} \frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}} \quad (2.23)$$

Analisando primeiramente o termo $\frac{\partial \varepsilon}{\partial S_i^{(k)}}$,

$$\frac{\partial \varepsilon}{\partial S_i^{(k)}} = \frac{\partial \left(\sum_{l=1}^{n_{k+1}} (d_l - X_l^{(k+1)})^2 \right)}{\partial S_i^{(k)}} \quad (2.24)$$

$$\frac{\partial \varepsilon}{\partial S_i^{(k)}} = -2 \sum_{l=1}^{n_{k+1}} (d_l - f(S_i^{(k+1)})) \frac{\partial f(S_i^{(k+1)})}{\partial S_i^{(k+1)}} \frac{\partial S_i^{(k+1)}}{\partial S_i^{(k)}} \quad (2.25)$$

Porém:

$$\frac{\partial S_i^{(k+1)}}{\partial S_i^{(k)}} = \frac{\partial \left(\sum_{j=0}^{n_k} w_{l,j}^{(k+1)} f(S_j^{(k)}) \right)}{\partial S_i^{(k)}} \quad (2.26)$$

Sabe-se que as derivadas serão nulas para $j \neq i$, portanto:

$$\frac{\partial S_i^{(k+1)}}{\partial S_i^{(k)}} = w_{l,i}^{(k+1)} f'(S_i^{(k)}) \quad (2.27)$$

então:

$$\frac{\partial \varepsilon}{\partial S_i^{(k)}} = -2 \sum_{l=1}^{n_{k+1}} (d_l - X_l^{(k+1)}) f'(S_l^{(k+1)}) w_{l,i}^{(k+1)} f'(S_i^{(k)}) \quad (2.28)$$

Pela equação (2.19), pode-se afirmar:

$$\delta_i^{(k+1)} = (d_l - X_l^{(k+1)}) f'(S_l^{(k+1)}) \quad (2.29)$$

Portanto:

$$\frac{\partial \varepsilon}{\partial S_i^{(k)}} = -2f'(S_i^{(k)}) \sum_{l=1}^{n_{k+1}} \delta_l^{(k+1)} w_{l,i}^{(k+1)} \quad (2.30)$$

Analisando agora o termo $\frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}}$.

$$\frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}} = \frac{\partial \left(\sum_{m=1}^{n_{k-1}} w_{i,m}^{(k)} X_m^{(k-1)} \right)}{\partial w_{i,j}^{(k)}} \quad (2.31)$$

As derivadas para $m \neq j$ são nulas, então:

$$\frac{\partial S_i^{(k)}}{\partial w_{i,j}^{(k)}} = X_j^{(k-1)} \quad (2.32)$$

Portanto a derivada do erro em relação aos pesos da camada oculta pode ser determinada da seguinte forma:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = -2f'(S_i^{(k)}) \sum_{l=1}^{n_{k+1}} (\delta_l^{(k+1)} w_{l,i}^{(k+1)}) X_j^{(k-1)} \quad (2.33)$$

Então:

$$\frac{\partial \varepsilon}{\partial w_{i,j}^{(k)}} = -2\delta_i^{(k)} X_j^{(k-1)} \quad (2.34)$$

onde tem-se para a camada intermediária:

$$\delta_i^{(k)} = f'_i(S_i^{(k)}) \sum_{l=1}^{n_{k+1}} (\delta_l^{(k+1)} w_{l,i}^{(k+1)}) \quad (2.35)$$

Esta forma para o cálculo de $\delta_i^{(k)}$ é usada para retropropagar o erro para as demais camadas ocultas, caso a rede neural apresentar $n > 1$ camadas ocultas. A equação para o ajuste dos pesos fica da mesma forma que apresentada na equação (2.22), ou seja:

$$w_{i,j}^{(k) \text{ new}} = w_{i,j}^{(k) \text{ old}} + 2\eta \delta_i^{(k)} X_j^{(k-1)} \quad (2.36)$$

Os passos de um treinamento via *backpropagation* é composto por duas fases: uma fase em que a entrada é apresentada e propagada para a frente através da rede, até computar os sinais de saída que são comparados com as saídas alvos, obtendo-se um erro; e uma fase em que o erro é propagado para trás e os pesos vão sendo atualizados.

A inicialização dos pesos das interconexões e os *biases* é realizada com números aleatórios pequenos; assim evitam-se problemas, como simetria (que haveria caso se começasse com pesos iguais) e saturação (produzidos por pesos grandes). No desenvolvimento desta tese adotaram-se pesos iniciais aleatórios no intervalo entre -1 e 1 (RUMELHART & McCLELLAND, 1986).

Para cada padrão p ($p=1, 2, \dots, N_p$; onde N_p é o número de padrões), tem-se um conjunto de entradas e um conjunto de saídas desejadas, d_i , que são usadas no critério de erro. Na implementação da aprendizagem *backpropagation*, pode-se atualizar os pesos após a apresentação de cada padrão (aprendizagem por padrão) ou após a apresentação de todos os padrões (aprendizagem batelada) (LEONARD & KRAMER, 1990; McCLELLAND & RUMELHART, 1988). A forma de implementação adotada foi a aprendizagem por padrão.

Depois do treinamento, pode-se usar a rede, onde tem-se apenas a fase em que as entradas passam através da rede para frente (*feedforward*), correspondendo às equações (2.1) e (2.2).

As redes neurais usadas, tanto como modelos de processos ou como controladores, nos trabalhos desenvolvidos foram submetidas a aprendizagem *off-line* num primeiro momento e aprendizagem *on-line* posteriormente. Entende-se por aprendizagem *off-line*, a aprendizagem da rede neural realizada com um conjunto de dados padrões obtidos do processo, os quais são apresentados à rede n vezes até a representação apropriada destes padrões; e por aprendizagem *on-line*, aquela realizada com dados obtidos do processo em tempo real e alimentados ao conjunto de padrões de aprendizagem. Esta aprendizagem permite a rede neural “aprender” o comportamento do processo num momento de não-linearidades fortes ou mapear uma região ainda não apresentada a rede neural, durante a sua utilização.

2.4 Aprendizagem *on-line* da ANN

A aprendizagem *on-line* apresenta limitações devido ao número de iterações necessárias para atender o critério de tolerância adotado e à necessidade de aprendizagem em tempo real. Para contornar estas dificuldades, limitou-se o número máximo de iterações, usando um vetor de padrões formado por um número preestabelecido das últimas informações passadas de perturbações/respostas do sistema em questão. Além disto, para garantir a boa representação dinâmica do processo através de redes neurais, adotou-se uma estratégia formada por três redes neurais que atuam paralelamente (ENDER & MACIEL FILHO, 2001^{a,b}). A primeira, formada por pesos da aprendizagem *off-line*, denominados de pesos padrões; a segunda, inicializada com os pesos padrões e submetida a aprendizagem a cada tempo de amostragem. Sempre que os pesos padrões apresentarem melhor desempenho, esta rede neural tem seus pesos substituídos pelos pesos padrões. A partir deste momento, esta começa a ser treinada novamente. A terceira é inicializada com os pesos padrões e continuamente é submetida a aprendizagem a cada tempo de amostragem. A rede neural que apresentar o menor erro quadrático na representação do vetor que contém as últimas perturbações/respostas do processo é utilizada naquele instante de amostragem.

2.5 Redes Neurais aplicadas a Controle de Processos

A aplicação de redes neurais em controle de processos tem-se mostrado a mais difundida na área da engenharia química (HOSKIN & HIMMELBLAU, 1988; HUNT et al., 1992; RAMIREZ-BELTRAN & JACKSON, 1999), pois os modelos neurais permitem levar em consideração de uma maneira adequada, não linearidades do processo, bem como as variáveis de interação (PSICHOGIOS & UNGAR, 1991).

Vários autores (BARTO (1990), WERBOS (1990), HUNT *et al.* (1992), e UNGAR *et al.* (1995), entre outros) realizaram classificações diferentes das redes neurais como aplicações em controle de processos. PSICHOGIOS & UNGAR (1991) classificaram a aplicação das redes neurais como *método direto* e *método indireto*. No método direto, a rede neural é treinada com dados de entrada/saída do sistema para representar a dinâmica inversa deste processo. Em outras palavras, dado o estado corrente do sistema dinâmico e o estado desejado (*setpoint*) para o próximo tempo de amostragem, a rede neural é treinada a produzir a ação de controle que leva o sistema para o estado desejado. O modelo inverso resultante poderá ser usado como controlador em uma estrutura *feedforward* típica. No método indireto a rede neural é treinada com dados do sistema dinâmico para representar dinâmicas futuras; conhecido o estado corrente e a ação de controle atual, a rede aprende a prever o novo estado do sistema.

BAUGHMAN & LIU (1995) classificaram as redes em três categorias, conforme descrito a seguir:

1. *Controle direto*: A rede neural é treinada a atuar como controlador e as ações de controle são determinadas diretamente;
2. *Controle inverso*: A rede neural é treinada como um modelo inverso do processo, predizendo a ação de controle necessária para produzir a resposta desejada do processo;
3. *Controle indireto*: A rede neural é treinada para atuar como modelo do processo, ou para determinar o *setpoint* para um controlador local, assim afetando indiretamente o processo controlado.

No desenvolvimento desta tese, adotou-se somente duas classificações: *controle direto* e *controle indireto*. Considerou-se controle direto sempre que a rede neural atua como controlador, seja como modelo representando a dinâmica inversa ou representando a estrutura do controle *feedback*. As redes neurais usadas em modelos de predição de estados ou dinâmicas futuras foram aqui denominadas de controle indireto.

Nos capítulos posteriores, serão apresentados os trabalhos desenvolvidos envolvendo redes neurais em estratégias de controle, seja como controlador, modelos dinâmicos preditivos e modelos estáticos nas estratégias envolvendo otimização das referências (*setpoints*) da malha fechada.

CAPÍTULO 3

REDES NEURAS APLICADAS EM ESTRATÉGIAS DE CONTROLE PREDITIVO

3.1 Introdução

As estratégias de controle preditivo caracterizam-se pelas previsões ao longo de um horizonte de tempos de amostragem. Estas previsões normalmente são realizados por modelos paramétricos, a exemplo do GPC (*Generalized Predictive Control*) (CLARKE et al., 1987), e por modelos de convolução, a exemplo do algoritmo DMC (*Dynamic Matrix Control*) (CUTLER & RAMAKER, 1979).

As habilidades das redes neurais em representar o comportamento dinâmico de processos vêm sendo utilizadas em uma gama de problemas aplicados à engenharia (BHAT & McAVOY, 1990), onde pode-se destacar a geração de modelos dinâmicos preditivos (PSICHOGIOS & UNGAR, 1991). As redes neurais apresentam a vantagem de serem inerentemente paralelas e como resultado podem ser resolvidas de forma mais rápida que um processamento digital computacional serial (BHAT et al., 1990). Dentro deste contexto, neste capítulo, será abordado o uso de redes neurais em estratégias de controle preditivo, enfatizando duas formas distintas. A primeira propõe maneiras alternativas de representar o comportamento dinâmico do processo através de redes neurais, modelos tipo NARMAX,

em substituição aos modelos clássicos lineares das estratégias de controle preditivo (ENDER & MACIEL FILHO, 1998, 1999^{a,b}, 2000^a, 2001^c). Esta rede neural é utilizada na geração dos coeficientes da resposta a um degrau, bem como das previsões ao longo de um horizonte de tempos de amostragem. A segunda forma aborda o uso de redes neurais em estratégias de controle preditivo com restrições. Neste caso, as ações de controle são obtidas pela minimização de uma função objetivo que leva em consideração as características desejáveis da estratégia de controle. Nos dois casos analisados, considera-se aprendizagem *on-line* das redes neurais, conforme descrito em capítulo anterior (ENDER & MACIEL FILHO, 2001^{a,b}).

Estas estratégias foram testadas no controle de um reator catalítico de leito fixo para obtenção de acetaldeído, descrito no Anexo 1 (TOLEDO, 1999). Estes reatores exibem complicados comportamentos estacionário e dinâmico, resultantes das não linearidades do sistema e à existência de resposta inversa nas variáveis controladas, gerando desta forma, um ambiente propício para avaliação de estratégias de controle.

3.2 Estratégias de identificação de processos utilizando Redes Neurais

Nos anos recentes, a habilidade das redes neurais em modelar uma função não linear para um grau de precisão tem atraído grande atenção da comunidade de controle de processos. BHAT & McAVOY (1990) estavam entre os primeiros a usar as ANN's para modelar processos químicos não lineares. Ainda outros, entre eles CYBENKO (1989), NARENDRA & PARTHASARATHY (1990), YDSTIE (1990), LEE & PARK (1992), CHEN & BILLINGS (1992), HUNT et al. (1992), DE SOUZA JR. (1993), SCOTT & RAY (1993), CHENG et al. (1995), propuseram o uso de redes neurais em identificação e/ou estratégias de controle não linear de processos. Recentes trabalhos, entre eles KRISHNAPURA & JUTAN (1997), SHAW et al. (1997), ATALLA & INMAN (1998), RUSNÁK et al. (1999), ENDER & MACIEL FILHO (2001^{a,b}), reforçam a aplicação de redes neurais na identificação de processos, gerando modelos a serem usados no projeto de controladores não lineares.

A fase de aprendizagem da rede neural caracteriza-se em otimizar os pesos, $w_{i,j}^{(k)}$, das conexões entre os neurônios das n-camadas. Para isto, um conjunto de dados de perturbações/respostas (denominados pares ou padrões), representando o comportamento dinâmico do processo, é apresentado inúmeras vezes a rede neural até que esta consiga representá-los. Duas arquiteturas diferentes para as redes neurais podem ser adotadas para serem utilizadas em estratégias de controle preditivo. O primeiro tipo fornece diretamente como saída as N (horizonte de predição) predições no futuro. O segundo tipo produz uma predição um passo adiante, que é realimentada à rede como uma entrada para o cálculo da próxima predição; os outros dados de entrada da rede são deslocados para trás, em uma unidade, e isto é feito iterativamente até que N predições tenham sido obtidas. Segundo DE SOUZA JR (1993), as redes do primeiro tipo apresentam várias desvantagens, como descritas abaixo:

- a) Quando da identificação do mesmo processo, a arquitetura do primeiro tipo produz redes maiores que a do segundo;
- b) As redes do primeiro tipo são menos flexíveis (PSICHOGIOS & UNGAR, 1991). Se N, por exemplo, fosse aumentado, uma abordagem do primeiro tipo exigiria um novo treinamento da rede; já para o segundo tipo, a rede já treinada para um passo adiante, teria apenas que, quando do seu uso, sofrer um maior número de iterações para produzir um número correspondentemente maior de predições no futuro;
- c) A predição de apenas um passo no futuro envolve um erro menor que a de N passos diretamente (PSICHOGIOS & UNGAR, 1991; SAINT-DONAT et al., 1991). Tal fato foi observado experimentalmente por LAPEDES & FARBER (1987).

Diante das razões expostas, as redes neurais foram treinadas para gerar respostas em um tempo futuro, de acordo com o segundo tipo. Adotou-se estratégias diferenciadas na forma de apresentar os parâmetros de perturbação/resposta do sistema à rede neural, de acordo com os casos citados a seguir:

Caso 1: Os pares de treinamento apresentados à rede neural são compostos por λ informações de perturbações/respostas passadas e a perturbação atual, conforme mostrado

na Figura 3.1. Desta forma, esta rede quando utilizada na predição, apresenta fluxo de informação do tipo *feedback*, pois as informações geradas no tempo (t) serão utilizadas nos tempos futuros. Neste caso, os pesos da rede neural são ajustados *on-line*, ao mesmo tempo em que é utilizada na estratégia de controle. Caso isto não seja realizado, os controladores e/ou os modelos de predição baseados nesta rede neural poderão gerar *off-set* na estratégia de controle, quando a rede não estiver devidamente treinada na região avaliada. Esta forma já foi testada por vários autores, dentre estes, NARENDRA & PARTHASARATHY (1990), RUSNAK et al. (1990), DE SOUZA JR. (1993).

Caso 2: Os pares de treinamento apresentam forma idêntica ao exposto no *Caso 1*, porém as informações estão na forma de desvios em relação ao instante anterior (forma incremental). Neste caso avaliou-se a possibilidade de aprendizagem *on-line* da rede neural.

Caso 3: Os pares de treinamento apresentados à rede neural são compostos além da perturbação atual, mais λ perturbações passadas. O vetor de entrada na rede neural é composto somente por ações de controle na forma de incrementos e não por saídas do processo. Desta forma, quando esta rede neural for utilizada nas predições, as saídas da rede neural não serão realimentadas, ou seja, não haverá fluxo de informação *feedback* na rede. Este caso possibilita trabalhar-se com redes neurais menores (menor número de neurônios por camada)

O Caso 1 é a forma clássica de apresentação dos padrões para a rede neural. As formas apresentadas nos casos 2 e 3 buscam tratar os padrões na forma incremental, gerando desta forma uma rede neural apresentando ação integral, eliminando problemas de *off-set*. Porém limitações devem ser respeitadas nos casos envolvendo resposta inversa. A diferença básica entre os casos estudados está na forma de apresentação das informações à rede neural e na possibilidade de treinamento *on-line* da rede.

Em todas as configurações de rede neural usadas nas predições dinâmicas, considerou-se apenas uma camada intermediária, por razões já discutidas no Capítulo 2.

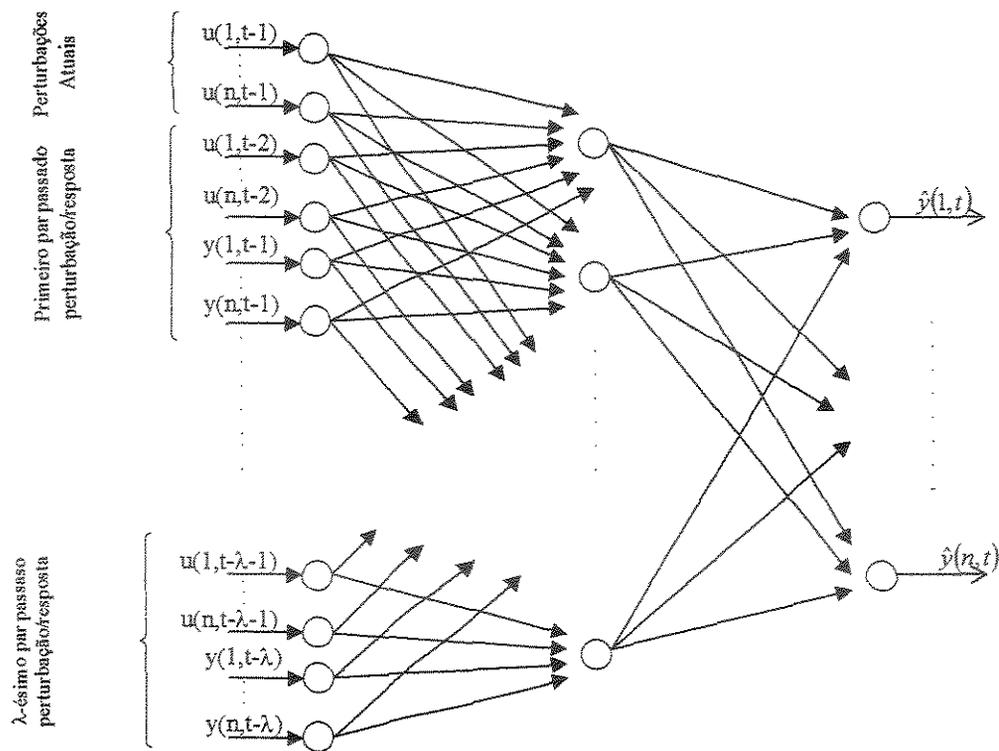


Figura 3.1 – Arquitetura *feedforward* multivariável utilizada – Caso 1

3.3 O comportamento dinâmico da Rede Neural

As redes neurais devidamente treinadas foram utilizadas para realizar as previsões em malha aberta, bem como a geração da matriz dinâmica da estratégia de controle preditivo. A literatura (NARENDRA & PARTHASARATHY, 1990; SU et al., 1992; DE SOUZA JR., 1993), classifica os métodos de identificação em método *série-paralelo* e método *paralelo*. É considerado método *série-paralelo* o método de identificação do modelo dinâmico onde a saída do processo é alimentada na rede para formar os padrões que representam o comportamento dinâmico passado. Já o método *paralelo* é aquele em que as respostas previstas pela rede neural são realimentadas na mesma, a exemplo das previsões ao longo de um horizonte de tempos de amostragem. As Figuras 3.2 e 3.3, apresentam os

resultados do comportamento dinâmico do reator catalítico de leito fixo, descrito no Anexo 1, através de redes neurais, considerando-se os 3 casos anteriormente expostos. Nestas figuras, as redes foram empregadas de tal forma a predizer 1 tempo futuro, ou seja, as informações passadas usadas como entradas da rede são dados reais obtidos por simulação, ou seja, método *série-paralelo*. A Figura 3.2 apresenta os resultados na representação de padrões usados na aprendizagem da rede neural, onde observa-se sobreposição das respostas. Estes dados foram obtidos via simulação, impondo perturbações do tipo degrau na vazão mássica de alimentação do reator, GM_o , e como variável resposta usou-se a temperatura média do terceiro ponto de colocação ortogonal axial do reator. A Figura 3.3 apresenta os resultados da generalização, ou seja, a representação de padrões não apresentados na fase de aprendizagem. Neste caso usou-se uma perturbação senoidal em GM_o .

A Figura 3.4 apresenta os resultados para a generalização de uma rede neural considerando como variável perturbação a temperatura do fluido refrigerante na alimentação, T_{RO} , e como resposta a temperatura do terceiro ponto de colocação ortogonal axial do reator. Apesar de a rede neural estar predizendo somente um tempo futuro, verifica-se que o *Caso 1* não responde adequadamente em alguns pontos, pois para esta situação física, as redes neurais não foram treinadas, propositadamente, em todas as regiões abrangidas na generalização. Tal fato não representa de forma alguma uma restrição, pois isto é facilmente contornado com a aprendizagem *on-line* da rede neural. A princípio, os casos 2 e 3 apresentam uma resposta mais apropriada, porém deve-se considerar a natureza incremental destes casos e que a Figura 3.4 apresenta os resultados preditos para um tempo futuro. Considerando que estas redes neurais serão usadas em estratégias de controle preditivo, e as predições ao longo do horizonte de tempos de amostragem, são realizadas acessando N (horizonte de predição) vezes a rede neural com a devida atualização das entradas, o acúmulo de erros pode gerar uma predição errônea em situações de variação brusca no comportamento dinâmico. Observou-se que o Caso 3 não representou adequadamente a dinâmica do processo ao longo de um horizonte de predições considerando método *paralelo* (que não é o caso da Figura 3.4, a representa os resultados para o método *série-paralelo*), em situações onde a variável perturbação (variável manipulada) gera resposta inversa na variável resposta. Tal fato representa uma restrição à

utilização do Caso 3, porém a escolha de uma variável manipulada que gera resposta inversa, também não é recomendada.

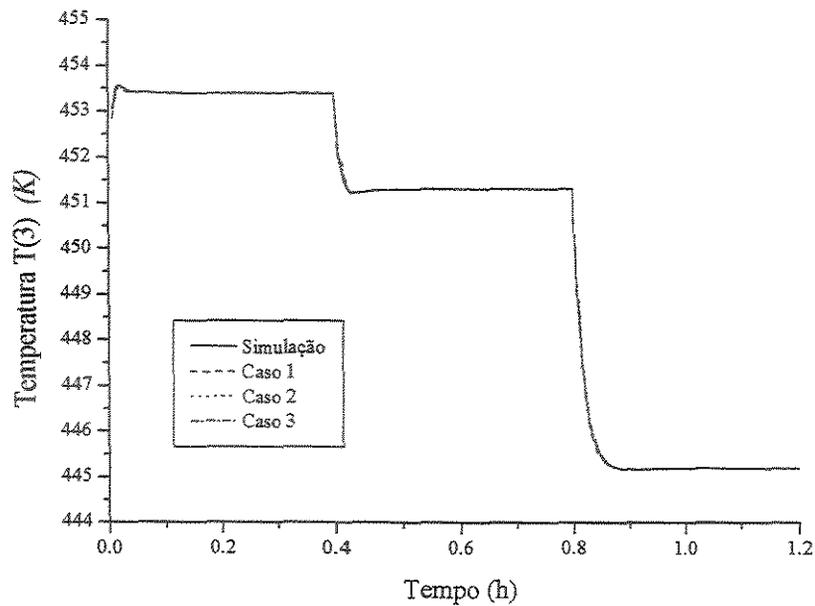


Figura 3.2 – Representação dos padrões para perturbação em GM_o e resposta a temperatura do terceiro ponto de colocação axial do reator

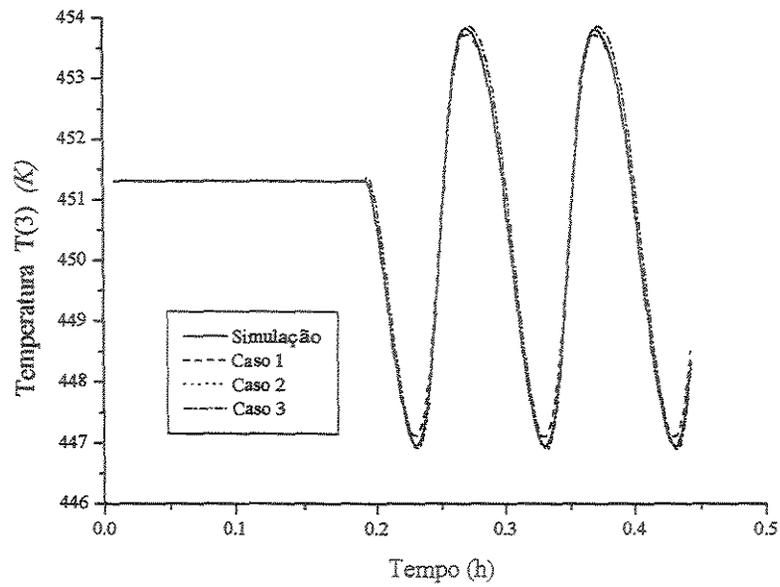


Figura 3.3 – Generalização para perturbação em GM_o e resposta a temperatura do terceiro ponto de colocação axial do reator

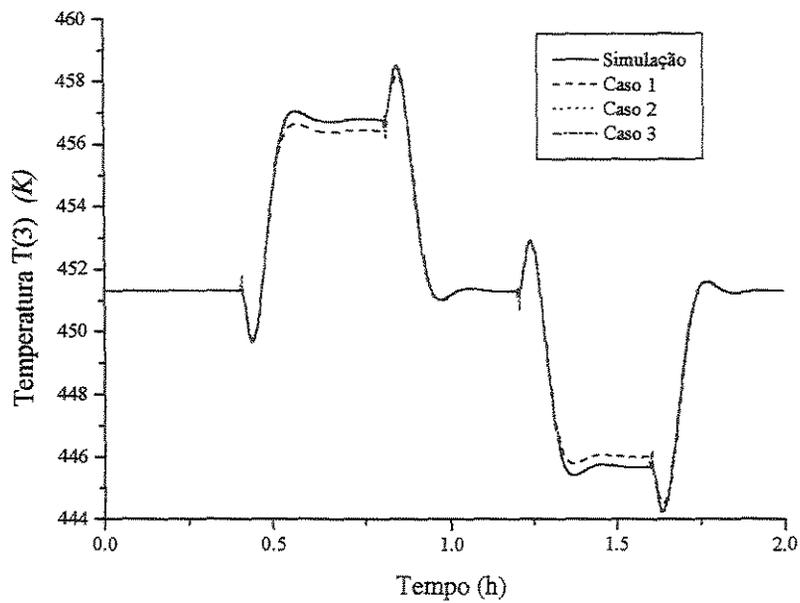


Figura 3.4 - Temperatura do terceiro ponto de colocação ortogonal axial do reator para perturbação em T_{R0}

3.4 Estratégias de controle preditivas

3.4.1 O algoritmo Modificado

A literatura apresenta uma série de trabalhos aplicando modelo dinâmico neural em estratégias de controle preditivas. Dentre estes podemos citar: ENDER & MACIEL FILHO (1998, 1999^{a,b}, 2000^a, 2001^c), PSICHOGIOS & UNGAR (1991), LEE & PARK (1992), DE SOUZA JR. (1993), MORRIS et al. (1994), DE SOUZA JR. et al. (1996), NAJIM et al. (1997), VEGA et al. (2000), SANTOS et al. (2000).

A estrutura básica do algoritmo de controle proposto esta fundamentada na estrutura clássica do algoritmo DMC (CUTLER & RAMAKER, 1979), onde a predição é realizada por um modelo de convolução baseado na resposta a um degrau, que muitas vezes não representa adequadamente as não-linearidades inerentes aos processos industriais. Para contornar este problema, utiliza-se as versões auto-ajustáveis, que melhoram a performance deste algoritmo, mas incorporam uma maior complexidade à malha de controle, e também aumento do tempo computacional requerido, além de incorporar os problemas de instabilidade da identificação *on-line*.

Diante destes fatos sugere-se a alteração da estratégia de predição do algoritmo, incorporando em seu lugar uma rede neural previamente treinada conforme proposto anteriormente. A rede neural também será utilizada na determinação dos coeficientes da resposta a um degrau, necessários para geração da matriz dinâmica do critério de custo quadrático da estratégia de controle preditivo. Seguem abaixo as etapas envolvidas:

a) Definição da trajetória de referência

O comportamento da planta em malha fechada é definido por uma trajetória de referência, utilizando um modelo de primeira ordem, inicializado com a resposta do processo, $y(i, t)$, no tempo considerado.

$$y_w(i, t) = y(i, t) \quad (3.1)$$

$$y_w(i, t + j) = \alpha_i y_w(i, t + j - 1) + (1 - \alpha_i) y_r(i) \quad (3.2)$$

y_w : vetor trajetória de referência;

i : índice referente a malha de controle;

α_i : parâmetro da trajetória de referência ($0 \leq \alpha_i < 1$);

$y_r(i)$: referência desejada para a malha de controle i (*setpoint*).

b) Definição da predição ao longo de um horizonte de tempos de amostragem

No caso particular do DMC, a predição é realizada através de um somatório de convolução baseada nos coeficientes da resposta ao degrau e das ações de controle passadas (CUTLER & RAMAKER, 1979; LUYBEN, 1989).

$$\hat{y}(i, t + m) = y(i, t) + \sum_{j=1}^{N_m} \sum_{k=0}^{(N_c-1)} [h_{i,j}(m+1+k) - h_{i,j}(1+k)] \Delta u(j, t-1-k) \quad (3.3)$$

para $1 \leq i \leq N_m$

$1 \leq m \leq N$

onde:

N = horizonte de predição;

N_m = número de malhas de controle;

N_c = número de termos de convolução;

$\Delta u(j, i)$ = vetor de incrementos na ação de controle, para a malha j ;

$h_{i,j}(l)$ = vetor de resposta ao degrau.

Para o Algoritmo Modificado, sugere-se a inserção da rede neural, previamente treinada para representar o comportamento dinâmico do processo a ser controlado, de tal forma a realizar as predições ao longo de um horizonte de tempos de amostragem. Neste momento é de fundamental importância que a rede seja treinada a representar adequadamente o processo e para isto os pares de treinamento devem conter informações significativas e consistentes em toda a faixa de operação. As redes neurais foram treinadas a prever um tempo futuro e em função disto, esta é acionada tantas vezes quanto for o valor do horizonte de predição.

A predição das “n” variáveis controladas é realizada simultaneamente, submetendo-se a rede neural as “m” variáveis manipuladas (não necessariamente $n = m$).

c) Critério de custo quadrático:

O objetivo da lei do controle preditivo é direcionar as saídas futuras da planta $\hat{y}(i, t + j)$ para a trajetória de referência $y_w(i, t + j)$.

Para isso, minimiza-se o critério de custo quadrático, chegando-se a (maiores detalhes para obtenção desta Lei de Controle, encontra-se no Anexo 3):

$$\underline{\tilde{u}} = \left[\underline{\tilde{A}}^T \underline{\tilde{A}} + \underline{\beta} \right]^{-1} \underline{\tilde{A}}^T (\underline{y}_w - \underline{\hat{y}}) \quad (3.4)$$

onde:

$$\underline{\tilde{A}} = \begin{bmatrix} \underline{A}_1 & \cdots & 0 \\ \vdots & & \vdots \\ \vdots & \ddots & \vdots \\ \underline{A}_N & \cdots & \underline{A}_{N-N_u+1} \end{bmatrix} \quad (3.5)$$

$$\begin{aligned}
\underline{y}_w &= [y_w(1, t+1), \dots, y_w(N_m, t+1), \dots, y_w(1, t+N), \dots, y_w(N_m, t+N)]^T \\
\underline{\hat{y}} &= [\hat{y}(1, t+1), \dots, \hat{y}(N_m, t+1), \dots, \hat{y}(1, t+N), \dots, \hat{y}(N_m, t+N)]^T \\
\underline{\tilde{u}} &= [\Delta u(1, t), \dots, \Delta u(N_m, t), \dots, \Delta u(1, t+Nu-1), \dots, \Delta u(N_m, t+Nu-1)]^T
\end{aligned} \tag{3.6}$$

Nu = horizonte de controle;

$\underline{\beta}$ = matriz que contém os fatores de supressão, β_i^2 , correspondendo a cada malha de controle i .

Para um sistema com 2 entradas e 2 respostas, $\underline{A}_i = \begin{bmatrix} h_{11}(1) & h_{12}(1) \\ h_{21}(1) & h_{22}(1) \end{bmatrix}$, onde $h_{i,j}(1)$

são os coeficientes da resposta a um degrau no instante inicial. No algoritmo modificado estes coeficientes são determinados pela mesma rede neural utilizada nas predições.

Usando a técnica de horizonte descendente, os primeiros N_m elementos do vetor $\underline{\tilde{u}}$ serão implementados, determinando as ações de controle correntes:

$$u(i, t) = u(i, t-1) + \Delta u(i, t) \tag{3.7}$$

O produto $\underline{\tilde{A}}^T \underline{\tilde{A}}$, terá como resultado uma matriz de ordem $(N_m \times Nu \text{ vs. } N_m \times Nu)$. Se todos os β_i^2 's forem iguais a zero, a inversão de $\underline{\tilde{A}}^T \underline{\tilde{A}}$ somente será possível se o posto desta matriz for igual a $N_m \times Nu$.

Executadas as alterações sugeridas, não há mais a necessidade de um modelo interno clássico, e sim, de uma rede neural apropriada para representar o comportamento dinâmico do processo. A rede neural poderá ser submetida a aprendizagem *on-line* conforme descrito no capítulo anterior. A avaliação da necessidade e benefícios incorporados por esta técnica passam a ser descritos a seguir.

3.4.1.1 Resultados

Objetivando verificar o desempenho do algoritmo de controle, foram impostas perturbações do tipo servo, regulador e estocásticas na malha fechada. A título de comparação dos resultados, foi usado o algoritmo DMC auto-ajustável, por ser este o algoritmo que motivou a alteração proposta. Para gerar a versão auto-ajustável do DMC, realizou-se identificação *on-line* do modelo interno, usando para isto mínimos quadrados recursivos (ÅSTRÖM & WITTENMARK, 1984) e fator de esquecimento variável escalar (HALLAGER et al., 1984).

O processo utilizado para testar o algoritmo preditivo modificado é um reator catalítico de leito fixo (TOLEDO, 1999) para obtenção de acetaldeído, descrito no Anexo 1. Estes reatores mostraram-se adequados pois apresentam comportamento dinâmico não linear e resposta inversa. Os resultados obtidos estão dentro de um contexto multivariável, apresentando duas malhas de controle. Em todas as situações analisadas, considerou-se como variáveis manipuladas a vazão mássica superficial de alimentação do reator, GM_o , e relação molar ar/etanol, R , com as respectivas variáveis controladas $T(1)$ e $T(3)$ (temperaturas do primeiro e terceiro pontos de colocação ortogonal axial do reator). Estas variáveis manipuladas e controladas foram adotadas devido ao comportamento dinâmico das respostas nestes pontos e a análise de sensibilidade realizada através de planejamento experimental, apresentado no Anexo 4. A Tabela 3.1 apresenta as variações nas referências desejadas das temperaturas do reator.

Tabela 3.1 – Problema Servo

Tempo (h)	T(1) (K) (Malha 1)	T(3) (K) (Malha 2)	Desvio em T(1)	Desvio em T(3)
0.00 – 0.25	449.71	451.30	-3.50	0.00
0.25 – 0.50	449.71	450.30	-3.50	-1.00
0.50 – 0.75	453.21	450.30	0.00	-1.00
0.75 – 1.00	453.21	451.30	0.00	0.00

Nos casos analisados considerou-se os mesmos parâmetros de ajuste do controlador, objetivando comparar as diferentes formas de predição adotadas.

A Figura 3.5.a apresenta os resultados da resposta em malha fechada para o problema servo, Tabela 3.1, considerando aprendizagem *on-line* das redes neurais usadas nas predições do algoritmo modificado. Observa-se um controle efetivo em todos os casos apresentados. Dentre os modelos de predição neural, verifica-se nesta figura, um comportamento menos robusto do Caso 3 em relação aos outros dois, nos momentos iniciais. Também o DMC auto-ajustável apresentou flutuações iniciais para a malha de controle 2. As Figuras 3.5.b e 3.5.c apresentam as ações de controle para o problema servo proposto, onde pode –se observar a menor robustez do Caso 3.

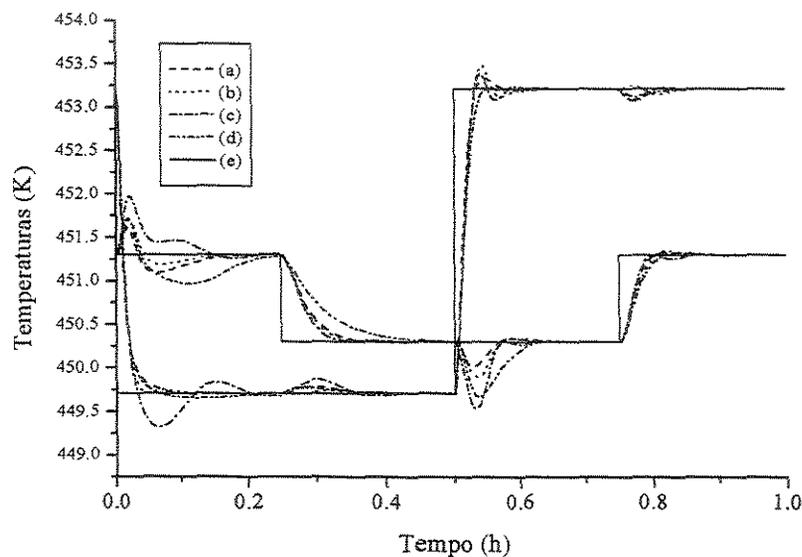


Figura 3.5.a – Problema servo com aprendizagem *on-line* das redes neurais. (a) Caso 1, $IAE(1)=0.1157$, $IAE(2)=0.0969$; (b) Caso 2, $IAE(1)=0.1183$, $IAE(2)=0.0979$; (c) Caso 3, $IAE(1)=0.1498$, $IAE(2)=0.1209$; (d) DMC Auto-Ajustável, $IAE(1)=0.1242$, $IAE(2)=0.1733$; (e) Referências da malha fechada.

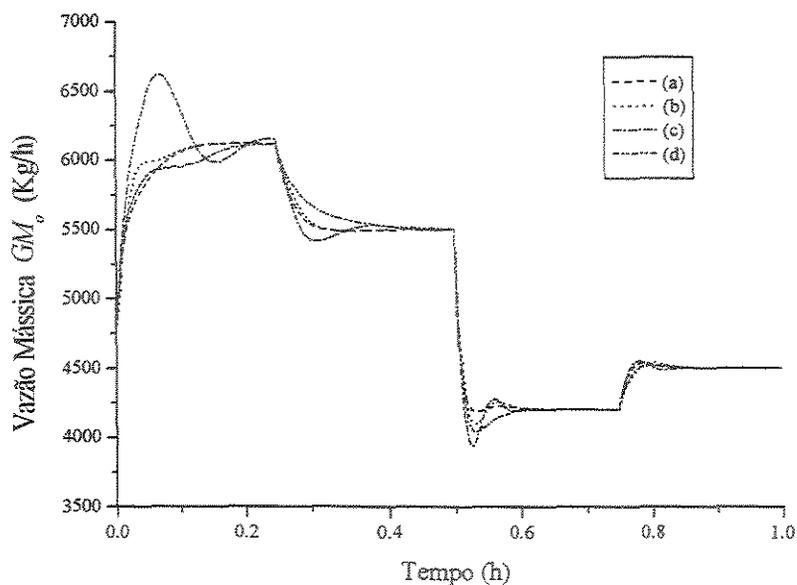


Figura 3.5.b – Ação de controle para problema servo com aprendizagem *on-line* das redes neurais. (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

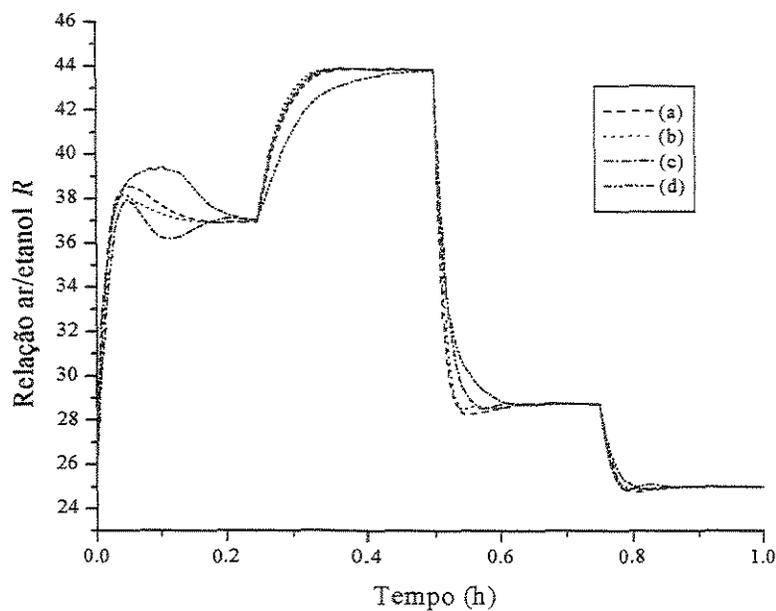


Figura 3.5.c – Ação de controle para problema servo com aprendizagem *on-line* das redes neurais. (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

A Figura 3.6.a – 3.6.c apresentam os resultados considerando predição usando redes neurais sem aprendizagem *on-line*. Pode-se observar para o Caso 1 a presença de *offset*, pois esta forma não incorpora ação integral ao algoritmo de controle. Porém, os resultados para o Caso 2 e 3 não apresentaram este problema, devido a forma incremental que os dados foram apresentados a rede neural, pois introduz-se ação integral ao controlador.

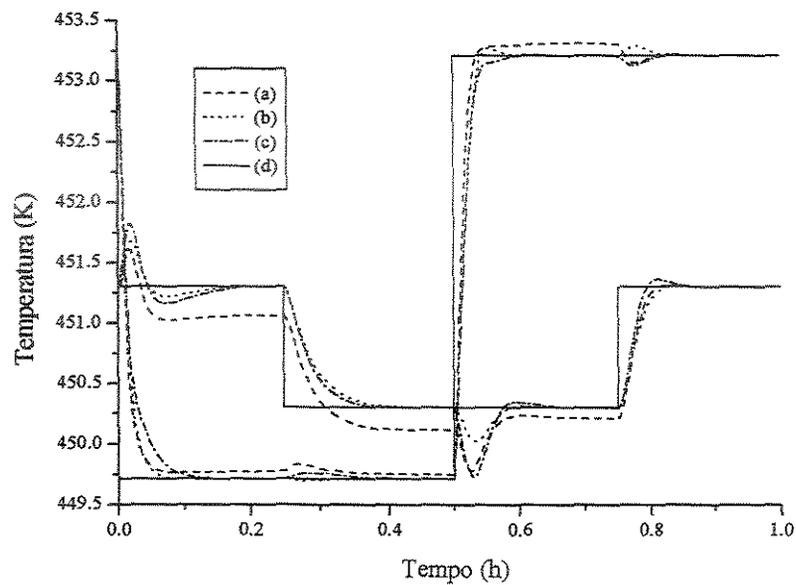


Figura 3.6.a – Problema servo sem aprendizagem *on-line* das redes neurais. (a) Caso 1, $IAE(1)=0.1434$, $IAE(2)=0.1773$; (b) Caso 2, $IAE(1)=0.1234$, $IAE(2)=0.0981$; (c) Caso 3, $IAE(1)=0.1392$, $IAE(2)=0.1100$; (d) Referências da malha fechada.

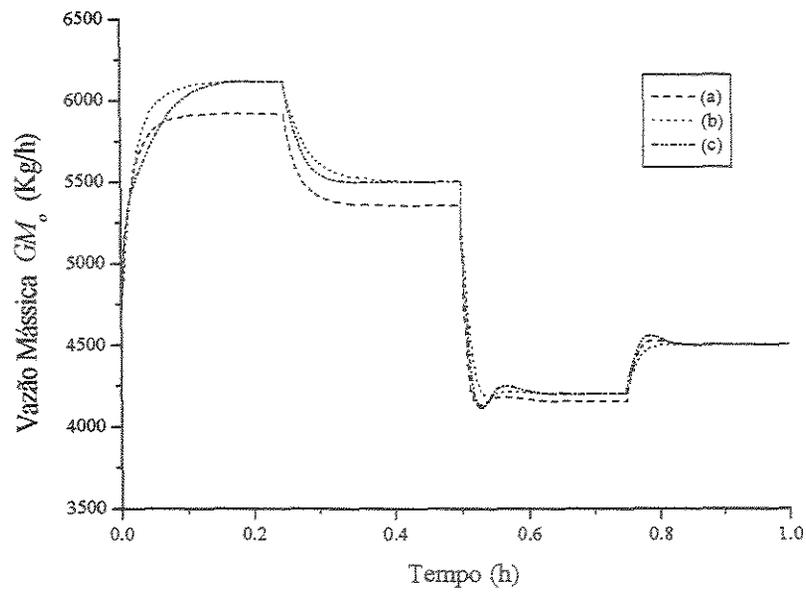


Figura 3.6.b – Ação de controle para problema servo sem aprendizagem *on-line* das redes neurais. (a) Caso 1; (b) Caso 2; (c) Caso 3.

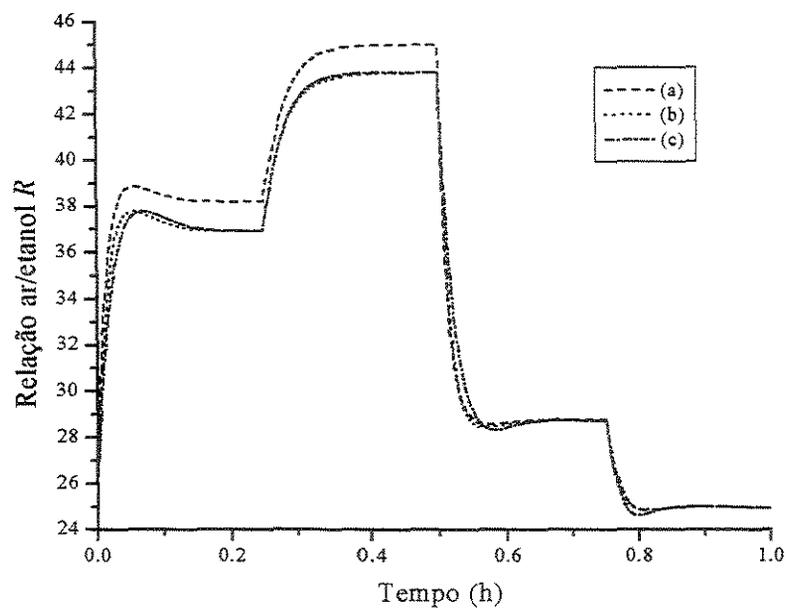


Figura 3.6.c – Ação de controle para problema servo sem aprendizagem *on-line* das redes neurais. (a) Caso 1; (b) Caso 2; (c) Caso 3.

A aprendizagem *on-line* é realizada com um conjunto de dados formado por um número preestabelecido de perturbações/respostas do processo, obtidos de forma *on-line*, a cada tempo de amostragem. As Figuras 3.7 e 3.8 apresentam os erros quadráticos das redes neurais na representação do conjunto de dados citado, usando os pesos da aprendizagem *off-line*, aqui denominados pesos padrão, e aprendizagem *on-line*, para o Caso 1 e Caso 2 respectivamente, correspondente a resposta em malha fechada apresentada na Figura 3.5.a. O conjunto de dados usados na aprendizagem da rede é formado por perturbações e respostas reais do sistema. Observando a Figura 3.6.a, onde não é considerada a aprendizagem *on-line*, ou seja, utiliza-se os pesos padrões em todos os momentos em que a rede neural é usada, pode-se verificar a presença de *off-set* na mesma região onde o erro quadrático deste conjunto de pesos apresenta maior magnitude, o que está representado na Figura 3.7. Isto deve-se ao fato desta região não estar devidamente mapeada na fase de aprendizagem *off-line*. A aprendizagem *on-line* da rede tem por objetivo contornar estas limitações, a qual tem-se mostrado eficiente, conforme pode ser observado nas Figuras 3.7 e 3.8. A forma incremental de apresentação dos dados à rede neural, realizada nos Casos 2 e 3, conforme já discutido anteriormente, permite a eliminação do *off-set*, mesmo sem a aprendizagem *on-line*. A Figura 3.8 apresenta o erro quadrático ao longo do tempo para o Caso 2, onde pode-se verificar um comportamento similar nos dois casos. Observa-se uma variação do erro quadrático nos momentos de mudança da referência da malha fechada, e uma rápida adaptação do modelo neural do processo.

A Figura 3.9 apresenta uma comparação da variação na variável manipulada GM_o , para o Caso 3, considerando o algoritmo de controle com e sem aprendizagem *on-line* das redes neurais usadas nas predições. Variações mais bruscas em GM_o são observadas na situação em que usou-se aprendizagem *on-line*. A aprendizagem *on-line* torna a rede menos genérica, representando mais apropriadamente a região de operação, após algum treinamento. Diante deste fato e em uma situação com variações bruscas nas variáveis do processo, é recomendado utilizar parâmetros do controlador mais conservativos.

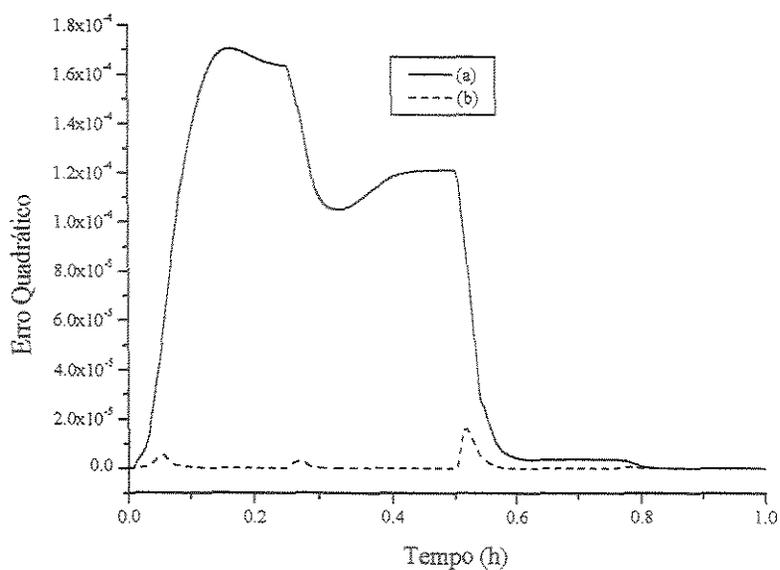


Figura 3.7 – Erro Quadrático para problema servo - *Caso 1*. (a) Pesos Padrão; (b) Pesos da aprendizagem *on-line*.

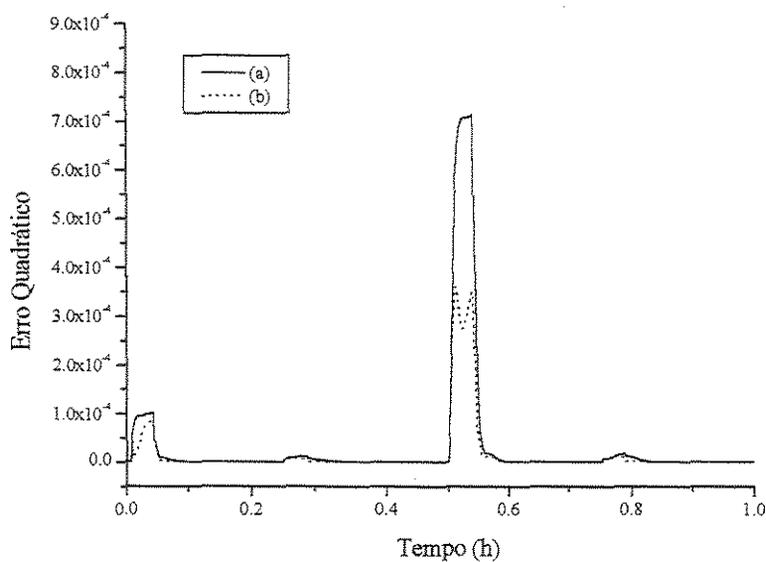


Figura 3.8 – Erro Quadrático para problema servo - *Caso 2*. (a) Pesos Padrão; (b) Pesos da aprendizagem *on-line*.

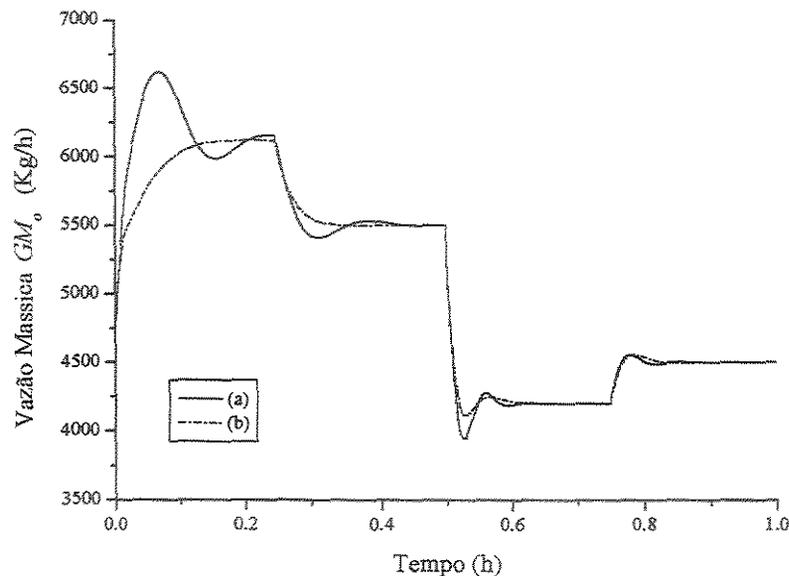


Figura 3.9 – Comparação das ações de controle – problema servo – Caso 3. (a) Com aprendizagem *on-line* da rede neural; (b) Sem aprendizagem *on-line* da rede neural.

As Figuras 3.10.a a 3.10.d apresentam os resultados para problema servo, Tabela 3.1, juntamente com perturbações estocásticas. Para amenizar os efeitos das perturbações estocásticas, as leituras geradas do processo foram filtradas através de um filtro passa baixa (Isermann, 1981). A presença de ruído estocástico acarreta problemas adicionais à malha fechada, porém um controle efetivo pode ser observado para o algoritmo modificado proposto. A identificação de um modelo interno por mínimos quadrados apresenta limitações quando usado com processos muito ruidosos. Por esta razão, observa-se uma resposta não muito adequada para a malha de controle 2, para a estratégia de controle baseada no DMC Auto-Ajustável. O uso de redes neurais mostrou-se mais apropriado.

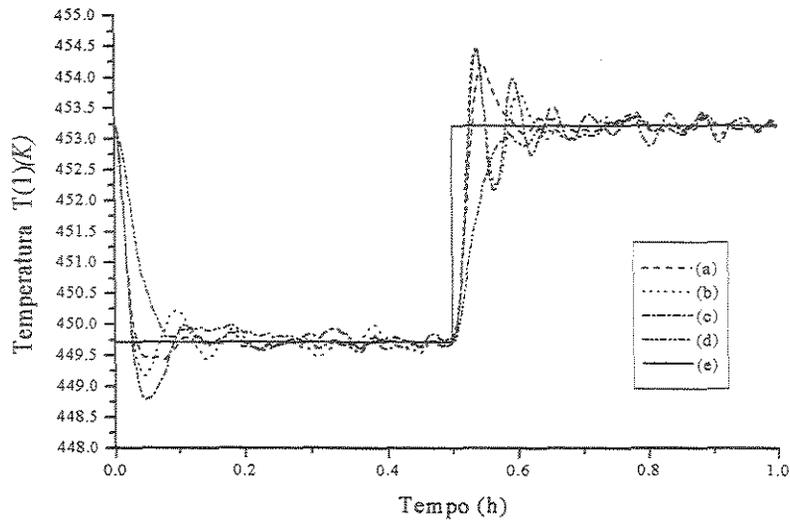


Figura 3.10.a – Problema servo com perturbação estocástica. (a) Caso 1, $IAE(1)=0.2198$; (b) Caso 2, $IAE(1)=0.2682$; (c) Caso 3, $IAE(1)=0.2995$; (d) DMC Auto-Ajustável, $IAE(1)=0.3324$; (e) Referência da malha fechada.

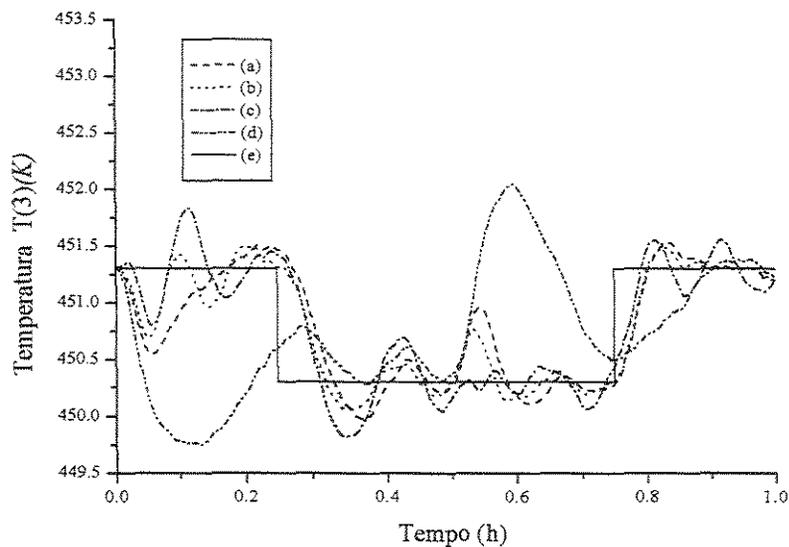


Figura 3.10.b – Problema servo com perturbação estocástica. (a) Caso 1, $IAE(2)=0.2368$; (b) Caso 2, $IAE(2)=0.1846$; (c) Caso 3, $IAE(2)=0.2181$; (d) DMC Auto-Ajustável, $IAE(2)=0.6397$; (e) Referência da malha fechada.

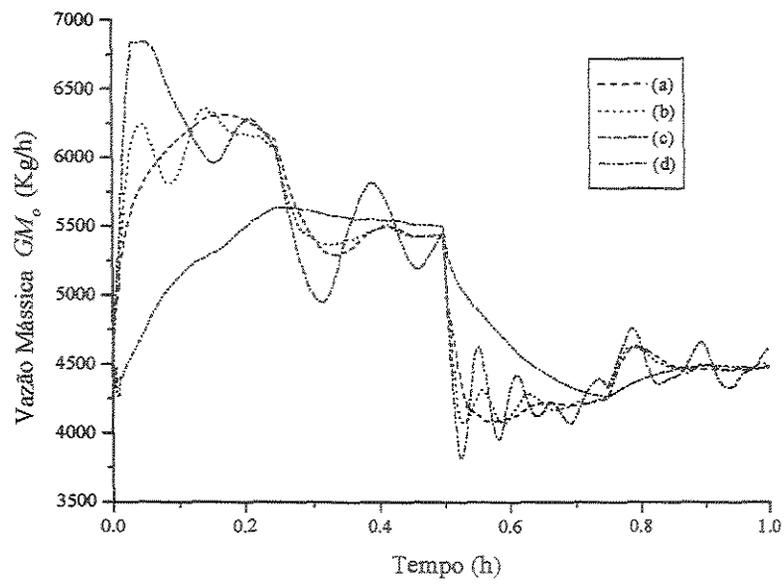


Figura 3.10.c – Ação de controle para problema servo com perturbação estocástica. (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

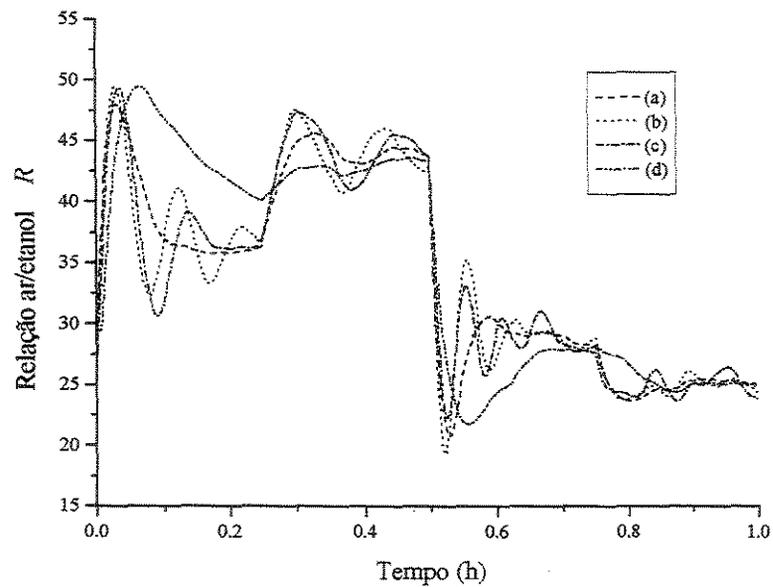


Figura 3.10.d – Ação de controle para problema servo com perturbação estocástica. (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

As Figuras 3.11.a a 3.11.d apresentam os resultados considerando problema regulador gerado pela perturbação na temperatura de alimentação do reator, T_{fo} , de acordo com a relação:

$$T_{fo} = T_{fo,ss} + 5 K; \text{ para } 0.1 \leq t \leq 0.4 \quad (3.8)$$

Estes resultados foram obtidos considerando aprendizagem *on-line* das redes neurais utilizadas. Verifica-se através destes resultados um desempenho superior do algoritmo modificado em relação ao DMC auto-ajustável.

As Figuras 3.12.a a 3.12.d apresentam as respostas da malha fechada para o problema regulador apresentado na equação (3.8), juntamente com uma perturbação estocástica. Verifica-se neste caso a superioridade do algoritmo modificado, o que vem comprovar o potencial das redes neurais na identificação de processos não lineares, mesmo na presença de ruídos. Neste caso particular observou-se problemas sérios no processo de identificação *on-line* usando mínimos quadrados, o que certamente comprometeu os resultados da malha fechada para o DMC auto-ajustável.

As Figuras 3.13.a a 3.13.d mostram os resultados para o problema regulador gerado pela perturbação da temperatura do fluido refrigerante do reator, T_{R0} , conforme equação (3.9). Esta perturbação produz resposta inversa em $T(3)$, o que torna o problema bastante interessante. Pelas figuras apresentadas observa-se um controle efetivo nas situações analisadas.

$$T_{R0} = T_{R0,ss} + 5 K; \text{ para } 0.05 \leq t \leq 0.5 \quad (3.9)$$

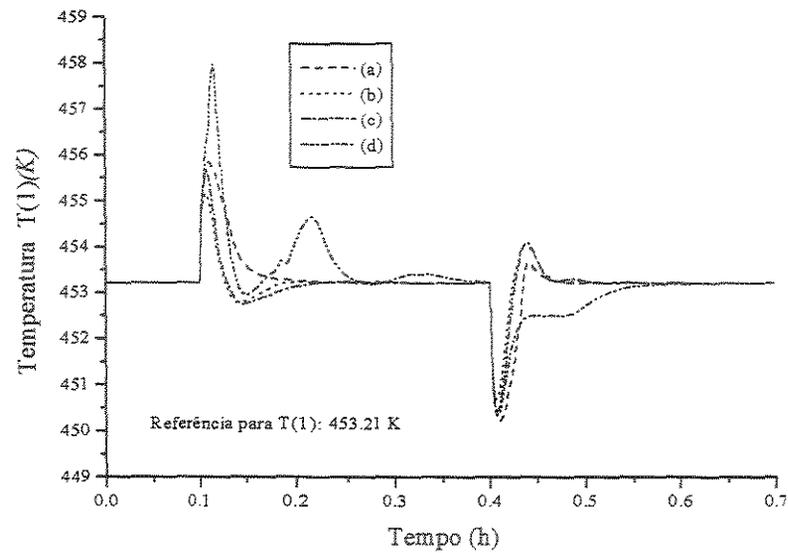


Figura 3.11.a - Problema Regulador – Perturbação em T_{fo} . (a) Caso 1, $IAE(1)=0.1657$; (b) Caso 2, $IAE(1)=0.1053$; (c) Caso 3, $IAE(1)=0.1263$; (d) DMC Auto-Ajustável, $IAE(1)=0.2874$.

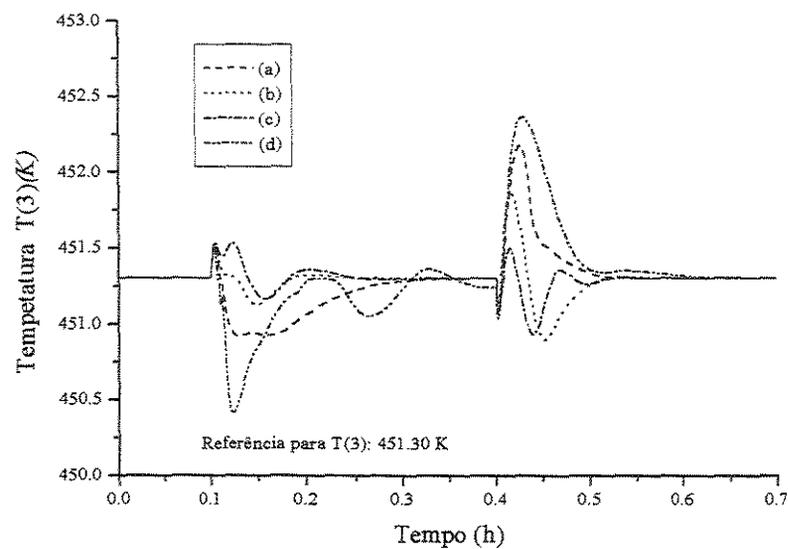


Figura 3.11.b - Problema Regulador – Perturbação em T_{fo} . (a) Caso 1, $IAE(2)=0.0745$; (b) Caso 2, $IAE(2)=0.0360$; (c) Caso 3, $IAE(2)=0.0278$; (d) DMC Auto-Ajustável, $IAE(2)=1113$.

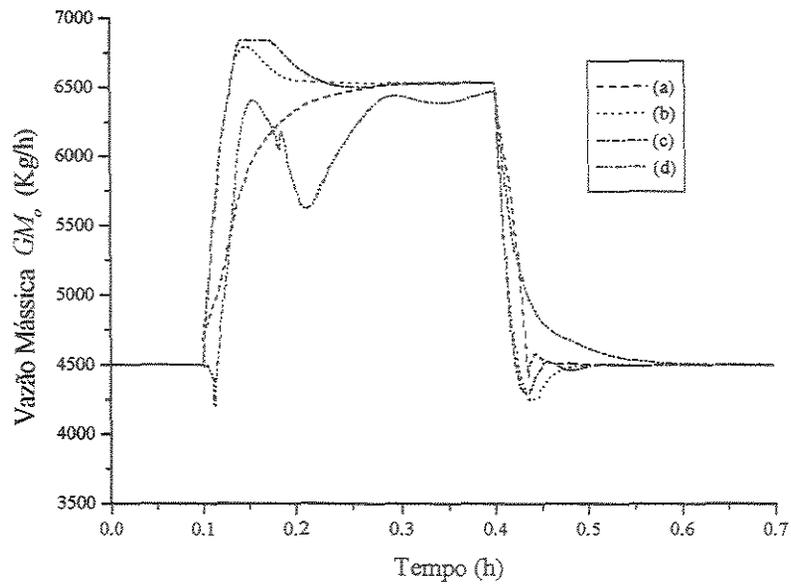


Figura 3.11.c – Ação de controle para problema Regulador – Perturbação em T_{fo} . (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

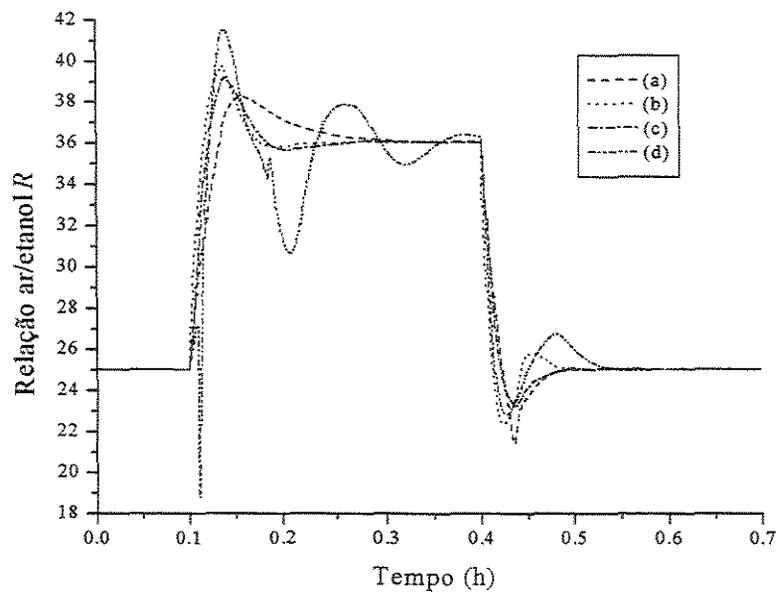


Figura 3.11.d – Ação de controle para problema Regulador – Perturbação em T_{fo} . (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

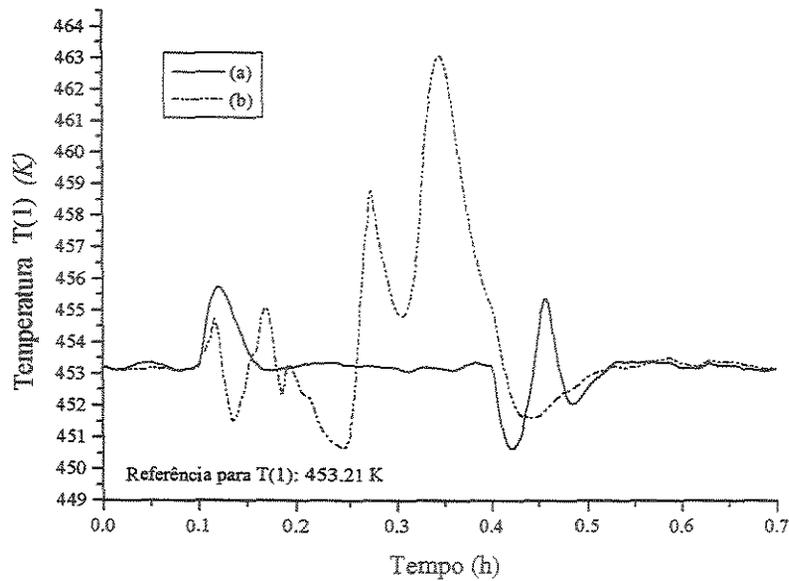


Figura 3.12.a – Problema Regulador pela perturbação em T_{fo} com perturbação estocástica. (a) Caso 1, $IAE(1)=0.2647$; (b) DMC Auto-Ajustável, $IAE(1)=0.9891$.

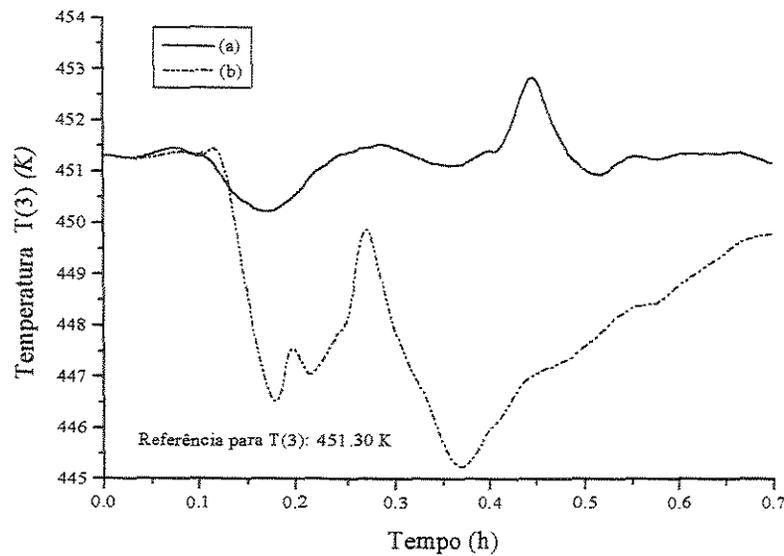


Figura 3.12.b – Problema Regulador pela perturbação em T_{fo} com perturbação estocástica T_{fo} . (a) Caso 1, $IAE(2)=0.1916$; (b) DMC Auto-Ajustável, $IAE(2)=2.002$.

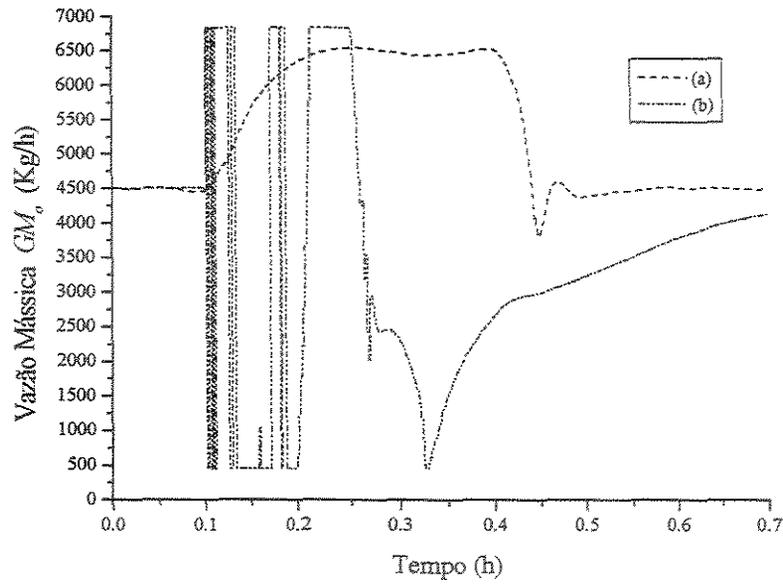


Figura 3.12.c – Ação de controle para problema Regulador pela perturbação em T_{fo} com perturbação estocástica T_{fo} . (a) Caso 1; (b) DMC Auto-Ajustável.

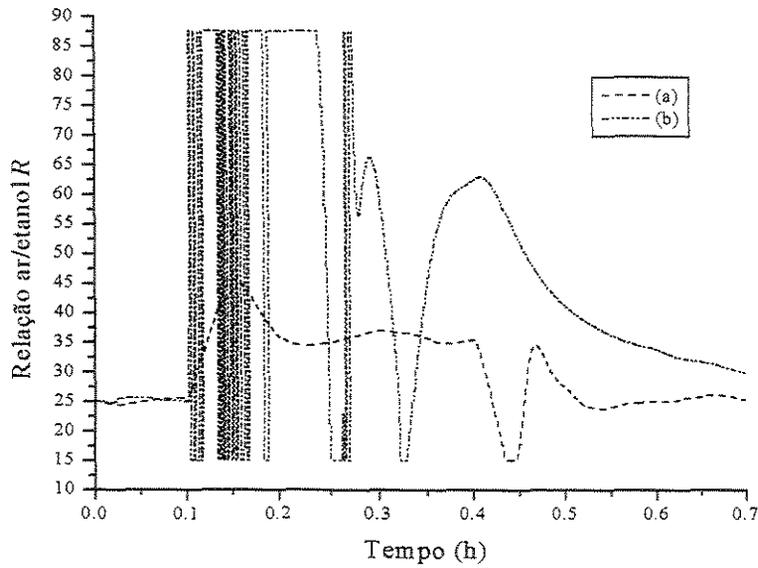


Figura 3.12.c – Ação de controle para problema Regulador pela perturbação em T_{fo} com perturbação estocástica T_{fo} . (a) Caso 1; (b) DMC Auto-Ajustável.

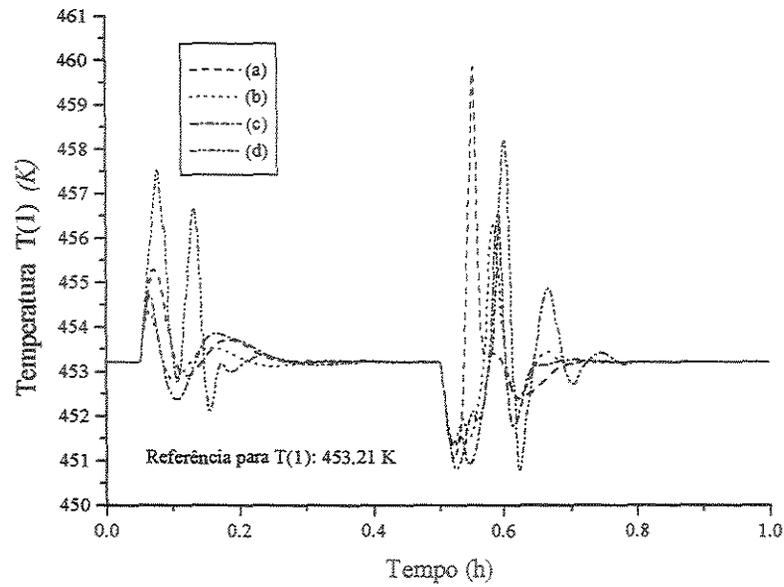


Figura 3.13.a – Problema regulador – perturbação em T_{RO} . (a) Caso 1, $IAE(1)=0.3056$; (b) Caso 2, $IAE(1)=0.2318$; (c) Caso 3, $IAE(1)=0.3097$; (d) DMC Auto-Ajustável, $IAE(1)=0.5182$.

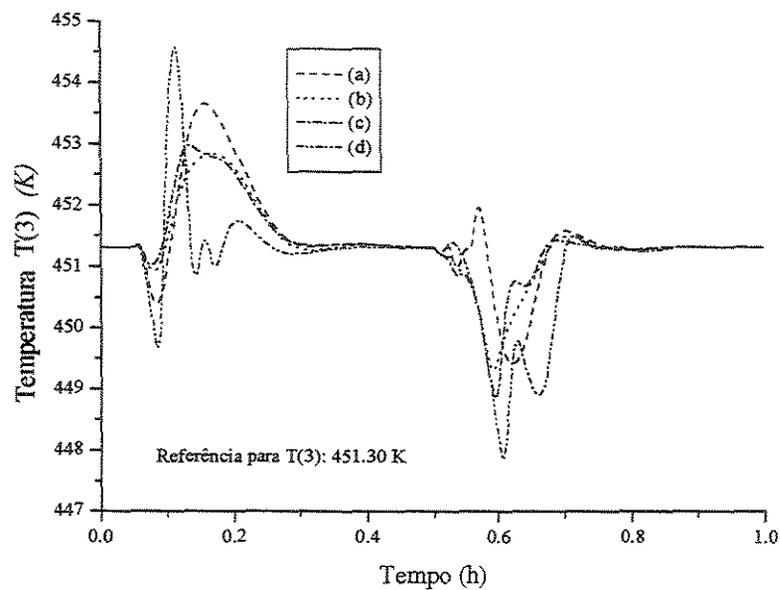


Figura 3.13.b – Problema regulador – perturbação em T_{RO} . (a) Caso 1, $IAE(2)=0.4061$; (b) Caso 2, $IAE(2)=0.3383$; (c) Caso 3, $IAE(2)=0.3443$; (d) DMC Auto-Ajustável, $IAE(2)=0.4339$.

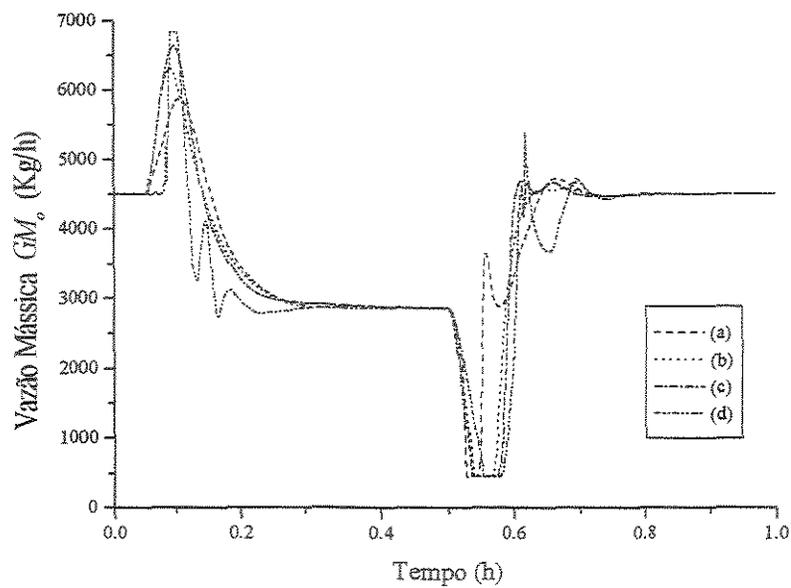


Figura 3.13.c – Ação de controle para problema regulador – perturbação em T_{R0} . (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

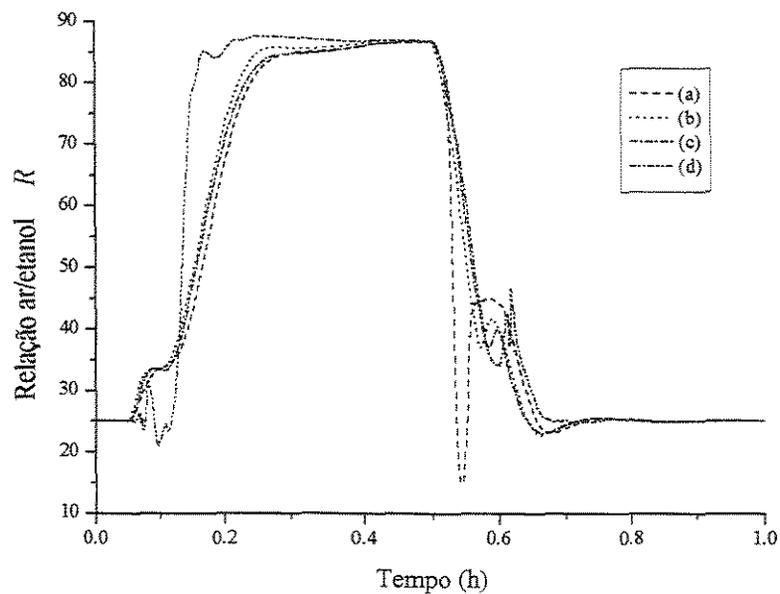


Figura 3.13.d – Ação de controle para problema regulador – perturbação em T_{R0} . (a) Caso 1; (b) Caso 2; (c) Caso 3; (d) DMC Auto-Ajustável.

A perturbação em T_{R0} gera uma forte não linearidade no processo, o que dificulta o controle do sistema. Pode-se verificar nas Figuras 3.13.c e 3.13.d ações de controle enérgicas, chegando aos limites operacionais do reator impostos. Mesmo diante das dificuldades geradas, verifica-se um controle efetivo, o que vem confirmar a eficiência das alterações propostas na estratégia de controle preditivo.

No próximo tópico será abordado a estratégia de controle preditivo com restrições inerentes ao processo de otimização. Para resolução deste problema de otimização *on-line* far-se-á uso do algoritmo SQP (*Successive Quadratic Control*). Esta técnica de controle explora mais uma importante aplicação de uso de redes neurais.

3.4.2 Algoritmo de Controle Preditivo com Restrições

O algoritmo de controle preditivo com restrições consiste na otimização *on-line* da função objetivo, considerando restrições nos vetores de predição e ações de controle futuras. A função objetivo do controlador preditivo é representada por:

$$J(k) = \sum_{j=1}^N \sum_{i=1}^{N_m} [\hat{y}(i, k+j) - y_w(i, k+j)]^2 + \sum_{j=1}^{Nu} \sum_{i=1}^{N_m} \beta^2(i) [\Delta u(i, k+j-1)]^2 \quad (3.10)$$

onde:

N = horizonte de predição;

N_m = número de malhas de controle;

Nu = horizonte de Controle;

$\beta^2(i)$ = fator de supressão;

$\hat{y}(i, k+j)$ = vetor de predição gerado por uma rede neural dinâmica;

$y_w(i, k+j)$ = trajetória de referência,

$\Delta u(i, k+j-1)$ = vetor de incrementos nas ações de controle.

Com as seguintes restrições:

$$y_{\min} \leq \hat{y}(k+i) \leq y_{\max}, \quad i = 1, \dots, N \quad (3.11)$$

$$u_{\min} \leq u(k+i-1) \leq u_{\max}, \quad i = 1, \dots, N \quad (3.12)$$

$$|u(k+i) - u(k+i-1)| \leq \Delta u_{\max}, \quad i = 1, \dots, Nu \quad (3.13)$$

Para resolver esse problema de otimização, usou-se a rotina DNCONF da biblioteca IMSL que emprega o método de programação quadrática sucessiva, o qual aproxima localmente a função objetivo por uma função quadrática e as restrições não lineares por funções lineares. Neste procedimento de otimização as variáveis independentes são as ações de controle futuras. Usando a técnica de horizonte descendente, apenas as primeiras N_m ações de controle calculadas são implementadas. Todos os cálculos são repetidos a cada tempo de amostragem.

O uso de uma rede neural para realizar previsões como modelo interno em um esquema de controle que minimiza, via SQP, a função objetivo já foi discutido, por exemplo, por PSICHOGIOS & UNGAR (1991), DE SOUZA JR. (1993), DE SOUZA JR. et al. (1996), NAJIM et al. (1997), VEGA et al. (2000) e SANTOS et al. (2000). A idéia básica do algoritmo implementado já foi bastante discutido na literatura, porém sugere-se o uso da aprendizagem *on-line* das redes neurais usadas nas previsões futuras.

No algoritmo proposto, o vetor de previsões futuras é gerado a partir de uma rede neural *feedforward* multivariável com aprendizagem *backpropagation*. Esta rede é treinada numa primeira etapa de uma forma *off-line*, formando um conjunto de pesos padrões que sempre é acionado quando a aprendizagem *on-line* não gerar bons resultados.

3.4.2.1 Resultados

Os resultados apresentados nesta seção buscam analisar o desempenho do algoritmo de controle preditivo com restrições. Os resultados serão comparados novamente com o DMC auto-ajustável, conforme descrito anteriormente. As previsões do algoritmo com restrições baseadas em redes neurais, estão de acordo com o Caso 1 considerando aprendizagem *on-line*, devido ao bom comportamento desta configuração em todas as situações anteriormente avaliadas. As malhas de controle são as mesmas usadas na seção anterior, ou seja:

- variáveis manipuladas: vazão mássica superficial de alimentação do reator, GM_o , e relação molar ar/etanol, R .
- variáveis controladas: $T(1)$ e $T(3)$ (temperaturas do primeiro e terceiro pontos de colocação ortogonal axial do reator).

As Figuras 3.14.a a 3.14.c apresentam os resultados para problema servo apresentado na Tabela 3.1, onde pode ser verificado um controle efetivo, com desempenho superior ao DMC auto-ajustável.

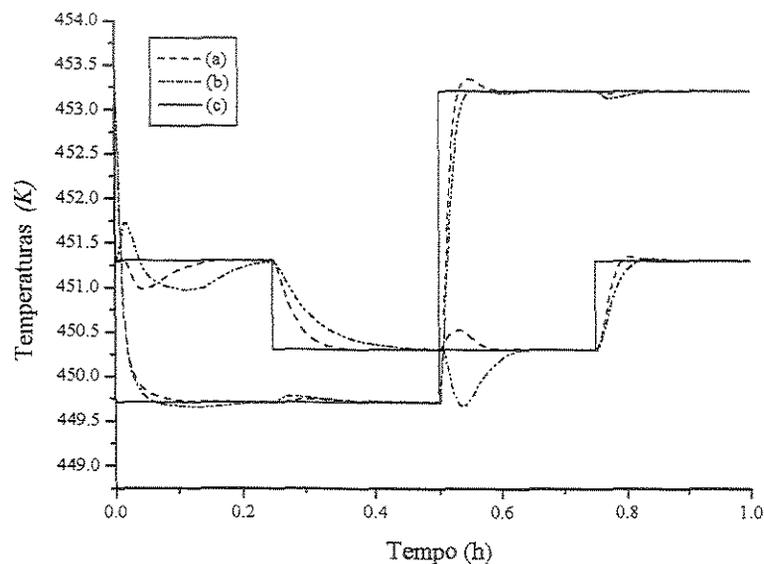


Figura 3.14.a – Problema Servo. (a) Controle Preditivo com restrições, $IAE(1)=0.1053$, $IAE(2)=0.0871$; (b) DMC auto-ajustável, $IAE(1)=0.1242$, $IAE(2)=0.1733$; (c) Referências da malha fechada.

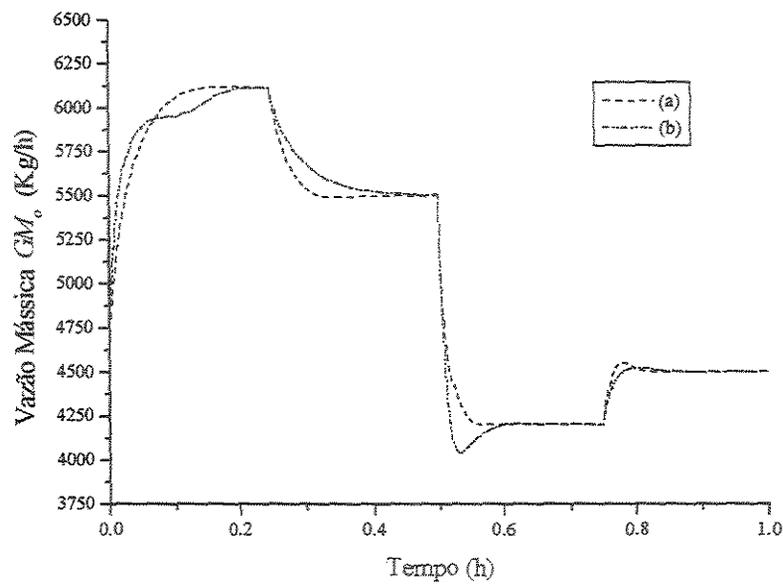


Figura 3.14.b – Ação de controle para problema Servo. (a) Controle Preditivo com restrições; (b) DMC auto-ajustável.

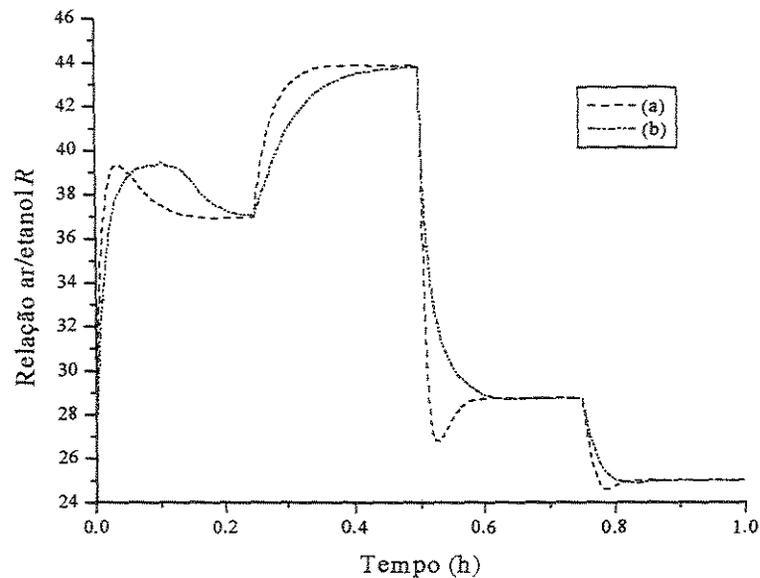


Figura 3.14.c – Ação de controle para problema Servo. (a) Controle Preditivo com restrições; (b) DMC auto-ajustável.

As Figuras 3.15.a a 3.15.d apresentam os resultados para o problema servo adicionado de perturbações estocásticas. Para o caso do DMC auto-ajustável verifica-se um controle não efetivo para a malha 2, Figura 3.15.b, provavelmente por problemas de identificação usando mínimos quadrados na presença de ruídos. Já o algoritmo preditivo com restrições tem-se mostrado eficiente nas duas malhas de controle. Neste caso, novamente as informações provenientes do processo foram filtradas por um filtro passa baixa, amenizando assim os efeitos do ruído estocástico.

As Figuras 3.16.a a 3.16.d apresentam os resultados para o problema regulador, Eq. 3.8, obtido pela perturbação da temperatura de alimentação do reator, T_{fo} , onde pode-se observar um controle efetivo para as duas estratégias de controle, porém um comportamento mais robusto do algoritmo preditivo com restrições é verificado.

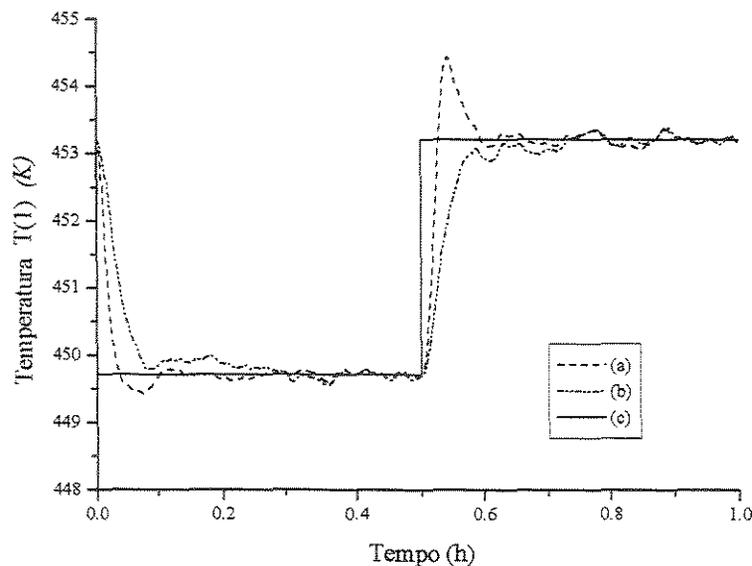


Figura 3.15.a – Problema servo com perturbação estocástica. (a) Controle Preditivo com restrições, $IAE(1)=0.2164$; (b) DMC auto-ajustável, $IAE(1)=0.3324$; (c) Referência da malha fechada.

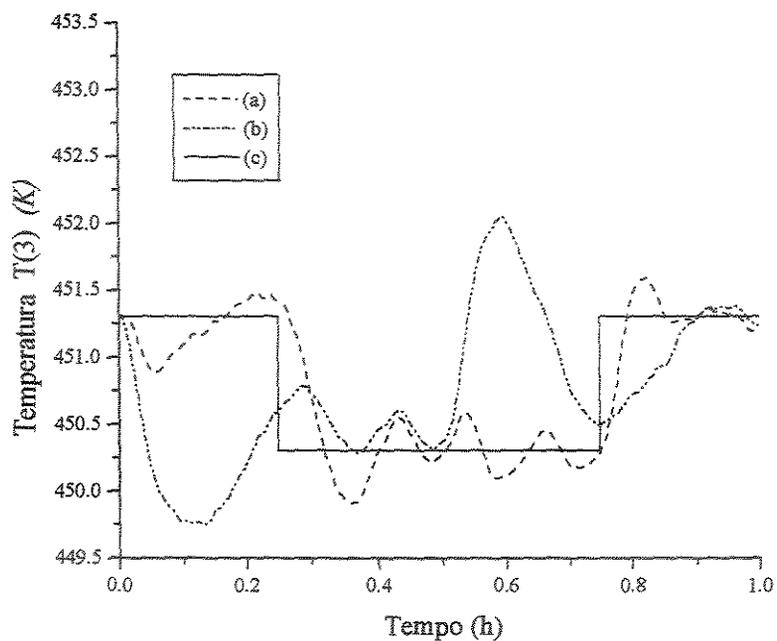


Figura 3.15.b – Problema servo com perturbação estocástica. (a) Controle Preditivo com restrições, $IAE(2)=0.1902$; (b) DMC auto-ajustável, $IAE(2)=0.6397$; (c) Referência da malha fechada.

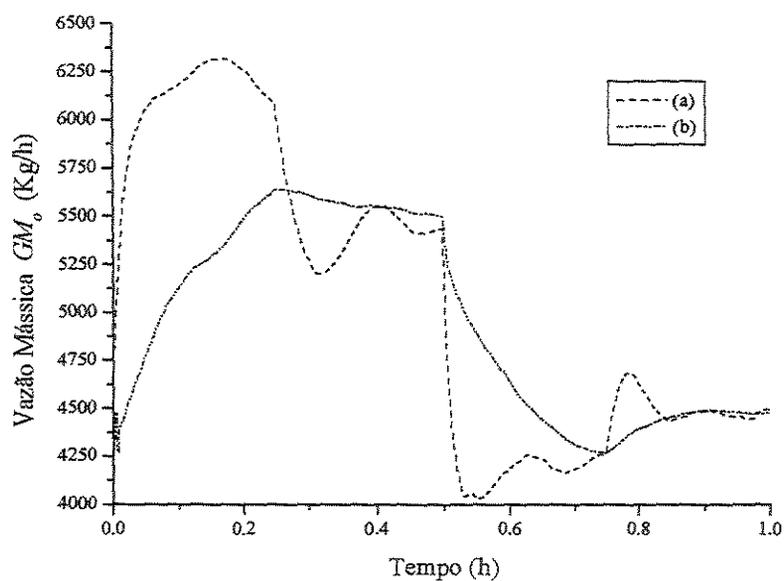


Figura 3.15.c – Ação de controle para problema servo com perturbação estocástica. (a) Controle Preditivo com restrições; (b) DMC auto-ajustável.

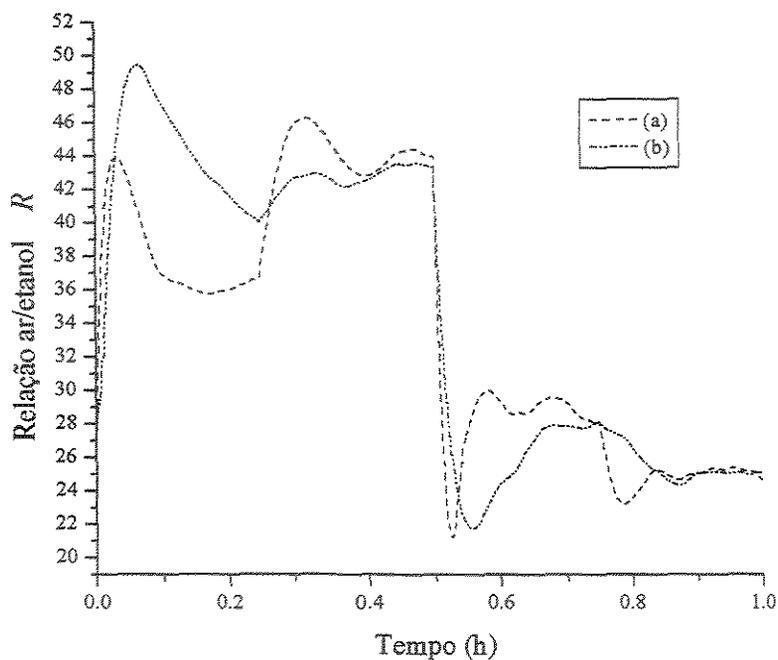


Figura 3.15.d – Ação de controle para problema servo com perturbação estocástica. (a) Controle Preditivo com restrições; (b) DMC auto-ajustável.

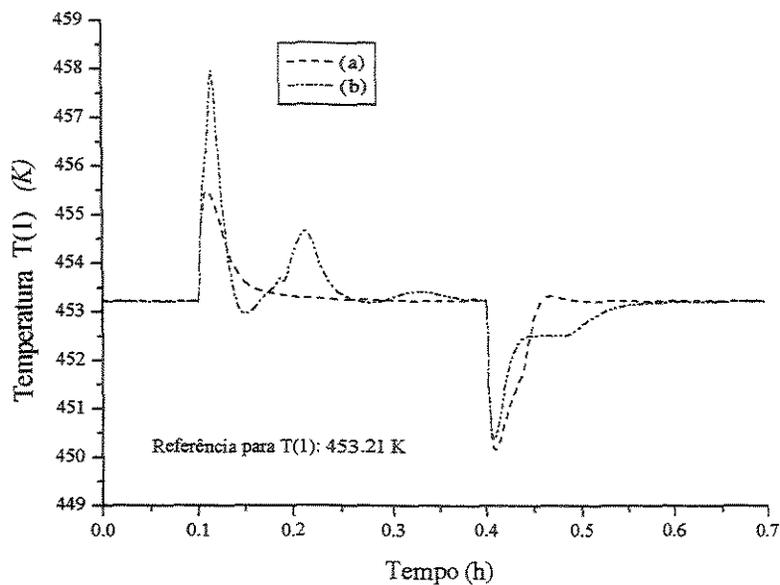


Figura 3.16.a - Problema Regulador – Perturbação em T_{fo} . (a) Controle Preditivo com restrições, $IAE(1)=0.1802$; (b) DMC auto-ajustável, $IAE(1)=0.2874$.

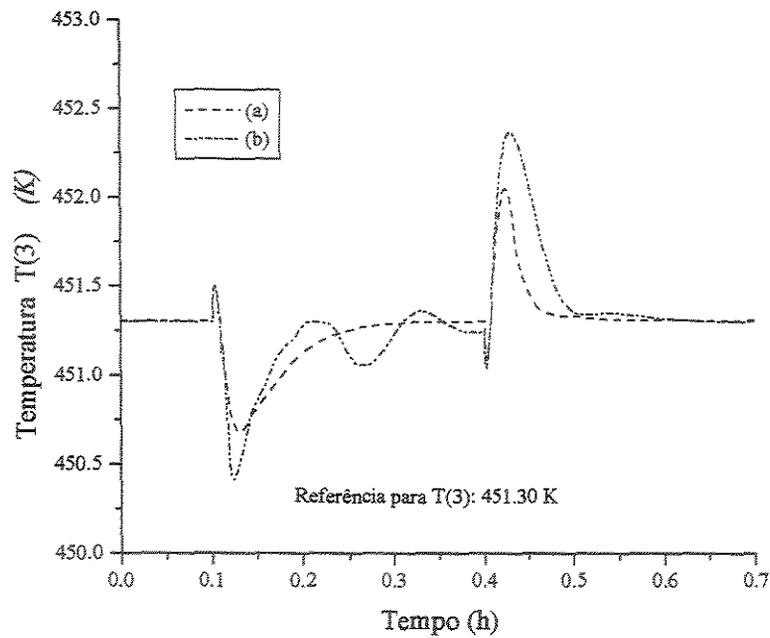


Figura 3.16.b - Problema Regulador – Perturbação em T_{fo} . (a) Controle Preditivo com restrições, IAE(2)=0.0696; (b) DMC auto-ajustável, IAE(2)=0.1113.

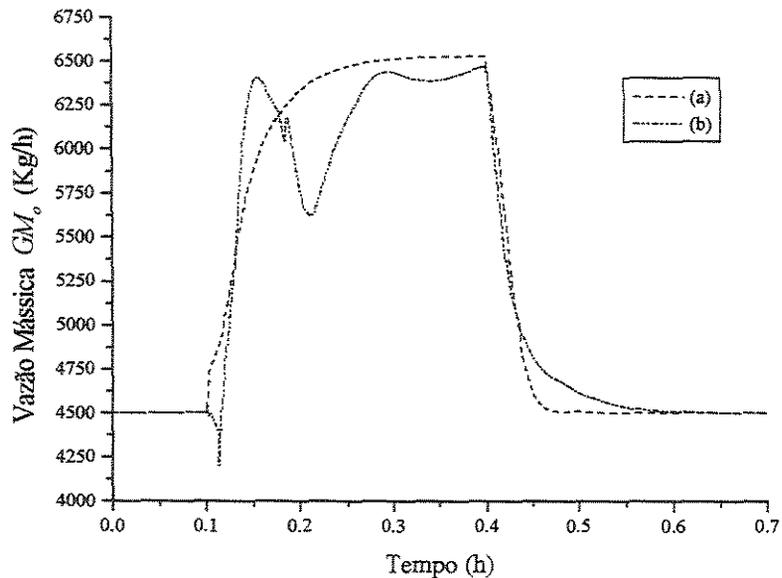


Figura 3.16.c – Ação de controle para problema Regulador – Perturbação em T_{fo} . (a) Controle Preditivo com restrições; (b) DMC auto-ajustável

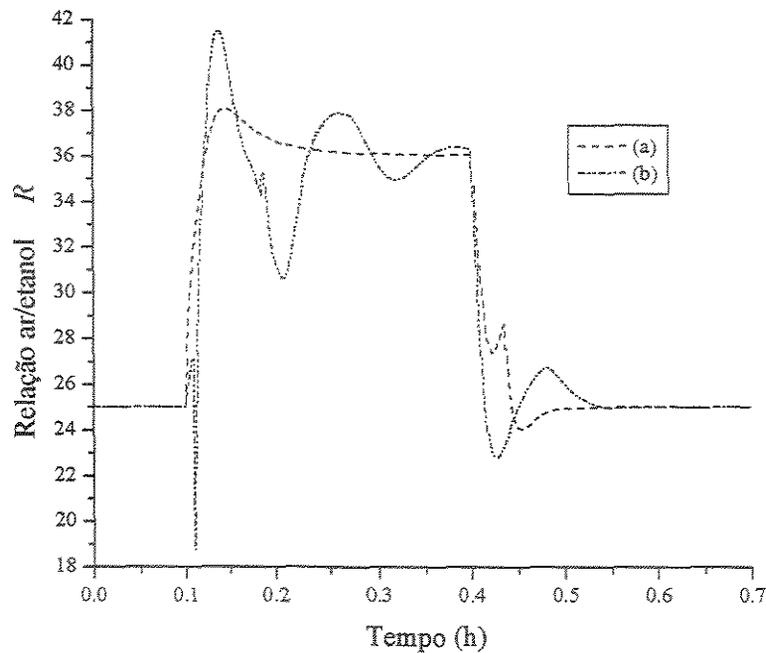


Figura 3.16.d – Ação de controle para problema Regulador – Perturbação em T_{f0} . (a) Controle Preditivo com restrições; (b) DMC auto-ajustável

As Figuras 3.17.a a 3.17.b mostram os resultados da malha fechada para a perturbação do tipo regulador, decorrente da perturbação da temperatura de refrigeração do reator, Equação 3.9. Apesar desta perturbação gerar resposta inversa em $T(3)$, observa-se um controle apropriado para as duas malhas de controle.

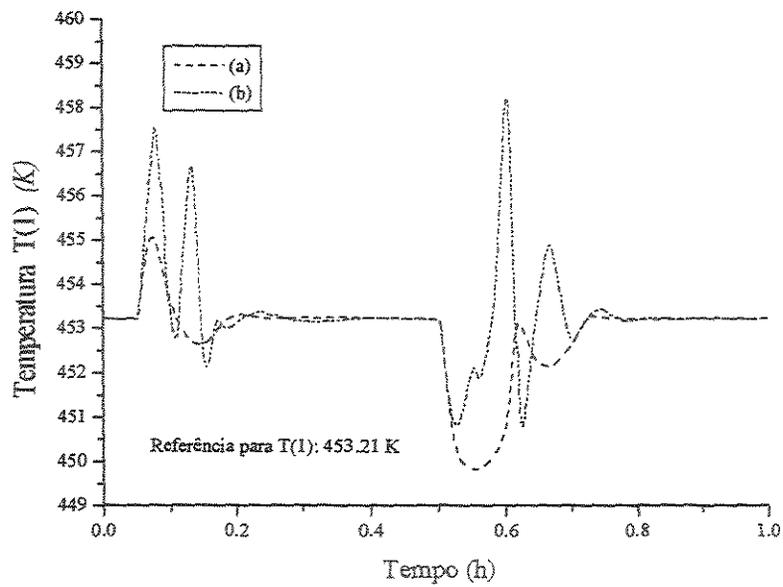


Figura 3.17.a - Problema Regulador – Perturbação em T_{R0} . (a) Controle Preditivo com restrições, $IAE(1)=0.4628$; (b) DMC auto-ajustável, $IAE(1)=0.5182$.

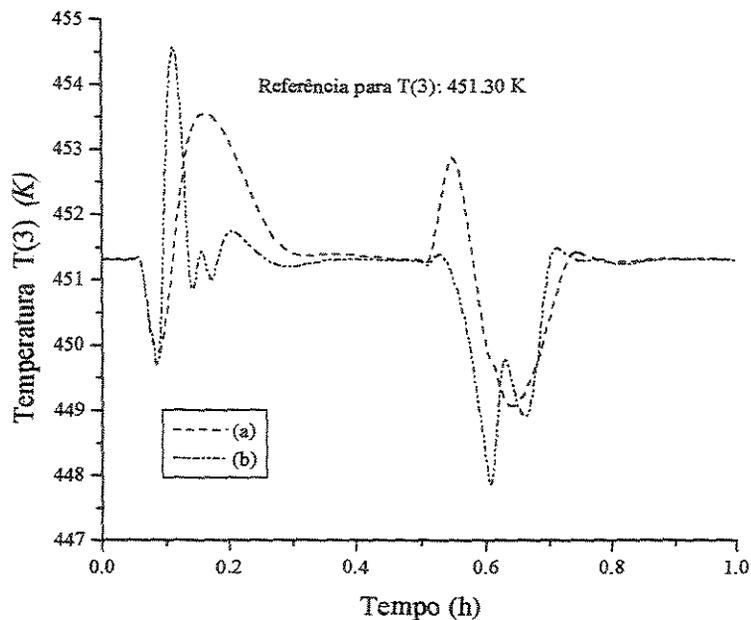


Figura 3.17.b - Problema Regulador – Perturbação em T_{R0} . (a) Controle Preditivo com restrições, $IAE(2)=0.5681$; (b) DMC auto-ajustável, $IAE(2)=0.4339$.

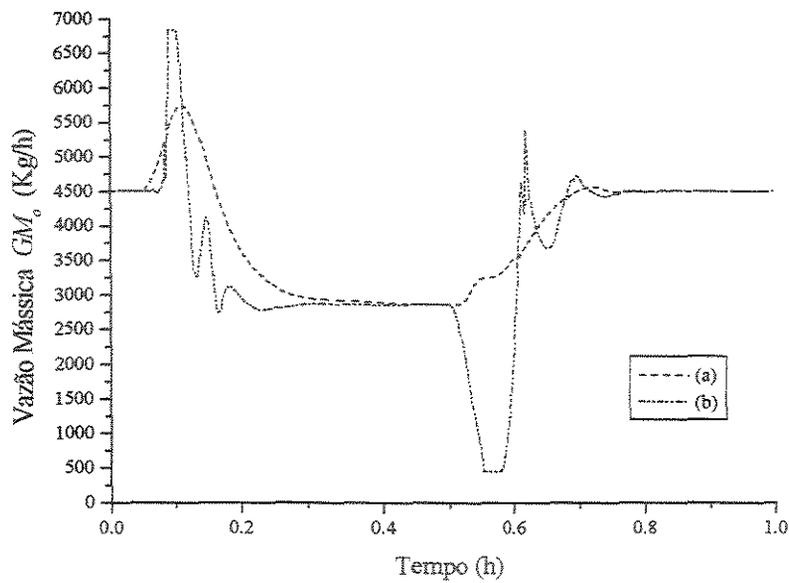


Figura 3.17.c – Ação de controle para problema regulador – Perturbação em T_{R0} . (a) Controle Preditivo com restrições; (b) DMC auto-ajustável.

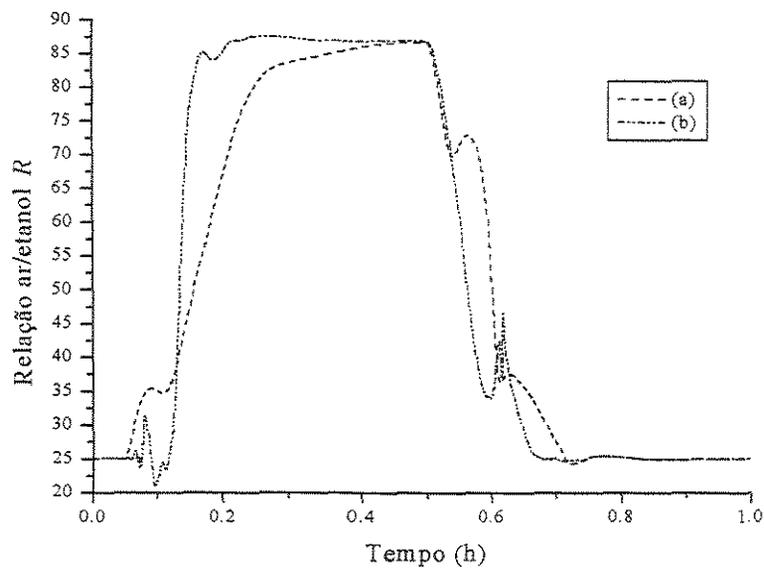


Figura 3.17.d – Ação de controle para problema regulador – Perturbação em T_{R0} . (a) Controle Preditivo com restrições; (b) DMC auto-ajustável.

3.5 Conclusões

Diante dos resultados alcançados nas formas apresentadas no uso de redes neurais em estratégias de controle preditivo, pode-se afirmar que as ANN's apresentam potencial na representação do comportamento dinâmico do processo para uso em estratégias de controle preditivo.

A alteração do modelo interno da estratégia de controle preditivo por um modelo dinâmico baseado em rede neural mostrou-se eficiente para as situações analisadas. A forma utilizada na aprendizagem *on-line* da rede neural foi capaz de contornar situações não contempladas na aprendizagem inicial (*off-line*), sendo desta maneira uma alternativa eficaz no uso de estratégias de controle.

A forma incremental de apresentação dos dados à rede neural, *Caso 2* e *Caso 3*, tem-se mostrado uma alternativa interessante, pois promove a inserção da ação integral na estratégia de controle, o que evita o aparecimento de *off-set* na malha fechada, quando se estiver atuando em uma região não apropriadamente mapeada pela rede neural em situações sem aprendizagem *on-line*. Nas análises realizadas para o algoritmo modificado, observou-se, de uma forma geral, uma performance superior deste algoritmo em relação ao DMC auto-ajustável. Isto deve-se principalmente a natureza não linear das redes utilizadas, permitindo representar apropriadamente o processo

A estratégia de controle preditivo usando restrições tem-se mostrado efetiva nas análises realizadas. Apesar da necessidade de um procedimento de otimização *on-line*, juntamente com um procedimento de aprendizagem das ANN's, isto não mostrou-se um fator limitante em termos do tempo necessário para executar estas tarefas.

Neste capítulo procurou-se abordar tópicos onde a rede neural é usada na representação do comportamento dinâmico do processo a ser controlado. Nos capítulos posteriores serão explorados estratégias de controle totalmente baseadas em redes neurais.

CAPÍTULO 4

REDES NEURAS ATUANDO COMO CONTROLADORES

4.1 Introdução

A habilidade das redes neurais para representar um mapeamento de dados e gerar modelos não lineares de sistemas é a principal característica explorada na síntese de controladores não lineares (HUNT et al., 1992). Segundo BAUGHMAN & LIU (1995), o uso de redes em controle de processos tem-se mostrado a mais difundida aplicação em engenharia química.

Nas aplicações em controle de processos, as redes neurais podem ser incorporadas na estratégia de controle através de métodos diretos e indiretos, já definidos no Capítulo 2. Entende-se, nesta tese, por método direto, sempre que a rede neural atua como controlador, seja representando a dinâmica inversa ou representando a estrutura de controle *feedback*. Por modelo indireto, considera-se as redes neurais usadas em modelos de predição de estados ou dinâmicas futuras de processos químicos.

A literatura apresenta várias configurações usando redes neurais como controladores e modelos dinâmicos a serem usados em estratégias de controle. PSALTIS et al. (1988) propuseram diversas arquiteturas de aprendizagem de redes neurais, com o

intuito de serem aplicadas em estratégias de controle. BHAT & McAVOY (1990), apresentam as redes neurais na identificação de processos e sua aplicação em estratégias de controle preditivo, conforme descrito no capítulo anterior, bem como a utilização de redes como controladores, em uma estrutura IMC empregando modelo inverso e modelo normal. Entende-se por modelo normal um modelo do processo, do tipo ARMA. PSICHOGIOS & UNGAR (1991) discutem em seu trabalho o uso de redes neurais como modelos diretos e indiretos em várias estruturas de controle. HUNT et al. (1992) apresentaram, para a época, uma excelente revisão dos principais aspectos abordando o uso de redes neurais. ICHIKAWA & SAWA (1992) propuseram uma estratégia de controle neural baseada na estrutura *feedback*, onde é proposta uma modificação do algoritmo genético para acomodar um procedimento de aprendizagem das redes neurais. Vários autores desenvolveram trabalhos aplicando as redes neurais em estratégias de controle adaptativas. Dentre estes podemos citar NARENDRA & MUKHOPADHYAY (1994), BOŠKOVIĆ & NARENDRA (1995), NARAYANAN et al. (1997), KAVCHAK & BUDMAN (1999), ZHANG & HANG (1999), KRISHNAPURA & JUTAN (2000), MELEIRO & MACIEL FILHO (2000). As redes neurais também tem sido aplicadas na avaliação de interações em processo MIMO, de acordo com o apresentado por OHBA & ISHIDA (1998).

As redes neurais multicamadas *feedforward* representam uma forma especial de modelo coneccionista que realiza um mapeamento a partir de uma região de entradas para uma região de saídas. As redes consistem em elementos de processamento simples altamente interconectados arranjados em camadas estruturadas; a intensidade de cada conexão é caracterizada pelos seus pesos. Os neurônios da camada de entrada estão conectados com os neurônios da camada de saída através dos neurônios das camadas internas. O processamento das informações em cada neurônio é efetuado através de sua função de ativação. Quando a função de ativação for não linear, a rede será não linear (PSICHOGIOS & UNGAR, 1991).

A arquitetura *feedforward* constitui provavelmente a arquitetura mais utilizada devido sua capacidade de aproximar funções complexas. Esta técnica permite o desenvolvimento de controladores baseados em rede neurais, mas isto requer o auxílio de computador para a geração das ações de controle, bem como tomar a decisão apropriada. Em geral, as estratégias de controle avançado tem os parâmetros do controlador

determinados *off-line*, o que significa que nestes casos a ação de controle é baseada em uma análise a qual pode não refletir as variações do processo, e obviamente as ações de controle não serão as melhores. Exceções são os algoritmos *self-tuning* que ainda não tem sido intensamente usados nas unidades industriais, talvez devido a sua complexidade inerente. Dentro deste contexto, torna-se importante a implementação de estratégias de controle que permitem adaptação de parâmetros e modelos ao longo do tempo.

Este capítulo aborda as redes neurais sendo utilizadas como controladores, ou seja, usadas como modelos diretos. Em algumas estratégias propostas as duas formas (modelo direto e indireto) são usadas simultaneamente. Estratégias diferenciadas de abordar a questão da aprendizagem *on-line* são propostas e discutidas, e os resultados alcançados mostraram-se bastante promissores.

4.2 A Rede Neural

O desenvolvimento das estratégias de controle propostas estão baseadas na arquitetura *feedforward* com aprendizagem *backpropagation*. Esta arquitetura tem-se mostrado bastante robusta quando comparada com outros possíveis arranjos e sua habilidade em aproximar funções complexas, bem como dados com ruídos, torna-a uma ferramenta muito interessante (BULSARI, 1995).

Em todas as configurações de redes neurais usadas, considerou-se a aprendizagem *on-line* com dados provenientes do processo simulado (ENDER & MACIEL FILHO, 2001^{a,b}), conforme descrito no Capítulo 2. Dados históricos de perturbações/resposta foram usados para treinar o modelo representando a dinâmica inversa e dados históricos de perturbações/erros da malha fechada foram utilizados para treinar a estrutura *feedback*.

4.3 Estratégia de controle usando Modelo Inverso

Neste tópico será feita uma abordagem usando redes neurais como controlador baseado no modelo inverso, onde a aprendizagem destas redes é realizada de acordo com a arquitetura de aprendizagem indireta proposto por PSALTIS et al. (1988). No trabalho citado, são indicados alguns problemas desta técnica no que diz respeito em eliminar o erro da malha fechada, os quais também foram verificados nos resultados apresentados nesta tese. São analisados duas estratégias distintas em se abordar um problema multivariável, ou seja a estratégia MIMO e a MultiSISO (ENDER & MACIEL FILHO, 1999^c, 2000^b).

4.3.1 Fundamentos

Um processo pode ser descrito na seguinte forma discreta (NARENDRA & PARTHASARATHY, 1990; CHEN & BILLINGS, 1992; NAHAS et al., 1992; SU et al., 1992; BULSARI, 1995):

$$Y(k+1) = f(Y(k), \dots, Y(k-n+1), U(k), \dots, U(k-m+1)) \quad (4.1)$$

com:

$$\begin{aligned} Y(k) &= [y(1, k), \dots, y(i, k)]^T \\ U(k) &= [u(1, k), \dots, u(i, k)]^T \quad i = \text{número de malhas de controle} \end{aligned} \quad (4.2)$$

onde $U(k)$ e $Y(k)$ ($k = 0, 1, \dots$) são os vetores de perturbações e respostas no instante de tempo k , respectivamente, e $f(\bullet)$ é uma função desconhecida. Considerando o desenvolvimento de uma função aproximada para $f(\bullet)$ pelo uso de redes neurais, onde $Y(k+1)$ e $Y(k), \dots, Y(k-n+1), U(k), \dots, U(k-m+1)$ são respostas e perturbações da rede

neural respectivamente, e W são os pesos das conexões e os limiares da rede neural. Esta rede neural pode gerar um modelo do processo, pelo ajuste de W , denominado modelo de controle indireto, como mostrado a seguir:

$$\hat{Y}(k+1) = \hat{f}(Y(k), \dots, Y(k-n+1), U(k), \dots, U(k-m+1); W) \quad (4.3)$$

Resolvendo a equação (4.1) em relação a $U(k)$, obtém-se:

$$U(k) = \phi(Y(k+1), Y(k), \dots, Y(k-n+1), U(k-1), \dots, U(k-m+1)) \quad (4.4)$$

onde $\phi(\bullet)$ é uma função desconhecida, que representa a dinâmica inversa do processo. Como a função $\phi(\bullet)$ é desconhecida, bem como a função $f(\bullet)$, não é possível avaliar a equação (4.4). Neste caso utiliza-se uma rede neural para gerar uma aproximação da função ϕ , da seguinte maneira:

$$\hat{U}(k) = \hat{\phi}(Y(k+1), Y(k), \dots, Y(k-n+1), U(k-1), \dots, U(k-m+1); W) \quad (4.5)$$

Considere um problema de controle em que a resposta $Y(k)$ procura acompanhar as referências da malha fechada, $Y_r(k)$. Então:

$$Y(k+1) = Y_r(k+1), \quad (4.6)$$

é desejado para o próximo tempo de amostragem. Substituindo a Equação (4.6) na Equação (4.5), a ação de controle desejada para o processo é dada por:

$$\hat{U}(k) = \hat{\phi}(Y_r(k+1), Y(k), \dots, Y(k-n+1), U(k-1), \dots, U(k-m+1); W) \quad (4.7)$$

Este modelo representa o mapeamento de perturbações/respostas não lineares através de redes neurais, denominado de modelo de controle direto, que terá suficiente exatidão se um grande número de dados estiver disponível (PSICHOGIOS & UNGAR, 1991; BULSARI, 1995). As entradas desta rede são formadas pelas referências das variáveis manipuladas para o próximo tempo de amostragem; variáveis manipuladas e controladas passadas e como resposta da rede, obtém-se as ações de controle para o próximo tempo de amostragem, conforme mostrado na Figura 4.1, onde λ representa o número de informações passadas do processo usadas como entrada na rede.

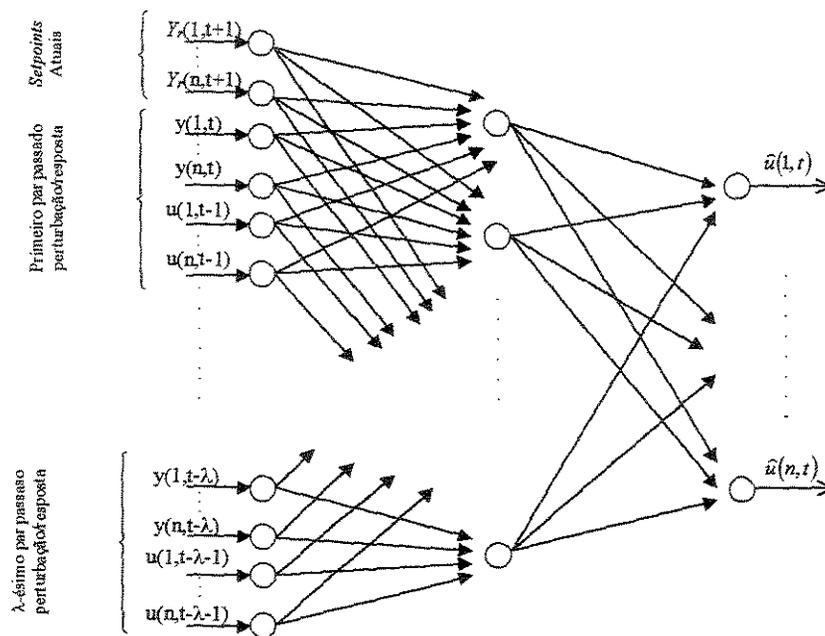


Figura 4.1 – Arquitetura *feedforward* multivariável – Modelo Inverso

A rede neural representando o modelo inverso é treinada numa primeira etapa de uma forma *off-line*, com um conjunto de dados históricos de perturbações/respostas do processo para gerar um conjunto de pesos para a partida do sistema de controle, uma vez

que posteriormente a rede neural é submetida a aprendizagem *on-line*. No modelo inverso, as entradas da rede neural são as respostas do processo e as respostas da rede neural são as entradas do processo.

O uso de controle neural baseado em modelo inverso está mostrado Figura 4.2, onde o objetivo da malha fechada é conduzir a variável controlada para a referência desejada do processo.

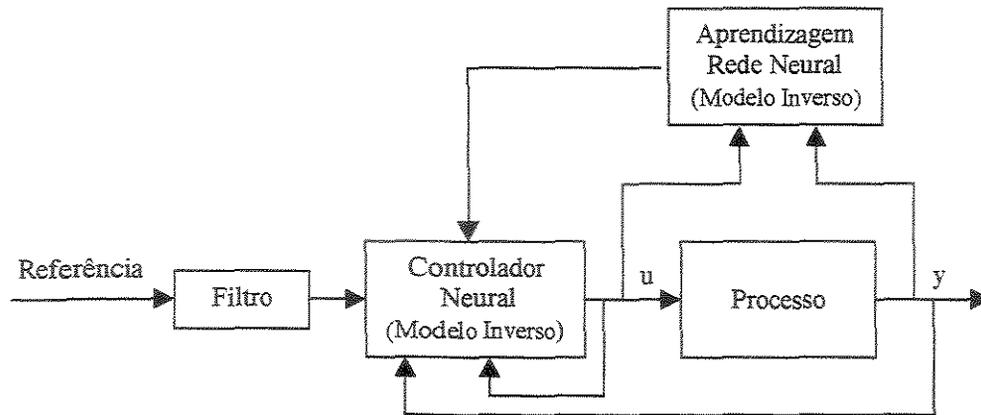


Figura 4.2 – Estratégia de controle usando modelo inverso do processo

Um filtro de primeira ordem é utilizado para evitar mudança brusca na referência do processo (*setpoint*), gerando desta forma uma trajetória de referências para o processo, similar a um modelo de primeira ordem. Também é realizada uma penalização nas ações de controle geradas para evitar variações enérgicas nas variáveis manipuladas, o que é uma característica desta técnica de controle, principalmente quando tem-se aprendizagem *on-line* da rede neural. A Equação (4.8) apresenta a equação de penalização usada, onde $\lambda(i)$ é o *fator de penalização*, para $0 \leq \lambda(i) < 1$. Valores de $\lambda(i)$ próximo de 0 (zero), possibilitam variações mais bruscas nas ações de controle, e valores de $\lambda(i)$ próximos da unidade, tornam o sistema mais conservativo.

$$u(i, t) = \lambda(i)u(i, t - 1) + (1 - \lambda(i))u(i, t) \quad (4.8)$$

Avaliou-se a possibilidade de tratar sistemas MIMO como n malhas SISO, denominados nesta tese de sistemas MultiSISO, usando a estratégia de controle citada. Esta forma de abordar o problema torna-se interessante quando as interações entre as malhas de controle é pequena, ou quando for apropriado tratar o problema usando n redes neurais menores, o que poderá facilitar a aprendizagem destas redes.

4.3.2 Resultados

Os resultados aqui apresentados foram gerados utilizando como processo um simulador do reator catalítico de leito fixo, descrito no Anexo 1.

Objetivando analisar as potencialidades destas técnicas de controle, foram geradas variações nas referências e em parâmetros do processo. Nos casos analisados, utilizou-se como variáveis manipuladas o fluxo mássico de alimentação (GM_o) e a relação ar/etanol (R); como variáveis controladas as temperaturas do primeiro, $T(1)$, e terceiro, $T(3)$, pontos de colocação axial do reator, o que é compatível com a estrutura de controle de sistemas industriais em operação.

Tabela 4.1 – Desvios nas referências do processo – Reator Catalítico

Tempo (hora)	Temperatura T(1) (K)	Temperatura T(3) (K)	Desvio em T(1)	Desvio em T(3)
0.00 – 0.20	449.71	451.30	-3.5	0.0
0.20 – 0.60	449.71	450.30	-3.5	-1.0
0.60 – 0.80	449.71	451.30	-3.5	0.0
0.80 – 1.00	453.21	451.30	0.0	0.0

As Figuras 4.3.a a 4.3.c apresentam os resultados comparativos das estratégias de controle usando modelo inverso do processo baseado em redes neurais como controlador, considerando o problema servo apresentado na Tabela 4.1.

Observa-se na Figura 4.3.a para o caso de sistema MIMO sem aprendizagem *on-line*, a presença de *off-set* na resposta da malha fechada. Devido a forma não incremental que as informações são apresentadas à rede neural, sua eficiência está diretamente relacionada a capacidade em mapear a região de interesse. Tal problema, presença de *off-set*, que também ocorre para o caso MultiSISO sem aprendizagem, é contornado realizando a aprendizagem *on-line* das redes neurais usadas.

Em sistemas de controle multivariáveis com forte acoplamento entre as malhas de controle, as estratégias de controle MultiSISO tem o desempenho prejudicado, gerando ações de controle enérgicas, a menos que o fator de penalização seja ajustado para um controle mais robusto, o que pode comprometer a velocidade de resposta da malha fechada.

As Figura 4.4.a a 4.4.c apresentam os resultados para as perturbações mostradas na Tabela 4.1, juntamente com perturbações estocásticas, as quais são adicionadas junto as respostas da malha fechada.

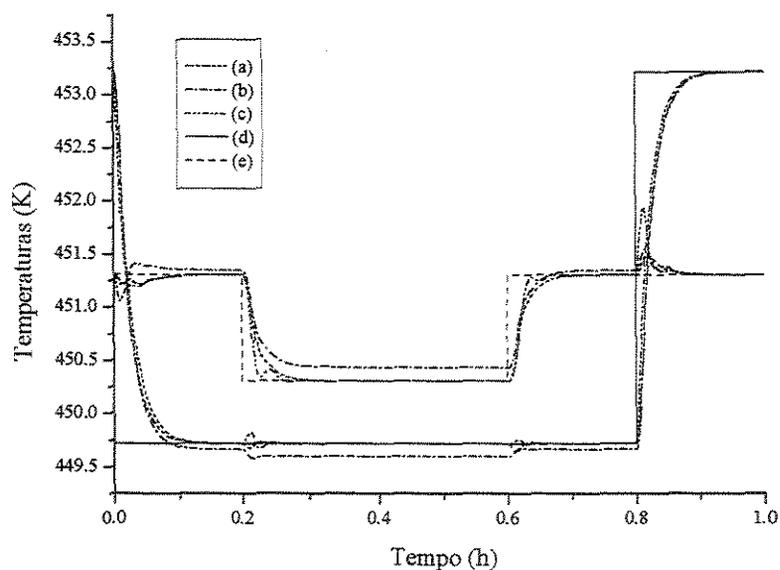


Figura 4.3.a – Respostas da malha fechada para problema servo. (a) Sistema MIMO com aprendizagem *on-line*, $IAE(1)=0.1834$, $IAE(2)=0.0614$; (b) Sistema MIMO sem aprendizagem *on-line*, $IAE(1)=0.2379$, $IAE(2)=0.1171$; (c) Sistema MultiSISO com aprendizagem *on-line*, $IAE(1)=0.1607$, $IAE(2)=0.0571$; (d) Setpoint para T(1); (e) Setpoint para T(3).

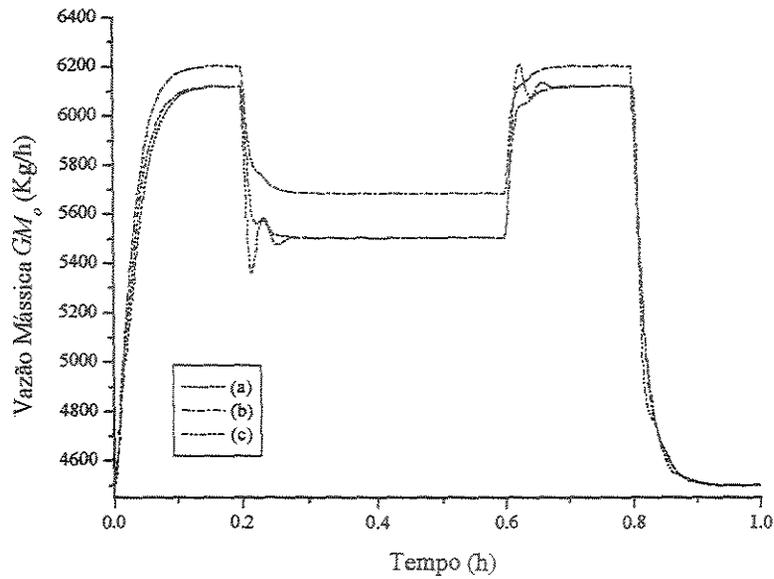


Figura 4.3.b – Ação de Controle para problema servo. (a) Sistema MIMO com aprendizagem *on-line*; (b) Sistema MIMO sem aprendizagem *on-line*; (c) Sistema MultiSISO com aprendizagem *on-line*.

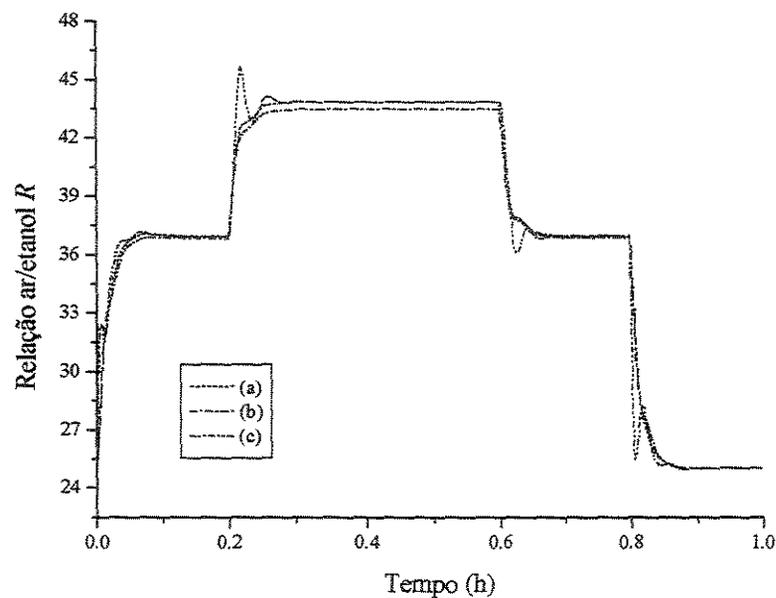


Figura 4.3.c – Ação de Controle para problema servo. (a) Sistema MIMO com aprendizagem *on-line*; (b) Sistema MIMO sem aprendizagem *on-line*; (c) Sistema MultiSISO com aprendizagem *on-line*.

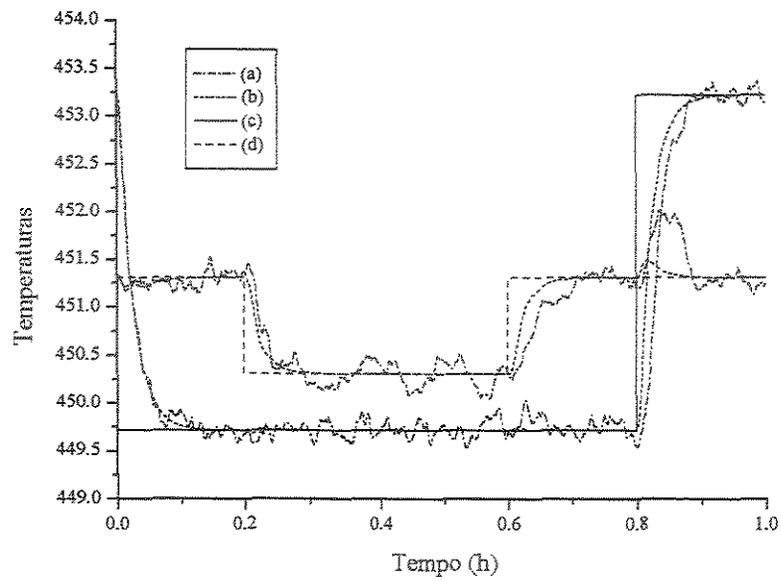


Figura 4.4.a – Resposta para a malha fechada para problema servo. (a) Sistema MIMO associado a perturbação estocástica, $IAE(1)=0.2780$, $IAE(2)=0.1794$; (b) Sistema MIMO, $IAE(1)=0.1834$, $IAE(2)=0.0614$; (c) Referência para T(1); (d) Referência para T(3).

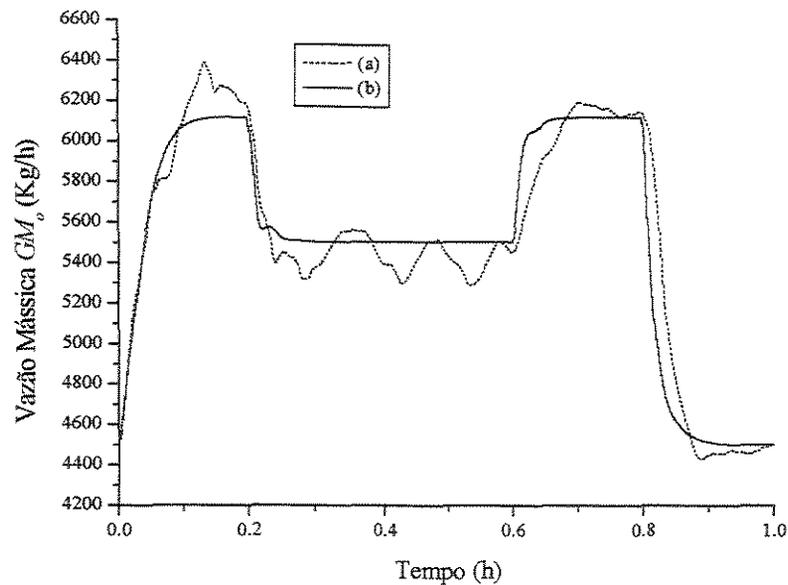


Figura 4.4.b – Ação de controle para problema servo. (a) Sistema MIMO associado a perturbação estocástica; (b) Sistema MIMO.

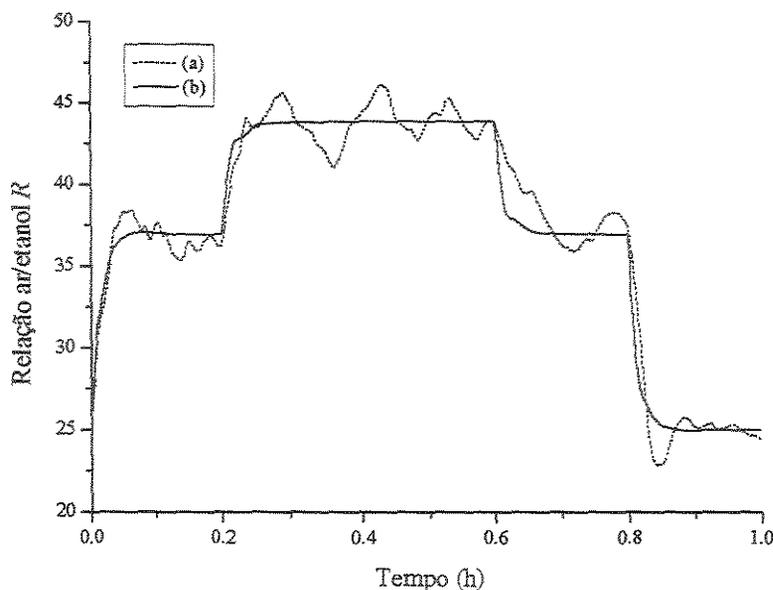


Figura 4.4.c – Ação de controle para problema servo. (a) Sistema MIMO associado a perturbação estocástica; (b) Sistema MIMO.

A Figura 4.5 apresenta o erro quadrático das redes neurais utilizadas como controladores para os sistemas MIMO analisados. De fato, o erro quadrático da rede neural considerando pesos padrão, ou seja, sem aprendizagem *on-line*, apresentam valores elevados, comparativamente aos erros quadráticos para o conjunto de pesos com aprendizagem *on-line*, o que vem comprovar a eficiência da estratégia de aprendizagem utilizada. Este fato vem explicar o *off-set* existente nas respostas em malha fechada obtidos na Figura 4.3.a. para o caso em que não utilizou-se aprendizagem *on-line*.

As Figuras 4.6.a a 4.6.d apresentam os resultados considerando uma perturbação do tipo regulador, variando bruscamente a temperatura de alimentação do reator, T_{fo} , de acordo com a relação:

$$T_{fo} = T_{foSS} + 5K ; \text{ para } 0.05 \leq t \leq 0.5 \quad (4.9)$$

Esta perturbação gera não linearidades no processo e uma resposta inversa pouca intensa em T(3). Apesar disto verifica-se um controle eficiente diante destas perturbações.

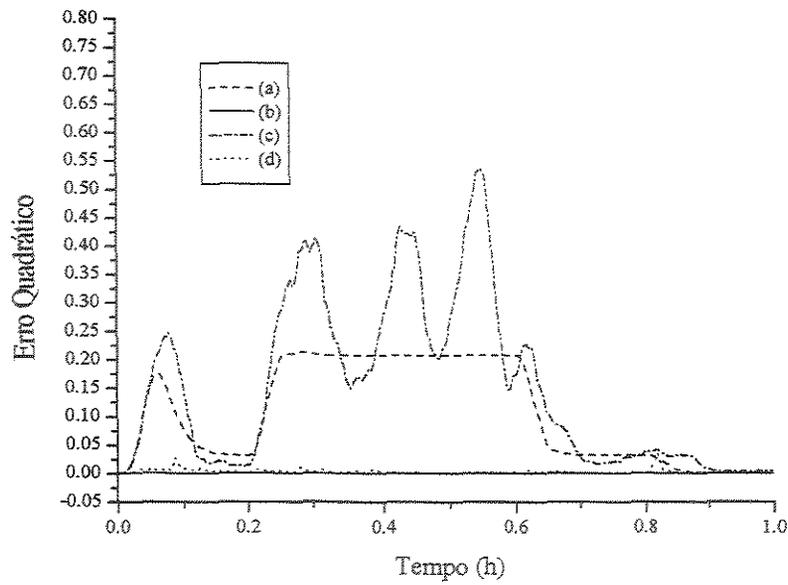


Figura 4.5 – Erro Quadrático da rede neural na representação do vetor de padrões obtidos *on-line*. (a) pesos padrão; (b) aprendizagem *on-line*; (c) pesos padrão + ruído; (d) aprendizagem *on-line* + ruído.

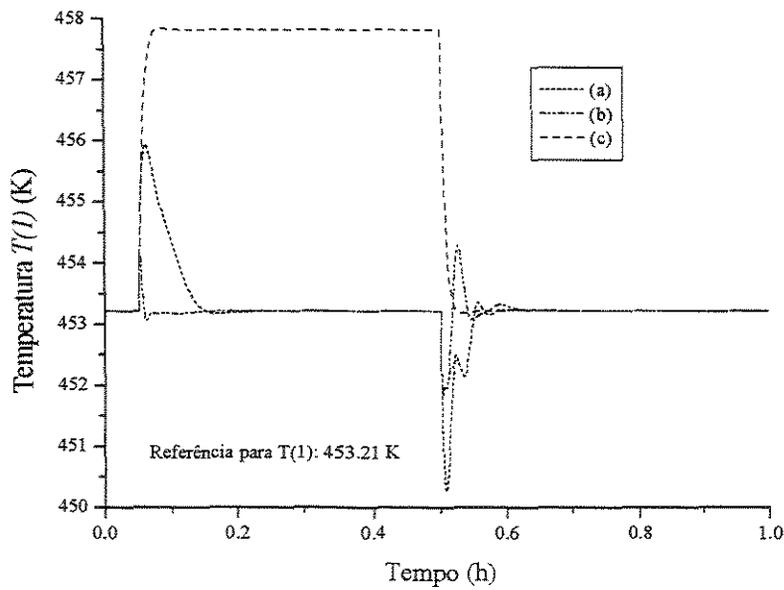


Figura 4.6.a – Resposta para a malha fechada para problema regulador, $T(1)$, pela perturbação de T_{fo} . (a) Sistema MIMO, $IAE(1)=0.1828$; (b) Sistema MultiSISO, $IAE(1)=0.0437$; (c) Resposta em Malha Aberta.

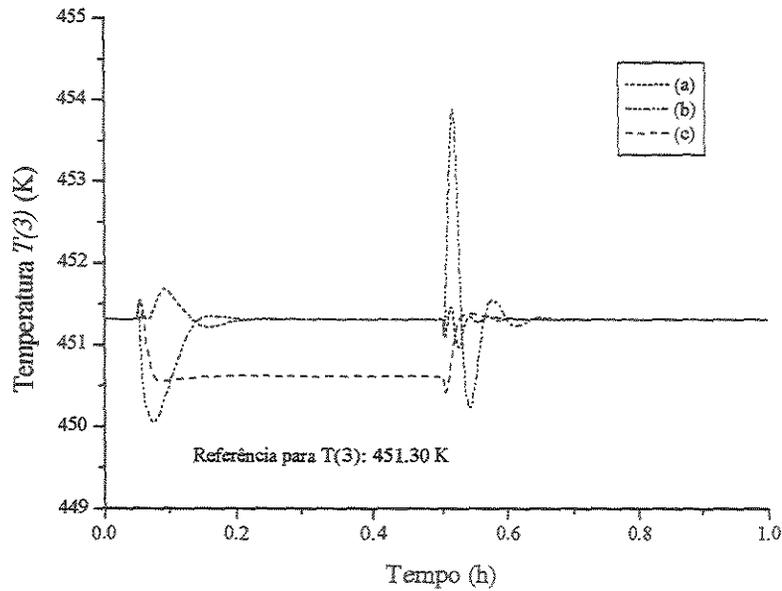


Figura 4.6.b – Resposta para a malha fechada para problema regulador, $T(3)$, pela perturbação de T_{fo} . (a) Sistema MIMO, $IAE(2)=0.0271$; (b) Sistema MultiSISO, $IAE(2)=0.1330$; (c) Resposta em Malha Aberta.

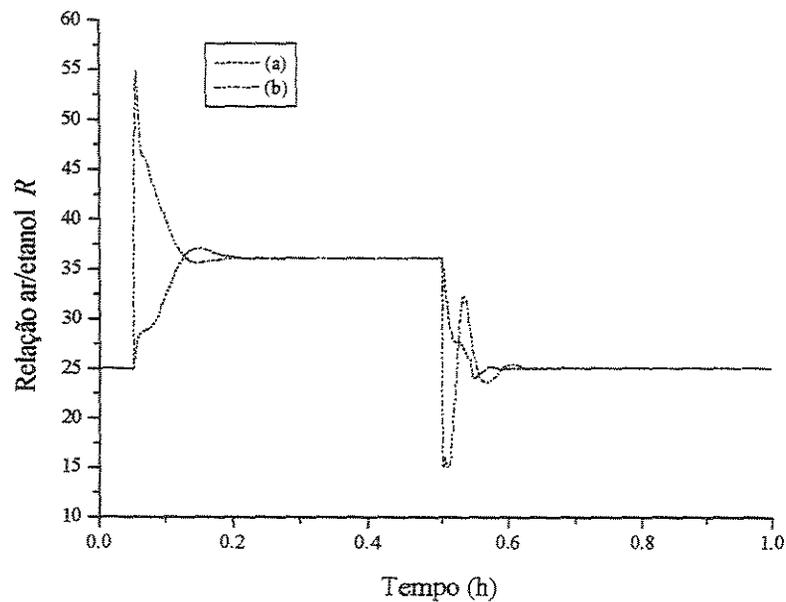


Figura 4.6.c – Ação de controle para problema regulador pela perturbação de T_{fo} . (a) Sistema MIMO; (b) Sistema MultiSISO.

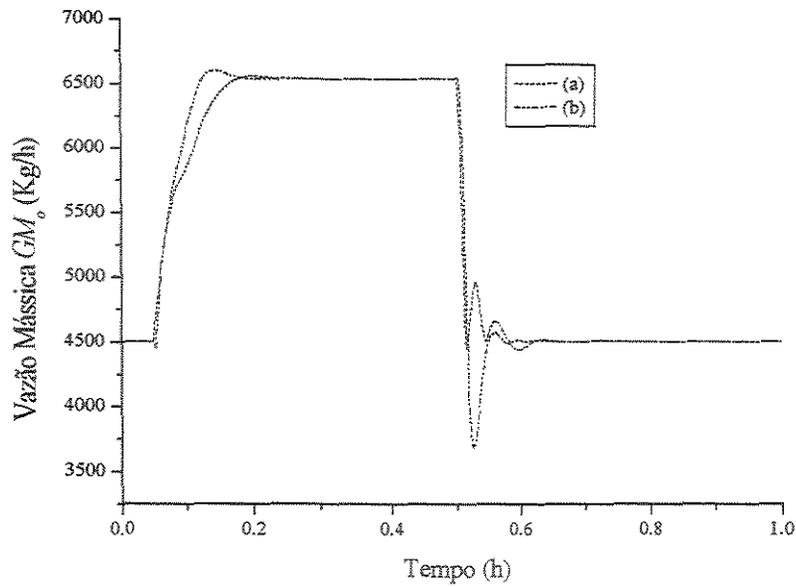


Figura 4.6.d – Ação de controle para problema regulador pela perturbação de T_{fo} . (a) Sistema MIMO; (b) Sistema MultiSISO.

Um controle efetivo não foi observado diante de perturbações em T_{RO} , pois esta perturbação gera uma resposta inversa considerável nas respostas da malha fechada, o que pode ser observado no Anexo 1. O controle baseado em modelo inverso, conforme estratégias de controle apresentada na Figura 4.2, não é adequado para este tipo de comportamento dinâmico, gerando resultados não eficientes.

4.4 Controle adaptativo baseado em redes neurais

A estratégia de controle proposta neste tópico apresenta um mecanismo diferenciado para ajustar a controlador neural, o qual esta baseado no modelo inverso. Esta estratégia de controle encontra-se esquematizada na Figura 4.7, a qual apresenta duas redes neurais que operam em conjunto. Uma é responsável na geração das ações de controle e a outra representa o comportamento dinâmico do processo, utilizada no ajuste dos pesos da primeira (ENDER & MACIEL FILHO, 2000^c, 2001^d).

Dados históricos de perturbações/resposta foram usados para treinar as duas redes neurais. A primeira rede neural é treinada a representar a dinâmica inversa do processo. As entradas desta rede são formadas pelos *setpoints* das variáveis manipuladas para o próximo tempo de amostragem; variáveis manipuladas e controladas passadas e como resposta, obtém-se as ações de controle para o próximo tempo de amostragem. A outra rede aprende a representar a dinâmica do processo. As entradas desta rede são formadas pelos valores correntes e passados das ações de controle e variáveis manipuladas; como respostas da rede neural, obtém-se as predições das respostas do processo para um tempo de amostragem futuro.

4.4.1 Fundamentos

A estratégia de controle adaptativa baseada em redes neurais é formada por um controlador neural, o qual representa a dinâmica inversa do processo (método direto) e um modelo neural do processo (método indireto) para gerar o modelo neural do processo, que é utilizado na rotina de otimização do controlador. A rede neural do controlador é otimizada *on-line*, a cada tempo de amostragem, usando o erro global estimado da malha fechada, obtido através do modelo neural do processo, na rotina de otimização. A rede neural representando o modelo dinâmico do processo também é treinada *on-line* com dados de

perturbação/resposta reais, objetivando acompanhar qualquer variação dinâmica deste processo.

O mapeamento de perturbações/respostas não lineares representado pelo método direto é descrito na equação (4.7).

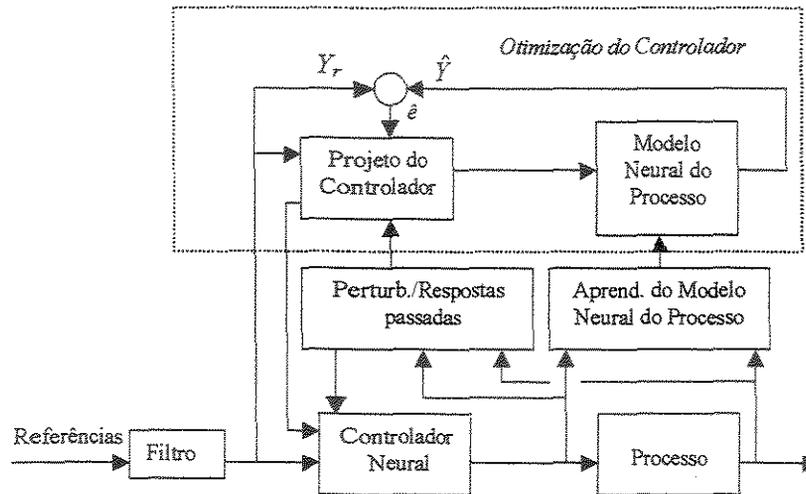


Figura 4.7 – Estratégia de Controle

O controlador é formado por uma rede neural multicamada e multivariável que representa a dinâmica inversa do sistema, com aprendizagem *on-line* realizada na rotina de otimização do controlador. A otimização do controlador consiste em ajustar os pesos das conexões entre os neurônios das redes neurais, usando o erro global estimado da malha fechada. Este erro global estimado, $\hat{e}(k)$, é obtido utilizando o modelo neural do processo, de acordo com a expressão:

$$\hat{e}(k) = Y_r(k) - \hat{Y}(k) \tag{4.10}$$

com:

$$\begin{aligned} Y_r(k) &= [y_r(1), \dots, y_r(i)]^T \\ \hat{Y}(k) &= [\hat{y}(1, k), \dots, \hat{y}(i, k)]^T \end{aligned} \tag{4.11}$$

onde: i : refere-se as malhas de controle;

$y_r(i, k)$: vetor que contém as referências do processo no instante de tempo k ;

$\hat{y}(i, k)$: vetor que contém a predição obtida a partir do modelo neural do processo.

O modelo neural do processo gera uma predição para um tempo de amostragem futuro. Este modelo do processo é treinado *on-line* com dados do processo, objetivando ter-se um modelo que de fato represente a realidade. A aprendizagem *on-line* das redes neurais apresentam limitações devido ao número de iterações necessárias para atingir-se as tolerâncias estabelecidas, bem como a necessidade deste procedimento ser executado em tempo real. Para conciliar a aprendizagem *on-line* e o tempo real, limitou-se o número de iterações permitidas na aprendizagem destas redes neurais, de acordo ao exposto no Capítulo 2. Este procedimento de aprendizagem é realizado para o modelo do processo, bem como, para o controlador, a cada tempo de amostragem.

Na rotina de otimização, o projeto do controlador utiliza as mesmas entradas que serão utilizadas no controlador neural para cada instante de amostragem. Esta rotina ajusta os pesos da rede neural do controlador de tal forma a minimizar o erro global estimado, equação (4.10), onde a resposta da malha fechada é proveniente do modelo neural do processo. Este erro global é gerado inúmeras vezes, até ser atingida a tolerância permitida para o erro quadrático da rede neural ou alcançar-se o número máximo de iterações preestabelecidas. Considerando que o ajuste da rede neural do controlador está baseado em predições, torna-se fundamental usar-se um modelo que representa com fidelidade o comportamento dinâmico do processo. Por esta razão, sempre que o erro quadrático, de pelo menos uma das redes neurais do modelo do processo, não atingir a tolerância desejada, o procedimento de aprendizagem das redes neurais do controlador não será executado. Nestas circunstâncias, o modelo do processo a ser usado, será aquele que apresentar menor erro quadrático na representação do vetor de padrões e a rede do controlador a ser usada na geração das ações de controle será aquela que apresentar menor erro quadrático para a situação atual. O modelo dinâmico do processo é utilizado para a geração do erro quadrático da rede neural de controlador.

O erro global apresentado na equação (4.10), não pode ser diretamente aplicado na aprendizagem das redes neurais do controlador, pois estes estão associados às respostas do processo e não às ações de controle, as quais são as respostas da rede neural a ser treinada. Porém estes erros na resposta do processo poderão ser retropropagados, através da matriz Jacobiana do processo. Esta matriz Jacobiana pode ser determinada a partir do modelo neural pela variação de cada variável perturbação, variável manipulada, em torno do seu ponto de operação e medindo a variação nas respostas do processo. Considerando a matriz jacobiana:

$$J = \begin{pmatrix} \frac{\partial y_1}{\partial u_1} & \dots & \frac{\partial y_1}{\partial u_i} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_i}{\partial u_1} & \dots & \frac{\partial y_i}{\partial u_i} \end{pmatrix} \quad (4.12)$$

considerando que as variáveis δy_j e δu_j são os desvios da resposta estimada da malha fechada em relação a referência desejada e desvio ação de controle que gerou esta resposta estimada da malha fechada e o suposto valor da ação de controle que levaria o sistema a referência desejada, respectivamente. Então:

$$\begin{pmatrix} \delta y_1 \\ \vdots \\ \delta y_i \end{pmatrix} = \begin{pmatrix} \frac{\partial y_1}{\partial u_1} & \dots & \frac{\partial y_1}{\partial u_i} \\ \vdots & \ddots & \vdots \\ \frac{\partial y_i}{\partial u_1} & \dots & \frac{\partial y_i}{\partial u_i} \end{pmatrix} * \begin{pmatrix} \delta u_1 \\ \vdots \\ \delta u_i \end{pmatrix} \quad (4.13)$$

$$\delta U = J^{-1} * \delta Y \quad (4.14)$$

Desta forma o vetor de erros estimados da malha fechada poderá ser convertido em erros nas ações de controle, os quais serão implementados no algoritmo de aprendizagem da rede neural do controlador na rotina de otimização do controlador.

Para evitar variações bruscas nas referências do processo um filtro de primeira ordem foi utilizado de tal forma a produzir-se uma trajetória de referências, da seguinte forma:

$$y_w(i, k+1) = \alpha_i y_w(i, k) + (1 - \alpha_i) y_r(i) \quad (4.15)$$

onde $y_r(i)$ é o valor desejado para a malha de controle i . Portanto o erro estimado apresentado na equação (4.10), assumirá a seguinte forma:

$$\hat{e}(k) = Y_w(k) - \hat{Y}(k) \quad (4.16)$$

onde $Y_w(k)$ contém os vetores de trajetória de referência para cada malha de controle.

O controlador baseado em redes neurais representando a dinâmica inversa, apresenta a característica de gerar ações de controle enérgicas. Por esta razão utiliza-se um filtro na saída do controlador amenizando os incrementos nas ações de controle gerados. O filtro a ser utilizado para penalizar as ações de controle apresenta a seguinte forma:

$$u(i, k) = \lambda_i u(i, k-1) + (1 - \lambda_i) u(i, k) \quad (4.17)$$

Inicialmente, as redes neurais do controlador e processo são treinadas de forma *off-line* com dados de perturbações e respostas obtidos a partir do processo. Esta aprendizagem inicial garante que o controlador esteja hábil à gerar ações de controle com relativa exatidão e tornar a malha fechada robusta. A aprendizagem *on-line* é responsável em acompanhar as variações do processo e ajustar o controlador para uma nova realidade.

Uma forma alternativa de aprendizagem inicial do controlador neural denominada *Specialized Learning Network Training*, proposta por PSALTIS (1988), poderia ser utilizada.

4.4.2 Resultados aplicados ao reator catalítico

Para verificar as potencialidades da estratégia de controle multivariável proposta, usou-se como processo o reator catalítico de leito fixo descrito no Anexo 1. Perturbações no estado estacionário do reator foram realizadas e variações do tipo carga na temperatura do fluido refrigerante (T_{R0}), bem como, na temperatura de alimentação (T_{f0}) foram impostas. Após analisar os resultados do planejamento fatorial aplicado ao reator em questão, bem como a viabilidade de implementação em caso real, foram adotadas como variáveis manipuladas a vazão mássica de alimentação (GM_o) e a relação ar/etanol na alimentação do reator (R). As variáveis controladas utilizadas foram as temperaturas do primeiro, $T(1)$, e terceiro, $T(3)$, pontos de colocação ortogonal axial do reator.

As Figuras 4.8.a e 4.8.b apresentam as respostas da malha fechada para o problema servo mostrado na Tabela 4.1. e as Figuras 4.8.c e 4.8.d apresentam as ações de controle associadas a este problema servo. Em ambos os casos é verificado um bom comportamento da malha fechada, confirmando a eficiência da estratégia de controle proposta, para este tipo de perturbação.

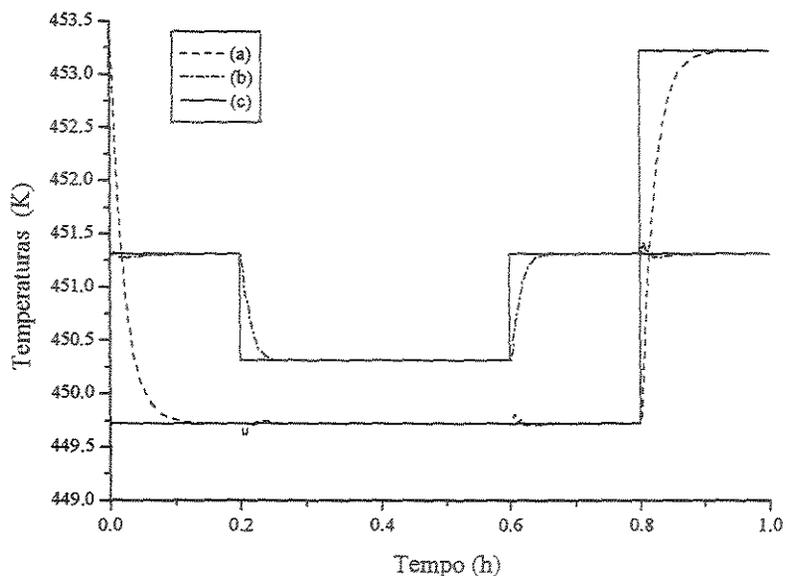


Figura 4.8.a – Resposta da malha fechada para problema servo. (a) Temperatura T(1), IAE(1)=0.1632; (b) Temperatura T(3), IAE(2)=0.0331; (c) Referências da malha fechada.

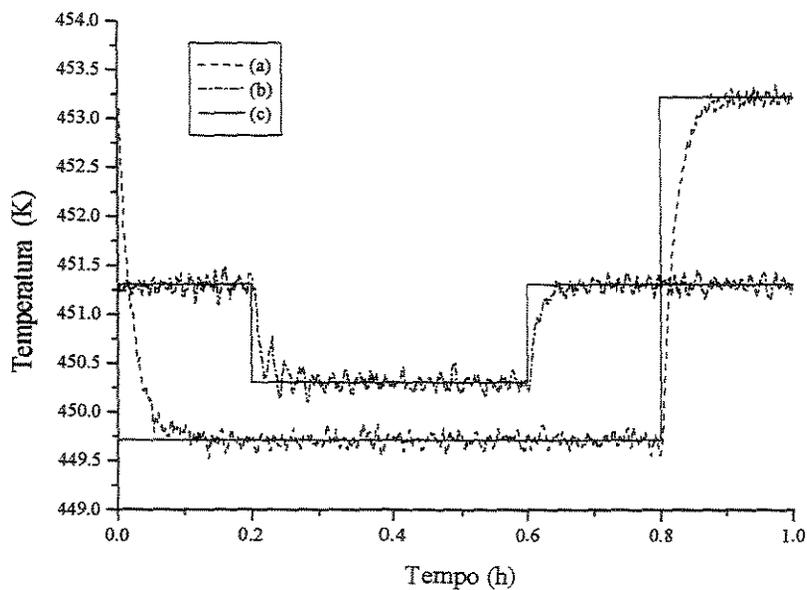


Figura 4.8.b – Resposta da malha fechada para problema servo associado a perturbação estocástica. (a) Temperatura T(1), IAE(1)=0.1962; (b) Temperatura T(3), IAE(2)=0.0844; (c) Referências da malha fechada.

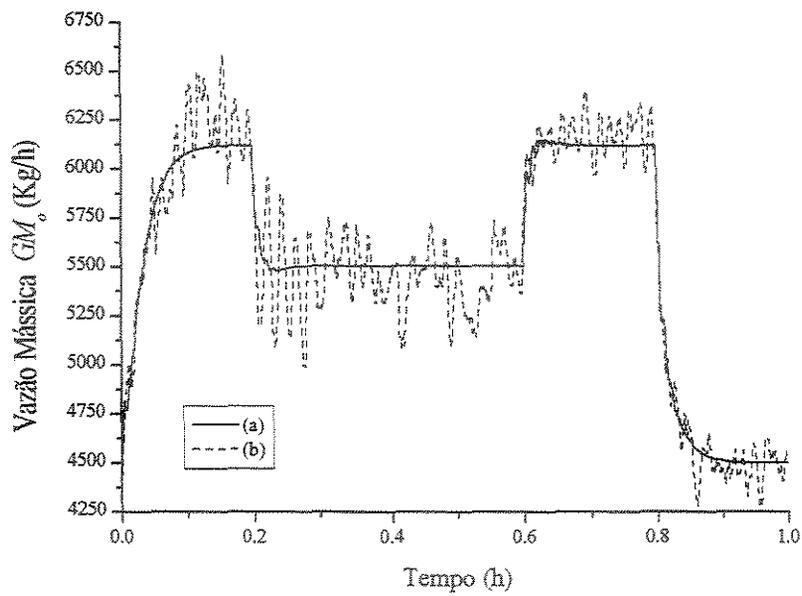


Figura 4.8.c – Ação de controle para problema servo.(a) Sistema MIMO; (b) Sistema MIMO associado a perturbação estocástica.

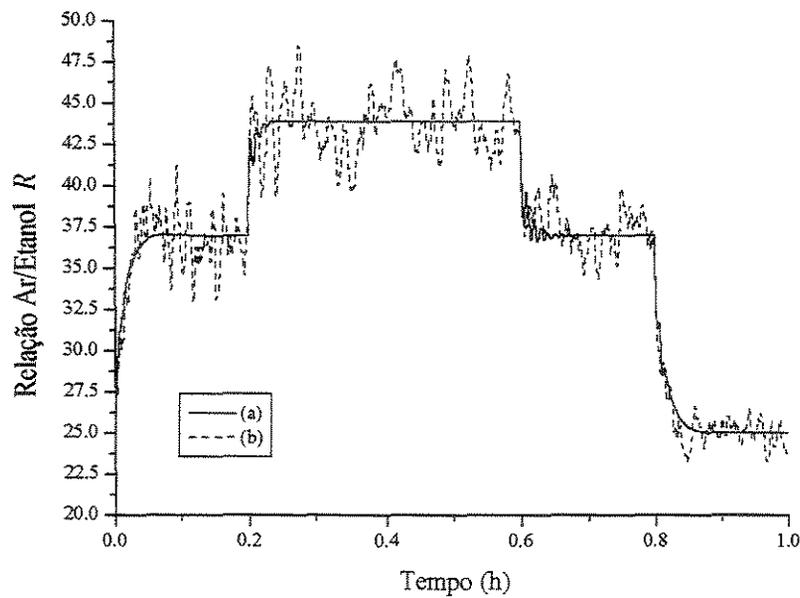


Figura 4.8.d – Ação de controle para problema servo. (a) Sistema MIMO; (b) Sistema MIMO associado a perturbação estocástica.

O modelo neural do processo apresenta como parâmetros de entrada informações passadas das variáveis manipuladas e controladas, bem como, das ações de controle atuais. Uma alternativa de inserir características *feedforward* à estratégia de controle é utilizar um modelo neural onde as principais variáveis mensuráveis do processo que apresentam forte influência na dinâmica, sejam incorporadas a este modelo, como parâmetro de entrada na rede neural. Esta alternativa apresenta dois aspectos importantes a serem considerados: (a) esta característica *feedforward* incorporada irá beneficiar a otimização do controlador, para o problema regulador, caso uma ou várias das variáveis consideradas forem perturbadas e o modelo do processo for confiável; (b) um maior número de variáveis consideradas como parâmetros da rede neural, irá aumentar consideravelmente o número de neurônios desta rede, e isto poderá dificultar a aprendizagem *on-line* do modelo do processo, o que irá prejudicar a otimização do controlador.

Os demais resultados apresentados neste caso de estudo, para esta estratégia de controle, irão considerar dois modelos para o processo conforme exposto anteriormente; um utilizando as variáveis T_{R0} e T_{fo} como parâmetros de entrada da rede, da mesma forma que as variáveis manipuladas, denominado aqui como modelo de perturbações mensuráveis, e outro modelo da mesma forma já descrito anteriormente, denominado modelo de perturbações não mensuráveis, o qual utiliza como parâmetros de entrada da rede neural somente variáveis manipuladas e controladas. Os resultados anteriores apresentados para o problema servo, consideraram o modelo de perturbações não mensuráveis para o processo.

As Figuras 4.9.a a 4.9.c apresentam os resultados para o problema regulador gerado a partir de perturbação na temperatura de alimentação do fluido refrigerante (T_{R0}), de acordo com a expressão:

$$T_{R0} = T_{R0SS} + 5K; \text{ para } 0.1 \leq t \leq 0.3 \quad (4.18)$$

onde T_{R0SS} é o valor estacionário inicial para a temperatura do fluido refrigerante.

Observa-se um excelente resultado da estratégia proposta, mesmo com a existência de resposta inversa considerável. Um comportamento ligeiramente mais robusto é

observado quando o modelo neural do processo considera o modelo de parâmetros mensuráveis, devido as características *feedforward* incorporadas na estratégia de controle, pela utilização deste modelo.

As Figuras 4.9.b e 4.9.c apresentam as ações de controle geradas pela estratégia de controle para o sistema diante, das perturbação impostas. Estas figuras também apresentam os resultados para os dois modelos neurais do processo, os quais vem demonstrar a eficiência dos dois casos apresentados, porém um comportamento ligeiramente mais robusto da malha fechada para o caso em que foi utilizado modelo neural do processo, com perturbações mensuráveis, é observado.

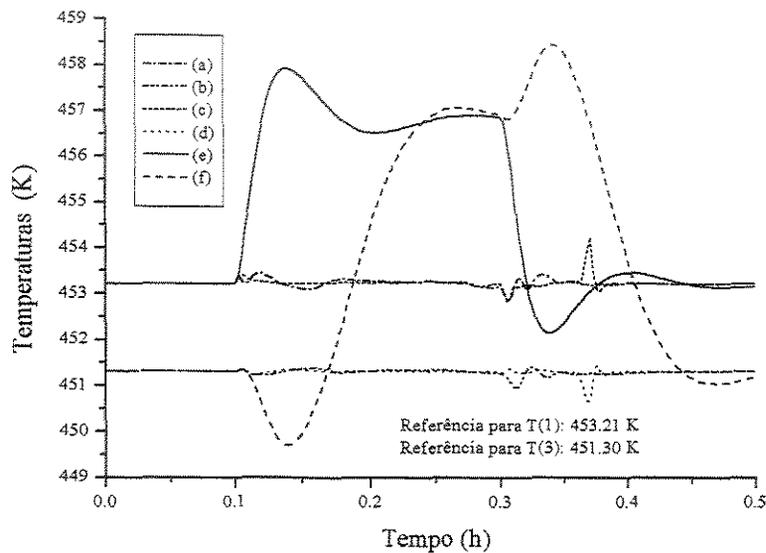


Figura 4.9.a – Problema regulador – perturbação na temperatura de alimentação do fluido refrigerante (T_{R0}). (a) Temperatura T(1) – perturbações mensuráveis, IAE(1)=0.0211; (b) Temperatura T(3) – perturbações mensuráveis, IAE(2)=0.0160; (c) Temperatura T(1) – perturbações não mensuráveis, IAE(1)=0.0175; (d) Temperatura T(3) – perturbações não mensuráveis, IAE(2)=0.0104; (e) Resposta em malha aberta para T(1); (f) Resposta em malha aberta para T(3).

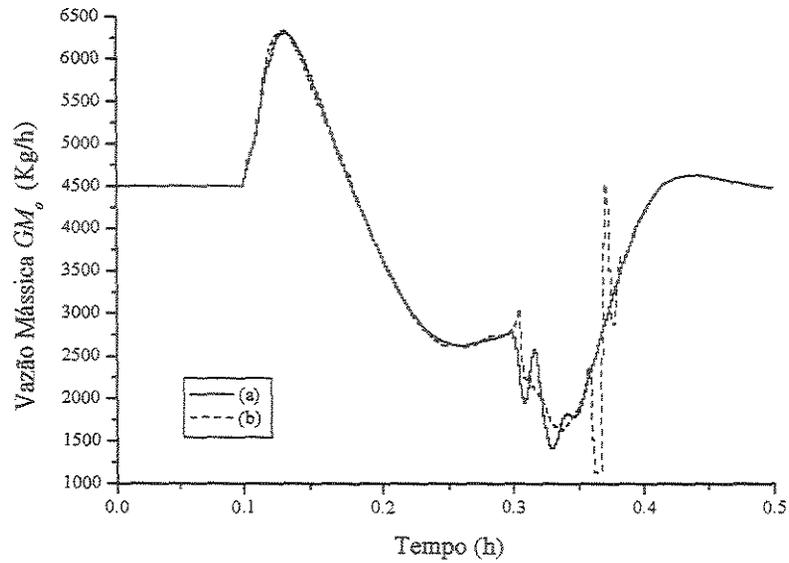


Figura 4.9.b – Problema regulador – perturbação na temperatura de alimentação do fluido refrigerante (T_{R0}). (a) Perturbações mensuráveis; (b) Perturbações não mensuráveis.

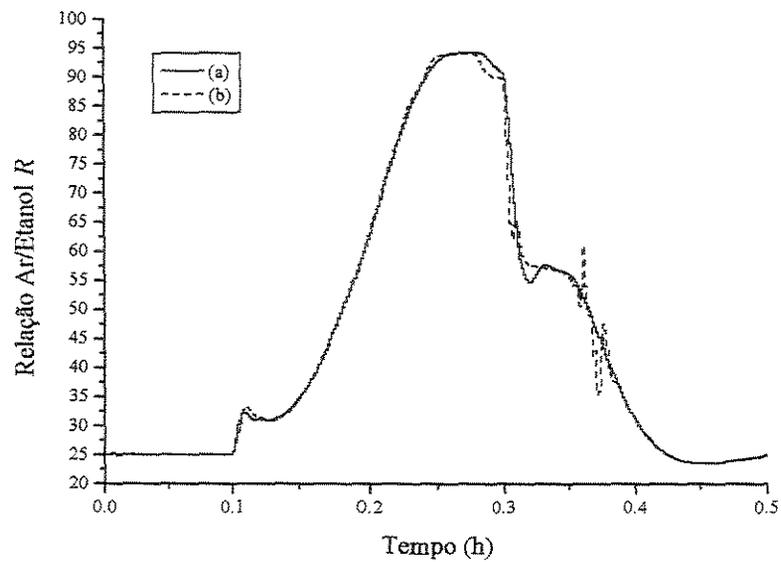


Figura 4.9.c – Problema regulador – perturbação na temperatura de alimentação do fluido refrigerante (T_{R0}). (a) Perturbações mensuráveis; (b) Perturbações não mensuráveis.

As Figuras 4.10.a a 4.10.c apresentam os resultados para problema regulador gerado a partir de perturbações na temperatura de alimentação do reator, de acordo com a expressão:

$$T_{fo} = T_{foSS} + 5 \text{ K} \quad \text{para } 0.05 \leq t \leq 0.2 \quad (4.18)$$

onde T_{foSS} refere-se a temperatura da alimentação do reator no estado estacionário inicial.

Analisando os resultados obtidos neste caso, Figuras 4.10.a a 4.10.c, e no caso anterior, Figuras 4.9.a a 4.9.c, pode-se observar que não existe um incremento significativo na qualidade de resposta para o modelo do processo que considera mensurável as perturbação tipo carga geradas. Tal fato deve-se a eficiência da aprendizagem *on-line*, a qual ajusta os pesos da rede neural de tal forma que o modelo considere esta perturbação como uma alteração na dinâmica do processo.

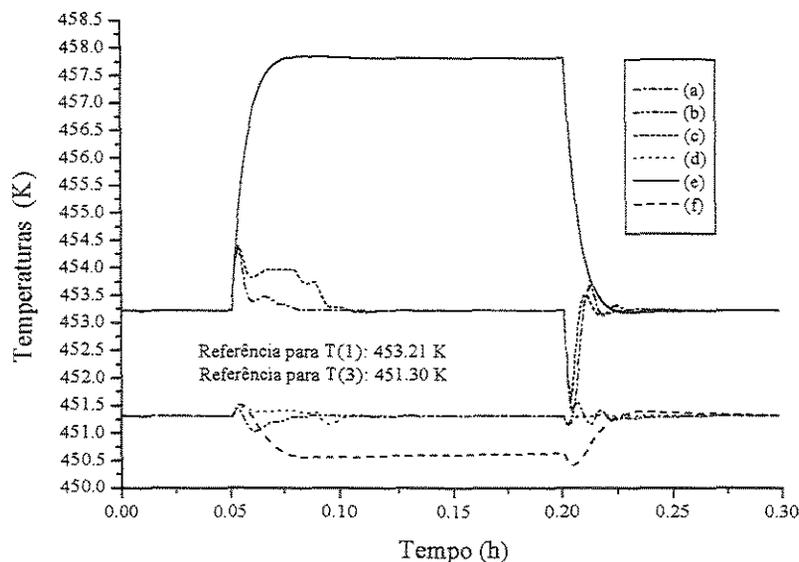


Figura 4.10.a – Problema regulador – perturbação em (T_{fo}). (a) Temperatura T(1) – perturbações mensuráveis; IAE(1)=0.0224; (b) Temperatura T(3) – perturbações mensuráveis, IAE(2)=0.0076; (c) Temperatura T(1) – perturbações não mensuráveis, IAE(1)=0.0377; (d) Temperatura T(3) – perturbações não mensuráveis, IAE(2)=0.0060; (e) Resposta em malha aberta para T(1); (f) Resposta em malha aberta para T(3).

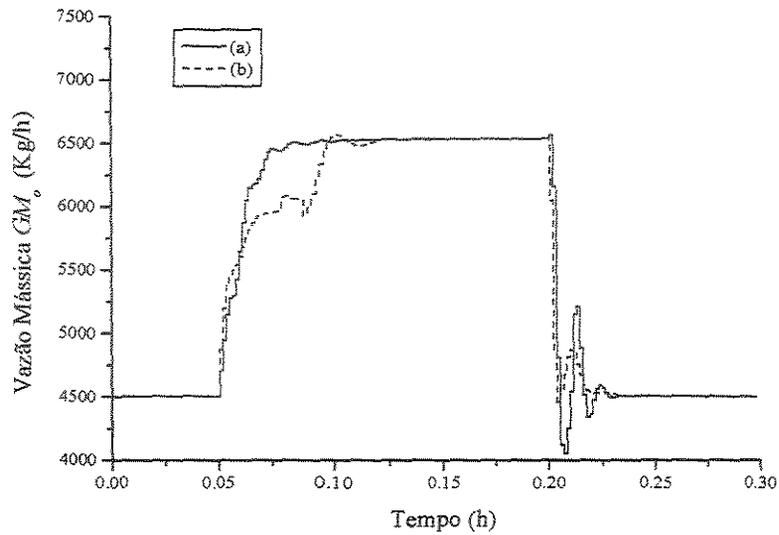


Figura 4.10.b – Problema regulador – perturbação na temperatura de alimentação do reator (T_{fo}). (a) Perturbações mensuráveis; (b) Perturbações não mensuráveis.

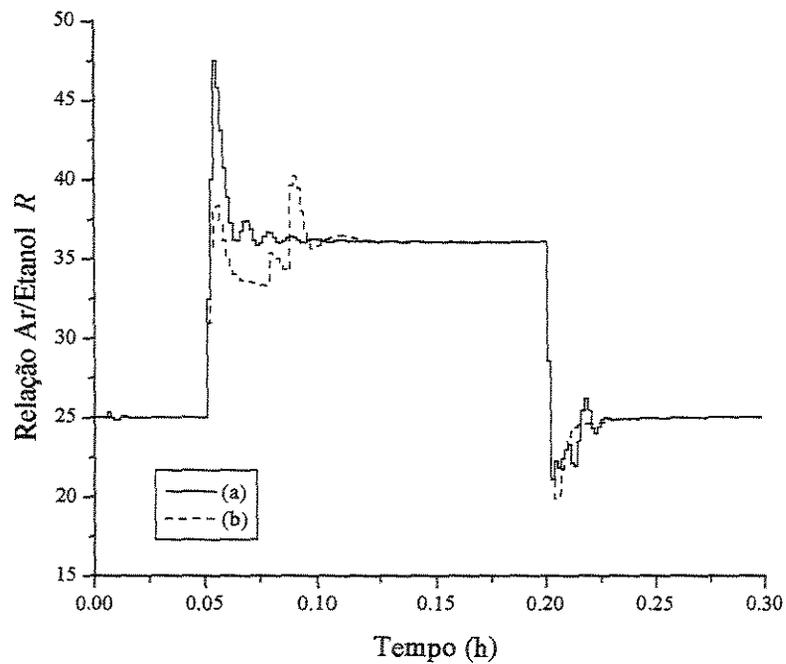


Figura 4.10.c – Problema regulador – perturbação na temperatura de alimentação do reator (T_{fo}). (a) Perturbações mensuráveis; (b) Perturbações não mensuráveis.

4.4.3 Resultados aplicados ao reator de penicilina

Objetivando analisar a performance da estratégia de controle proposta diante de um processo com dinâmica lenta, usou-se um processo fermentativo para produção de penicilina como caso de estudo (RODRIGUES, 1999). A descrição deste processo encontra-se no Anexo 2.

A estratégia de controle proposta foi implementada na forma monovariável, considerando como variável controlada, a concentração relativa de oxigênio dissolvido e como variável manipulada, a velocidade de rotação do agitador.

As Figuras 4.11.a e 4.11.b apresentam os resultados para um problema regulador complexo, onde uma série de perturbações no fluxo de alimentação de substrato é realizada. A concentração de substrato afeta a concentração de oxigênio dissolvido devido o crescimento celular e a energia requerida no metabolismo para produção de penicilina. Mesmo com as perturbações no fluxo de alimentação de substrato ocorrendo a cada 6 horas, o processo mantém-se dentro dos valores especificados.

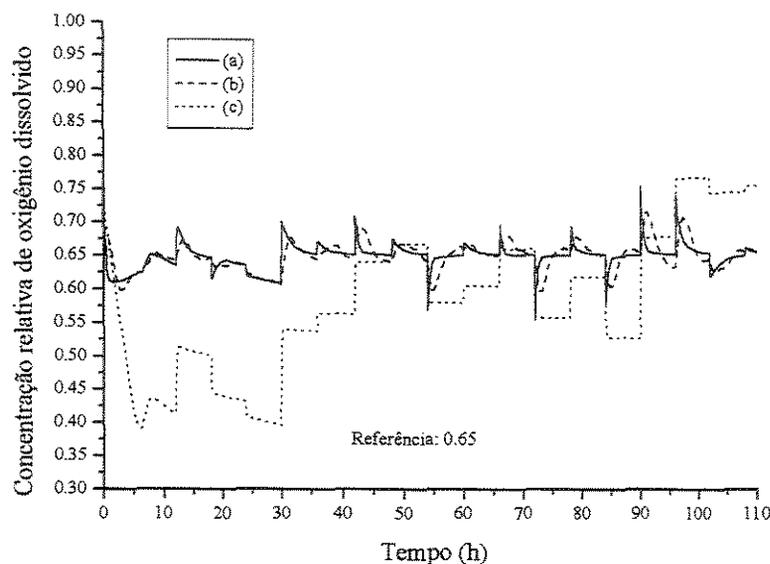


Figura 4.11.a – Resposta da malha fechada – Perturbação no fluxo de alimentação de substrato. (a) Sistema sem ruído estocástico, $IAE(1)=1.2847$; (b) Sistema com ruído estocástico, $IAE(1)=3.3798$; (c) Resposta da malha aberta.

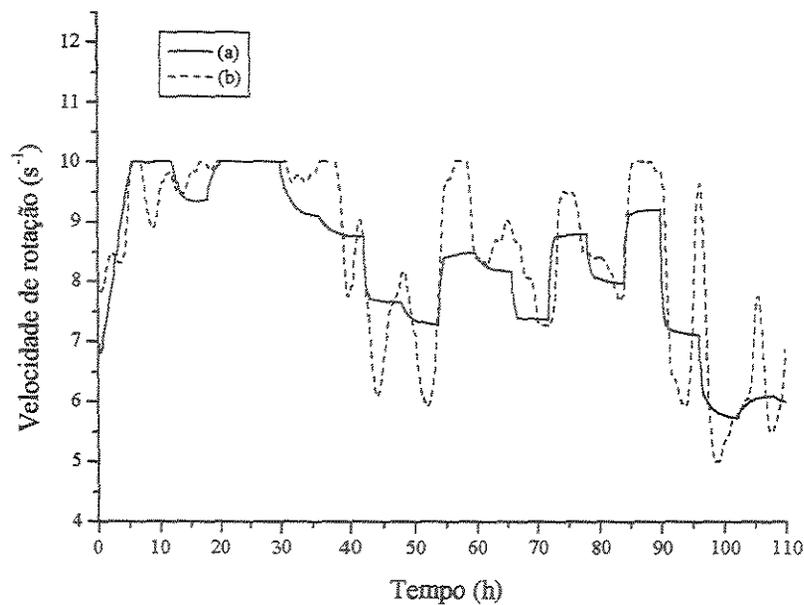


Figura 4.11.b – Ação de controle – Perturbação no fluxo de alimentação de substrato.
 (a) Sistema sem ruído estocástico; (b) Sistema com ruído estocástico.

Tabela 4.2 – Problema Servo – Processo Fermentativo

Tempo (hora)	Concentração relativa de oxigênio dissolvido
0 – 15	0.65
15 – 30	0.75
30 – 45	0.65
45 – 60	0.55
60 – 75	0.65

As Figuras 4.12.a e 4.12.b apresentam os resultados para este problema servo, onde um bom comportamento da malha fechada é observado, mesmo diante de ruído estocástico.

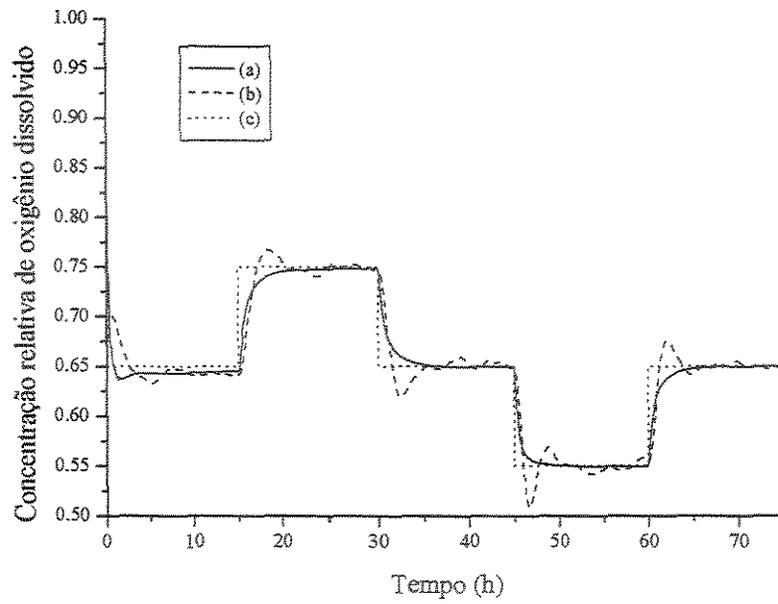


Figura 4.12.a – Resposta da malha fechada – Problema Servo, Tabela 4.2. (a) Sistema sem ruído estocástico, $IAE(1)=0.5985$; (b) Sistema com ruído estocástico, $IAE(1)=0.8653$; (c) Referência da malha fechada.

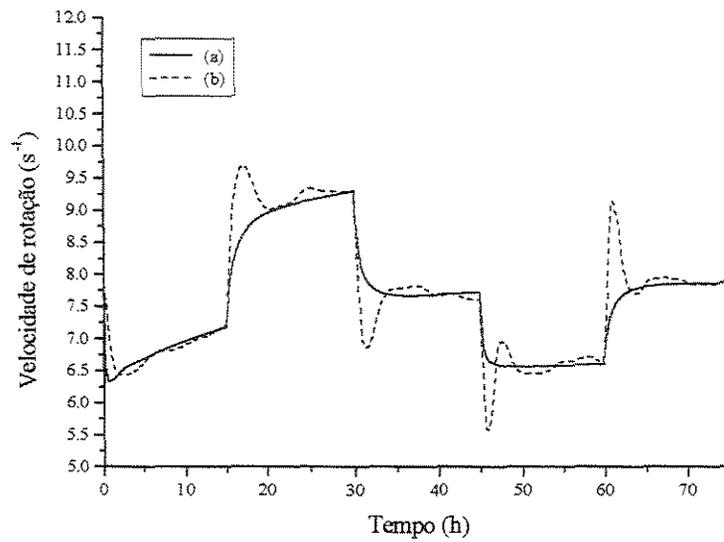


Figura 4.12.b – Ação de controle – Problema Servo. (a) Sistema sem ruído estocástico; (b) Sistema com ruído estocástico.

O desempenho da estratégia de controle mostrou-se muito boa para uma grande faixa de condições de operação. Isto vem confirmar o potencial da estratégia de controle proposta para ser implementada em processos com dinâmica complexa e não linear, mesmo em condições severas de operação. Além do mais, vem confirmar a eficiência da técnica de aprendizagem *on-line* das redes neurais propostas.

4.5 Estratégia baseada na estrutura de controle *Feedback*

4.5.1 Fundamentos

As redes neurais artificiais podem ser utilizadas na geração de ações de controle diretamente, representando a estrutura de controle *feedback* (ICHIKAWA & SAWA, 1992). Neste caso, dado um erro na resposta, ou seja, uma diferença entre a saída desejada do processo, $y_r(k)$, e a resposta da malha fechada, $y(i, k)$, o controlador *feedback* produz uma ação de controle para minimizar este erro, $e(i, k)$. O controlador neural representa a estrutura *feedback*, usando o mesmo erro como entrada da rede, de acordo com a equação (4.20):

$$\Delta \hat{U}(k) = \hat{\Phi}(E(k), \dots, E(k-n+1), \Delta U(k-1), \dots, \Delta U(k-m+1); W) \quad (4.20)$$

onde:

$$\begin{aligned} E(k) &= [e(1, k), \dots, e(i, k)] \\ \Delta U(k) &= [\Delta u(1, k), \dots, \Delta u(i, k)] \\ e(i, k) &= y_r(i, k) - y(i, k) \\ \Delta u(i, k) &= u(i, k) - u(i, k-1) \end{aligned} \quad i = \text{número de perturbações/respostas} \quad (4.21)$$

Inicialmente, o controlador neural é treinado com dados de entradas/saídas de um controlador *feedback* existente. Esta aprendizagem inicial deverá assegurar que o controlador neural seja capaz de gerar ações de controle apropriadas, para manter estável a malha fechada. A Figura 4.13 apresenta a configuração da aprendizagem *off-line* utilizada.

Esta técnica de controle pode ser implementada usando ou não a aprendizagem *on-line* da rede neural do controlador. Para o caso em que não adotar-se a aprendizagem *on-line*, a eficiência desta técnica está diretamente relacionada a capacidade da rede em mapear a faixa de operação desejada para o processo, na aprendizagem *off-line*.

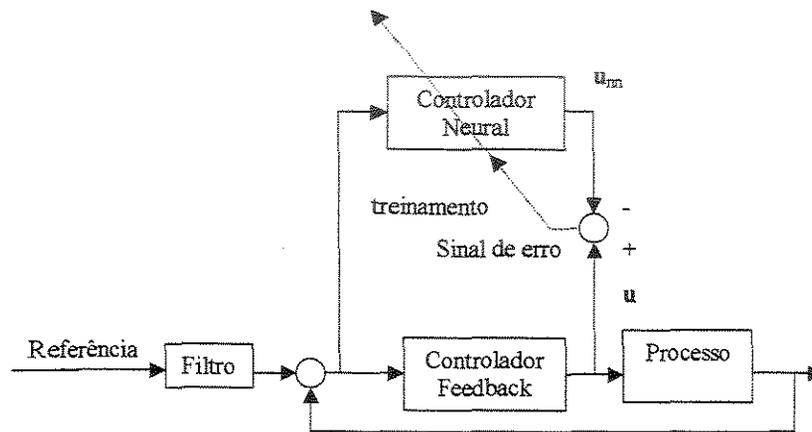


Figure 4.13 - Controle Direto (*feedback*) – aprendizagem *off-line*

4.5.1.1 Técnicas de aprendizagem *on-line* propostas

A dificuldade para realizar-se a aprendizagem *on-line* reside na definição dos padrões de aprendizagem para as ações de controle a serem usados. Dentro deste contexto, são propostas duas formas de se realizar a aprendizagem *on-line*. A primeira faz uso do modelo inverso para gerar o sinal para a aprendizagem da rede neural (ENDER & MACIEL FILHO, 1999^o), e a segunda forma faz uso de uma estrutura similar a usada na estratégia de controle adaptativa apresentada anteriormente (ENDER & MACIEL FILHO, 2000^o).

Adotando-se a aprendizagem *on-line* da estrutura *feedback*, baseada no modelo inverso, propõe-se o uso da arquitetura de aprendizagem indireta (PSALTIS, 1988), usando o sinal gerado por uma rede neural representando o modelo inverso, para compor o padrão de aprendizagem da rede neural do controlador, conforme Figura 4.14. Com isto, a eficiência desta técnica de controle estará diretamente relacionada à eficiência do modelo usado para gerar o sinal de aprendizagem. Neste caso, para a aprendizagem *on-line*, o vetor de entradas da rede neural é composto pelo erro global atual da malha fechada, e um conjunto preestabelecido de informações formadas por erros da malha fechada e incrementos nas ações de controle passados. Os padrões a serem alcançados pela rede neural do controlador são formados por ações de controle geradas por um modelo inverso do processo, com aprendizagem *on-line*.

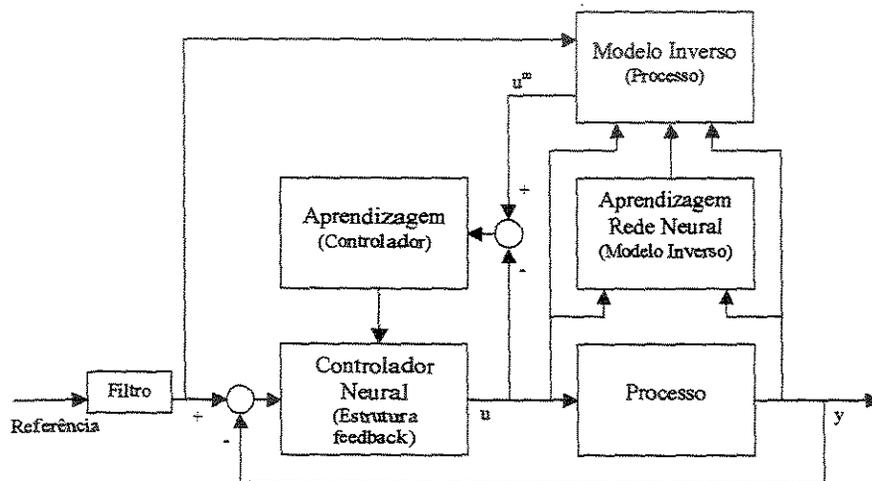


Figura 4.14 – Estratégia de controle direto (*feedback*), com aprendizagem *on-line* baseada em modelo inverso

A outra técnica de aprendizagem *on-line* proposta está apresentada na Figura 4.15. Neste caso, a cada tempo de amostragem, um procedimento de treinamento é executado, ajustando os pesos da rede neural do controlador, baseado no erro global da malha fechada. Neste procedimento de aprendizagem, o erro global é obtido a partir de um modelo neural do processo, também com aprendizagem *on-line*.

Na rotina de otimização do controlador, o erro estimado gerado $\hat{e}(k) = Y_r - \hat{Y}$, não é o valor utilizado na aprendizagem das redes neurais, pois estes erros estão associadas as respostas do processo e não a erros nas ações de controle. Utilizou-se o mesmo procedimento apresentado nas equações 4.12 a 4.14 para gerar-se o sinal de aprendizagem. O vetor de entradas da rede neural é formado pelo erro atual da malha fechada, mais um conjunto de informações passadas, predefinido, de erros da malha fechada e incrementos na ação de controle que geraram estes erros. As respostas desejadas (alvos), não são obtidas diretamente nas sim por um procedimento similar as equações 4.12 a 4.14.

Para as duas propostas apresentadas, adotou-se a técnica de aprendizagem *on-line* considerando tres redes neurais atuando paralelamente, conforme técnica descrita no Capítulo 2.

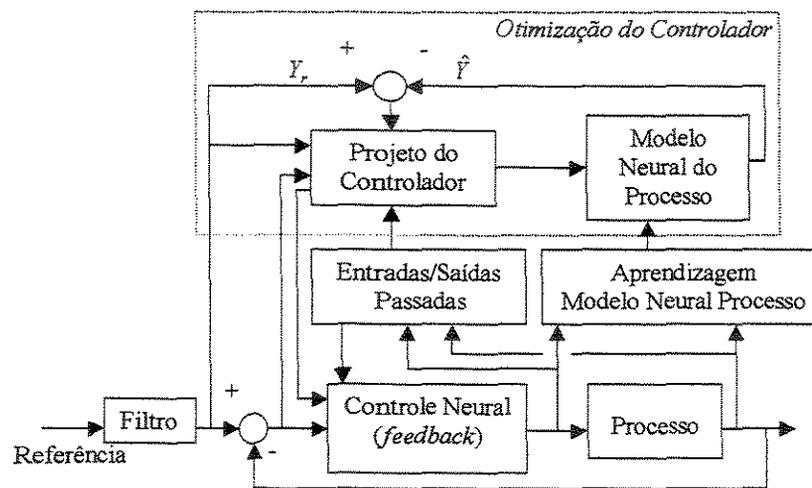


Figura 4.15 – Controle Direto usando estrutura *feedback* com aprendizagem *on-line* baseada no erro global

4.5.2 Resultados

Objetivando analisar as potencialidades destas técnicas de controle e confrontar o desempenho das estratégias de controle propostas, utilizou-se as mesmas perturbações das seções anteriores. Nos casos analisados, utilizou-se como variáveis manipuladas o fluxo mássico de alimentação (GM_o) e a relação ar/etanol (R); como variáveis controladas as temperaturas do primeiro, $T(1)$, e terceiro, $T(3)$, pontos de colocação axial do reator. As Figuras 4.16.a a 4.16.d mostram os resultados para o problema servo apresentados na Tabela 4.1.

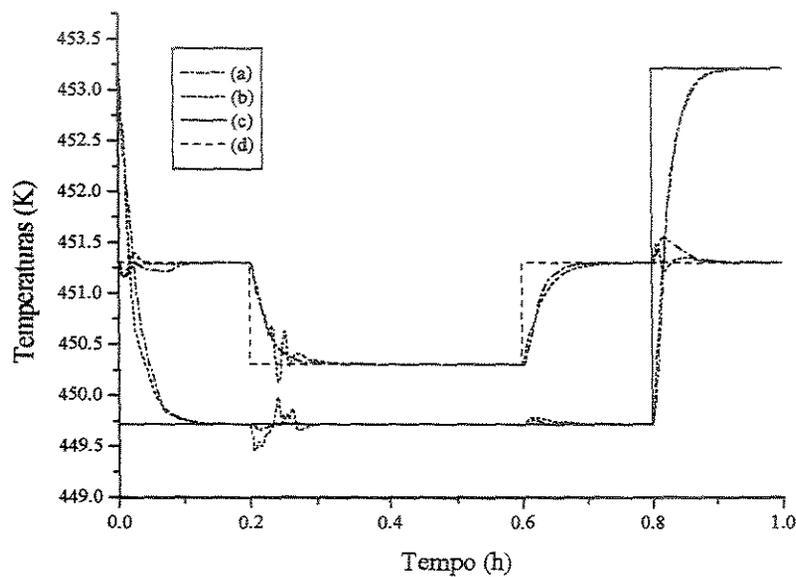


Figura 4.16.a – Resposta da malha fechada para problema servo. (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso, $IAE(1)=0.1898$, $IAE(2)=0.0683$; (b) Controle direto com aprendizagem *on-line* baseada no erro global, $IAE(1)=0.1824$, $IAE(2)=0.0665$; (c) Referência para $T(1)$; (d) Referência para $T(3)$.

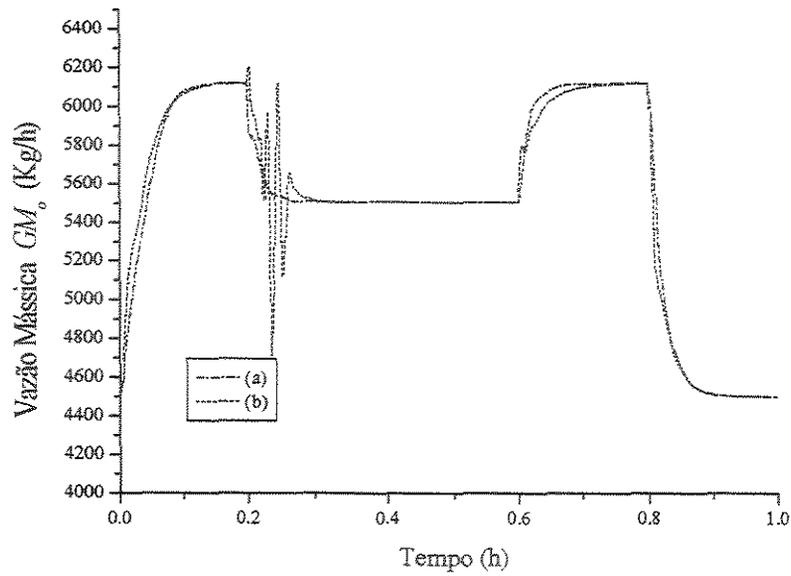


Figura 4.16.b – Ação de controle para problema servo. (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso; (b) Controle direto com aprendizagem *on-line* baseada no erro global.

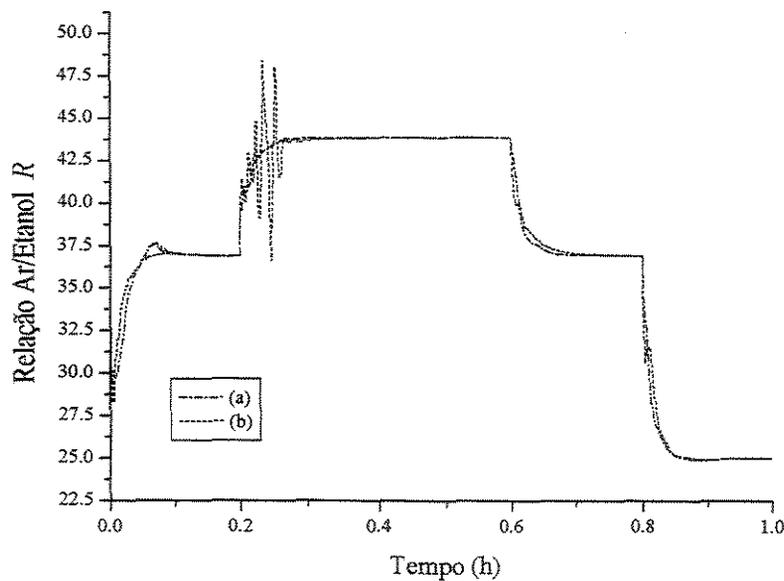


Figura 4.16.c – Ação de controle para problema servo. (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso; (b) Controle direto com aprendizagem *on-line* baseada no erro global.

Pode-se verificar nas Figuras 4.16.b e 4.16.c, ações de controle enérgicas por ocasião de mudanças na referência da malha fechada. Isto deve-se ao fato do conjunto de pesos das redes neurais do controlador não estarem devidamente treinadas para a perturbação imposta. Porém observa-se uma rápida adaptação destes pesos, através da aprendizagem *on-line*, o que pode ser comprovado na estratégia usando erro global, na Figura 4.16.d, onde os erros quadráticos tendem a zero após ocorrida a perturbação. Isto deve-se ao fato da natureza incremental do controlador feedback implementado. Para o caso da aprendizagem *on-line*, verifica-se a redução mais brusca do erro quadrático em função do ajuste dos pesos da rede neural.

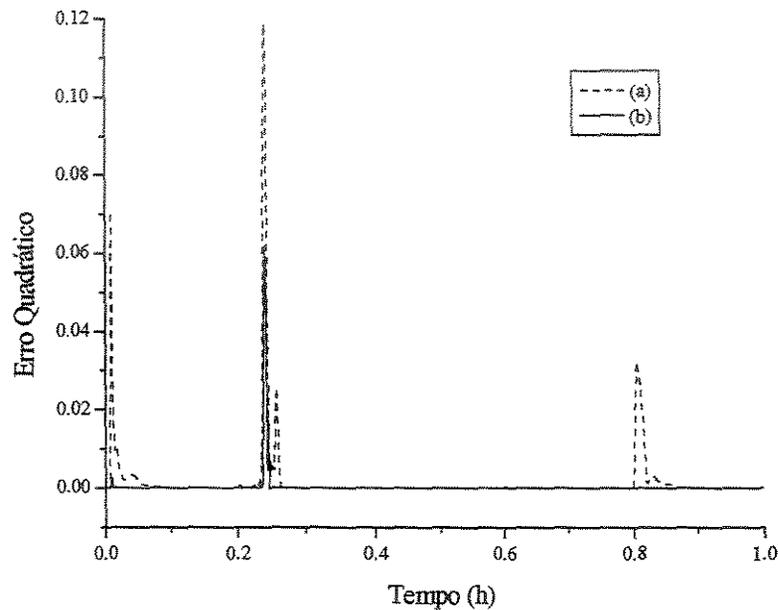


Figura 4.16.d – Erro quadrático das redes neurais do controlador – estratégia usando erro global. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

As Figuras 4.17.a a 4.17.c, apresentam os resultados para problema regulador gerado pela perturbação na temperatura de alimentação do reator, T_{fo} , de acordo com a equação (4.9).

As Figuras 4.18.a a 4.18.d apresentam os resultados para o problema regulador gerado pela perturbação na temperatura de alimentação do fluido refrigerante, T_{RO} , de acordo com a equação (4.22).

$$T_{RO} = T_{RO,SS} + 5 K \quad \text{para } 0.05 \leq t \leq 0.5 \quad (4.22)$$

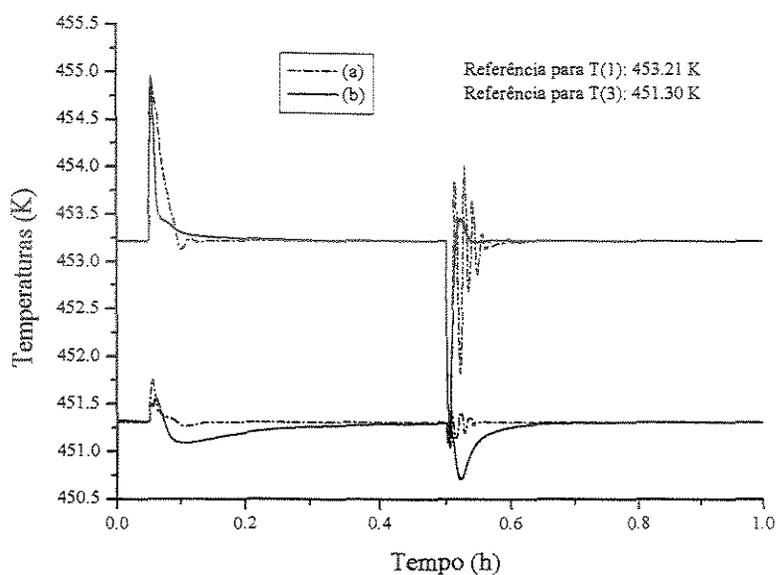


Figura 4.17.a – Resposta da malha fechada para perturbação em T_{fo} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso, IAE(1)=0.0752, IAE(2)=0.0103; (b) Controle direto com aprendizagem *on-line* baseada no erro global, IAE(1)=0.0485, IAE(2)=0.0647.

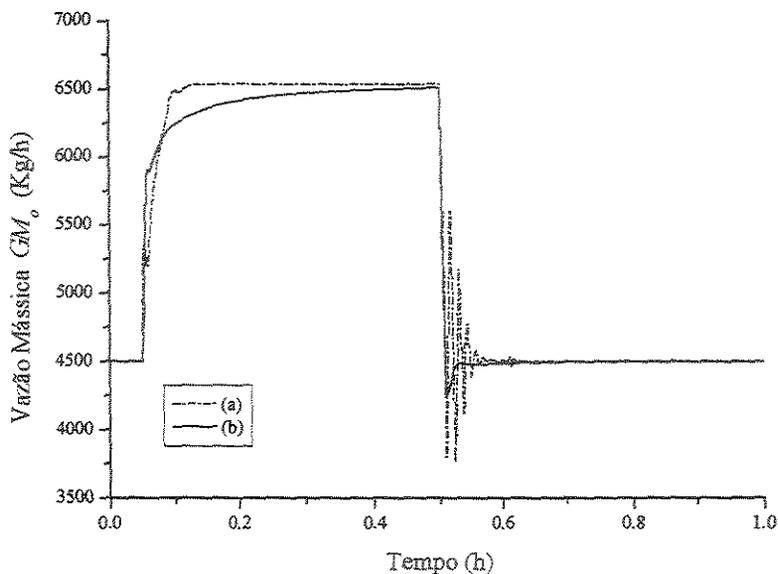


Figura 4.17.b – Ações de controle para perturbação em T_{fo} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso; (b) Controle direto com aprendizagem *on-line* baseada no erro global.

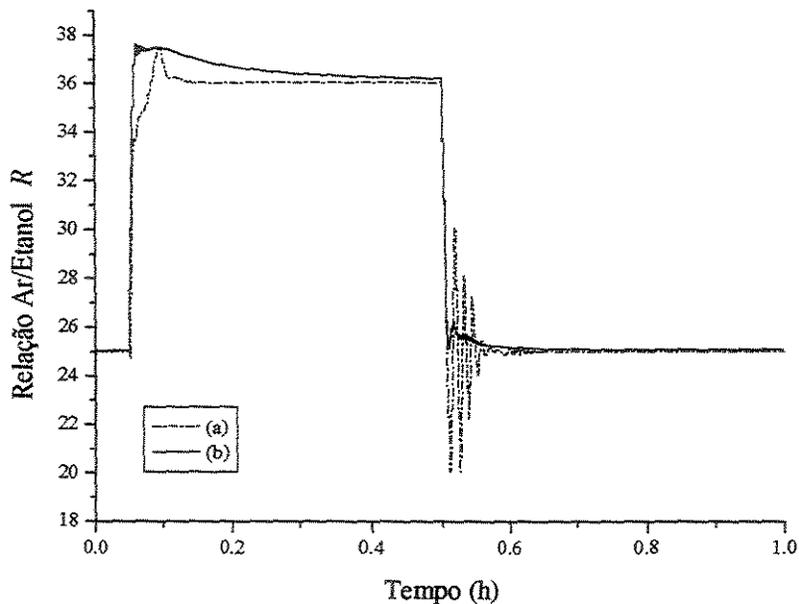


Figura 4.17.c – Ações de controle para perturbação em T_{fo} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso; (b) Controle direto com aprendizagem *on-line* baseada no erro global.

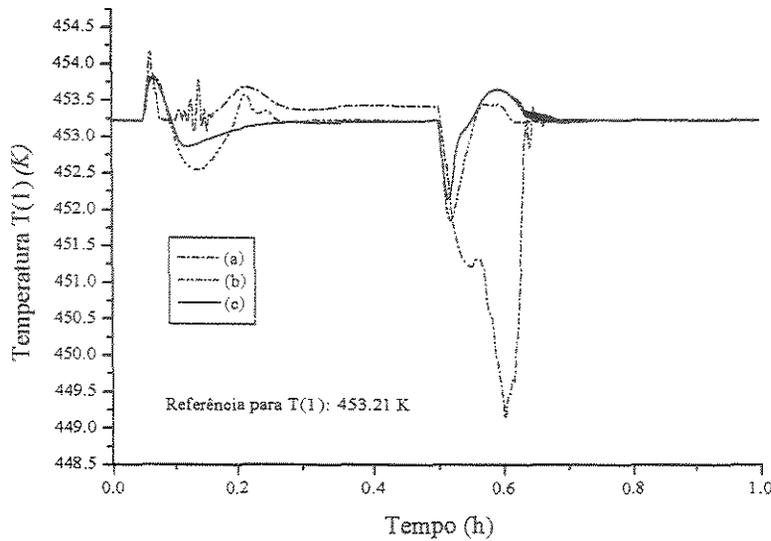


Figura 4.18.a – Resposta da malha fechada para perturbação em T_{R0} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso, $IAE(1)=0.3950$; (b) Controle direto sem aprendizagem *on-line*, $IAE(1)=0.1294$; (c) Controle direto com aprendizagem *on-line* baseada no erro global, $IAE(1)=0.1044$.

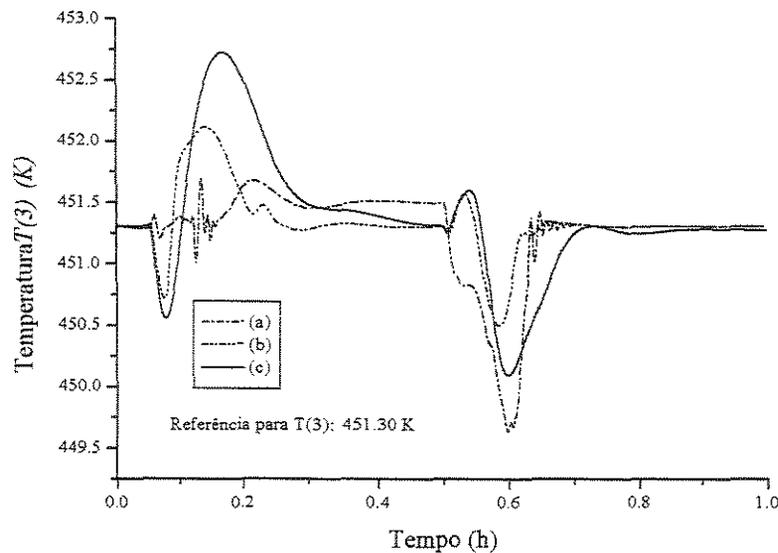


Figura 4.18.b – Resposta da malha fechada para perturbação em T_{R0} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso, $IAE(2)=0.1941$; (b) Controle direto sem aprendizagem *on-line*, $IAE(2)=0.1355$; (c) Controle direto com aprendizagem *on-line* baseada no erro global, $IAE(2)=0.3192$.

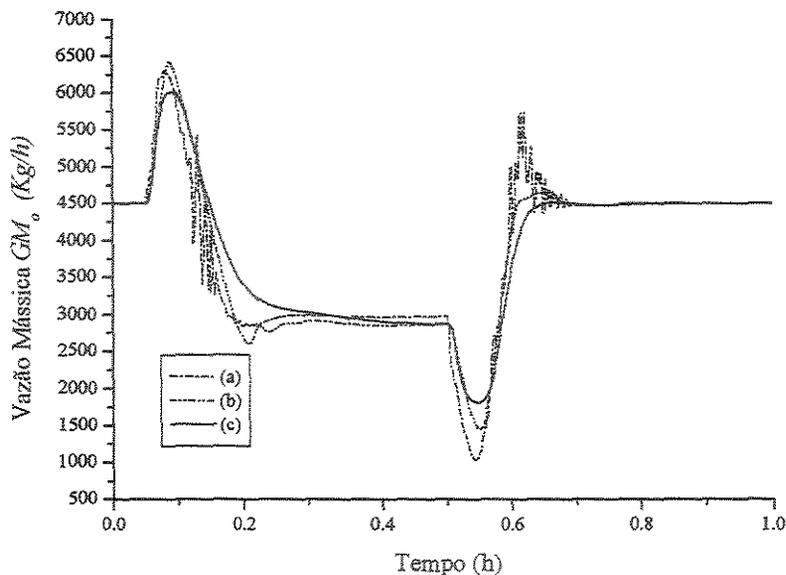


Figura 4.18.c – Ação de controle para perturbação em T_{R0} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso; (b) Controle direto sem aprendizagem *on-line*; (c) Controle direto com aprendizagem *on-line* baseada no erro global.

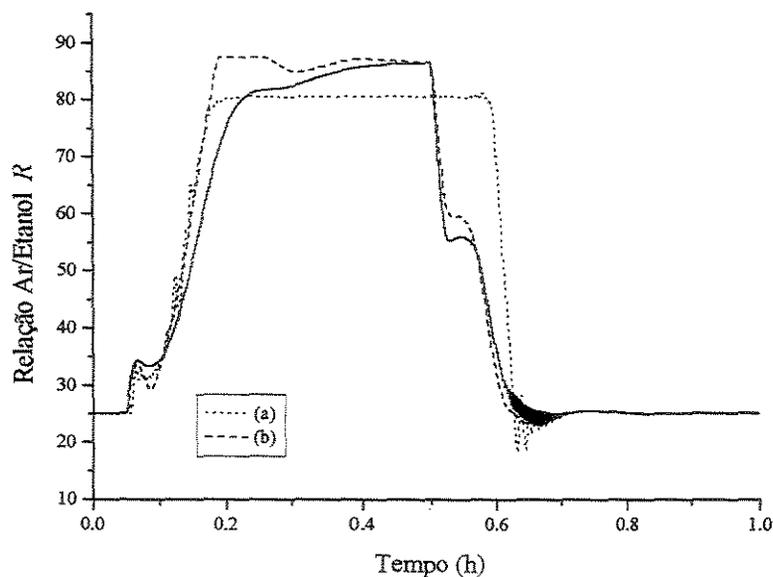


Figura 4.18.d – Ação de controle para perturbação em T_{R0} . (a) Controle direto com aprendizagem *on-line* baseada em modelo inverso; (b) Controle direto sem aprendizagem *on-line*; (c) Controle direto com aprendizagem *on-line* baseada no erro global.

Os resultados obtidos para o problema regulador numa estrutura *feedback* mostraram-se relativamente bons. Para a estratégia usando erro global estimado para ajustar a rede neural do controlador não observou-se maiores problemas, apesar que para a perturbação em T_{R0} , o algoritmo não apresentou a robustez desejada. Já a estratégia usando o modelo inverso para gerar o sinal de aprendizagem da rede neural do controlador *feedback*, incorporou, o que é natural, as deficiências da estratégia usando modelo inverso, diante de perturbação que gerar resposta inversa. Neste caso, esta estratégia levou a resultados com desempenho inferior ao algoritmo sem aprendizagem, o qual eliminou o *off-set*, Figuras 4.18.a e 4.18.b. Na presença de resposta inversa, o sinal gerado pelo modelo inverso não mostrou-se apropriado para a aprendizagem *on-line*. Quando da aprendizagem *off-line* da rede do controlador *feedback*, esta é treinada a responder um desvio zero nas ações de controle, quando o erro da malha fechada for zero. A aprendizagem *on-line* torna a rede neural menos genérica, treinando-a para a nova realidade física apresentada. Neste caso, na presença de resposta inversa, os padrões gerados conduzem a um novo comportamento da rede, o que leva a um *off-set* da malha fechada. Já a estratégia sem aprendizagem *on-line* elimina o *off-set*, devido a natureza incremental do algoritmo e as características impostas na aprendizagem *off-line*.

4.6 Conclusões

Neste capítulo procurou-se explorar o uso de redes neurais em estratégias de controle atuando como controlador. Diversas formas foram abordadas, permitindo-nos afirmar que as redes neurais oferecem uma alternativa possível para utilização em sistemas de identificação ou projeto de controladores, quando não existe um modelo matemático determinístico confiável, ou somente dados históricos do processo estão disponíveis.

As estratégias de controle baseada em modelo inverso, usando diretamente dados de perturbação/resposta do processo na aprendizagem das redes neurais, aprendizagem indireta, tanto na forma MIMO ou MultiSISO, apresentaram bons resultados, com exceção dos casos em que a perturbação apresentou resposta inversa considerável, onde esta

estratégia deixou *off-set*. As estratégias MultiSISO são alternativas interessantes nos casos em que não existirem muitas interações entre as malhas de controle. Esta técnica permite trabalhar-se com várias redes neurais menores, o que facilita a aprendizagem individual de cada rede, e não necessariamente, implica em tempo computacional menor. As estratégias MIMO permitem gerenciar melhor as n malhas de controle, levando em consideração possíveis interações entre as malhas de controle, as quais são incorporadas no controlador durante a aprendizagem. Porém, tem-se a desvantagem em gerar-se redes neurais maiores, o que pode comprometer a aprendizagem neural. Para as duas estratégias avaliadas, não observou-se problemas que venham comprometer o desempenho global do controlador, no que diz respeito a aprendizagem *on-line*.

A estratégia adaptativa proposta, apresentou excelente resultados em todas os casos avaliados, o que vem comprovar a eficiência da metodologia de aprendizagem das redes neurais do controlador usada. A utilização do erro global estimado mostrou-se eficiente na aprendizagem do controlador, baseado na dinâmica inversa do processo. Apesar de vários procedimentos serem executados a cada tempo de amostragem, não verificou-se problemas com o tempo computacional requerido para uma implementação em tempo real.

A utilização de redes neurais como controladores *feedback* apresenta a dificuldade adicional na geração do sinal para a aprendizagem. As duas técnicas de aprendizagem *on-line* das redes do controlador mostraram-se eficientes, a exceção dos problemas gerados devido o sinal de aprendizagem das redes neurais usando modelo inverso, na presença de resposta inversa do processo. Este problema não está associado a técnica de aprendizagem das redes do controlador, mas sim, da rede neural que gera o sinal de aprendizagem.

De uma forma geral, pode-se afirmar que as redes neurais apresentam um excelente potencial para serem usadas como controladores, mesmo na presença de fortes não linearidades no processo, desde que utilizada uma técnica apropriada na aprendizagem *on-line*.

Apesar da estratégia de controle usar várias redes neurais simultaneamente, não observou-se problemas com a aprendizagem *on-line* destas, o que permite afirmar que a capacidade de aprendizagem das redes neurais possibilita sua incorporação em estratégias de controle adaptativo.

CAPÍTULO 5

ESTRATÉGIA DE CONTROLE DE CONCENTRAÇÃO USANDO *SOFT SENSOR*

5.1 Introdução

Diante das dificuldades de obtenção de medidas de concentrações *on-line* em tempo real e a um custo acessível para serem utilizadas em estratégias de controle, propõe-se uma forma alternativa, usando para isto um *software sensor* baseado em redes neurais e estratégias de otimização de referências de temperatura, associada a uma estratégia de controle.

Software sensor é a associação de um sensor (*hardware*), que permite medidas *on-line* de variáveis do processo, com um algoritmo de estimação (*software*), objetivando estimar variáveis de difícil medida, parâmetros de modelos ou para superar problemas com tempos mortos na medida (ASSIS & MACIEL FILHO, 2000). Várias técnicas de estimação tem sido propostas na literatura, onde pode-se citar o uso de ANN.

O objetivo das estratégias propostas é o controle da concentração de um reator catalítico de leito fixo, descrito no Anexo 1, pela otimização *on-line* das referências de

temperaturas das malhas de controle. A otimização utiliza um modelo neural para representação estática do processo e o algoritmo SQP (Sucessive Quadratic Programming).

Alguns trabalhos usando *soft sensor* baseados em redes neurais podem ser encontrados na literatura. Dentre estes pode-se citar os trabalhos de ZYNGIER et al.(2000), ASSIS & MACIEL FILHO (2000), PARK & HAN (2000), ENDER & MACIEL FILHO (2001^{e,fs}).

As estratégias propostas estão fundamentadas no algoritmo de controle adaptativo baseado em redes neurais, e algoritmo de controle preditivo com restrições, ambos já descritos nos capítulos anteriores.

5.2 Estratégias de controle da concentração

As estratégias de controle da concentração propostas consistem em determinar-se através de um procedimento de otimização usando o SQP, as melhores referências de temperatura para a malha fechada, que levam o sistema à concentração desejada. Para isto, estas estratégias são formadas pelo controlador propriamente dito e pelo procedimento de otimização das referências de temperatura (*otimização do setpoint*). O módulo de *otimização do setpoint* é formado pelos procedimentos de otimização *on-line* das referências de temperatura da malha fechada.

5.2.1 Estratégia de controle baseada em redes neurais

A Figura 5.1 apresenta a estratégia de controle de concentração proposta. O módulo de *otimização do controlador* consiste no procedimento de aprendizagem *on-line* das redes neurais usadas no controlador e nas previsões dinâmicas do processo, para

geração do erro estimado da malha fechada, similar ao descrito no capítulo anterior (ENDER & MACIEL FILHO, 2000^o).

A estratégia de otimização das referências de temperatura é baseada em uma rede neural representando o comportamento estacionário do processo com aprendizagem *on-line* e um procedimento de otimização usando programação quadrática sucessiva (SQP). O modelo neural representando o comportamento estacionário do processo é usado na predição da concentração e temperaturas do reator, atuando como *soft sensor* (ENDER & MACIEL FILHO, 2001^{fs}). Os parâmetros de entrada desta rede são variáveis de estado que apresentam forte influência no processo, e as variáveis manipuladas, e as saídas são concentrações e temperaturas do reator nos pontos desejados.

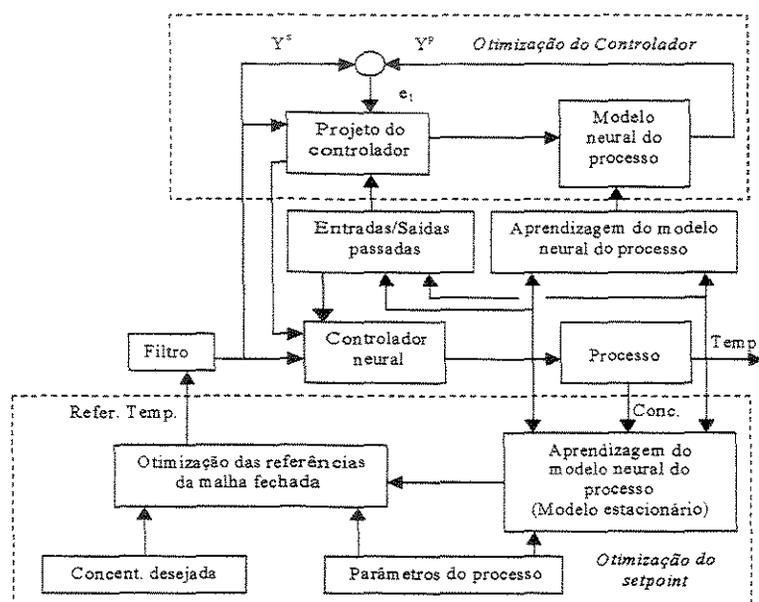


Fig. 5.1 – Estratégia de controle adaptativa baseada em redes neurais, acoplada ao módulo de *Otimização do Setpoint*

5.2.2 Estratégia baseada no algoritmo de controle preditivo com restrições

A Figura 5.2 apresenta o esquema do acoplamento da estratégia de controle preditivo com restrições e a rotina de *otimização do setpoints* (ENDER & MACIEL FILHO, 2001⁶).

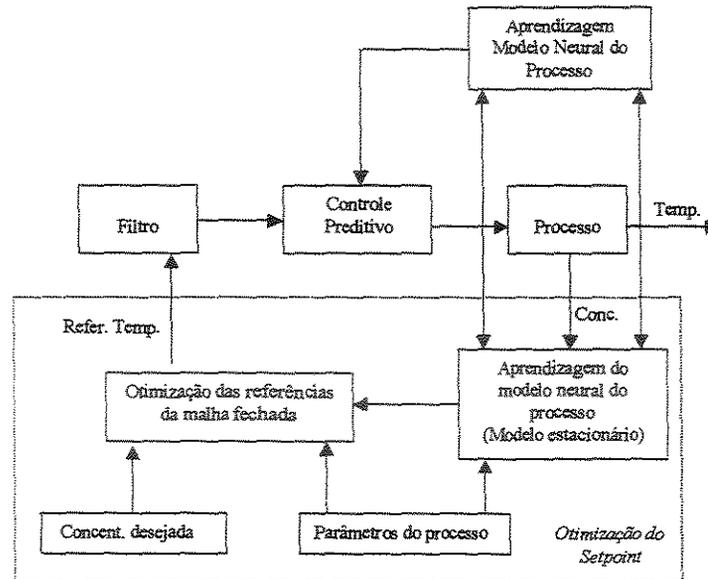


Figura 5.2 – Estratégia de controle preditivo associada a *otimização do setpoint*.

A estratégia de controle preditivo com restrições utilizada encontra-se detalhadamente descrita no Capítulo 3.

Convém ressaltar que nas duas estratégias de controle propostas, a malha fechada atua no controle da temperatura do reator. A definição das referências de temperatura são provenientes de um processo de otimização, usando SQP, que ocorre paralelamente a malha fechada, onde são considerados as características do processo e a concentração desejada para o reator.

No caso particular da estratégia de controle preditivo com restrições, acoplada ao procedimento de otimização das referências de temperatura, tem-se dois procedimentos de otimização, baseados no SQP, sendo executados paralelamente. Apesar do esforço computacional exigido, não observou-se problemas com o tempo de processamento computacional requerido.

5.2.3 A otimização das referências de temperatura

A otimização da referência de temperatura é realizada para manter as concentrações do reator em valores preestabelecidos (referências de concentração). Esta técnica é formada por uma rede neural que representa o estado estacionário do processo e um procedimento de otimização. A rede neural estacionária apresenta como variáveis de entrada as variáveis manipuladas pelo controlador, GM_o e R – neste caso de estudo, e as temperaturas de alimentação, T_{fo} , e fluido refrigerante, T_{R0} , do reator, por serem variáveis de forte influência na dinâmica do processo em questão. Estes parâmetros, bem como as variáveis manipuladas da malha fechada foram definidos pela análise de sensibilidade através de planejamento fatorial, apresentado no Anexo 4. O vetor de saída desta rede é formado pelas previsões das concentrações e temperaturas nos respectivos pontos de colocação ortogonal, definidos considerando-se a necessidade da estratégia de controle, comportamento dinâmico das variáveis em questão nestes pontos e a sensibilidade destes pontos em relação as variáveis manipuladas da malha fechada.

Analisando o comportamento do processo diante de perturbações a serem utilizadas como parâmetros de entrada na rede neural, pode-se verificar nas Figuras 5.3.a – 5.3.c, uma maior variação na temperatura nos primeiros pontos de colocação e uma maior variação na concentração nos últimos pontos de colocação ortogonal. Diante destas informações e o objetivo de controlar as concentração na saída e em um ponto intermediário do reator, adotar-se-á a estratégia de controlar a temperatura no primeiro e terceiro pontos de colocação, por serem pontos sensíveis as variáveis manipuladas a serem usadas.

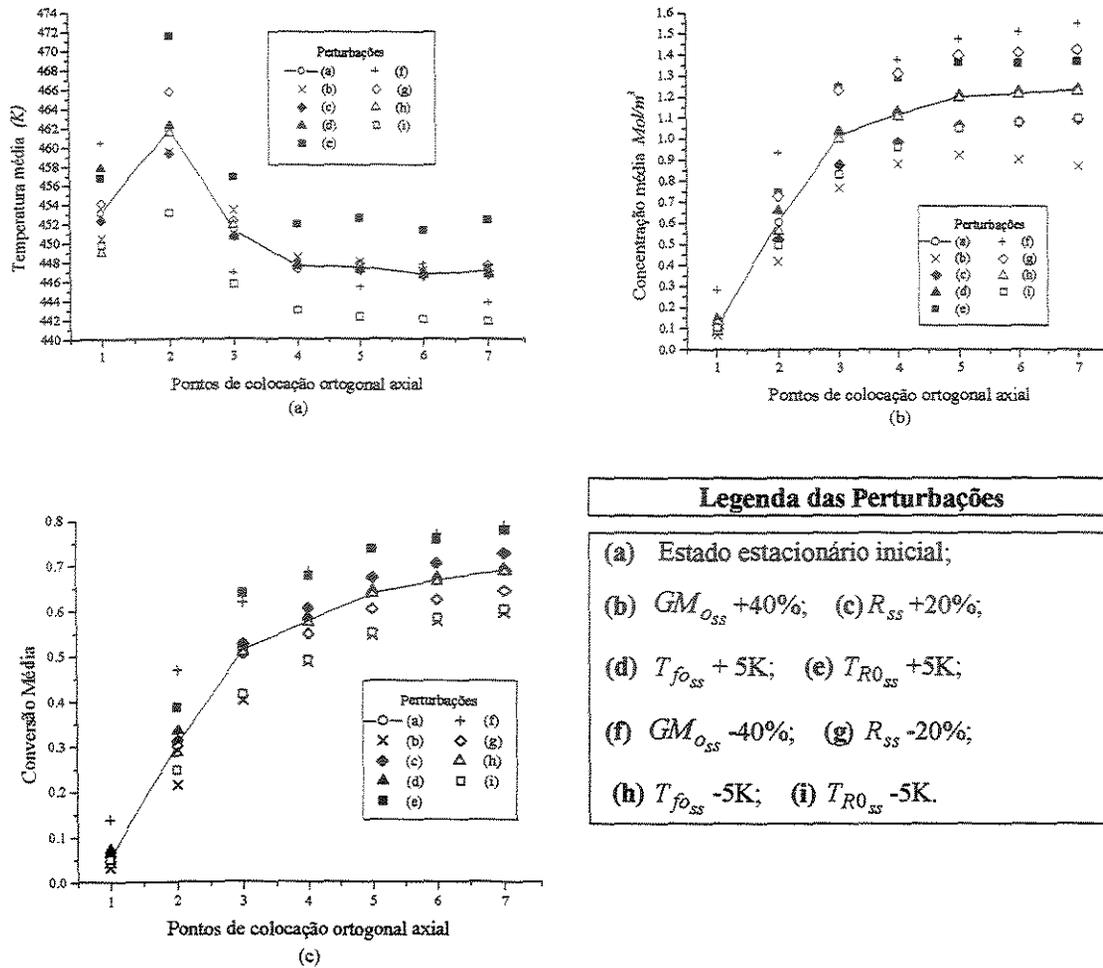


Figura 5.3 – (a)Variação da temperatura média do reator nos pontos de colocação ortogonal axial; (b)Variação da concentração média do reator nos pontos de colocação ortogonal axial; (c)Variação da conversão média do reator nos pontos de colocação ortogonal axial.

Diante destes resultados, adotou-se as saídas da rede neural representando o modelo estacionário do processo, compostas pelas concentrações do reator, nos pontos de controle, ou seja, concentrações do terceiro, $C(3)$, e sétimo, $C(7)$, pontos de colocação ortogonal axial do reator, bem como as variáveis controladas pela malha fechada, ou seja, temperaturas dos pontos de colocação desejados ($T_{estimado}(1)$ e $T_{estimado}(3)$), as quais serão as referências da malha fechada, uma vez otimizado o sistema. A Figura 5.4 mostra a rede

neural representando o modelo estacionário do reator, usado no processo de otimização das referências de temperatura.

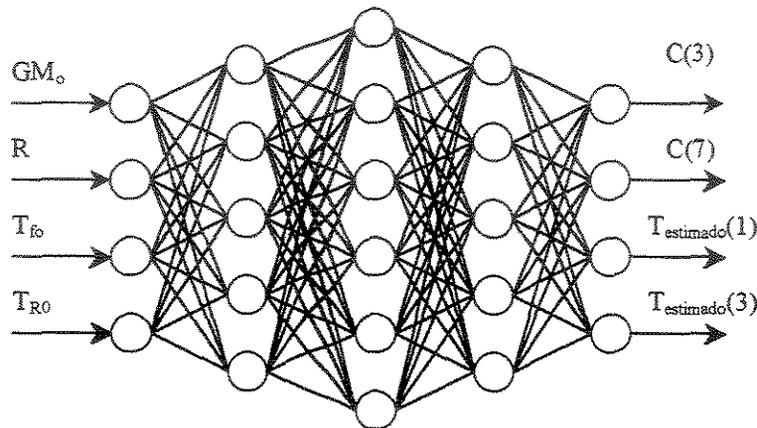


Figura 5.4 – Rede Neural – Modelo Estacionário

A estratégia de controle usa como variáveis manipuladas GM_o e R , e como variáveis controladas as temperatura do primeiro e terceiro ponto de colocação, cujas referências da malha fechada são determinados em procedimento de otimização *on-line*.

A função objetivo da otimização das referências das variáveis controladas foi definida para minimizar o erro entre a concentração estimada pelo modelo estático do processo, $\bar{C}(i, k)$, e a referência de concentração desejada, $\bar{C}_r(i)$, bem como minimizar o desvio na temperatura dos pontos onde esta é controlada, $\bar{T}(i, k)$. Diante destas considerações, a função objetivo é dada por:

$$J(k) = \sum_{i=1}^{N_m} \left(\left| \bar{C}_r(i) - \bar{C}(i, k) \right| \right)^3 * \bar{T}(i, k)^2 * 10^3 \quad (5.1)$$

sujeito as seguintes restrições:

$$\bar{C}_{\min} \leq \bar{C}(i, k) \leq \bar{C}_{\max}, \quad i = 1, \dots, N_m \quad (5.2)$$

$$\bar{T}_{\min} \leq \bar{T}(i, k) \leq \bar{T}_{\max}, \quad i = 1, \dots, N_m \quad (5.3)$$

onde:

N_m : número de malhas de controle;

\bar{x} : indica desvio em relação ao estado estacionário inicial;

$$\bar{T}(i, k) = \bar{T}_{estimado}(j), \text{ para } \bar{T}_{estimado}(j) \geq 1 \quad (5.4)$$

$$\bar{T}(i, k) = 1, \text{ para } \bar{T}_{estimado}(j) < 1 \quad (5.5)$$

$$\bar{C}(i, k) = \bar{C}(j) \quad (5.6)$$

\bar{C}_r : referência de concentração

j : ponto de colocação ortogonal axial

Para resolver esse problema de otimização, usou-se a rotina DNCONF da biblioteca IMSL que emprega o método de programação quadrática sucessiva, com variáveis independentes as vazão mássica de alimentação (GM_o) e relação ar/etanol (R).

Uma vez minimizada a função objetivo, os valores estimados das temperaturas nos pontos em análise, através dos procedimentos de otimização, serão usados como respectivas referências na estratégia de controle da temperatura do reator. Os valores obtidos para GM_o e R obtidos neste procedimento de otimização não irão interferir na malha fechada, porém são os supostos valores para as ações de controle, uma vez atingido o estado estacionário.

Este procedimento de otimização das referências da malha fechada não é realizado a cada tempo de amostragem da estratégia de controle, mas sim, periodicamente em tempos mais esparsos ou em momentos em que for detectado uma mudança considerável no processo, objetivando não perturbar em demasia o sistema.

A aprendizagem *on-line* do modelo neural estático requer alguns cuidados na escolha de padrões estacionários, uma vez que o processo é dinâmico. Para verificar se o padrão representa um ponto estacionário, um procedimento analisando a derivada das variáveis envolvidas é realizado. Esta análise das derivadas é composta por uma avaliação das derivadas temporais das principais variáveis do processo (variáveis manipuladas,

variáveis controladas e neste caso de estudo, T_{R0} e T_{fo}), computadas para o tempo atual e um valor predefinido de tempos passados. Se todas as derivadas (presentes e passadas) atenderem a uma tolerância especificada, o conjunto de padrões representa um estado estacionário deste processo, e poderá ser usada na aprendizagem *on-line* do modelo neural estacionário. Caso este critério não seja atendido, o conjunto de dados amostrado é rejeitado. Outro aspecto importante é o número de padrões estacionários que devem ser considerados no conjunto de informações usadas na aprendizagem *on-line*. Um número elevado de padrões no conjunto de informações, a princípio, parece ser interessante pois a aprendizagem irá abordar o comportamento do processo de uma forma mais ampla. Porém em situações não previstas, como por exemplo, a falha de um sensor, onde o processo real responde a uma perturbação que está ocorrendo, mas a rede neural não está sendo informada a respeito desta perturbação, em função da falha do sensor, um conjunto formado por vários padrões de aprendizagem não se torna interessante, pois a rede terá que se readaptar para incorporar esta falha do sensor e, a princípio, somente a última amostragem contém informações do processo que consideram esta falha do sensor. Nestas circunstâncias, o novo comportamento dinâmico/estático da rede neural deverá incorporar o efeito desta perturbação, como uma alteração no comportamento daquele processo. Diante disto e considerando que a amostragem de concentração ocorre em intervalos de tempos maiores do que os intervalos de controle, um conjunto de informações a ser usado na aprendizagem *on-line* composto por vários padrões, poderá comprometer a velocidade de adaptação da rede neural, pois a renovação deste conjunto de padrões representando o novo comportamento do processo será mais lenta. Após vários testes realizados e considerando várias perturbações possíveis de ocorrer no sistema de controle, adotou-se um conjunto formado com somente um padrão de aprendizagem, o qual representa os valores obtidos da última amostragem não rejeitada do processo. A aprendizagem *on-line* somente irá ocorrer se o erro quadrático da rede em questão ao representar este padrão não atender um critério de erro preestabelecido. Esta técnica não pode ser chamada de aprendizagem, mas sim, uma adaptação do modelo neural estático para a situação atual do processo.

5.3 Resultados

Para verificar as potencialidades das estratégias de controle multivariável propostas, foram geradas perturbações do tipo servo e regulador. Em todos os casos analisados usou-se como variáveis manipuladas, a vazão mássica GM_o e a relação ar/etanol R , e como variáveis controladas, as temperaturas $T(1)$ e $T(3)$, objetivando manter as concentrações $C(3)$ e $C(7)$ em seus valores desejados. Seguem abaixo, os tempos envolvidos na estratégia de controle:

- Tempo de amostragem de concentração: 10 minutos;
- Tempo a partir da amostragem da concentração, em que os dados de concentração estão disponíveis: 2 minutos;
- Tempo de atualização das referências de temperatura: 2 minutos ou quando ocorrer uma perturbação;
- Tempo de amostragem do controlador: 0.09 min \Rightarrow 5.4 segundos.

A Tabela 5.1 apresenta o problema servo usado e as Figuras 5.5 a 5.7 mostram os resultados alcançados.

Tabela 5.1 – Problema Servo

Tempo (h)	C(3) (Mol/m ³)	C(7) (Mol/m ³)	$\bar{C}(3)$	$\bar{C}(7)$
0.00 – 0.10	1.017	1.233	0.000	0.000
0.10 – 0.55	1.284	1.519	0.266	0.286
0.55 – 1.00	1.017	1.233	0.000	0.000

A apresentação dos resultados mostra um paralelo entre as estratégias de controle anteriormente citadas, ou seja, estratégia baseada em redes neurais, aqui denominada *Estratégia A* e a estratégia baseada no algoritmo de controle preditivo com restrições, aqui

denominada *Estratégia B*. As Figuras 5.5.a a 5.5.f apresentam os resultados para o problema servo, Tabela 5.1, usando a estratégia baseada em redes neurais (*Estratégia A*); as Figuras 5.6.a a 5.6.e, usando a estratégia de controle preditivo (*Estratégia B*) e as Figuras 5.7.a e 5.7.b as ações de controle envolvidas. Convém ressaltar que o procedimento de otimização da referência da temperatura da malha fechada é exatamente o mesmo nas duas estratégias de controle, porém é observado um comportamento diferenciado entre as duas estratégias, devido a aquisição *on-line* de dados do processo para a aprendizagem da rede neural, os quais são reflexos da estratégia de controle empregada. Analisando as Figuras 5.5.a e 5.5.b, que apresentam os resultados da malha fechada, referência de temperatura e concentrações do reator, pode-se verificar um excelente comportamento da estratégia proposta. Observa-se que a referência de temperatura é ajustada ao longo do tempo em função de novas informações obtidas do processo as quais são utilizadas na aprendizagem *on-line* das redes neurais. Observa-se também, que as concentrações desejadas são alcançadas com erro porcentual muito próximo a zero, o que pode ser comprovado na Figura 5.5.c As Figuras 5.5.c a 5.5.f apresentam os erros quadráticos das redes neurais utilizadas, onde pode ser verificada a eficiência da aprendizagem *on-line* proposta.

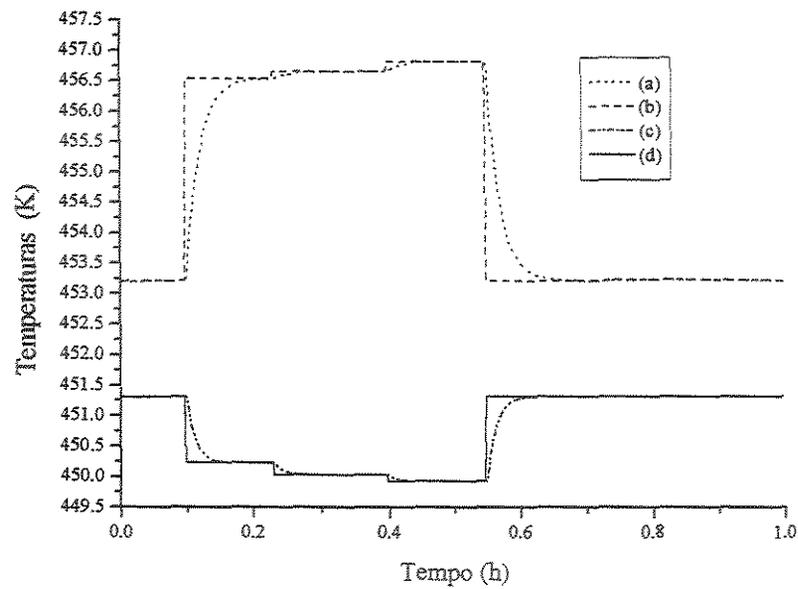


Figura 5.5.a – Resposta da malha fechada e referências de temperatura para problema servo – Estratégia A. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

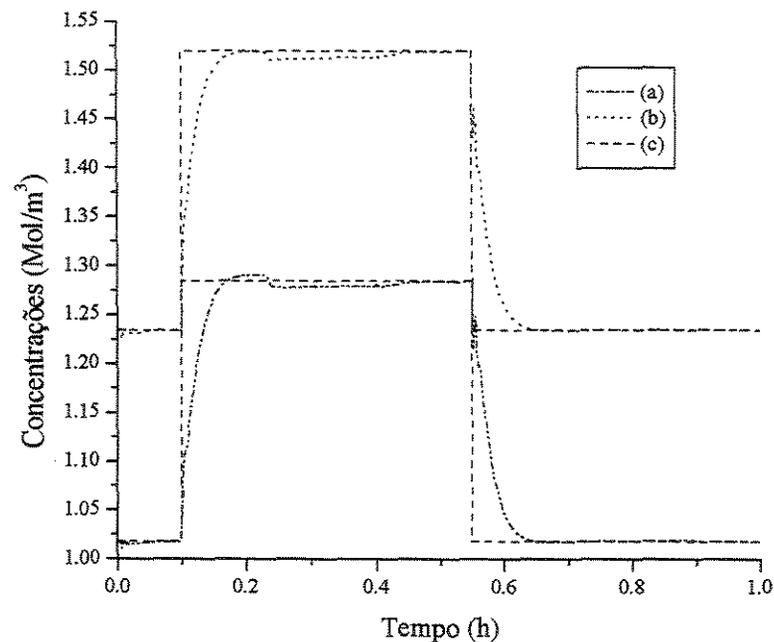


Figura 5.5.b – Concentrações para problema servo – Estratégia A. (a) Concentração C(3); (b) Concentração C(7); (c) Referências de concentração – Tabela 5.1

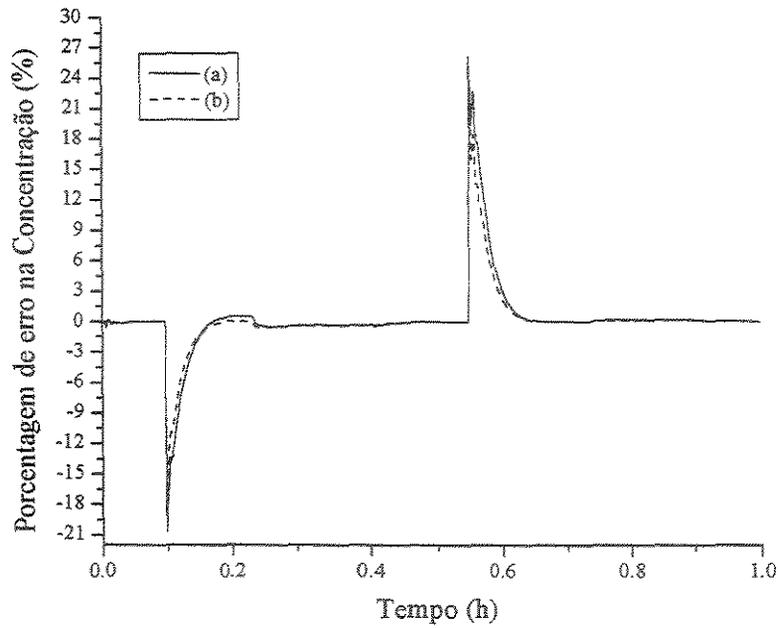


Figura 5.5.c – Erros na concentração para problema Servo – Estratégia A. (a) Concentração C(3); (b) Concentração C(7).

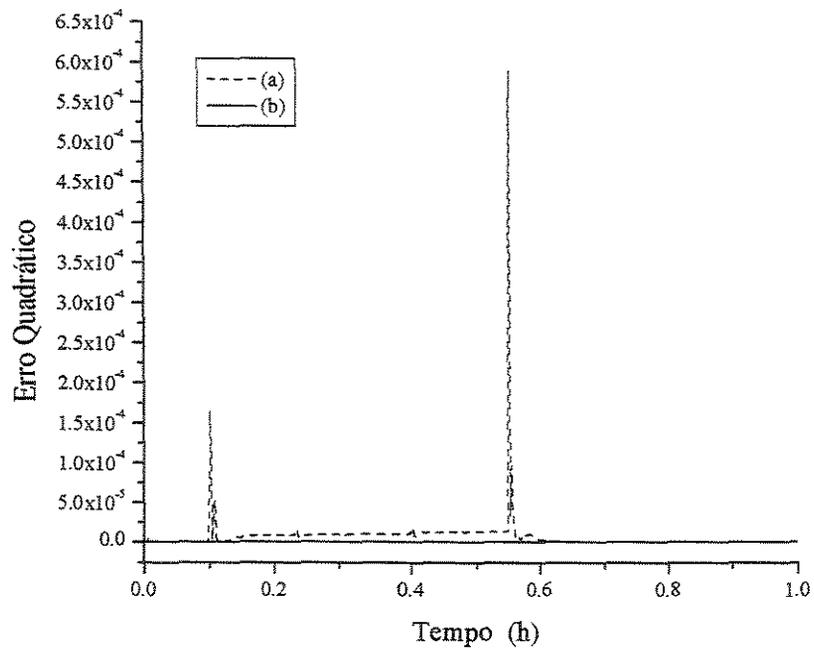


Figura 5.5.d – Erros quadráticos das redes neurais do controlador para problema servo – Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

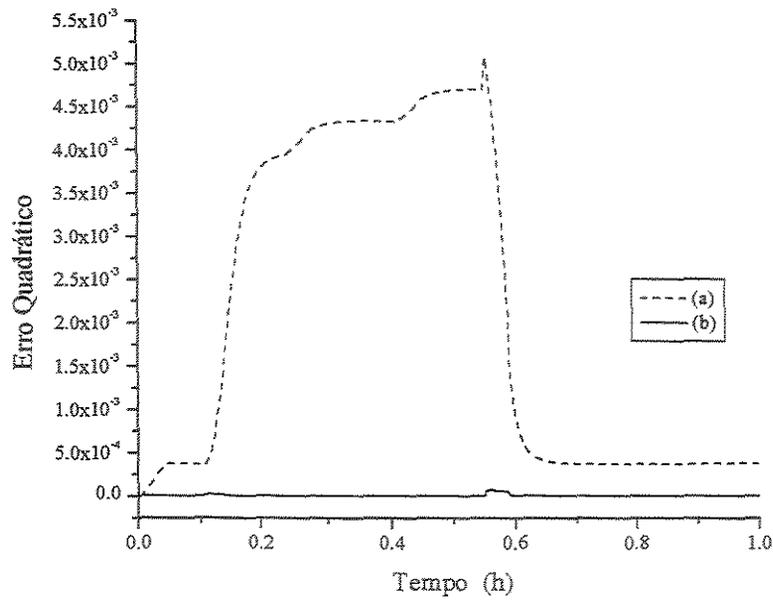


Figura 5.5.e - Erros quadráticos das redes neurais do modelo dinâmico para problema servo – Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

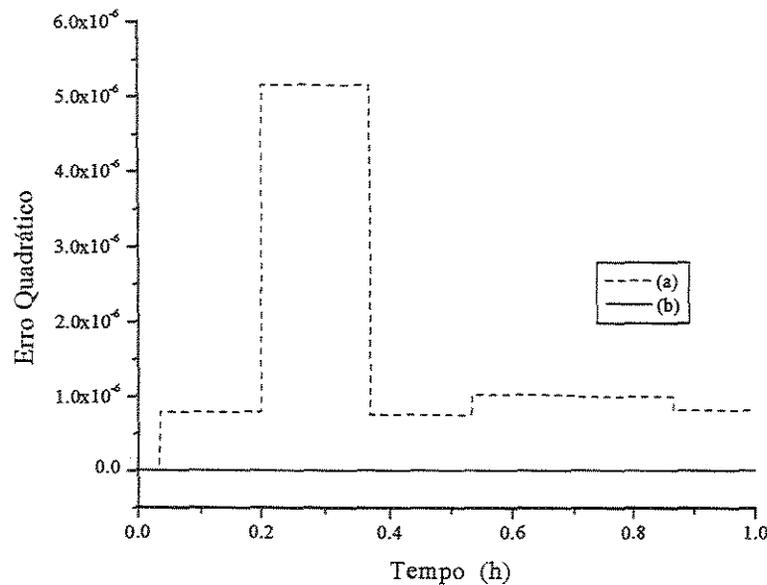


Figura 5.5.f - Erros quadráticos das redes neurais do modelo estático para problema servo – Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

A Figura 5.5.f e as demais representando o erro quadrático das redes neurais do modelo estático, apresentam esta forma de degraus devido um único padrão ser utilizado na adaptação da rede neural e a renovação deste padrão ocorrer a cada tempo de amostragem da concentração, se este representar um ponto estacionário.

As Figuras 5.6.a a 5.6.e apresentam os resultados para o problema servo proposto, Tabela 5.1, usando a estratégia de controle preditivo com restrições (Estratégia B).

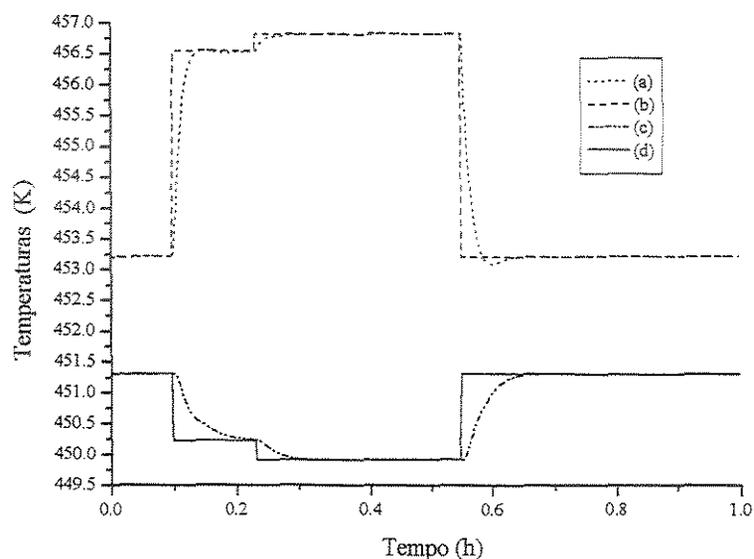


Figura 5.6.a – Resposta da malha fechada e referências de temperatura para problema servo – Estratégia B. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

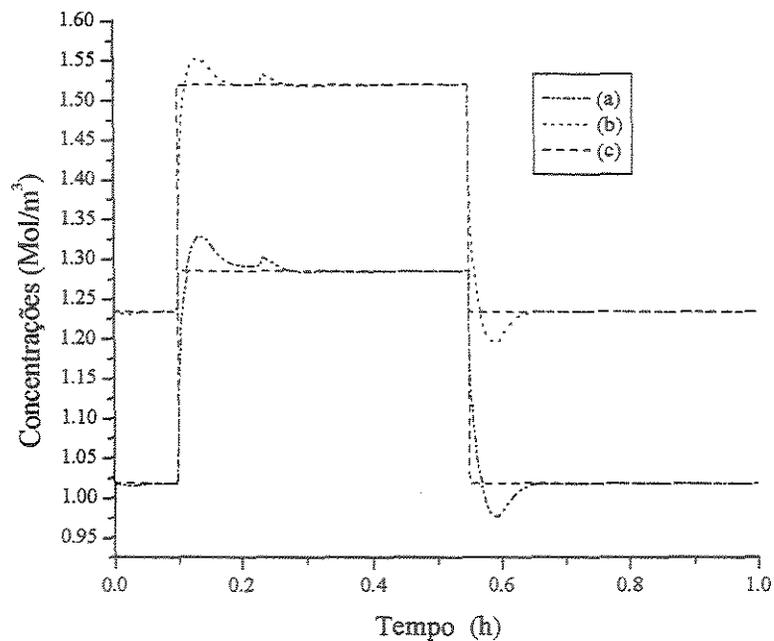


Figura 5.6.b – Concentrações para problema servo – Estratégia B. (a) Concentração C(3); (b) Concentração C(7); (c) Referências de concentração – Tabela 5.1

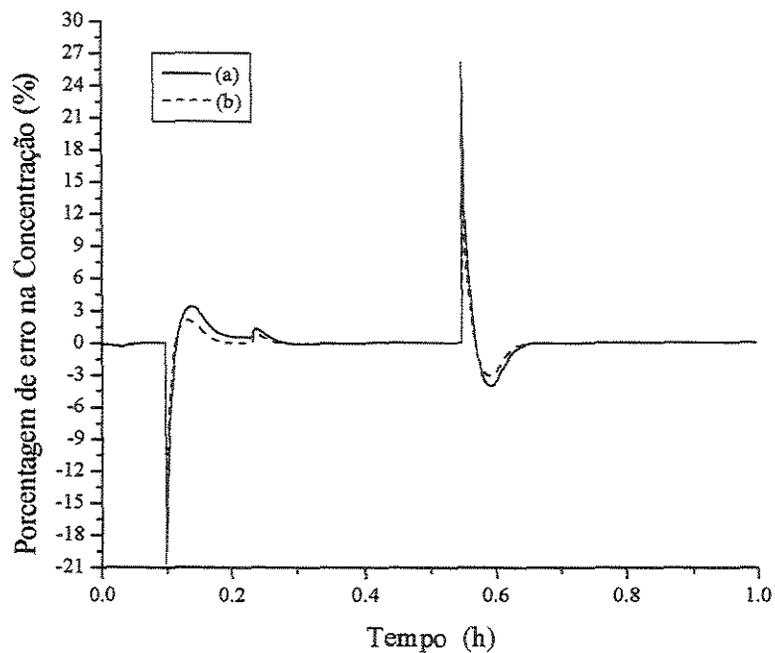


Figura 5.6.c – Erros na concentração para problema Servo – Estratégia B. (a) Concentração C(3); (b) Concentração C(7).

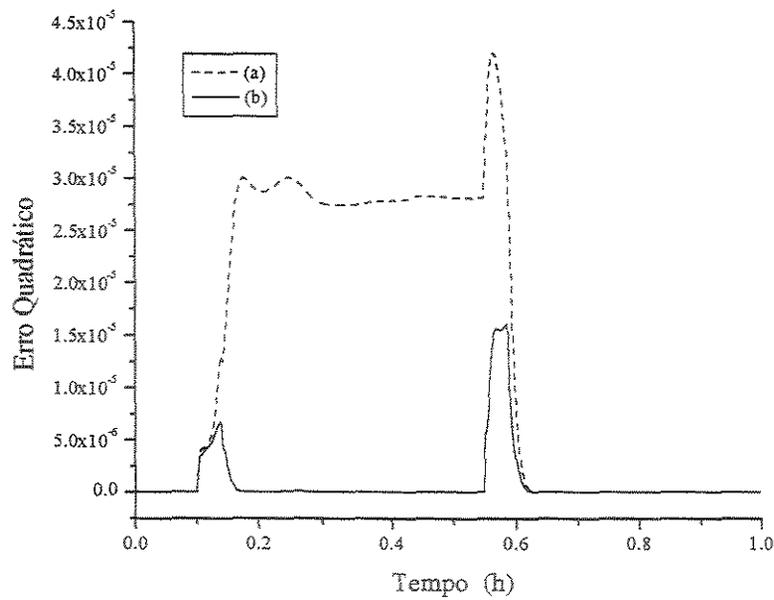


Figura 5.6.d - Erros quadráticos das redes neurais do modelo dinâmico para problema servo – Estratégia B. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

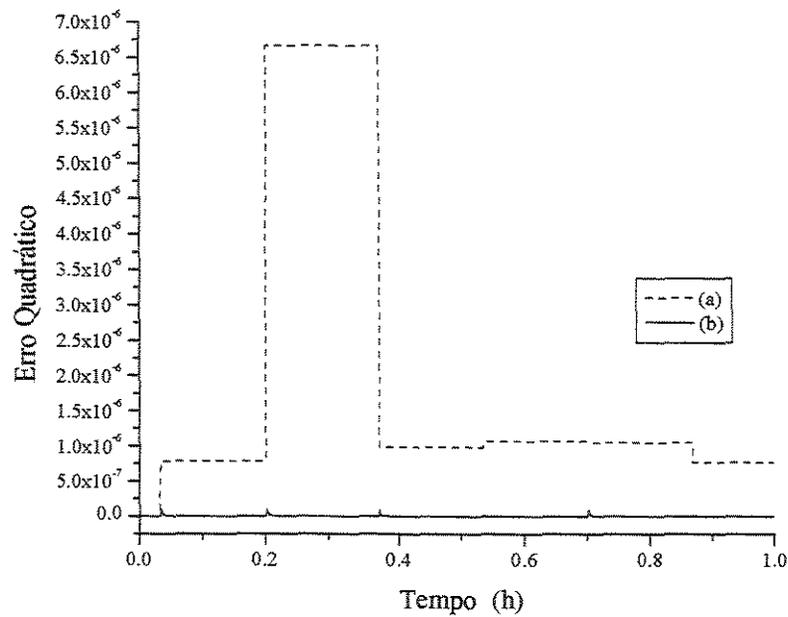


Figura 5.6.e - Erros quadráticos das redes neurais do modelo estático para problema servo – Estratégia B. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

As Figuras 5.7.a e 5.7.b apresentam as ações de controle geradas pelas duas estratégias de controle usadas.

O comportamento da malha fechada apresenta variações em função do ajuste de seus parâmetros de controle. Para os casos analisados e observando as Figuras 5.7.a e 5.7.b, pode-se verificar ações de controle mais enérgicas para a Estratégia A, em função do ajuste utilizado. Porém pode-se afirmar que as duas estratégias apresentam excelentes resultados para o problema servo.

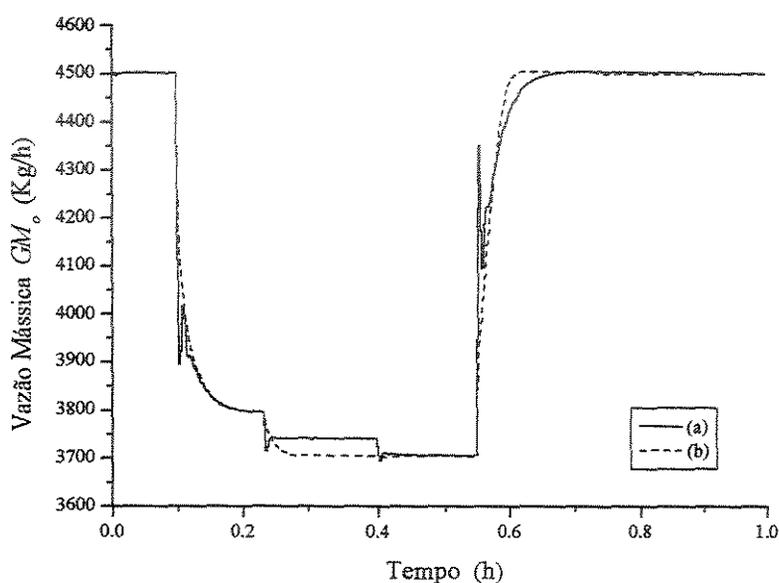


Figura 5.7.a – Ação de controle para problema servo. (a) Estratégia A; (b) Estratégia B.

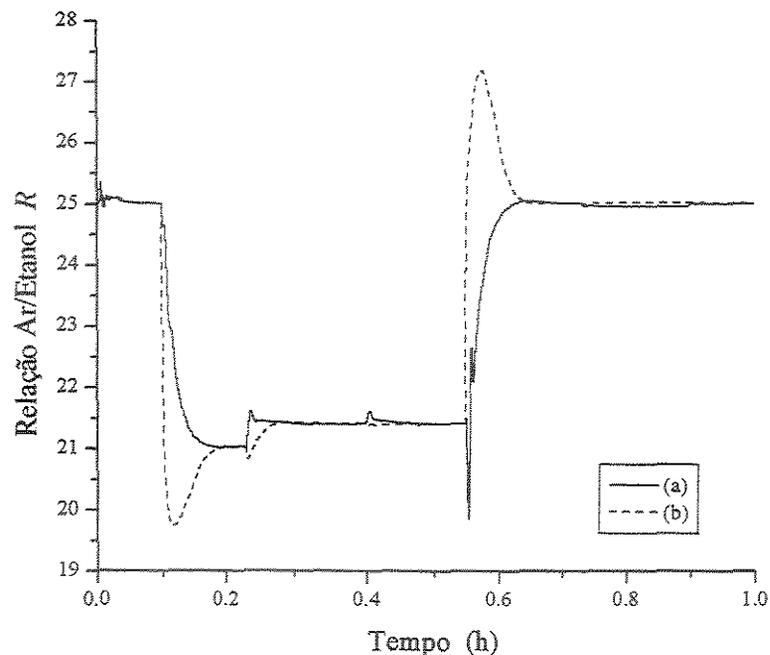


Figura 5.17 – Ação de controle para problema servo. (a) Estratégia A; (b) Estratégia B.

Este mesmo processo, dentro do mesmo objetivo de controle foi submetido a perturbações do tipo regulador, causadas pela variação nas temperaturas de alimentação do reator, T_{fo} , e de refrigeração, T_{R0} .

Considerando inicialmente uma perturbação em T_{fo} de acordo com a equação (5.7), as Figuras 5.8.a a 5.8.f apresentam os resultados para a estratégia de controle baseada em redes neurais; as Figuras 5.9.a a 5.9.e os resultados para as estratégias de controle preditivo com restrições e as Figuras 5.10.a e 5.10.b apresentam o resultado comparativo das ações de controle para as duas estratégias de controle.

$$T_{fo} = T_{foSS} + 5 K \quad \text{para } 0.1 \leq t \leq 0.75 \quad (5.7)$$

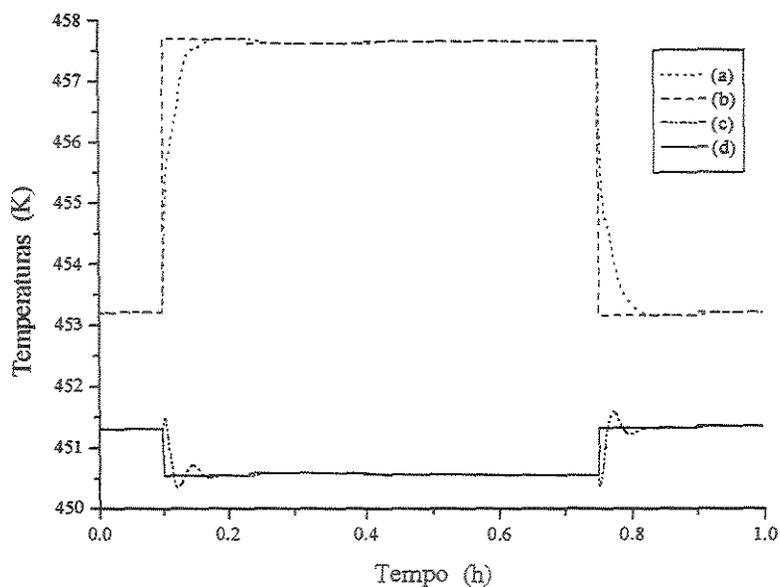


Figura 5.8.a -- Resposta da malha fechada e referências de temperatura para problema regulador - T_{fo} - Estratégia A. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

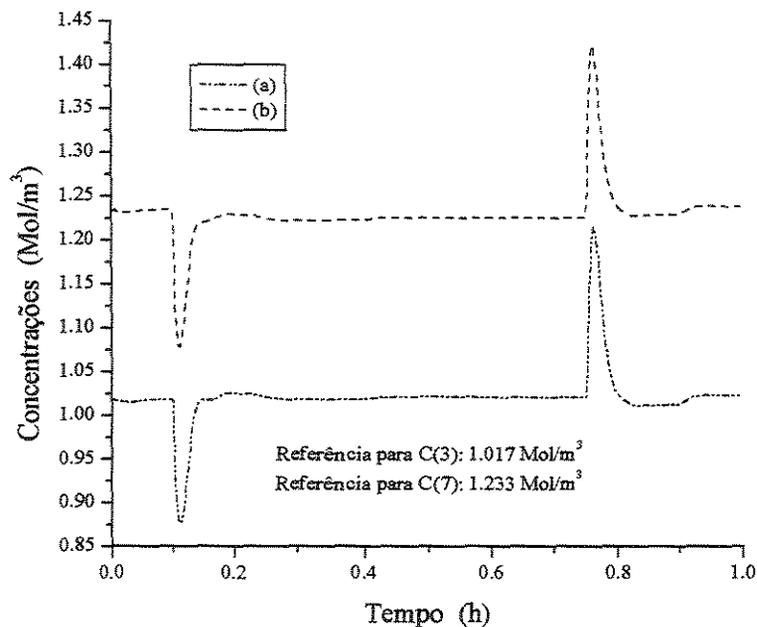


Figura 5.8.b – Concentrações para problema regulador - T_{fo} - Estratégia A. (a) Concentração C(3); (b) Concentração C(7).

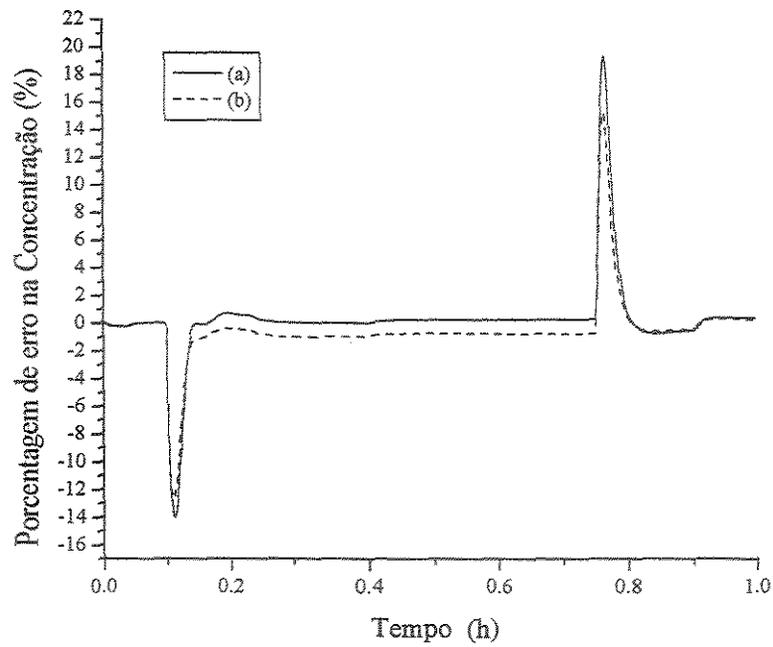


Figura 5.8.c – Erros nas concentrações para problema regulador - T_{fo} - Estratégia A. (a) Concentração C(3); (b) Concentração C(7).

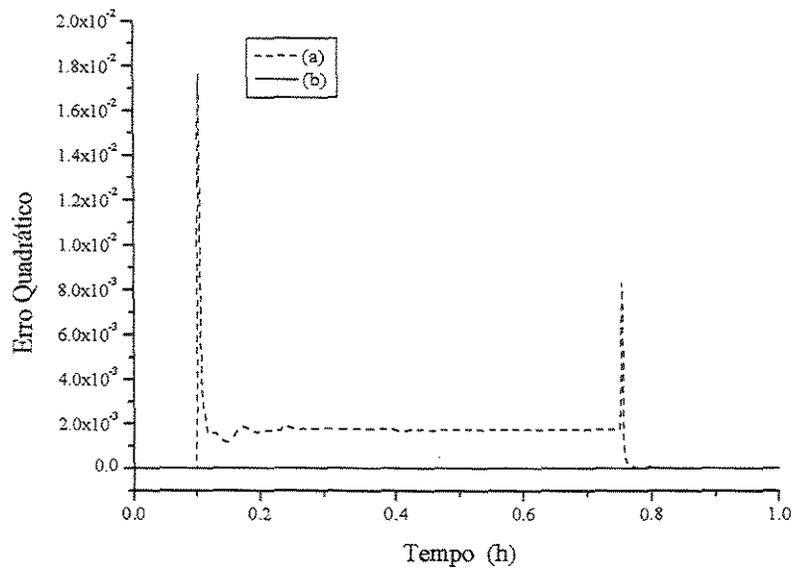


Figura 5.8.d – Erros quadráticos das redes neurais do controlador para problema regulador - T_{fo} - Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

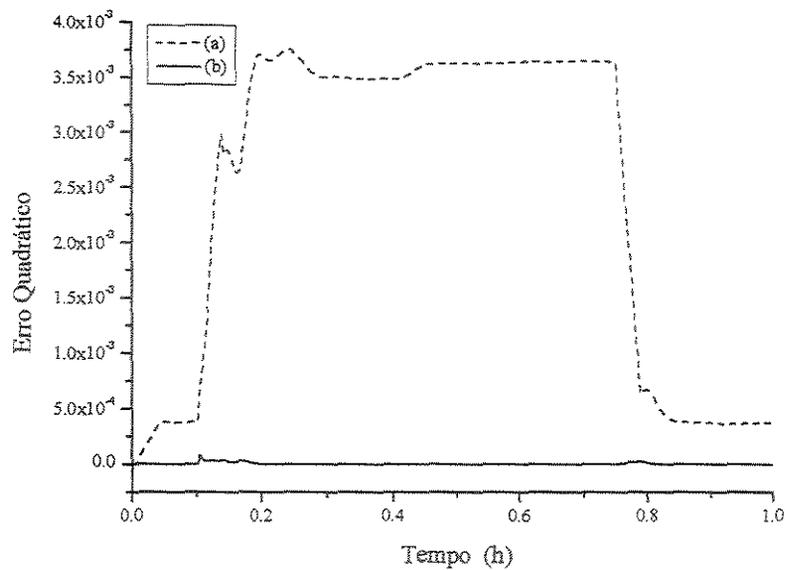


Figura 5.8.e - Erros quadráticos das redes neurais do modelo dinâmico para problema regulador - T_{fo} - Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

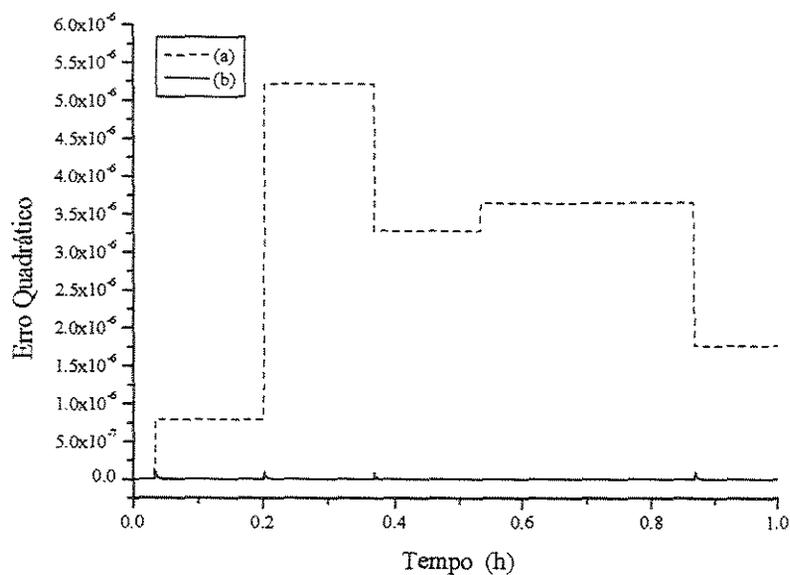


Figura 5.8.f - Erros quadráticos das redes neurais do modelo estático para problema regulador - T_{fo} - Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

As Figuras 5.9.a a 5.9.e apresentam os resultados para o problema regulador pela perturbação de T_{fo} , para a estratégia de controle preditivo com restrições (Estratégia B), nas mesmas condições de operação apresentadas para a estratégia baseada em redes neurais (Estratégia A).

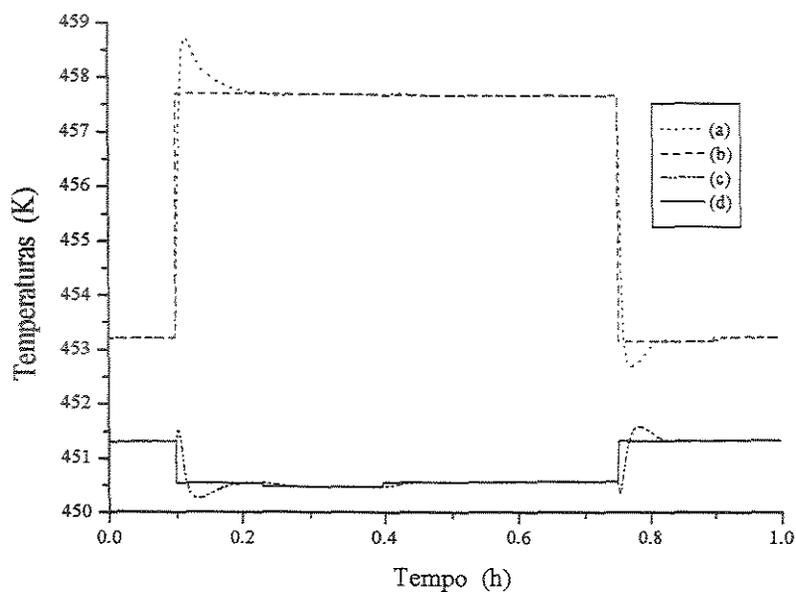


Figura 5.9.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{fo} - Estratégia B. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

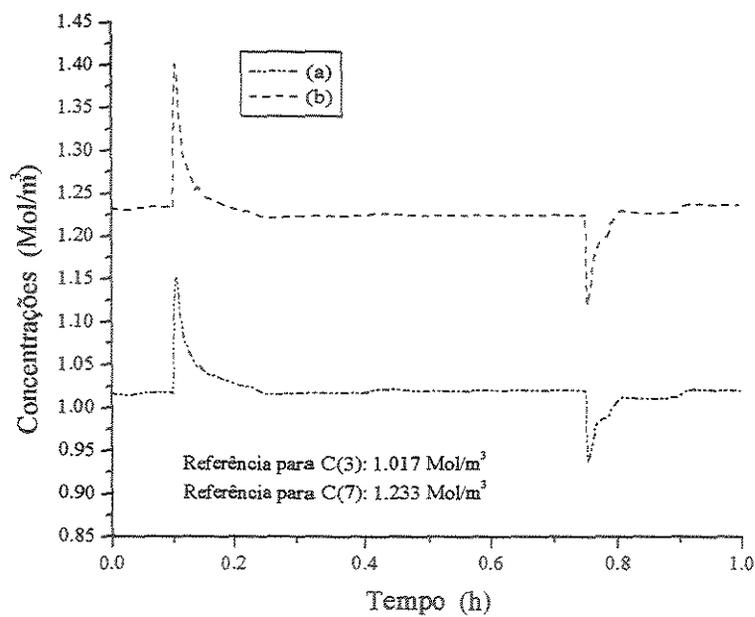


Figura 5.9.b – Concentrações para problema regulador - T_{fo} - Estratégia B. (a) Concentração C(3); (b) Concentração C(7).

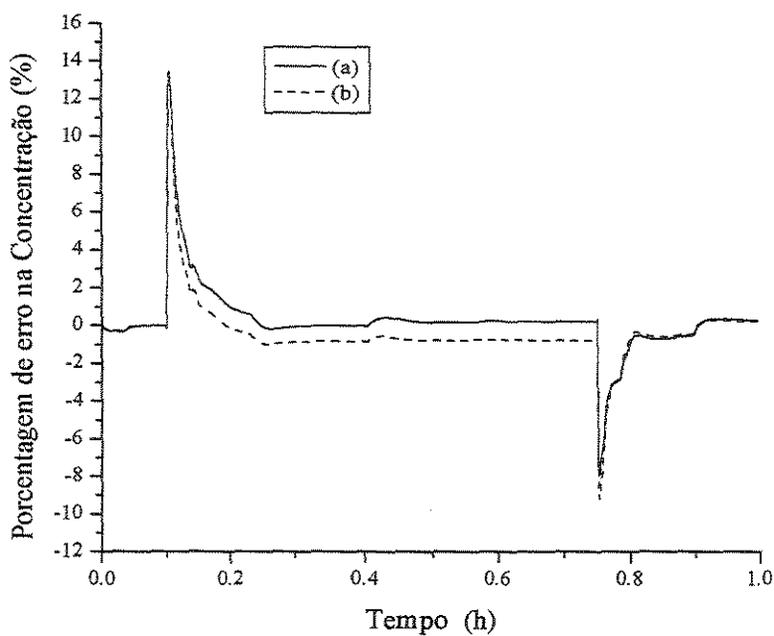


Figura 5.9.c - Erros nas concentrações para problema regulador - T_{fo} - Estratégia B. (a) Concentração C(3); (b) Concentração C(7).

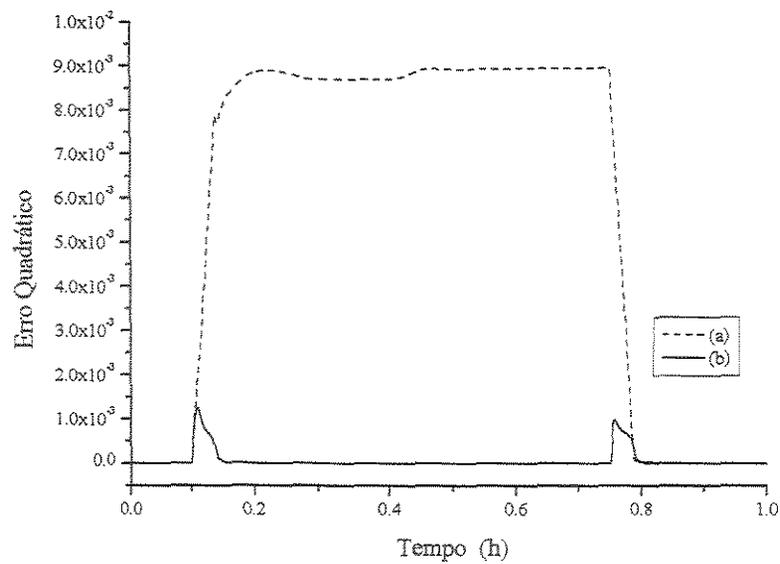


Figura 5.9.d - Erros quadráticos das redes neurais do modelo dinâmico para problema regulador - T_{fo} - Estratégia B. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

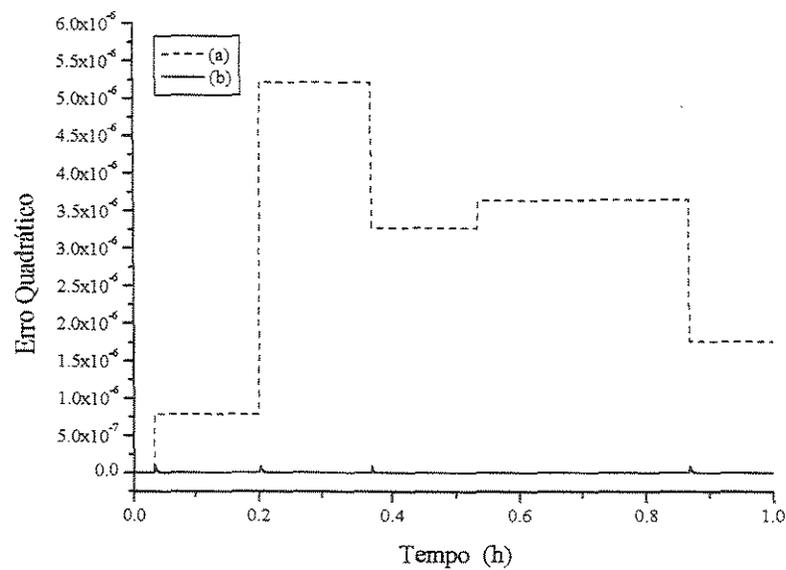


Figura 5.9.e - Erros quadráticos das redes neurais do modelo estático para problema regulador - T_{fo} - Estratégia B. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

As Figuras 5.10.a e 5.10.b apresentam um paralelo das ações de controle geradas pelas estratégias de controle citadas.

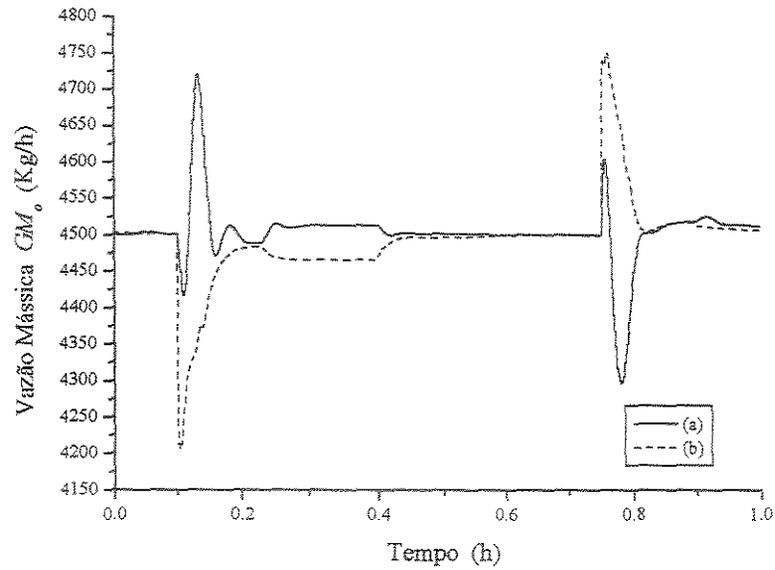


Figura 5.10.a – A o de controle para problema regulador - T_{fo} . (a) Estrat gia A; (b) Estrat gia B.

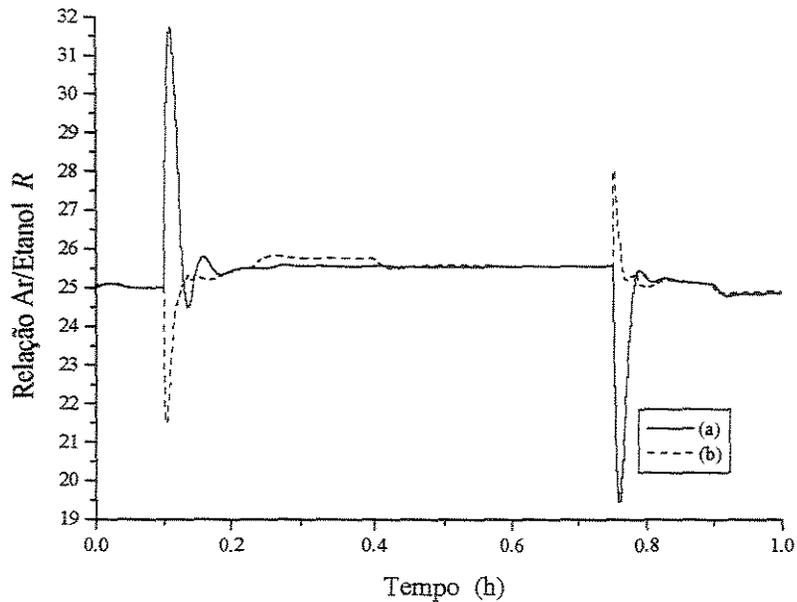


Figura 5.10.b – A o de controle para problema regulador - T_{fo} . (a) Estrat gia A; (b) Estrat gia B.

As Figuras 5.11.a a 5.13.b apresentam os resultados para o problema regulador devido a perturbação na temperatura de refrigeração do reator, T_{R0} . Dentre estas, as Figuras 5.11.a a 5.11.f apresentam os resultados usando a estratégia de controle baseada em redes neurais; as Figuras 5.12.a a 5.12.e apresentam os resultados usando estratégia de controle preditivo com restrições e as Figuras 5.13.a e 5.13.b mostram um comparativo das ações de controle geradas para as duas estratégias para o problema regulador em pauta. As perturbações geradas em T_{R0} respeitam a equação 5.8.

$$T_{R0} = T_{R0SS} + 4 K \quad \text{para } 0.1 \leq t \leq 1.0 \quad (5.8)$$

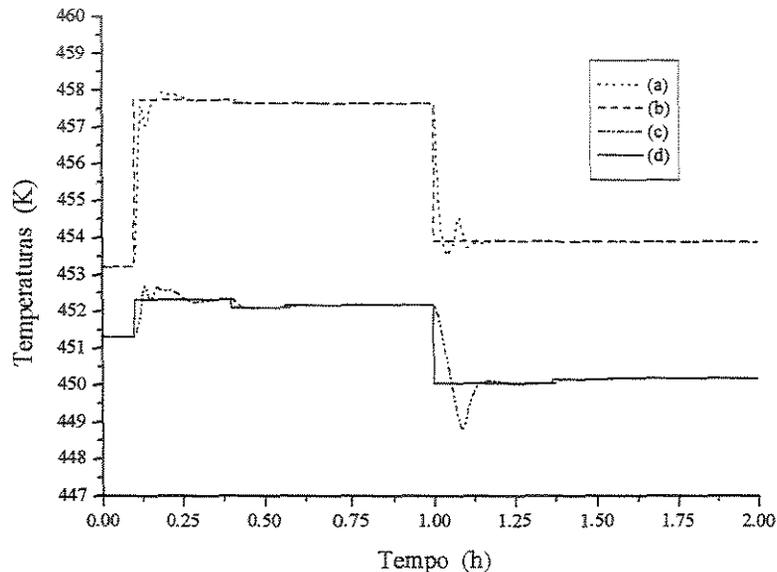


Figura 5.11.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{R0} - Estratégia A. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

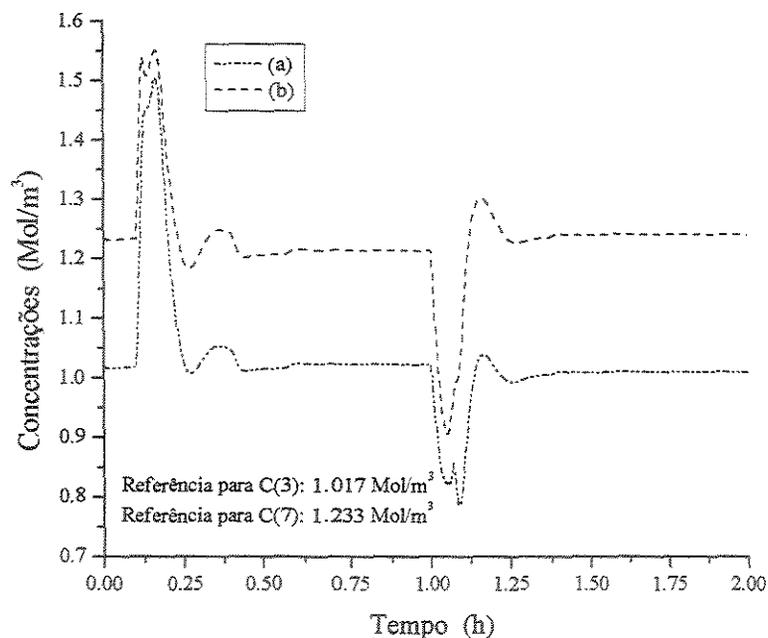


Figura 5.11.b – Concentrações para problema regulador - T_{R0} - Estratégia A. (a) Concentração C(3); (b) Concentração C(7).

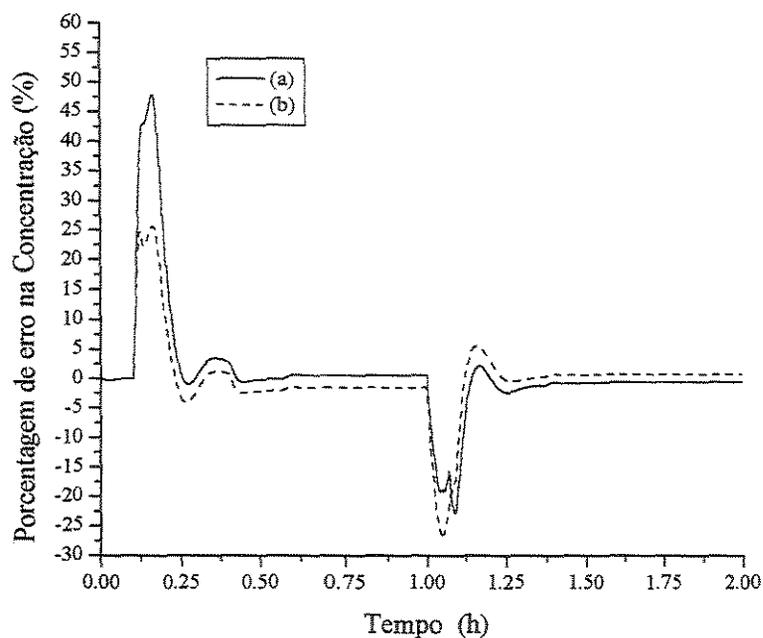


Figura 5.11.c - Erros nas concentrações para problema regulador - T_{R0} - Estratégia A. (a) Concentração C(3); (b) Concentração C(7).

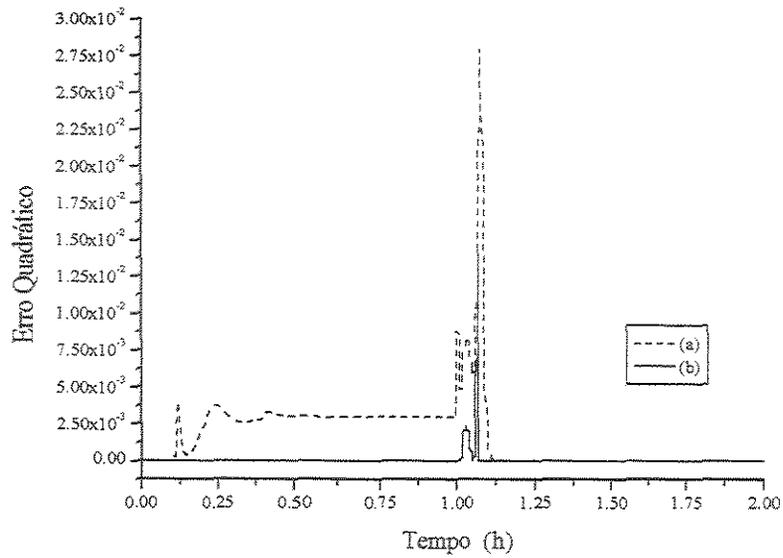


Figura 5.11.d – Erros quadráticos das redes neurais do controlador para problema regulador - T_{R0} - Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

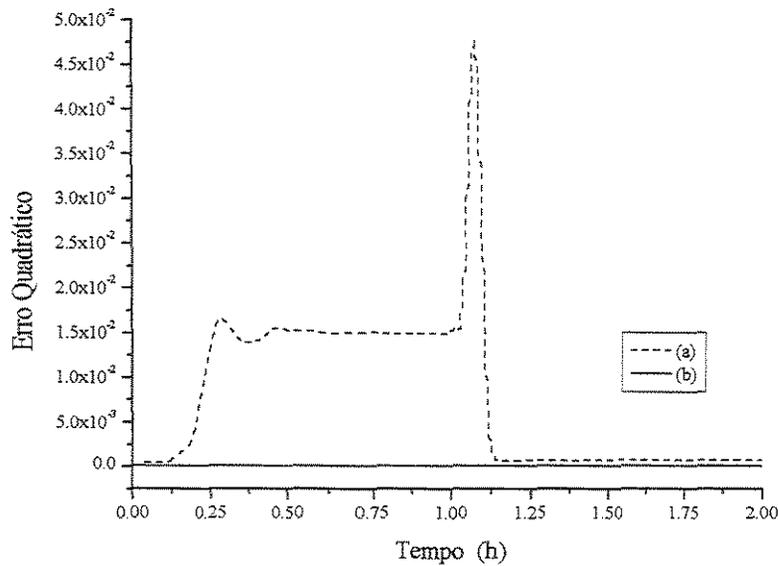


Figura 5.11.e - Erros quadráticos das redes neurais do modelo dinâmico para problema regulador - T_{R0} - Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

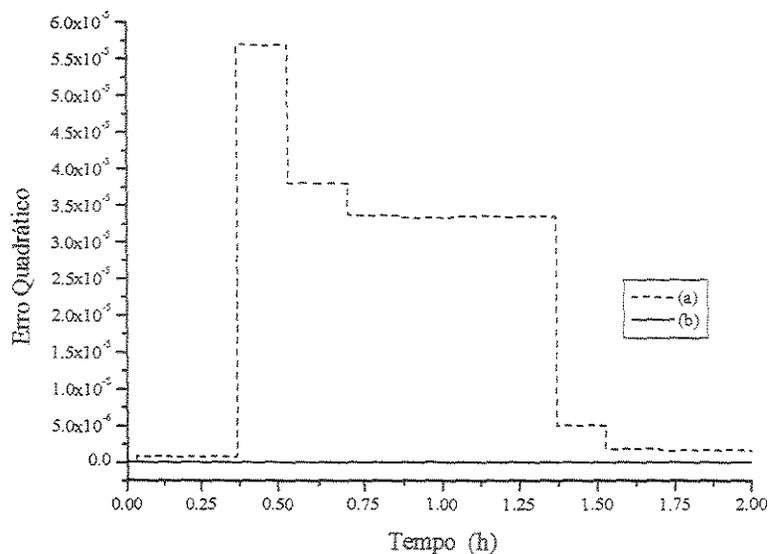


Figura 5.11.f - Erros quadráticos das redes neurais do modelo estático para problema regulador - T_{R0} - Estratégia A. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

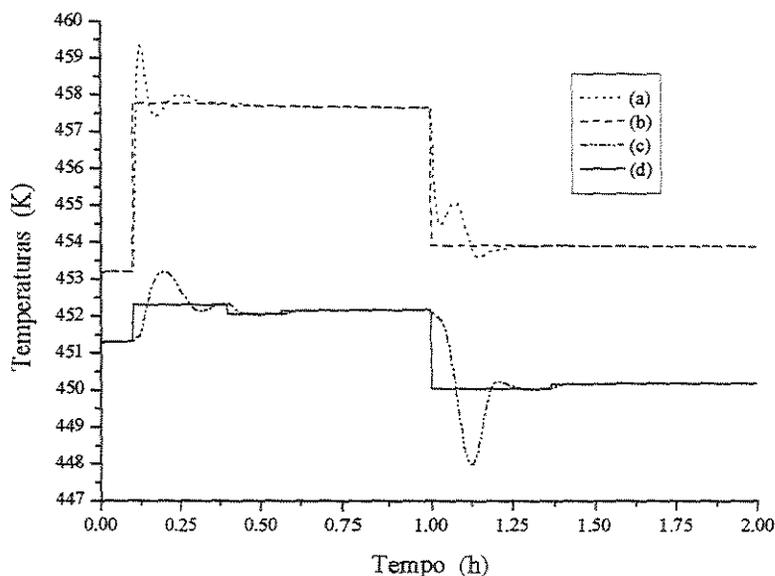


Figura 5.12.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{R0} - Estratégia B. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

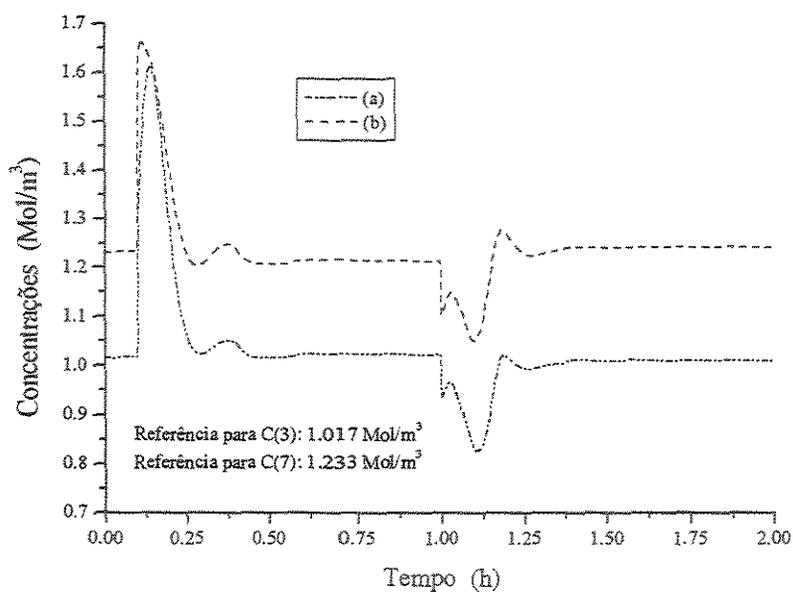


Figura 5.12.b – Concentrações para problema regulador - T_{R0} - Estratégia B. (a) Concentração C(3); (b) Concentração C(7).

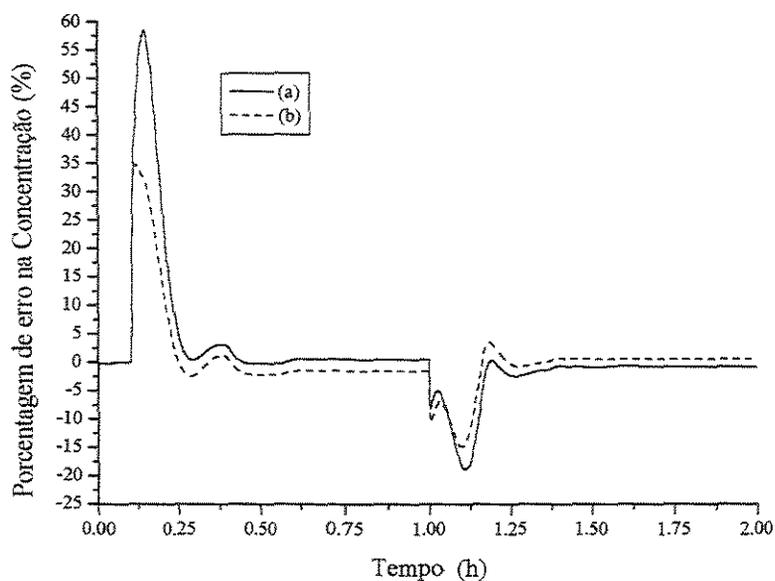


Figura 5.12.c - Erros nas concentrações para problema regulador - T_{R0} - Estratégia B. (a) Concentração C(3); (b) Concentração C(7).

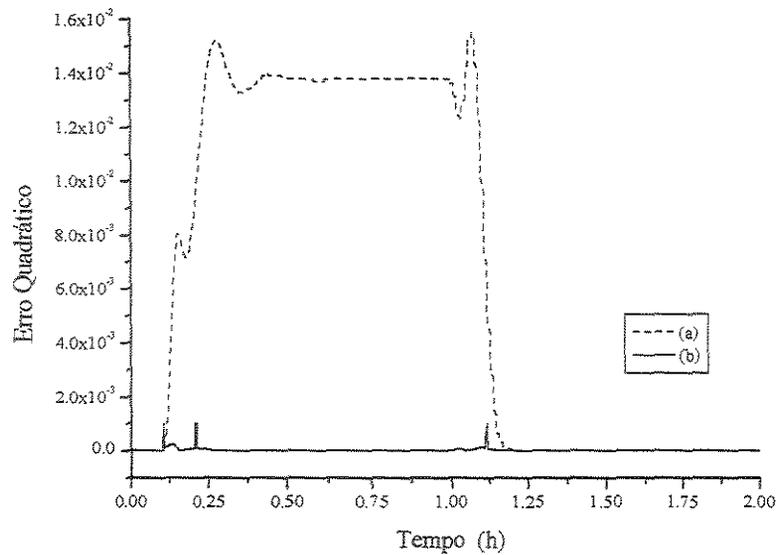


Figura 5.12.d - Erros quadráticos das redes neurais do modelo dinâmico para problema regulador - T_{R0} - Estratégia B. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

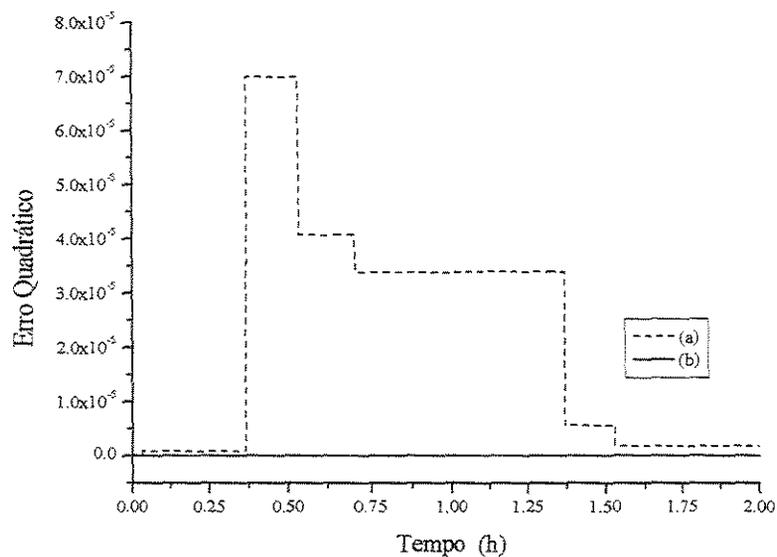


Figura 5.12.e - Erros quadráticos das redes neurais do modelo estático para problema regulador - T_{R0} - Estratégia B. (a) pesos da aprendizagem *off-line*; (b) pesos da aprendizagem *on-line*.

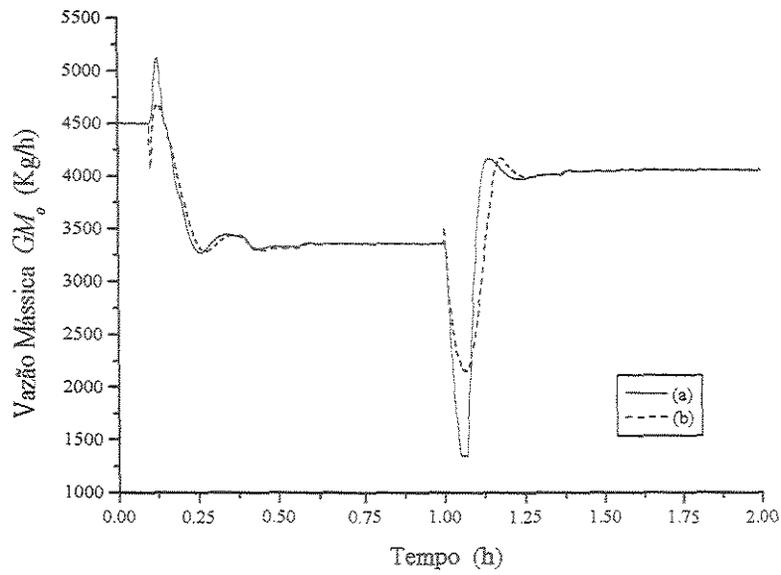


Figura 5.13.a – Ação de controle para problema regulador - T_{R0} . (a) Estratégia A; (b) Estratégia B

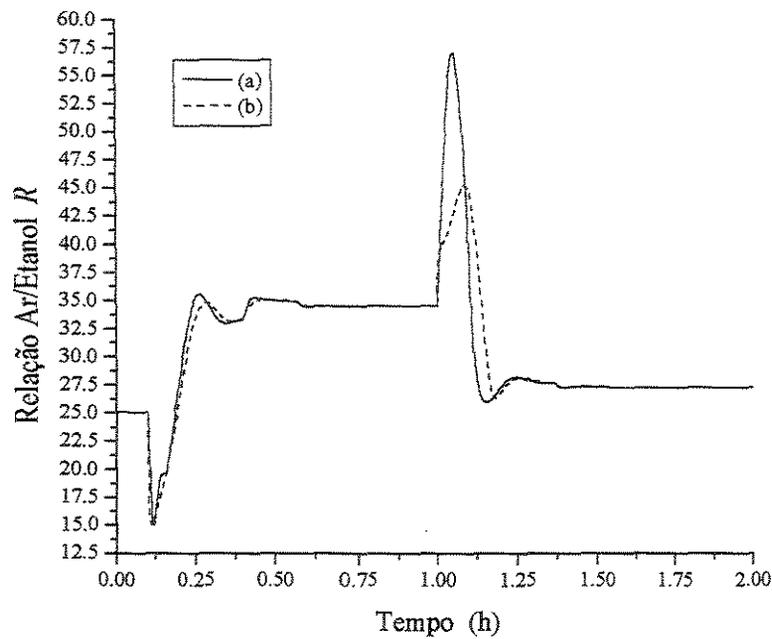


Figura 5.13.b – Ação de controle para problema regulador - T_{R0} . (a) Estratégia A; (b) Estratégia B.

Para os casos analisados até o momento, pode-se observar um excelente desempenho das estratégias de controle propostas, considerando o comportamento dinâmico complexo do reator em análise. Isto vem confirmar a eficiência das estratégias usadas, mostrando seu potencial e validando o uso de redes neurais em estratégias de controle, seja como controlador e/ou modelo do processo.

Objetivando explorar ainda mais as estratégias de controle de concentração propostas, simulou-se uma falha do sensor da temperatura de alimentação do reator, ou seja, foi gerada uma perturbação em T_{fo} , porém esta perturbação não é informada à estratégia de controle, simulando algo similar a uma falha na leitura deste parâmetro. Convém ressaltar que esta temperatura é parâmetro de entrada nas redes do modelo dinâmico e estático na Estratégia A e da rede neural do modelo estático na Estratégia B. Com isto, as redes neurais deverão incorporar esta perturbação como uma variação no comportamento do processo. Como perturbação em T_{fo} , adotou-se o seguinte:

$$T_{fo} = T_{foSS} + 5 K \quad \text{para} \quad t \geq 0.1 \quad (5.9)$$

Nos casos anteriores foi considerado uma amostragem de concentração a cada 10 minutos. Estas concentrações, juntamente com outros parâmetros do processo irão formar um padrão de aprendizagem, um vez satisfeito o critério adotado para definir se este conjunto de dados representa um estado estacionário. No caso de uma falha de sensor, a mudança no comportamento da rede neural estática que é usada na otimização da referência de temperatura, usa a informação do processo para reaprender um novo comportamento, considerando a falha de leitura e ajustar uma nova referência de temperatura que venha levar as concentrações do reator para os valores desejados. Durante o intervalo em que não ocorrer uma amostragem de concentração, porém o processo já sofreu a perturbação, nada irá ocorrer na estratégia de controle para corrigir esta falha de leitura, pois a rede ainda não está “ciente” deste fato.

As Figuras 5.14.a a 5.16.b apresentam os resultados considerando amostragem de concentração a cada 10 minutos e as Figuras 5.17.a a 5.19.b considera amostragem de

concentração a cada 5 minutos. A diminuição no tempo de amostragem de concentração está diretamente associado a velocidade de adaptação do modelo estático neural e por sua vez a velocidade do ajuste da referência de temperatura da malha fechada, corrigindo a falha do sensor.

As Figuras 5.14.a a 5.14.c apresentam os resultados considerando estratégia de controle baseada em redes neurais (Estratégia A) e tempo de amostragem de concentração de 10 minutos.

De acordo com a equação (5.9), a perturbação em T_{fo} ocorre para $t \geq 0.1$, e o que se observa na Figura 5.14.a, é um ajuste na referência da temperatura para um tempo superior a isto. Tal fato deve-se a necessidade da leitura de um conjunto de dados do processo que represente um estado estacionário já perturbado. As Figuras 5.14.b e 5.14.c apresentam os resultados para as concentrações do reator, bem como, o erro associado a esta resposta. Verifica-se um desvio acentuado na concentração nos instantes após ocorrer a perturbação e uma convergência para o valor desejado, uma vez adaptado o modelo do processo. A adaptação do modelo é extremamente rápida; o que demanda um tempo considerável é a obtenção de dados do processo.

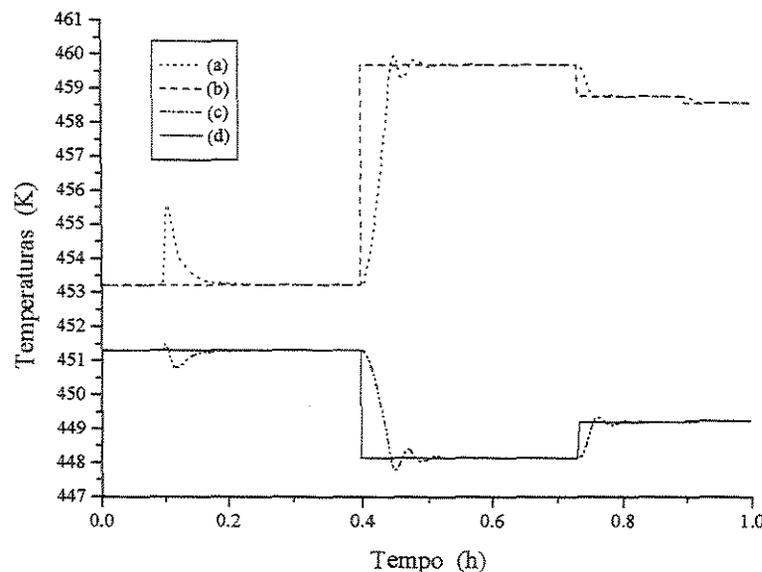


Figura 5.14.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{fo} (Estratégia A), considerando falha de sensor. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

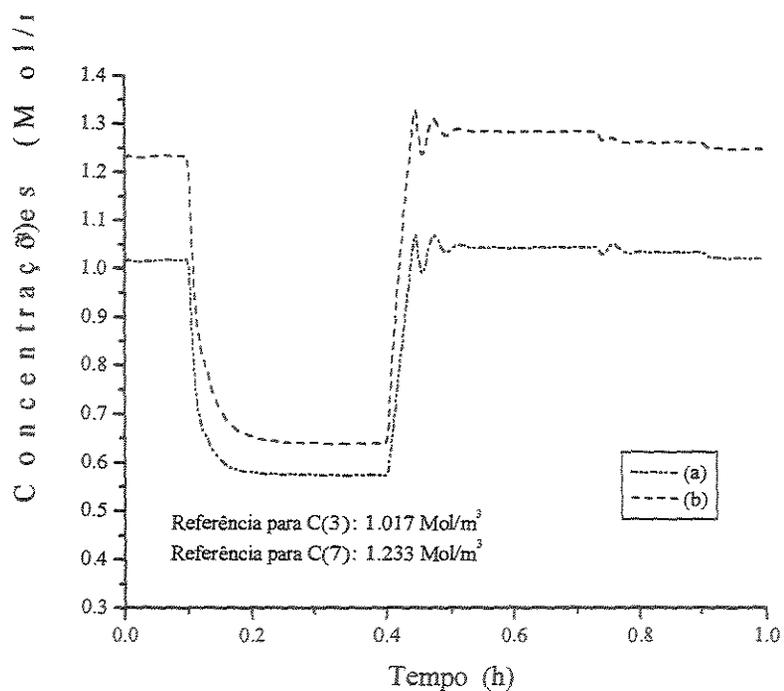


Figura 5.14.b – Concentrações para problema regulador - T_{fo} (Estratégia A), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

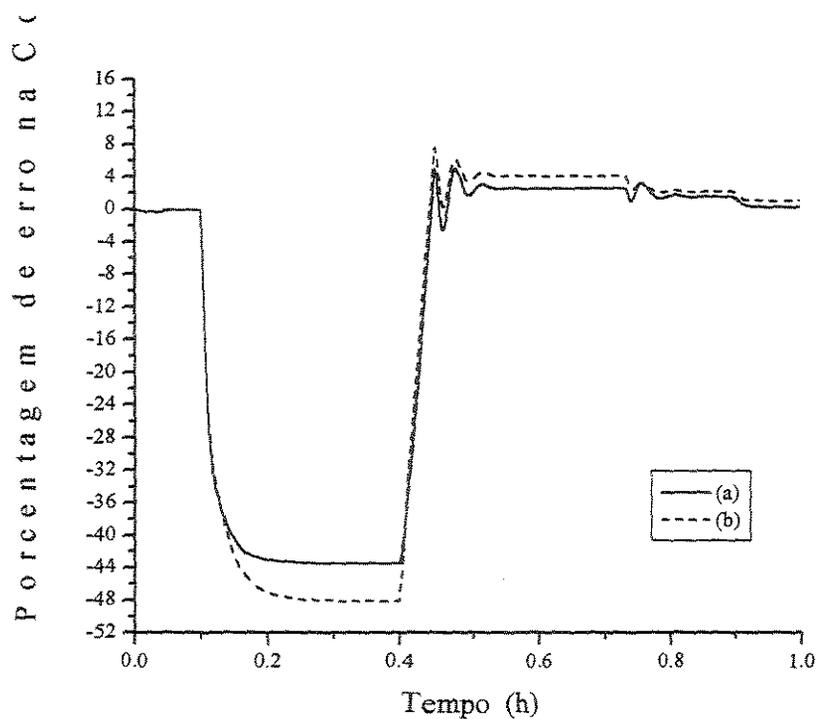


Figura 5.14.c - Erros nas concentrações para problema regulador - T_{fo} (Estratégia A), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

As Figuras 5.15.a a 5.15.c apresentam os resultados para a estratégia de controle preditivo com restrições (Estratégia B), nas mesmas condições apresentadas para o caso anterior.

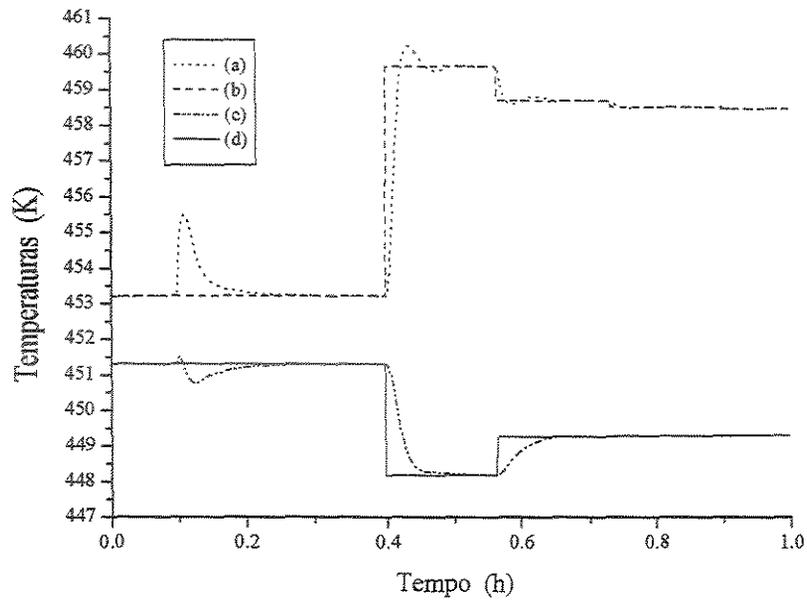


Figura 5.15.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{fo} (Estratégia B), considerando falha de sensor. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

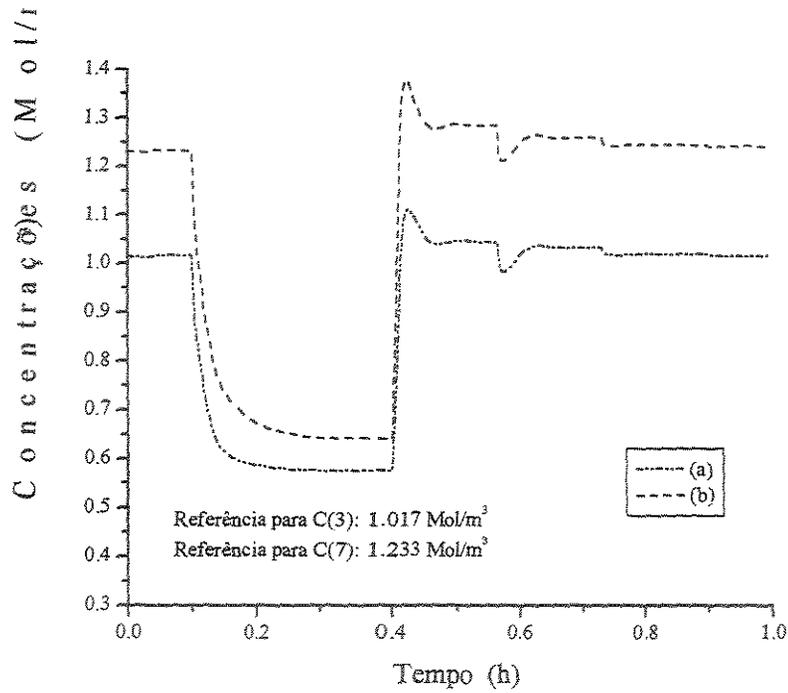


Figura 5.15.b – Concentrações para problema regulador - T_{fo} (Estratégia B), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

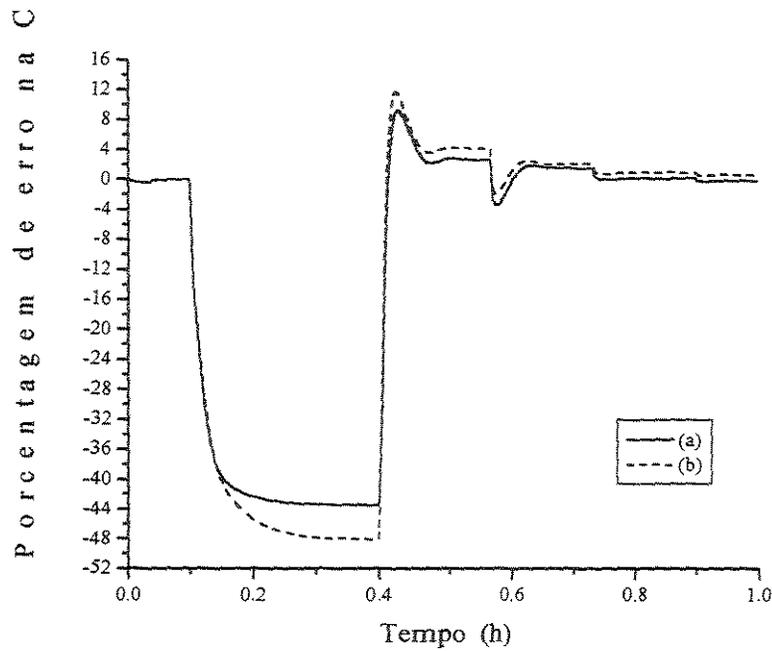


Figura 5.15.c - Erros nas concentrações para problema regulador - T_{fo} (Estratégia B), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

As Figuras 5.16.a e 5.16.b apresentam as ações de controle geradas para as duas estratégias de controle considerando o problema em questão

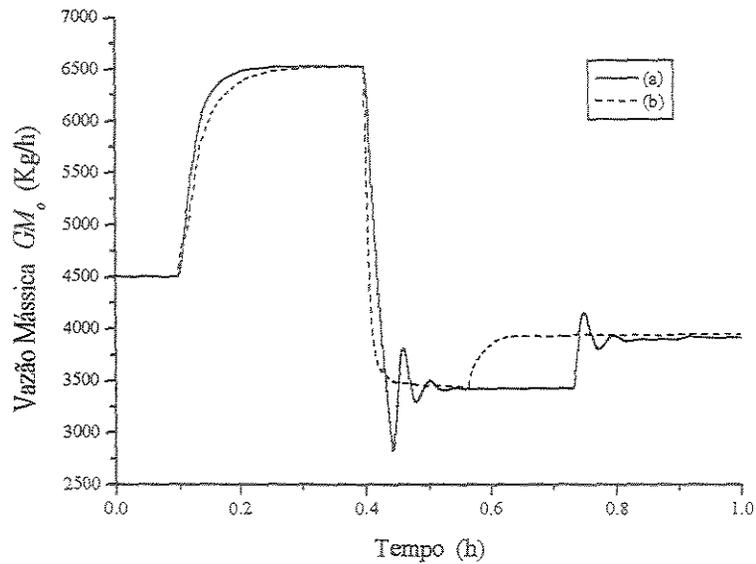


Figura 5.16.a – Ação de controle para problema regulador - T_{fe} , considerando falha de sensor. (a) Estratégia A; (b) Estratégia B.

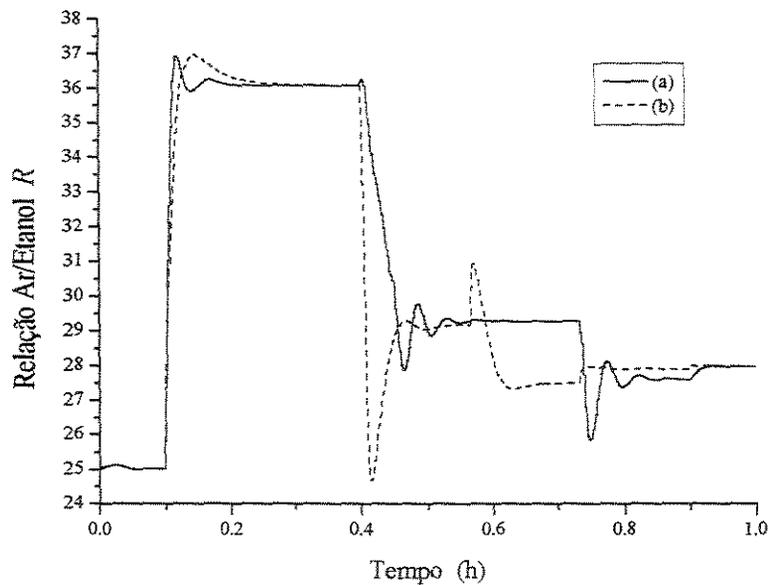


Figura 5.16.b – Ação de controle para problema regulador - T_{fe} , considerando falha de sensor. (a) Estratégia A; (b) Estratégia B.

As Figuras 5.17.a a 5.19.b apresentam os resultados considerando o tempo de amostragem de concentração igual a 5 minutos. Com isto ganha-se em qualidade das respostas devido a adaptação mais rápida das redes neurais representando o modelo estático. Dentre os resultados apresentados, as Figuras 5.17.a a 5.17.c referem-se a estratégia de controle baseada em redes neurais; as Figuras 5.18.a a 5.18.c referem-se a estratégia de controle preditivo e as Figuras 5.19.a e 5.19.b mostram as ações de controle obtidas.

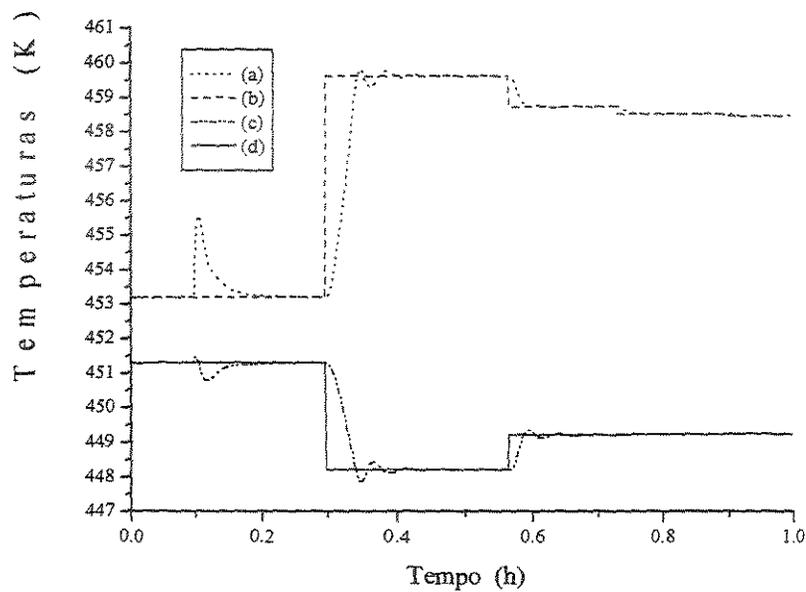


Figura 5.17.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{fo} (Estratégia A), considerando falha de sensor. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

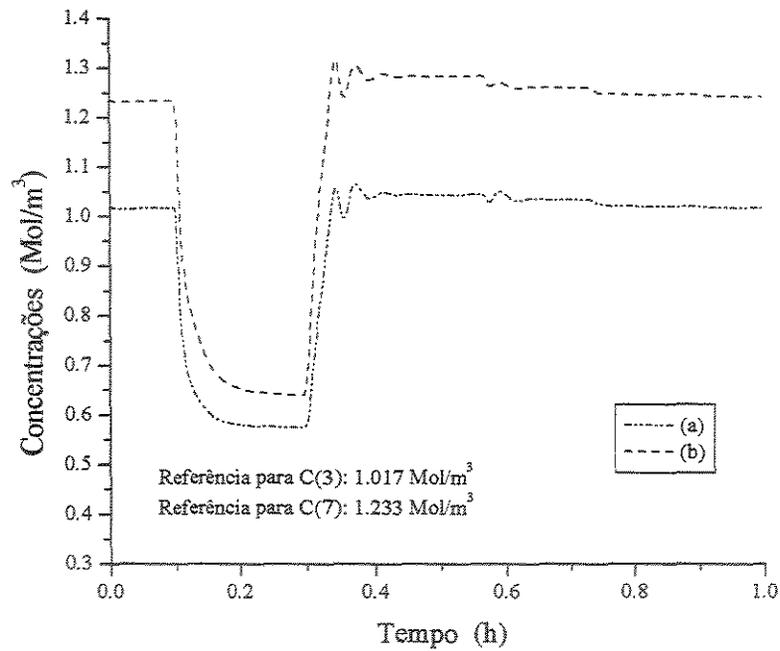


Figura 5.17.b – Concentrações para problema regulador - T_{fo} (Estratégia A), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

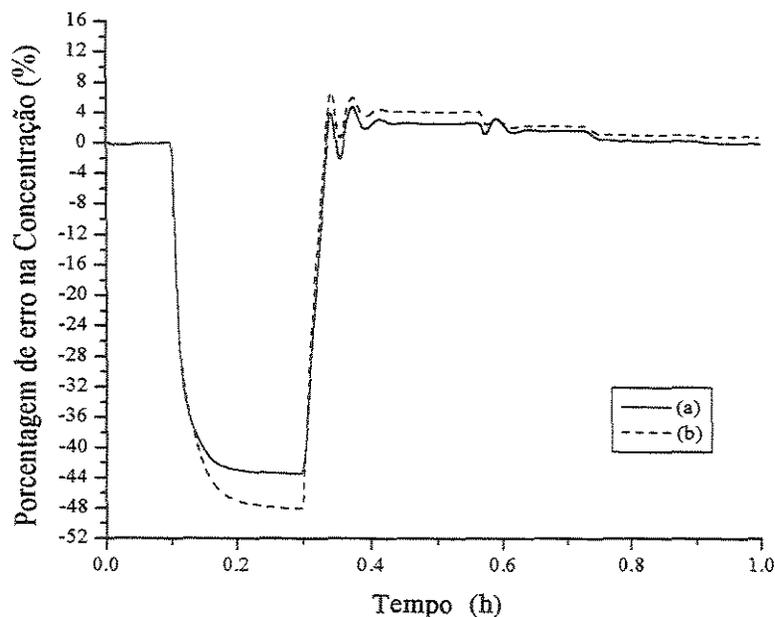


Figura 5.17.c - Erros nas concentrações para problema regulador - T_{fo} (Estratégia A), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

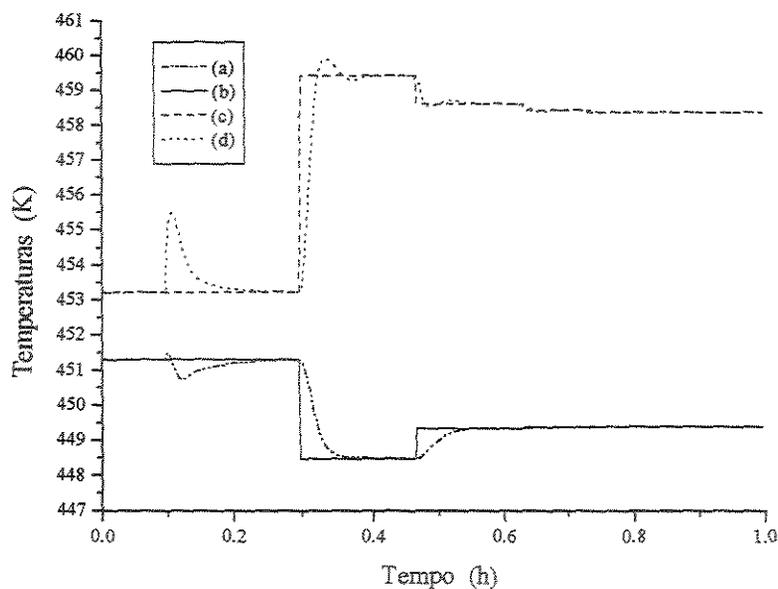


Figura 5.18.a - Resposta da malha fechada e referências de temperatura para problema regulador - T_{fc} (Estratégia B), considerando falha de sensor. (a) Temperatura T(1); (b) Referência para T(1); (c) Temperatura T(3); (d) Referência para T(3).

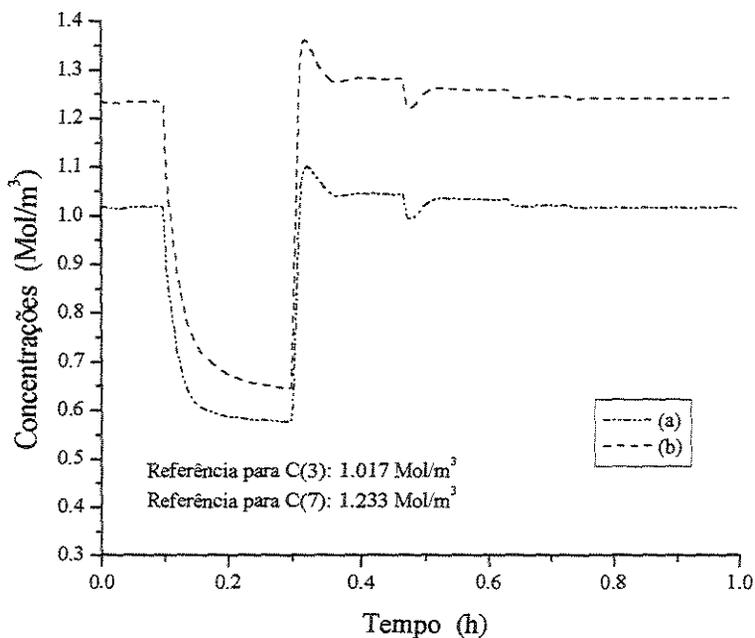


Figura 5.18.b - Concentrações para problema regulador - T_{fc} (Estratégia B), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

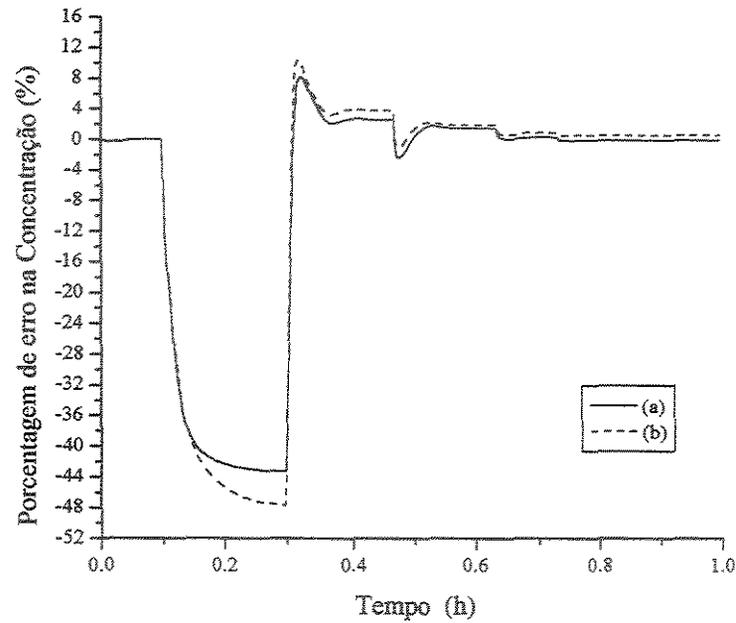


Figura 5.18.c - Erros nas concentrações para problema regulador - T_{fp} (Estratégia B), considerando falha de sensor. (a) Concentração C(3); (b) Concentração C(7).

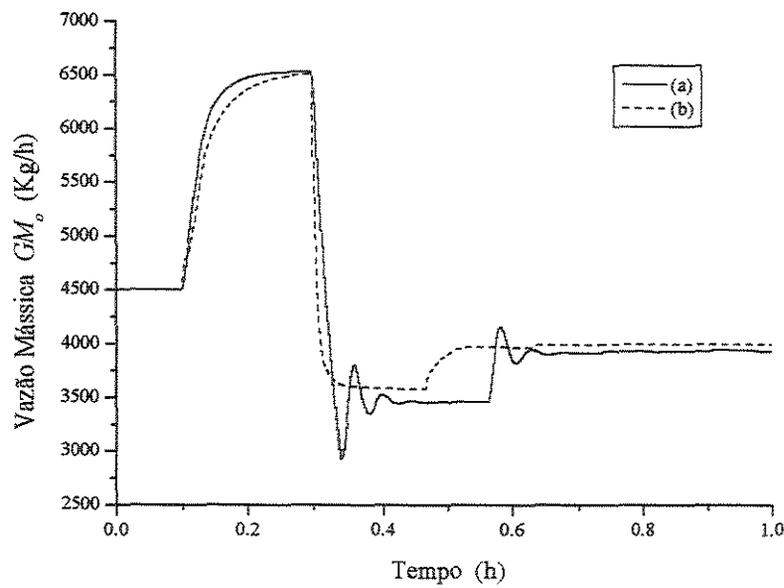


Figura 5.19.a - Ação de controle para problema regulador - T_{fp} , considerando falha de sensor. (a) Estratégia A; (b) Estratégia B.

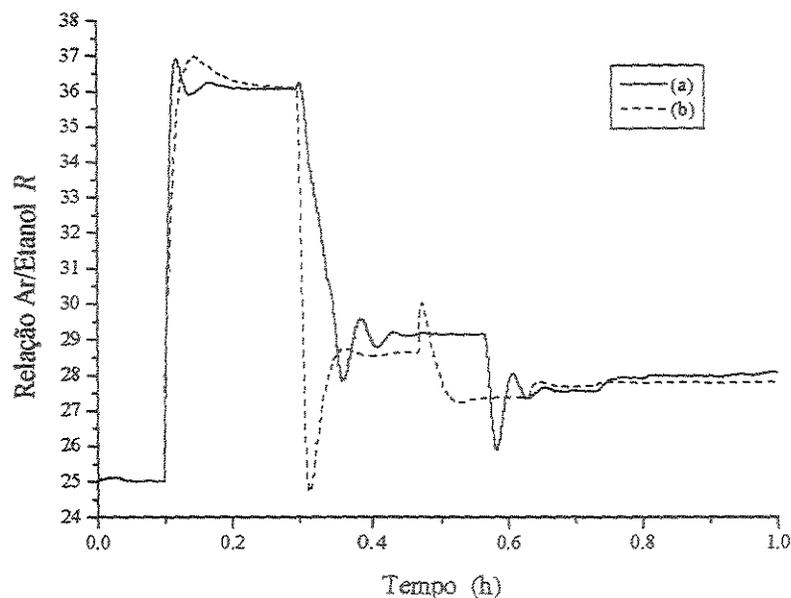


Figura 5.19.b – Ação de controle para problema regulador - T_{fc} , considerando falha de sensor. (a) Estratégia A; (b) Estratégia B.

5.4 Conclusões

O controle de um reator catalítico de leito fixo foi considerado neste caso de estudo. Estes sistemas apresentam comportamento dinâmico complexo e mesmo assim o desempenho da estratégia de controle de concentração tem apresentado resultados muito interessantes. Isto leva a concluir, que a abordagem considerada apresenta grande potencial para ser aplicada em processos com dinâmicas não lineares e complexas, mesma quando condições de operação severas são requeridas.

As redes neurais oferecem uma excelente alternativa como *soft sensor* ou controlador do sistema quando não existe um modelo matemático válido em todas as condições de operação, pois os modelos neurais usados necessitam somente de dados históricos de entrada/saída do processo.

A estratégia de aprendizagem *on-line* das redes neurais empregada tem apresentado resultados muito bons. No caso particular do modelo estacionário usado na predição das concentrações e temperaturas do processo para a definição das referências da malha fechada, optou-se em utilizar o conjunto de padrões contendo somente uma informação representando o estado estacionário atual do processo. A princípio isto pode significar uma limitação, pois a rede neural estaria sofrendo um treinamento muito pontual. Porém, deve-se levar em consideração que este procedimento é somente realizado se a rede neural não representar o padrão. Esta forma de implementar a aprendizagem ou “adaptação” *on-line* tem funcionado bem em todos os casos avaliados, pois permite uma adaptação mais rápida à realidade estacionária do processo, comparativamente se fosse usado um conjunto de padrões com mais informações. Além do mais, deve-se lembrar que a amostragem da concentração ocorre em tempos esparsos. Diante dos resultados obtidos e pelo exposto, pode-se afirmar que a estratégia proposta, apresenta potencial para ser implementada junto a estratégias de controle de parâmetros de difícil mensuração; neste caso, concentração.

A forma que as estratégias de controle foram desenvolvidas, considerando aprendizagem *on-line* das redes neurais, e com isto acompanhando as alterações do processo, permite contornar problemas que necessitam uma adaptação contínua do processo devido alterações dinâmicas, como desativação catalítica, ou problemas que possam ocorrer na malha fechada, tal como falha de sensor.

CAPÍTULO 6

CONCLUSÕES E SUGESTÕES PARA TRABALHOS FUTUROS

6.1 Conclusões

Após o estudo de redes neurais como ferramenta para modelagem de processos químicos e elaboração de estratégias de controle, pode-se concluir o seguinte:

- As redes neurais apresentam potencial na representação do comportamento dinâmico/estacionário do processo para uso em estratégias de controle;
- O uso do modelo dinâmico neural em substituição aos modelos clássicos (modelos de convolução) tem-se mostrado uma alternativa interessante a ser usada na geração das previsões futuras em estratégias de controle preditivo. As formas propostas (Caso 1, Caso 2, Caso 3) de apresentação dos dados à rede neural promove a inserção de ação integral devido a forma incremental de tratamento dos dados;
- A aprendizagem *on-line* proposta apresentou resultados apropriados nas diversas simulações realizadas, seja como modelo do processo ou como controlador;

- A estratégia de controle preditivo com restrições, usando modelo dinâmico neural mostrou-se uma excelente estratégia de controle. Apesar dos procedimentos de aprendizagem *on-line* das redes neurais envolvidas e o procedimento de otimização usando SQP, a cada tempo de amostragem, isto não mostrou-se um fator limitante em termos do tempo necessário para executar estas tarefas;
- Várias formas de utilizar as redes neurais em estratégias de controle, como controladores e/ou modelos de processo, foram avaliadas, o que permite afirmar que as redes neurais oferecem uma alternativa viável para utilização em sistemas de identificação ou projeto de controladores, quando não existe um modelo matemático determinístico confiável, ou somente dados históricos do processo estão disponíveis.
- As estratégias de controle baseadas no modelo inverso, usando diretamente dados de perturbação/resposta do processo na aprendizagem das redes neurais, aprendizagem indireta, apresentam o problema de deixar *off-set* na malha fechada, nos casos em que a variável controlada apresentar resposta inversa considerável diante de perturbações ocorridas. Este fato está relacionado à aprendizagem *on-line*, porém nesta tese, é proposta uma forma de aprendizagem, denominada de “projeto do controlador”, contornando perfeitamente estes problemas.
- Avaliou-se o uso de estratégias MultiSISO baseado no modelo inverso e aprendizagem *on-line* indireta, e pode-se afirmar que esta estratégia é uma alternativa interessante nos casos em que não existirem muitas interações entre as malhas de controle. Esta técnica permite trabalhar-se com várias redes neurais menores, o que facilita a aprendizagem individual de cada rede, e não necessariamente, implica em tempo computacional menor;
- A estratégia adaptativa proposta, apresentou excelente resultados em todas os casos avaliados, o que vem comprovar a eficiência da metodologia usada de aprendizagem das redes neurais do controlador. A utilização do erro global estimado mostrou-se eficiente na aprendizagem do controlador baseado na dinâmica inversa do processo;
- As estratégias propostas de aprendizagem *on-line* das redes neurais atuando como controladores *feedback* tem-se mostrado eficientes, a exceção dos problemas gerados devido o sinal de aprendizagem das redes neurais usando modelo inverso, na presença

de resposta inversa do processo. Este problema não diz respeito a técnica de aprendizagem das redes do controlador, mas sim, da rede neural que gera o sinal de aprendizagem;

- As redes neurais oferecem uma excelente alternativa como *soft sensor* para ser incorporados em estratégias de controle de variáveis de difícil mensuração. A forma implementada, considerando-se o número reduzido de padrões de aprendizagem, permite representar adequadamente o processo, gerando referências apropriadas para a malha fechada;

6.2 Sugestões para Trabalhos Futuros

Com o objetivo de dar continuidade aos estudos realizados, alguns trabalhos interessantes são sugeridos para serem explorados, tais como:

- Estudo e avaliação de outras arquiteturas de redes neurais e técnicas de aprendizagem, a serem implementadas dentro das mesmas configurações propostas nesta tese;
- Estudo e avaliação de técnicas de análise de estabilidade da malha fechada usando as estratégias de controle propostas. Alguns trabalhos que podem ser adotados como base teórica inicial encontram-se publicados, dentre estes LEVIN & NARENDRA (1993), NIKRAVESH et al. (1997), CAÑETE et al. (1998) e ZHANG & HANG (1999);
- No desenvolvimento desta tese, teve-se a preocupação em gerar algoritmos com poucos parâmetros ajustados pelo operador. Dentro deste contexto, seria interessante explorar técnicas de ajuste automático destes parâmetros. Uma idéia que vem sendo explorada é a utilização do filtro de Kalman (Ender et al., 2001⁶);
- Com o intuito de otimizar a quantidade e qualidade das conexões em uma rede neural, sugere-se o estudo e implementação de técnicas apropriadas de eliminação de entradas e neurônios insignificantes nas respostas, de acordo com trabalho apresentado por HENRIQUE et al (2000);

- Uma sugestão que provavelmente será a continuidade natural deste trabalho de pesquisa é a implementação em plantas piloto das estratégias avaliadas nesta tese.

CAPÍTULO 7

REFERÊNCIAS BIBLIOGRÁFICAS

AIBA, S., HUMPHREY, A.E., MILLIS, N.F. *Engenharia Bioquímica*. Campinas: Fundação Centro de Pesquisas e Tecnologia de Alimentos, 1971.

ASSIS, A. J., MACIEL FILHO, R. Soft sensor development for on-line bioreactor state estimation. *Computers & Chemical Engineering*, v.24, p.1099-1103, 2000.

ÅSTRÖM, K. J., WITTENMARK, B. *Computer Controlled Systems – Theory and Design*. New Jersey: Prentice Hall, 1984.

ATALLA, M. J., INMAN, D. J. On model updating using neural networks. *Mechanical Systems and Signal Processing*, v. 12, n. 1, p. 135 – 161, 1998.

- BADINO Jr., A.C. *Contribuição ao Estudo de Agitação e Aeração de Caldos de Fermentação Penicilínica*, São Carlos, SP, 1991. Dissertação (Mestrado em Engenharia Química). PPG-EQ/UFSCar.
- BADINO Jr., A.C.; BARBOZA, M.; BELLUZZO, R.P.; HOKKA, C.O. Caracterização Reológica e Verificação da Influência da Massa Celular na Reologia de Caldos de Fermentação com *Penicillium chrysogenum* e *Cephalosporium acremonium*. *Revista Brasileira de Engenharia, Caderno de Engenharia Química*, v. 11, n. 1, p. 43 - 60, 1994.
- BARROS NETO, B., SCARMINIO, I. S., BRUNS, R. E. *Planejamento e Otimização de Experimentos*, Campinas: Editora da Universidade Estadual de Campinas, 1995, 299p.
- BARTO, A. G. Connectionist Learning for Control: An Overview, p. 5-56, in *Neural Networks for Control*, W. T. Miller, R. S. Sutton and P. J. Werbos, Editors, MIT Press, Cambridge, MA, 1990.
- BAJPAI, R. K., REUSS, M. A Mechanistic Model for Penicillin Production. *Journal of Chemical Technology Biotechnology*, v. 30, p. 332 - 340, 1980.
- BAKSHI, B. R., STEPHANOPOULOS, G. Wave-net: a Multiresolution, Hierarchical Neural Network with Localized Learning. *AIChE Journal*, v. 39, n. 1, 1993.
- BAUGHMAN, D. R., LIU, Y. A. *Neural Networks in Bioprocessing and Chemical Engineering*, London: Academic Press, 1995, 488p.

- BHAT, N., MINDERMAN, P. A., McAVOY, T., WANG, N. S. Modeling chemical process systems via neural computation. *IEEE Control Systems Magazine*, v. 10, n. 3, p. 24-30, April, 1990.
- BHAT, N., McAVOY, T. Use of neural nets for dynamic modeling and control of chemical process systems. *Computer Chemical Engineering*, v. 14, n. 4/5, p. 573-582, 1990.
- BOŠKOVIĆ, J. D., NARENDRA, K. Comparison of Linear, Nonlinear and Neural-network-based Adaptive Controllers for a Class of Fed-batch Fermentation Process. *Automatica*, v. 31, n. 6, p. 817 – 840, 1995.
- BOX, G. E. P., HUNTER, W. G. HUNTER, J. S. *Statistics for experimenters. Na introduction to design, data analysis and model building*. Nova York: Wiley, 1978.
- BULSARI, A. B. *Neural Networks for Chemical Engineers*. Amsterdam: Elsevier, 1995. 680p.
- CAÑETE, J. F., GARCIA-CEREZO, A., GARCIA-MORAL, I. Direct control with radial basis function network: analysis and applications. *Journal of Systems Architecture*, v. 44, p. 583 – 596, 1998.
- CHEN, F.-C. Back-Propagation Neural Networks for Nonlinear Self-Tuning Adaptive Control. *IEEE Control Systems Magazine*, v. 10, n. 3, p. 44 – 48, April, 1990
- CHEN, S., BILLINGS, S. A. Neural networks for nonlinear dynamic system modeling and identification. *Int. J. Control*, v. 56, n. 2, p. 319 – 346, 1992.

- CHENG, Y., KARJALA, T. W., HIMMELBLAU, D. M. Identification of nonlinear dynamic processes with unknown and variable dead time using an internal recurrent neural network. *Ind. Eng. Chem. Res.*, v. 34, n. 5, p. 1735 – 1742, 1995.
- CLARKE, D. W., MOHTADI, C., TUFFS, P. S. Generalized Predictive Control - Parts I and II. *Automatic*, v. 23, n. 2, p. 137-160, 1987.
- CONTOIS, D.E. Kinetics of Bacterial Growth: Relationship between Population Density and Specific Growth Rate of Continuous Cultures. *Journal of General Microbiology*, v. 21, p. 40 - 50, 1959.
- CUTLER, C. R., RAMAKER, B. L., Dynamic Matrix Control - A Computer Control Algorithm, *AIChE 86th National Meeting*, Houston, TX, April, 1979.
- CYBENKO, G. Approximation by Superpositions of a Sigmoidal Function. *Mathematics of Control, Signals, and Systems*, v. 2, p. 303 – 314, 1989.
- DE SOUZA JR., M. B. *Redes Neurais Multicamadas Aplicadas a Modelagem e Controle de Processos Químicos*. Rio de Janeiro, RJ, 1993. 124p. Tese (Doutorado em Engenharia Química). PEQ/COPPE/UFRJ.
- DE SOUZA JR., M. B., PINTO, J. C., LIMA, E. L. Control of a chaotic polymerization reactor: a neural network based model predictive approach. *Polymer Engineering and Science*. v. 36, n. 4, p. 448 – 457, February, 1996.
- DOMINGUES, A., MACIEL FILHO, R. A Multitubular Reactor for obtain of Acetaldehyde by Oxidation of Ethyl Alcohol. *Chemical Engineering Science*, v. 47, p. 2571-2576, 1992.

ENDER, L., MACIEL FILHO, R. Utilização de redes neurais em estratégias de controle preditivo aplicado a um reator catalítico de leito fixo – Caso SISO, Anais do *XVIII Interamerican Congress of Chemical Engineering*, San Juan, Puerto Rico, 1998.

ENDER, L., MACIEL FILHO, R. Use of Neural Networks in Control Strategies Applied to a Fixed Bed Catalytic Reactor, Proceedings of *The Fourth Italian Conference on Chemical and Process Engineering*, Florence, Italy, 1999a.

ENDER, L., MACIEL FILHO, R. Design a New Multivariable Predictive Controller Coupling Neural Networks Applied to a Fixed Bed Catalytic Reactor, Anais do *II Congresso de Engenharia de Processos do Mercosul*, Florianópolis, Brasil, 1999b.

ENDER, L., MACIEL FILHO, R. Sistema de controle multivariável a partir de redes neurais, Anais do *II Congresso de Engenharia de Processos do Mercosul*, Florianópolis, Brasil, 1999c.

ENDER, L., MACIEL FILHO, R. Utilização de modelo dinâmico neural em estratégia de controle preditivo multivariável, Anais do *XIII Congresso Brasileiro de Engenharia Química*, Águas de São Pedro/SP, Brasil, 2000a.

ENDER, L., MACIEL FILHO, R. Design of multivariable controller aided by computer based on neural networks, Anais do *XIII Congresso Brasileiro de Engenharia Química*, Águas de São Pedro/SP, Brasil, 2000b.

ENDER, L., MACIEL FILHO, R. Design of multivariable controller based on neural networks. *Computers & Chemical Engineering*, v. 24, p. 937-943, 2000c.

ENDER, L., MACIEL FILHO, R. A Robust Procedure for on-line Learning of Neural Networks in Control Strategy Development, Proceedings of *3rd European Congress of Chemical Engineering*, Germany, 2001a.

ENDER, L., MACIEL FILHO, R. A robust procedure for on-line learning of neural networks in control strategy development – Part II, Proceedings of *6th World Congress of Chemical Engineering*, Melbourne, Austrália, 2001b.

ENDER, L., SCHEFFER, R., MACIEL FILHO, R. Computer design of a new predictive adaptive controller coupling neural networks and kalman filter, in *Computer Aided Chemical Engineering*, v. 9, p. 639 – 644, Elsevier, 2001.

ENDER, L., MACIEL FILHO, R. An innovative nonlinear adaptive control algorithm: Application to a fermentation process, Proceedings of *III Congresso de Engenharia de Processos do Mercosul*, Santa Fé, Argentina, 2001d.

ENDER, L., MACIEL FILHO, R. A predictive control strategy with constrains and soft sensor for concentration control of a fixed bed catalytic reactor, Proceedings of *The fifth Italian Conference on Chemical and Process Engineering*, Florence, Italy, 2001e.

ENDER, L., MACIEL FILHO, R. Procedure for the development soft sensors coupled a generic advanced control algorithm based on neural networks, Proceedings of *6th World Congress of Chemical Engineering*, Melbourne, Austrália, 2001f.

ENDER, L., MACIEL FILHO, R. Development of soft sensor Coupled to a generic advanced control algorithm based on neural networks, Proceedings of *6th IFAC Symposium on Dynamics and Control of Process Systems*, Cheju Island, Korea, 2001g.

- GIARRATANO, J. C., VILLARREAL, J. A., SAVELY, R. T. Future Impacts of Artificial Neural Systems on Industry, *ISA Transactions*, v. 29, n. 1, p. 9 – 14, 1990.
- GOZDZIEJEWSKI Jr, R., ENDER, L. Desenvolvimento de software didático em redes neurais artificiais. Anais do *Congresso Brasileiro de Engenharia Química*, Porto Alegre/RS, Brasil, 1998.
- HALLAGER, L., GOLDSCHMIDT, L., JØRGENSEN, S. B. Multivariable Adaptive Identification and Control of a Distributed Chemical Reactor., *Large Scale Systems*, v. 6, p. 323 – 336, 1984.
- HAYKIN, S. *Neural Networks - A comprehensive Foundation*. New Jersey: Prentice Hall, 1999.
- HECHT-NIELSEN, R. Theory of the Backpropagation Neural Network, *IEEE Int. Conf. On Neural Networks*, v. 1, p.593 - 605, 1989.
- HECHT-NIELSEN, R. *Neurocomputing*. New York: Addison-Wesley Publishing Company, Reading, MA, 1990.
- HORNIK, K., STINCHCOMBE, M., WHITE, H. Multi-layered feedforward neural networks are universal approximations. *Neural Networks*, v. 2, p. 359 – 366, 1990.
- HOSKINS, J. C. e HIMMELBLAU, D. M. Artificial Neural Network Models of Knowledge representation in chemical engineering, *Comput. Chem. Eng.*, v. 12, n. 9/10, p. 881-890, 1988.

HUMBERTO, H. M., LIMA, E. L., SEBORG, D. E. Model structure determination in neural network models, *Chemical Engineering Science*, v. 55, p. 5457 – 5469, 2000.

HUNT, K. J., SBARBARO, D., ZBIKOWSKI, R., GAWATHROP, P. J. Neural networks in Control Systems: A survey, *Automatica*, v. 28, n. 6, p.1083-1112, 1992.

ICHIKAWA, Y., SAWA, T. Neural network application for direct feedback controllers, *IEEE Transactions on Neural Networks*, v.3, n. 2, p. 224 – 231, March, 1992.

ISERMANN, R. *Digital Control Systems*. Berlin: Springer-Verlag, 1981.

KAVCHAK, M., BUDMAN, H. Adaptive neural network structures for non-linear process estimation and control, *Computers and Chemical Engineering*, v. 23, p. 1209 – 1228, 1999.

KRISHNAPURA, V. G., JUTAN, A. Arma neuron networks for modeling nonlinear dynamical systems. *The Canadian Journal of Chemical Engineering*, v. 75, p. 574 – 582, June, 1997.

KRISHNAPURA, V. G., JUTAN, A. A neural adaptive controller. *Chemical Engineering Science*, v. 55, p. 3803 – 3812, 2000.

KOLMOGOROV, A. N. *On the Representation of Continuous Functions*. URSS: Dokl. Acad. Nauk, 1957.

KOVÁCS, Z. L. *Redes Neurais Artificiais – Fundamentos e Aplcações*. São Paulo: Edição Acadêmica, 1996.

- LAPEDES, A., FARBER, R. Nonlinear Signal Processing Using Neural Networks: Prediction and System Modeling. *Los Alamos National Laboratory Technical Report LA-UR-87-2662*, July, 1987.
- LEE, M., PARK, S. A new scheme combining neural feedforward control with model-predictive control. *AIChE Journal*, v. 38, n. 2, p. 193 – 200, February, 1992.
- LEVIN, A. U., NARENDRA, K. S. Control of Nonlinear Dynamical Systems Using Neural Networks: Controllability and Stabilization. *IEEE Transactions on Neural Networks*, v.4, n. 2, p. 192 – 206, March, 1993.
- LOESCH, C., SARI, S. T. *Redes neurais: fundamentos e modelos*. Blumenau: FURB, 1996.
- LUYBEN, W. L. *Process Modeling Simulation and Control for Chemical Engineers*. New York: McGraw-Hill, USA, 1989.
- MACIEL FILHO, R. *Oxidação Catalítica de Etanol a Acetaldeído sobre Catalizador Óxido de Ferro-Molibdênio*, Campinas, SP, 1985. 117p. Dissertação (Mestrado em Engenharia Química). Faculdade de Engenharia Química, UNICAMP.
- MACIEL FILHO, R. *Modeling and Control of Multitubular Reactors*, Leeds, 1989. 300p. Thesis (Ph.D. in Chemical Engineering). Leeds.
- MACIEL FILHO, R., MCGREAVY, C., Dynamic of Fixed Bed Catalytic Reactor, Proceedings of *IFAC – Symposium of Dynamic and Control of Chemical Reactor, Distillation Columns and Batch process*, Maastricht, The Netherlands, p. 119 – 124, 1989.

- McCULLOCH, W. S. and W. PITTS. A Logical Calculus of the Ideas Immanent in Neurons Activity. *Bull. Math. Biophys.*, v. 5, p. 115, 1943.
- METZNER, A.B.; FEEHS, R.H.; RAMOS, H.P.; OTTO, R.E.; TUTHILL, J.D. Agitation of Viscous Newtonian and Non-Newtonian Fluids. *AIChE Journal*, v. 7, n. 1, p. 3 - 9, 1961.
- MELEIRO, L. A. C., MACIEL FILHO, R., A self-tuning adaptive control applied to an industrial large scale ethanol production. *Computers and Chemical Engineering*, v. 24, p. 925 – 930, 2000.
- MICHEL, B.J. & MILLER, S.A. Power Requirements of Gas-Liquid Agitated Systems. *AIChE Journal*, v. 8, v. 2, p. 262 - 266, 1962.
- MORRIS, A. J., MONTAGUE, G. A., WILLIS, M. J. Artificial neural networks: Studies in process modeling and control. *Trans. Inst. Chemical Eng.*, v. 72, n. 1, p. 3 – 19, January, 1994.
- NAJIM, K., RUSNAK, A., MESZAROS, A., FIKAR, M. Constrained long-range predictive control based on artificial neural networks. *International Journal of Systems Science*. v. 28, n. 12, p. 1211 – 1226, 1997.
- NAHAS, E. P., HENSON, M. A., SEBORG, D. E. Nonlinear internal model control strategy for neural network models. *Computers chem. Engng.*, v.16, n. 12, pp. 1039 – 1057, 1992.
- NARAYANAN, N. R. L., KRISHNASWAMY, P. R., RANGAIAH, G. P. An adaptive internal model control strategy for pH neutralization. *Chemical Engineering Science*, v. 52, n. 18, p. 3067 – 3074, 1997.

- NARENDRA, K., PARTHASARATHY, K. Identification and Control of Dynamical Systems Using Neural Networks. *IEEE Trans. On Neural Networks*, v. 1, n.1, p. 4 – 27, March, 1990.
- NARENDRA, K. S., MUKHOPADHYAY, S. Adaptive control of nonlinear multivariable systems using neural networks. *Neural Networks*, v. 7, n. 5, p. 737 – 752, 1994.
- NGUYEN, D. H., WIDROW, B. Neural Networks for Self-Learning Control Systems. *IEEE Control Systems Magazine*, p. 18 – 23, April, 1990.
- NICOLAI, B. M., VAN IMPE, J. F., VANROLLEGHEM, P. A., VANDEWALLE, J. A Modified Unstructured Mathematical Model for the Penicillin G Fed Batch Fermentation. *Biotechnology Letters*, v. 13, n. 7, p. 489 - 494, 1991.
- NIKRAVESH, M., FARELL, A. E., STANFORD, T. G. Dynamic neural network control for non-linear systems: optimal neural network structure and stability analysis. *Chemical Engineering Journal*, v. 68, p. 41 – 50, 1997.
- OHBA, T., ISHIDA, M. Application of a neural network to evaluation of interactions in a MIMO process. *AIChE Journal*, v. 44, n. 9, September, 1998.
- PARK, S., HAN, C. A nonlinear soft sensor based on multivariable smoothing procedure for quality estimation in distillation columns. *Computers and Chemical Engineering*, v. 24, p. 871 – 877, 2000.
- PSALTIS, D., SIDERIS, A., YAMAMURA, A. A Multilayered Neural Network Controller, *IEEE Control Systems Magazine*, p.17 – 21, April, 1988.

- PSICHOGIOS, D. C. and UNGAR L. H. Direct and Indirect Model-based Control Using Artificial Networks, *Industrial Engineering & Chemical Research*, v.30, n. 12, p. 2564-2573, 1991.
- RAMIREZ-BELTRAN, N. D., JACKSON, H. Application of Neural Networks to Chemical Process Control. *Computers & Industrial Engineering*, v. 37, p. 387 – 390, 1999.
- RODRIGUES, J.A.D. *Contribuição ao Estudo Cinético da Fase de Crescimento do Processo de Produção de Penicilina*, São Carlos, SP, 1992. Dissertação (Mestrado em Engenharia Química). PPG-EQ/UFSCar.
- RODRIGUES J. A. D., MACIEL FILHO, R. Production Optimization with Operating Constraints for a Feed-Batch Reactor with DMC Predictive Controller. *Chemical Engineering Science*, v.54, n. 13/14, p. 2745-2751, July, 1999.
- RODRIGUES, J. A. D. *Dinâmica e Estratégias de otimização e Controle do Processo de Penicilina*, Campinas, SP, 1996. Tese (Doutorado em Engenharia Química). Faculdade de Engenharia Química, UNICAMP.
- RUMELHART, D. E., McCLELLAND, J. L. *Parallel Distributed Processing: Explorations in the Microstructure of cognition - Volume 1: Foundations*. Cambridge: The Mit Press, 1986.
- RUSHTON, J.H.; COSTICH, E.W.; EVERETT, H.J. Power Characteristics of Mixing Impellers - Part I. *Chemical Engineering Progress*, v. 46, n. 8, p. 395 - 404, 1950a.

RUSHTON, J.H.; COSTICH, E.W.; EVERETT, H.J. Power Characteristics of Mixing Impellers - Part II. *Chemical Engineering Progress*, v.46, n. 9, p. 467 - 476, 1950b.

RUSNÁK, A., FIKAR, M., MÉSZÁROS. Receding horizon control using modified iterative dynamic programming and neural network models. *Computers and Chemical Engineering Supplement*. p. S297 – S300, 1999.

SAINT-DONAT, J., BHAT, N., McAVOY, T. Neural Net Based Model Predictive Control. *Int. J. Control*, v. 54, n. , 6, p. 1453 – 1468, 1991.

SANTOS, V. M. L., CARVALHO, F. R., De SOUZA Jr, M. B. Predictive control based on neural networks: na application to a fluid catalytic cracking industrial unit. *Brazilian Journal of Chemical Engineering*. v. 17, n. 04 – 07, p. 879 – 905, December, 2000.

SCHUMPE, A.; ADLER, I.; DECKWER, W.D. Solubility of Oxygen in Eletrolyte Solutions. *Biotechnology Bioengineering*, v. 20, p. 145 - 150, 1978.

SCOTT, G. M., RAY, W. H. Experiences with model-based controllers based on neural network process models. *J. Proc. Cont.*, v. 3, n. 3, p. 179 – 196, 1993.

SHAW, A. M., DOYLE III, F. J., SCHWABER, J. S. A dynamic neural network approach to nonlinear process modeling. *Computers chem. Engng*, v. 21, n. 4, p. 371 – 385, 1997.

SIMPSON, P. K. *Artificial Neural Systems: Foundations, Paradigms, Applications and Implementations*. New York: Pergamom Press, Inc., 1990.

- SMITH, J.J.; LILLY, M.D.; FOX, R.I. The Effect of Agitation on the Morphology and Penicillin Production of *Penicillium chrysogenum*. *Biotechnology Bioengineering*, v. 35, p. 1011-1023, 1990.
- SU, H. T., McAVOY, T. J., WERBOS, P. Long-term predictions of chemical process using recurrent neural networks: A parallel training approach. *Ind. Eng. Chem. Res.*, v. 31, n. 5, 1992.
- TOLEDO, E. C. V. *Modelagem, Simulação e Controle de Reatores Catalíticos de Leito Fixo*, Campinas, SP, 1999. Tese (Doutorado em Engenharia Química). Faculdade de Engenharia Química, UNICAMP.
- UNGAR, L. H., HARTMAN, E. J., KEELER, J. D., MARTIN, G. D., Process Modeling and Control Using Neural Networks, Proceedings of the *International Conference on Intelligent Systems in Process Engineering*, Snowmass, CO, 1995.
- QUANTRILLE, T. E. e LIU, Y. A. *Artificial Intelligence in Chemical Engineering*. San Diego, CA: *Academic Press*, 1991.
- QUICKER, G.; SCHUMPE, A.; KONIG, B.; DECKWER, W.D. Comparison of Measured and Calculated Oxygen Solubilities in Fermentation Media. *Biotechnology Bioengineering*, v. 23, p. 635 - 650, 1981.
- VEGA, M. P., LIMA, E. L., PINTO, J. C. Control of a loop polymerization reactor using neural networks. *Brazilian Journal of Chemical Engineering*, v. 17, n. 04 – 7, p. 471 – 481, December, 2000.

- VOGEL, H.C. *Fermentation and Biochemical Engineering Handbook*. New Jersey: Noyes Publications, 1983.
- WERBOS, P. J. Overview of Designs and Capabilities, in *Neural Networks for Control*, W. T. Miller, R. S. Sutton, P. J. Werbos, Editors, MIT Press, p.59-66, Cambridge, MA (1990).
- YDSTIE, B. E. Forecasting and Control Using Adaptive Connectionist Networks. *Comp. Chem. Engng.*, v. 14, n. 4/5, p. 583 – 599, 1990.
- ZHANG, T., GE, S. S., HANG, C. C. Design and performance analysis of a direct adaptive controller for nonlinear systems. *Automatica*, v. 35, p. 1809 – 1817, 1999.
- ZLOKARNIK, M. Sorption Characteristics for Gas-Liquid Contacting in Mixing Vessels. *Advances Biochemical Engineering*, v. 8, p. 133 - 151, 1978.
- ZYGIER, D., ARAÚJO, O Q. F., LIMA, E. L. Soft Sensors With White- and Black- Box Approaches for a Wastewater Treatment Process. *Brazilian Journal of Chemical Engineering*, v. 17, n. 04 – 07, p. 433 – 440, December, 2000.

ANEXO 1

MODELO DO REATOR CATALÍTICO DE LEITO FIXO

Os reatores de leito fixo são equipamentos muito utilizados na indústria química, principalmente em reações catalíticas. O controle efetivo desses reatores é fundamental para se obter operações seguras, especialmente quando alta performance é desejada. Estes reatores exibem complicado comportamento estacionário e dinâmico resultante das não-linearidades do sistema e da diferença de capacidades térmicas do sólido e do fluido, as quais entre outras coisas levam ao surgimento da resposta inversa e “runaway” da temperatura do reator (TOLEDO, 1999).

As equações que descrevem o comportamento dinâmico do reator catalítico de leito fixo foram baseados no modelo por MACIEL FILHO (1989) e TOLEDO (1999), desenvolvido com as seguintes considerações:

- Variação das propriedades físicas do fluido (densidade, viscosidade, condutividade térmica, capacidade calorífica, entalpia da reação, peso molecular, velocidade superficial), e coeficientes de transferência de calor e massa ao longo do reator;

- Perfil de velocidade “plug-flow”;
- Dispersão axial desprezível.

Diante destas considerações, tem-se o seguinte modelo matemático para o comportamento dinâmico do reator:

Balço de Massa:

$$\frac{\partial X}{\partial t} = \frac{D_{ef}}{R_i^2} \frac{1}{r} \frac{\partial}{\partial r} \left[r \frac{\partial X}{\partial r} \right] - \frac{GM_o}{\rho_f L} \frac{\partial X}{\partial z} + \frac{(1-\varepsilon)PM\rho_B}{\varepsilon\rho_f} R_w \quad (A1.1)$$

Balço de Energia:

$$\frac{\partial T}{\partial t} = \frac{\lambda_{ef}}{C_m R_i^2} \frac{1}{r} \frac{\partial}{\partial r} \left[r \frac{\partial T}{\partial r} \right] - \frac{\varepsilon GM_o C_{pf}}{C_m L} \frac{\partial T}{\partial z} + \frac{(1-\varepsilon)\rho_B(-\Delta H_R)}{C_m T_{ref}} \frac{R_w}{(R+1)} \quad (A1.2)$$

onde: $C_m = \varepsilon (\rho_f C_{pf}) + (1 - \varepsilon) (\rho_s C_{ps})$

Balço de Momento:

$$\frac{\partial p}{\partial t} = - \frac{GM_o}{\rho_f L} \left[\frac{\partial p}{\partial z} + f \right] \quad (A1.3)$$

Equação Continuidade

$$\frac{\partial}{\partial z}(\rho_f V_f) = 0 \quad (\text{A1.4})$$

Equação do Fluido Refrigerante

$$\frac{\partial T_R}{\partial t} = \frac{-u_R}{L} \frac{\partial T_R}{\partial z} + \frac{h_w}{\rho_R C_{PR}} (T(1, z) - T_R) \quad (\text{A1.5})$$

com as seguintes condições iniciais e de contorno:

$$r = 0 \quad \frac{\partial X}{\partial r} = \frac{\partial T}{\partial r} = 0 \quad (\text{simetria}) \quad (\text{A1.6})$$

$$r = 1 \quad \frac{\partial X}{\partial r} = 0, \frac{\partial T}{\partial r} = Bi_h(T_w - T_R) \quad (\text{A1.7})$$

para todo z

$$z = 0 \quad X = 0, T = \frac{T_{fo}}{T_{ref}}, p = \frac{P_o}{P_{ref}}, T_R = \frac{T_{R0}}{T_{ref}} \quad \text{para todo } r \quad (\text{A1.8})$$

R_w é a taxa de oxidação de etanol à acetaldeído sobre catalisador de Fe-Mo, (Kmol de mistura reagente/h Kg de cat.), MACIEL FILHO, 1985:

$$R_w = \frac{2K_1 K_2 P_{O_2} P_{ET}}{K_3 K_1 P_{ET} P_{AC} + K_1 P_{ET} + 2K_2 P_{O_2} + K_3 K_4 P_{AC} P_{H_2O}} \quad (\text{A1.9})$$

$$P_{N_2} = \frac{0.79R}{R+1+0.5X} \quad (\text{A1.10})$$

$$P_{O_2} = \frac{0.21R-0.5X}{R+1+0.5X} \quad (\text{A1.11})$$

$$P_{H_2O} = \frac{X}{R+1+0.5X} \quad (\text{A1.12})$$

$$P_{ET} = \frac{1-X}{R+1+0.5X} \quad (\text{A1.13})$$

$$P_{AC} = \frac{X}{R+1+0.5X} \quad (\text{A1.14})$$

onde P_{O_2} , P_{ET} , P_{H_2O} , P_{AC} são pressões parciais do oxigênio, etanol, água e acetaldeído respectivamente, P é a pressão total do sistema, e K_i , constantes cinéticas na forma de Arrhenius. Estas equações são resolvidas pelo método das linhas, onde as coordenadas axial e radial foram discretizadas por colocação ortogonal e a integração em relação ao tempo foi resolvida pelo método de Gear (TOLEDO, 1999).

Foram adotados os seguintes parâmetros estacionários para as simulações que seguem (DOMINGUES & MACIEL, 1992; TOLEDO, 1999), apresentados na Tabela A1.

Tabela A1 – Parâmetros estacionários do reator

Parâmetros	Valor estacionário
T_{fo}	445,15 K
T_{Ro}	445,15 K
GM_o	4500 kg/h
R	25
U_r	3,0 m/h
P_o	1 atm
D_t	0,017 m
D_p	0,002 m
L	1,0 m
$T(1)^*$	453,21 K
$T(3)^*$	451,30 K

(*) Variáveis resposta

As figuras a seguir apresentam o comportamento dinâmico do reator para os parâmetros mencionados, onde a resposta inversa e as não linearidades podem ser observadas. Tais características geram dificuldades adicionais no controle deste sistema, onde técnicas convencionais de controle tendem a falhar quando elevada performance operacional é requerida. A coordenada axial do reator foi discretizada usando 7 pontos de colocação ortogonal, e os pontos T(1) e T(3), são as temperaturas do primeiro e terceiro pontos de colocação ortogonal axial ao longo do reator.

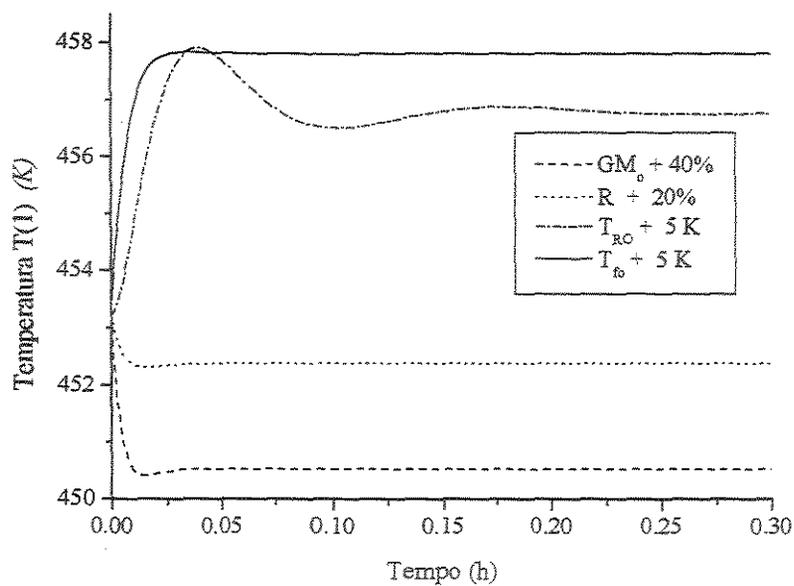


Figura A1.a – Temperaturas do primeiro ponto de colocação ortogonal axial para as perturbações especificadas

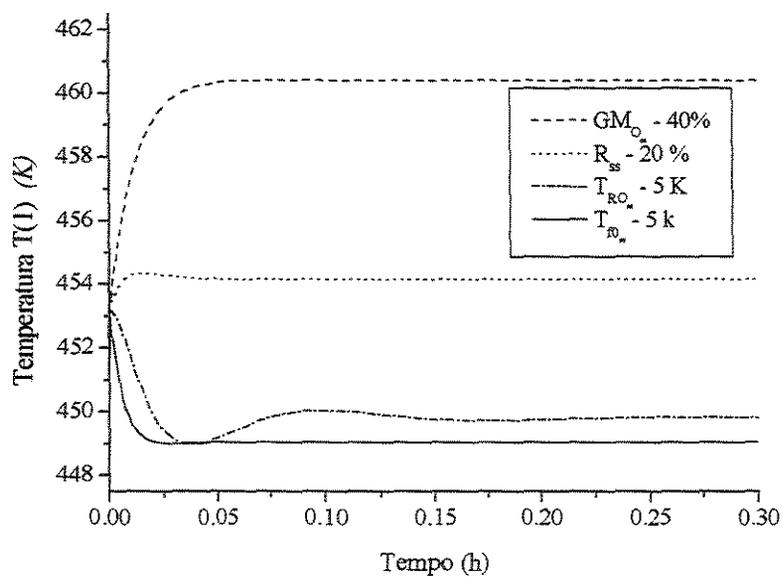


Figura A1.b – Temperaturas do primeiro ponto de colocação ortogonal axial para as perturbações especificadas

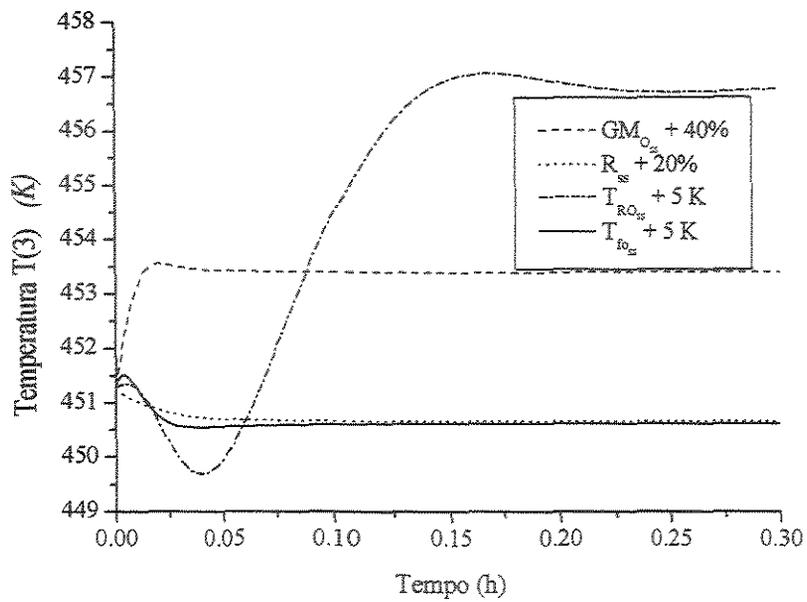


Figura A2.a – Temperaturas do terceiro ponto de colocação ortogonal axial para as perturbações especificadas

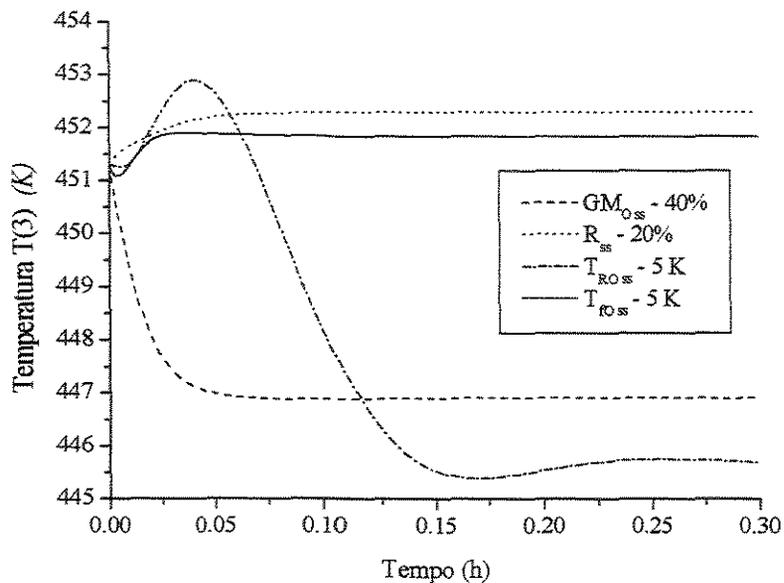


Figura A2.b – Temperaturas do terceiro ponto de colocação ortogonal axial para as perturbações especificadas

ANEXO 2

MODELO DO REATOR DE PRODUÇÃO DE PENICILINA

Este anexo apresenta a modelagem matemática do processo de produção de penicilina, considerando-as as fases de crescimento e produção, assumindo como quantidades fundamentais o crescimento celular, consumo de substrato, demanda de oxigênio, síntese do antibiótico, volume do meio fermentativo, considerando o sistema isotérmico.

O processo de produção de penicilina pode ser dividido em duas fases: fase de crescimento e a fase de produção propriamente dita. O modelo adotado nesta tese é o proposto por RODRIGUES (1996), o qual é descrito a seguir.

A fase de crescimento é a fase na qual o objetivo é o aumento da massa celular para, desta forma, obter quantidade suficiente de microrganismo de modo a tornar viável a fase seguinte. Tem-se o processo operando em batelada, com uma quantidade inicial considerável de substrato e outros nutrientes secundários, e uma pequena quantidade inicial de microrganismos (inóculo). A fluidodinâmica é a de um reator descontínuo ideal, perfeitamente agitado com densidade constante, havendo uma contínua aeração do sistema.

Obtido a massa celular desejável, parte-se para a fase de produção, onde ocorre a formação da substância de interesse – a penicilina.

A fase de produção se caracteriza por manter as condições operacionais da fermentação tais que a penicilina comece a ser sintetizada. O processo é realizado em batelada alimentada com relação ao substrato, uma vez que é de primordial interesse a manutenção de quantidades residuais mínimas desse substrato no caldo de fermentação, sendo este um fator crucial para a produção do antibiótico, pois é percebido que o microrganismo começa efetivamente a produzir esta substância quando se encontra em condições de carência nutricional. Também nesta etapa o processo é aeróbio, com a fluidodinâmica sendo descrita por uma mistura perfeita e densidade constante, sendo as condições do meio relevantes à estabilização do metabolismo no ponto ótimo, principalmente no tocante à concentração de oxigênio dissolvido, pH e temperatura.

9.1 Desenvolvimento do Modelo Matemático

As considerações assumidas para o desenvolvimento do modelo proposto são esquematizadas para as fases de crescimento e produção.

9.1.1 Fase de Crescimento

a) A velocidade específica de conversão de substrato à biomassa é fixada como sendo a equação cinética proposta por Contois (CONTOIS, 1959), pois esta se mostra mais adequada para processos onde, com o aumento da concentração celular ocorrem limitações difusionais com relação à transferência de massa, havendo a necessidade da variação da constante de saturação da equação, sendo o caso de sistema de fermentação com fungos típico desta ocorrência (BAJPAI & REUSS, 1980);

b) Nesta fase não se considera nem o conceito de metabolismo endógeno nem o de manutenção, pois devido ao tempo reduzido com que esta etapa ocorre e as condições do meio apresentadas, não é verificado a necessidade de assumir estes mecanismos (RODRIGUES, 1992);

c) A demanda específica de oxigênio por parte do microrganismo é assumida como sendo proporcional à velocidade específica de crescimento, considerando-se um fator de rendimento entre ambas análogo àquele da conversão de substrato à células, enfatizando que nesta fase não foi assumido a demanda de oxigênio devido aos metabolismos endógenos e/ou manutenção, nem devido a produção de antibiótico, justificando-se com base nos mesmos motivos do item anterior (RODRIGUES, 1992);

d) A produção específica de penicilina não foi considerada neste ponto, pois tem-se evidências que na fase de crescimento, onde a concentração de substrato é considerável na maior parte do processo, a síntese de antibiótico, caso ocorra, é mínima (RODRIGUES, 1992).

Portanto o modelo apresenta o seguinte conjunto de equações:

$$\frac{dX}{dt} = \mu \cdot X \quad (\text{A.2.1})$$

$$\frac{dS}{dt} = -\sigma \cdot X \quad (\text{A.2.2})$$

$$Q_{O_2} = \phi \cdot X \quad (\text{A.2.3})$$

$$\mu = \mu_{substr} = \mu_x \cdot \frac{C_S}{K_X \cdot C_X + C_S} \quad (\text{A.2.4})$$

$$\sigma = \frac{\mu}{Y_{X/S}} \quad (\text{A.2.5})$$

$$\phi = \frac{\mu}{Y_{X/O}} \quad (\text{A.2.6})$$

$$C_X = \frac{X}{V} \quad C_S = \frac{S}{V} \quad q_{O_2} = \frac{Q_{O_2}}{V} \quad (\text{A.2.7})$$

9.1.2 Fase de Produção

- (a) A velocidade específica de conversão de substrato à biomassa também é fixada como sendo a equação proposta por Contois, principalmente por ser nesta etapa do processo que ocorre a maior influência da concentração celular, causando as limitações difusionais citadas, dado o maior tempo de duração (BAJPAI & REUSS, 1980);
- (b) Existe uma gradativa transição entre os metabolismos de manutenção e endógeno como função da concentração de substrato, sendo que para concentrações desse substrato tendendo a zero o metabolismo endógeno é requerido, e para altos valores de concentração o metabolismo requerido é o de manutenção. Além disso, é possível ajustar a fração endógena para certos valores de concentração usando, tão poucos quanto possíveis, parâmetros adicionais, afim de evitar complicações desnecessárias no estudo de estimação de parâmetros (NICOLAI et al., 1991);
- (c) A demanda específica de oxigênio, nesta etapa, continua sendo considerada como proporcional ao crescimento, mas também considera-se a proporcionalidade entre esta demanda e o consumo de substrato devido aos metabolismos endógenos e/ou manutenção bem como devido à síntese de antibiótico, assumindo novamente os fatores de rendimento entre massa celular, oxigênio e substrato assimilados;

(d) Evidências bioquímicas sugerem que a biossíntese de penicilina pode estar sujeita a repressão por glicose. Embora o mecanismo exato, isto é, repressão ou inibição, não seja conhecido ainda, foi escolhido o modelo cuja velocidade específica de produção apresenta uma cinética de inibição pelo substrato. Além disso, foi considerado no modelo a hidrólise de penicilina à ácido penicilóico conforme uma cinética de primeira ordem com relação à penicilina (NICOLAI et al., 1991).

O modelo usado apresenta o seguinte conjunto de equações, entre cinéticas e de balanço:

$$\frac{dX}{dt} = \mu \cdot X \quad (\text{A.2.8})$$

$$\frac{dS}{dt} = -\sigma \cdot X + F \cdot S_F \quad (\text{A.2.9})$$

$$\frac{dP}{dt} = \pi \cdot X - Kh \cdot P \quad (\text{A.2.10})$$

$$\frac{dV}{dt} = F \quad (\text{A.2.11})$$

$$Q_{O_2} = \phi \cdot X \quad (\text{A.2.12})$$

$$\mu_{substr} = \mu_X \cdot \frac{C_S}{K_X \cdot C_X + C_S} \quad (\text{A.2.13})$$

$$\mu = \mu_{substr} - Y_{x/s} \cdot \left\{ F_m \cdot M_S + \frac{F_p \cdot \pi}{Y_{p/s}} \right\} \quad (\text{A.2.14})$$

$$\sigma = \frac{\mu_{substr}}{Y_{x/s}} + M_S \cdot \{1 - F_m\} + \pi \cdot \left\{ \frac{1 - F_p}{Y_{p/s}} \right\} \quad (\text{A.2.15})$$

$$\phi = \frac{\mu_{substr} + Y_{x/s} \cdot \left[M_S \cdot \{1 - F_m\} + \pi \cdot \left\{ \frac{1 - F_p}{Y_{p/s}} \right\} \right]}{Y_{x/o}} \quad (\text{A.2.16})$$

$$\pi = \mu_p \cdot \frac{C_S}{K_p + C_S + \frac{C_S^2}{K_i}} \quad (\text{A.2.17})$$

$$F_m = e^{\left(\frac{-C_S}{E_m} \right)} \quad (\text{A.2.18})$$

$$F_p = e^{\left(\frac{-C_S}{E_p} \right)} \quad (\text{A.2.19})$$

$$C_X = \frac{X}{V} \quad C_S = \frac{S}{V} \quad q_{O_2} = \frac{Q_{O_2}}{V} \quad (\text{A.2.20})$$

Os valores dos parâmetros envolvidos nestas equações encontram-se no Quadro A.2.1.

Quadro A.2.1 – Parâmetros envolvidos nas equações (A.2.1) à (A.2.20) (NICOLAI et al., 1991; RODRIGUES, 1992).

$\mu_x = 0.11 \text{ h}^{-1}$
$K_x = 0.006 \text{ g/g}$
$\mu_p = 0.00506 \text{ h}^{-1}$
$K_p = 0.0001 \text{ g/l}$
$K_i = 0.1 \text{ g/l}$
$K_h = 0.01 \text{ h}^{-1}$
$M_s = 0.029 \text{ g/g.h}$
$Y_{x/s} = 0.47 \text{ g/g}$
$Y_{p/s} = 1.2 \text{ g/g}$
$Y_{x/o} = 1.72 \text{ g/g}$
$E_m = 1.0 \text{ g/l}$
$E_p = 1.0 \text{ g/l}$

9.1.3 Fatores comuns às duas fases

Além das equações mostradas anteriormente, tem-se duas considerações que são comuns à ambas as fases: o perfil de concentração de oxigênio dissolvido e o perfil de temperatura.

9.1.3.1 Concentração de oxigênio dissolvido

Quanto ao perfil de oxigênio dissolvido no meio de fermentação, este é obtido fazendo-se um balanço de oxigênio, considerando-se o mesmo consumido devido a demanda metabólica por parte do microrganismo presente no processo e àquele dissolvido

no meio, sendo este proporcional ao gradiente de concentração de oxigênio na fase líquida. A constante de proporcionalidade, o chamado coeficiente volumétrico de transferência de oxigênio, é tido como função da geometria do fermentador (como razão entre o tamanho das aletas e diâmetros do tanque e do impelidor, tipo de impelidor, tipo de dispersor, etc.), condições operacionais (como velocidade de agitação e vazão de ar de entrada) e das propriedades físicas e morfológicas do caldo (como viscosidade aparente e densidade), sendo o meio, no caso em questão, de comportamento não newtoniano com características de pseudoplástico (AIBA et al., 1971).

A equação de balanço toma a forma:

$$\frac{d[C_{O_2} \cdot (C_{O_2}^* \cdot V)]}{dt} = -\phi \cdot X + K_L a \cdot C_{O_2}^* \cdot (1 - C_{O_2}) \cdot V \quad (\text{A.2.21})$$

sendo $K_L a$ o coeficiente de transferência de oxigênio, função da velocidade de rotação, vazão de aeração, fatores geométricos, e propriedades físicas. Ressalta-se que é usual a concentração de oxigênio dissolvido ser expressa em termos relativos da concentração de saturação (denotada pelo asterisco).

Esta equação é de fundamental importância para a aplicação de estratégias de controle com respeito à manutenção de quantidades pré-determinadas de oxigênio dissolvido, relativas ao valor de saturação, superiores a 30%, valor considerado limite mínimo para o bom andamento do processo em termos de comportamento metabólico RODRIGUES & MACIEL FILHO (1999). Além disso, existe uma restrição no sentido de se evitar o excesso de agitação, devido a acentuada ruptura causada nos micélios, o que acredita-se diminuir a produção (SMITH et al., 1990).

A viscosidade aparente foi estimada com base em duas equações. Àquela utilizada por METZNER et al. (1961) para o cálculo da taxa de cisalhamento em tanques agitados para fluidos com comportamento pseudoplástico (sendo $K' = 11.5$):

$$\gamma = K' \cdot N \quad (\text{A.2.22})$$

onde N representa a velocidade de rotação (RPS).

O cálculo da viscosidade aparente, baseada na lei da potência (AIBA et al., 1971), com os valores dos índices de consistência e comportamento ajustados por BADINO Jr. Et al. (1994) adaptada aos dados de BADINO Jr. (1991) em função da concentração celular, para caldos de fermentação de penicilina pode ser dada por:

$$\begin{aligned} \mu_{ap} &= K' \cdot \gamma^{n-1} \\ K &= 1.30 \cdot C_x^{1.02} \\ n &= 0.39 \end{aligned} \quad (\text{A.2.23})$$

A potência aplicada ao fluido de fermentação sem aeração foi determinada pela metodologia proposta por METZNER et al. (1961), através da função que relaciona o número de Reynolds modificado (Nrem) à fluidos não Newtonianos com o número de potência (NPO). Ressalta-se que esta função foi inicialmente ajustada à fluidos Newtonianos pelo trabalho de RUSHTON et al. (1950^{a,b}):

$$NRem = \frac{D_i^2 \cdot N \cdot \rho}{\mu_{ap}} \quad (\text{A.2.24})$$

$$NPO = \frac{Po}{N^3 \cdot D_i^5 \cdot \rho} \quad (\text{A.2.25})$$

Na determinação da potência requerida para a agitação do caldo de fermentação com aeração, utilizou-se a correlação do tipo proposta por MICHEL & MILLER (1962) ajustada no trabalho de BADINO Jr (1991).

$$Pg = 1.09 \cdot \left[\frac{Po^2 \cdot N \cdot D_i^3}{Q^{0.56}} \right]^{0.44} \quad (\text{A.2.26})$$

Finalmente, a determinação do coeficiente volumétrico de transferência de oxigênio foi feita com base na correlação proposta por ZLOKARNIZ (1978), também ajustada por BADINO Jr. (1991).

$$K_L a \cdot \frac{V}{Q} = 0.07 \cdot \left[\frac{Pg}{Q \cdot \rho \cdot (v_{ap} \cdot g)^{2/3}} \right]^{0.48} \quad (\text{A.2.27})$$

O Quadro A.2.2 apresenta os parâmetros envolvidos nas equações (A.2.21) à (A.2.33), conforme RODRIGUES (1996).

Quadro A.2.2 – Parâmetros usados nas simulações

$Co_2^* = 0.23 \text{ mmol } O_2/l = 7.36 \text{ mg/l}$ (SHUMPE et al., 1978; QUICKER et al., 1981)

$\phi \cdot X =$ calculada por (A.2.3) ou (A.2.12)

$K' = 11.5$

$D_i = 0.12 \text{ m}$

$g = 9.81 \text{ m/s}^2$

$Q(O_2) =$ calculada por (A.2.3) ou (A.2.12)

$Pg =$ calculado por (A.2.26) com 92% eficiência (VOGEL, 1983)

$\rho = \rho_e = \rho_j = 1000 \text{ Kg/m}^3$

$A = p \cdot 0.275 \cdot 0.4 = 0.346 \text{ m}^2$ (BADINO Jr., 1991)

$V_j = 0.346 \cdot 0.02 = 6.92 \text{ l}$ (BADINO Jr., 1991)

ANEXO 3

DMC - MIMO

O controle preditivo DMC (*Dynamic Matrix Control*) é considerado pela literatura como sendo um algoritmo de controle de grande potencial para aplicações industriais. O DMC é tido como uma estrutura de controle robusta e de simples implementação. O DMC caracteriza-se em usar uma modelagem de convolução discreta que relaciona as variáveis de saída com as variáveis de entrada, através de coeficientes de convolução, que são obtidos pôr variações degraus nas variáveis de entrada no processo. Com esta modelagem o DMC torna-se capaz de considerar a influência direta das interações entre as variáveis envolvidas, ou seja, é considerado o efeito da variação de uma determinada variável manipulada sobre todas as variáveis de saída. O modelo de convolução é obtido através de perturbações degraus unitários, aplicadas nas variáveis de entrada do sistema em malha aberta.

De posse dos elementos do modelo de convolução de cada par (variável de entrada versus variável de saída), pode-se obter o modelo de convolução, representado pela equação (A.3.1).

$$\hat{y}_{OL}(i, t+m) = \sum_{j=1}^{N_m} \sum_{k=0}^{(N_c-1)} h_{i,j}(m+1+k) \Delta u(j, t-1-k) \quad (\text{A.3.1})$$

No instante de amostragem presente ($m=0$), tem-se as medidas reais das variáveis controladas, $y(i, t)$, o qual também pode ser predito pelo modelo de convolução. A diferença entre os dois é usada para corrigir o dinamicamente modelo, proporcionando uma melhor predição no futuro. Então, a predição para $m=0$, é dada por:

$$\hat{y}_{OL}(i, t) = \sum_{j=1}^{N_m} \sum_{k=0}^{(N_c-1)} h_{i,j}(1+k) \Delta u(j, t-1-k) \quad (\text{A.3.2})$$

A diferença entre estas predições, equações (A.3.2), são adicionadas a equação (a.3.1), chegando-se a equação usada na predição da malha aberta da a estratégia de controle DMC, a qual leva somente em consideração, valores da ações de controle passadas.

$$\hat{y}_{OL}(i, t+m) = y(i, t) + \sum_{j=1}^{N_m} \sum_{k=0}^{(N_c-1)} [h_{i,j}(m+1+k) - h_{i,j}(1+k)] \Delta u(j, t-1-k) \quad (\text{A.3.3})$$

Esta modelagem também é empregada para predizer o comportamento das variáveis de saída, em malha fechada. A resposta em malha fechada considera os efeitos futuros das variáveis de saída, devido aos efeitos passados da variável manipulada, resposta em malha aberta $\hat{y}_{OL}(i, t+m)$, adicionado aos efeitos futuros das variações na variável manipulada, \tilde{u} .

$$\hat{y}(i, t + m) = \hat{y}_{OL}(i, t + m) + \sum_{j=1}^{N_m} \sum_{k=0}^{Nu-1} h_{i,j}(m+k) \Delta u(j, t+k) \quad (\text{A.3.4})$$

O objetivo do Algoritmo DMC é encontrar os melhores valores futuros de mudanças nas variáveis manipuladas, para minimizar a função objetivo apresentada na equação (A.3.5), sujeito a penalização nos incrementos nas ações de controle.

$$J(k) = \sum_{i=1}^{N_m} \sum_{m=1}^N [\hat{y}(i, k+m) - y_w(i, k+m)]^2 + \sum_{i=1}^{N_m} \sum_{j=0}^{Nu-1} \beta^2(i) [\Delta u(i, k+j)]^2 \quad (\text{A.3.5})$$

onde o vetor $y_w(i, k+m)$ é o vetor que contém as trajetória de referência da malha i , conforme já apresentado no Capítulo 3 desta tese.

Substituindo a equação (A.3.4) na equação (A.3.5) e minimizando a função objetivo em relação ao vetor de ações de controle futuras, chega-se:

$$\tilde{u} = \left[\tilde{A}^T \tilde{A} + \underline{\beta} \right]^{-1} \tilde{A}^T (y_w - \hat{y}) \quad (\text{A.3.6})$$

onde:

$$\tilde{A} = \begin{bmatrix} \underline{A}_1 & \cdots & 0 \\ \vdots & & \vdots \\ \vdots & \ddots & \vdots \\ \underline{A}_N & \cdots & \underline{A}_{N-Nu+1} \end{bmatrix} \quad (\text{A.3.7})$$

$$\begin{aligned}
 \underline{y}_w &= [y_w(1, t+1), \dots, y_w(N_m, t+1), \dots, y_w(1, t+N), \dots, y_w(N_m, t+N)]^T \\
 \underline{\hat{y}} &= [\hat{y}(1, t+1), \dots, \hat{y}(N_m, t+1), \dots, \hat{y}(1, t+N), \dots, \hat{y}(N_m, t+N)]^T \\
 \underline{\tilde{u}} &= [\Delta u(1, t), \dots, \Delta u(N_m, t), \dots, \Delta u(1, t+N_u-1), \dots, \Delta u(N_m, t+N_u-1)]^T
 \end{aligned} \tag{A.3.8}$$

N_u = horizonte de controle;

$\underline{\beta}$ = matriz que contém os fatores de supressão, β_i^2 , correspondendo a cada malha de controle i .

ANEXO 4

ANÁLISE DE SENSIBILIDADE DOS PARÂMETROS DO PROCESSO

Uma das etapas importantes no sucesso de uma estratégia de controle é a definição das variáveis manipuladas e controladas da malha fechada. Diferentes técnicas são propostas. Nesta tese baseou-se no procedimento utilizado por TOLEDO (1999), que utiliza o planejamento fatorial (BARROS NETO et. al., 1995).

Um dos problemas mais comuns em estudos de sistemas é a determinação da influência de uma ou mais variáveis sobre uma outra variável de interesse, ou seja, tem-se um certo número de fatores atuando sobre o sistema em estudo, produzindo um certo número de respostas. O sistema é considerado como uma função desconhecida cuja atuação sobre as variáveis de entrada (fatores) produzem como saída as variáveis observadas. Tendo definido os fatores e respostas de interesse para o sistema a ser investigado, deve-se definir claramente o objetivo de estudo, seja ele experimental ou através de simulações do modelo, porque isso determinará que tipo de planejamento deve ser utilizado.

Neste ponto tem-se um tipo particular de planejamento experimental, o planejamento fatorial de dois níveis (BOX et al., 1978). Planejamentos deste tipo são de grande utilidade em investigações preliminares, quando se deseja saber se determinados fatores tem ou não influência sobre a resposta, e não se está preocupado ainda com uma descrição muito rigorosa dessa influência.

O método de planejamento fatorial consiste em selecionar um número fixo de níveis e então executar experimentos com todas as possíveis combinações destes. Geralmente faz-se um planejamento fatorial com dois níveis (nível -1 e nível +1) em relação a um padrão (nível 0 ou ponto central) para cada variável. Assim, para n variáveis envolvidas no estudo, o número de experimentos que devem ser realizados para investigar todas as combinações possíveis é igual a 2^n . Com os resultados obtidos do planejamento fatorial é possível calcular os efeitos principais e de interação das variáveis de interesse (fatores) sobre as respostas, determinando assim quais são os efeitos mais significativos e os seus respectivos modos de influenciar as respostas.

Desta forma, realizou-se uma análise de sensibilidade das variáveis de operação do reator catalítico de leito fixo sobre o perfil de temperatura do mesmo, utilizando-se a metodologia de planejamento fatorial completo, cuja descrição acerca de seu procedimento de cálculo dos efeitos são mostrados em BOX et al. (1978), BARROS NETO et al. (1995) e TOLEDO (1999).

O planejamento fatorial completo para o reator catalítico de leito fixo visou a determinação dos efeitos principais que as variáveis R (relação ar/etanol), GM_o (fluxo mássica de alimentação) e U_r (velocidade do fluido refrigerante) exercem sobre as temperaturas médias do primeiro e terceiro ponto de colocação ortogonal axial do reator. Através desta análise pretende-se definir as variáveis manipuladas para as variáveis controladas, já especificadas.

A Tabela A.2 apresenta os níveis das variáveis usadas no planejamento fatorial.

Tabela A2 – Níveis das variáveis usadas na análise de sensibilidade do processo

Variável	Nível Inferior (-1)	Ponto Central	Nível Superior (+1)
1: GM_o	4275	4500	4725
2: R	23.75	25.00	26.25
5: U_r	2.85	3.00	3.15

A Tabela A.3 apresenta os efeitos dos parâmetros do processo nas variáveis de saída $T(1)$ e $T(3)$.

Tabela A3 – Efeitos das variáveis na análise de sensibilidade

Efeitos	T(1)	T(3)
1	-1.047	0.822
2	-0.462	-0.377
3	0.002	-0.032
12	0.062	-0.042
13	-0.002	0.002
23	0.002	0.002
123	-0.002	-0.002

Através destes resultados é possível obter as seguintes conclusões: (a) A variável 1 (GM_o) tem maior influência nas variáveis resposta analisadas. GM_o apresenta uma relação inversa com $T(1)$ e uma relação direta com $T(3)$; (b) A variável 2 (R) tem uma relação inversa com as variáveis resposta analisadas; (c) A variável U_r apresenta uma pequena influência nas variáveis resposta e não é uma variável manipulada apropriada para este caso; (e) As variáveis GM_o e R apresentam interações, o que dificulta o controle.

ANEXO 5

DESCRIÇÃO DOS PROGRAMAS COMPUTACIONAIS DESENVOLVIDOS

Segue abaixo a relação dos programas desenvolvidos, baseados nas estratégias de controle analisadas/propostas nesta teste. Os programas foram desenvolvidos em linguagem de programação Fortran 90.

CoNDEG – Controle Neural Direto baseado no Erro Global. Estratégia de controle neural baseada em modelo inverso, com aprendizagem *on-line* realizado em rotina de projeto de controlador, que utiliza a estimativa do erro global da malha fechada (Controle adaptativo baseado em redes neurais – Cap. 4).

CoNDEGSS – Controle Neural Direto baseado no Erro Global usando *soft sensor*. Estratégia para controle de concentração usando *soft sensor* acoplado a rotina CoNDEG. O objetivo desta estratégia é controlar a concentração do processo, pela definição das

melhores referências de temperatura da malha fechada (Estratégias de controle baseada em redes neurais – Cap. 5).

CoNDI – Controle Neural Direto baseado no modelo inverso. Estratégia de controle neural com aprendizagem *on-line*, baseado em modelo inverso com aprendizagem indireta.

CoNFeedI – Controle Neural *Feedback* com aprendizagem *on-line* usando modelo inverso. Estratégia de controle neural baseada na estrutura de controle *feedback*. A redes neurais empregadas nesta estratégia de controle são submetidas a aprendizagem *on-line* e o sinal de aprendizagem do controlador é gerado por um modelo inverso (Estratégia baseada na estrutura de controle *Feedback* – Cap. 4).

CoNFeedG – Controle Neural *Feedback* com aprendizagem *on-line* usando erro global estimado da malha fechada. Controle neural *feedback* com aprendizagem *on-line*, baseado em uma estrutura que prediz o erro global da malha fechada. Esta predição usa modelo dinâmicos neurais com aprendizagem *on-line* (Estratégia baseada na estrutura de controle *Feedback* – Cap. 4).

CoPreO – Controle Preditivo com aprendizagem *On-line*. Estratégia de controle preditivo, baseado na estrutura do algoritmo DMC, usando redes neurais com aprendizagem *on-line*, como modelo dinâmico do processo (Algoritmo Modificado – Cap. 3).

CoPreOR – Controle Preditivo com Restrições usando modelo dinâmico baseado em redes neurais, com aprendizagem *on-line*. Estratégia de controle preditivo usando otimização *on-line* da função objetivo da estratégia de controle preditivo, usando SQP (Algoritmo de Controle Preditivo com Restrições – Cap. 3).

CoPreRSS – Controle Preditivo com Restrições usando modelo dinâmico baseado em redes neurais, com aprendizagem *on-line* e *soft sensor*. Utiliza o mesmo princípio da rotina CoNDEGSS, porém esta baseado na rotina CoPREOR.

DMCSS – Algoritmo DMC Auto Ajustável. Estratégia de controle preditivo DMC com auto-ajuste do modelo interno usando mínimos quadrados recursivos e fator de esquecimento variável escalar (Anexo 3).

LearnNet – Aprendizagem de Redes Neurais. Rotina de redes neurais usando arquitetura *feedforward* com aprendizagem *backpropagation*.

ANEXO 6

PARÂMETROS UTILIZADOS

Este anexo tem como objetivo apresentar os parâmetros usados na definição da arquitetura das redes neurais e parâmetros das estratégias de controle. Estes parâmetros estão definidos por capítulo, conforme abaixo.

Capítulo 3:

- Rede Neural:

3 camadas (22-12-2);

Horizonte passado: $\lambda=5$;

Função de transferência Sigmoidal.

- Parâmetros dos Controladores: Considerando as seguintes variáveis:

Variável controlada(1): Temperatura do primeiro ponto de colocação ortogonal;

Variável controlada(2): Temperatura do terceiro ponto de colocação ortogonal;

Variável manipulada(1): Vazão Mássica de alimentação do reator;

Variável manipulada(2): Relação ar/etanol.

Parâmetros usados:

$N=20$;

$Nu=5$;

$N_c=50$ (somente para o DMC);

$\alpha(1)=0.5$;

$\alpha(2)=0.5$;

$\beta^2(1)=0.1$;

$\beta^2(2)=6.0$;

Capítulo 4:

ESTRATÉGIA DE CONTROLE USANDO MODELO INVERSO MIMO

- Rede Neural:

3 camadas (10-6-2);

Número de perturbações: 2;

Número de respostas: 2;

Horizonte passado: $\lambda=2$;

Função de transferência Tangente Hiperbólica.

- Parâmetros do Controlador: Considerando as seguintes variáveis:

Variável controlada(1): Temperatura do primeiro ponto de colocação ortogonal;

Variável controlada(2): Temperatura do terceiro ponto de colocação ortogonal;

Variável manipulada(1): Vazão Mássica de alimentação do reator;

Variável manipulada(2): Relação ar/etanol.

Parâmetros usados:

Penalização na ação de controle $\lambda(1)=0.87$;

Penalização na ação de controle $\lambda(2)=0.87$;

Parâmetro da trajetória do *setpoint* $\alpha(1)=0.93$;

Parâmetro da trajetória do *setpoint* $\alpha(2)=0.93$;

MultiSISO

- Parâmetros do Controlador: Considerando as seguintes malhas de controle:

Malha de Controle 1:

Variável controlada: Temperatura do primeiro ponto de colocação ortogonal;

Variável manipulada: Vazão Mássica de alimentação do reator.

Rede Neural:

3 camadas (5-3-1);

Número de perturbações: 1;

Número de respostas: 1;

Horizonte passado: $\lambda=2$;

Função de transferência Tangente Hiperbólica.

Parâmetros usados:

Penalização na ação de controle $\lambda(1)=0.95$;

Parâmetro da trajetória do *setpoint* $\alpha(1)=0.93$.

Malha de Controle 2:

Variável controlada: Temperatura do terceiro ponto de colocação ortogonal;

Variável manipulada: Relação ar/etanol.

Rede Neural:

3 camadas (11-6-1);

Número de perturbações: 1;

Número de respostas: 1;

Horizonte passado: $\lambda=5$;

Função de transferência Tangente Hiperbólica.

Parâmetros usados:

Penalização na ação de controle $\lambda(2)=0.8$

Parâmetro da trajetória do *setpoint* $\alpha(2)=0.93$

ESTRATÉGIA DE CONTROLE ADAPTATIVO BASEADO EM REDES NEURAIAS

Reator Catalítico

- Rede Neural do Controlador:

3 camadas (10-6-2);

Número de perturbações: 2;

Número de respostas: 2;

Horizonte passado: $\lambda=2$;

Função de transferência Tangente Hiperbólica.

- Rede Neural do Modelo do Processos (Parâmetros mensuráveis):

3 camadas (28-15-2);

Número de perturbações: 4;

Número de respostas: 2;

Horizonte passado: $\lambda=4$;

Função de transferência: Sigmoidal.

- Rede Neural do Modelo do Processos (Parâmetros não mensuráveis):

3 camadas (10-6-2);

Número de perturbações: 2;

Número de respostas: 2;

Horizonte passado: $\lambda=2$;

Função de transferência: Tangente Hiperbólica.

- Parâmetros do Controlador: Considerando as seguintes variáveis:

Variável controlada(1): Temperatura do primeiro ponto de colocação ortogonal

Variável controlada(2): Temperatura do terceiro ponto de colocação ortogonal

Variável manipulada(1): Vazão Mássica de alimentação do reator

Variável manipulada(2): Relação ar/etanol

Parâmetros usados:

Penalização na ação de controle $\lambda(1)=0.9$

Penalização na ação de controle $\lambda(2)=0.5$

Parâmetro da trajetória do *setpoint* $\alpha(1)=0.93$

Parâmetro da trajetória do *setpoint* $\alpha(2)=0.9$

Reator de Penicilina

• Rede Neural do Controlador:

3 camadas (7-4-1);

Número de perturbações: 1;

Número de respostas: 1;

Horizonte passado: $\lambda=3$;

Função de transferência Tangente Hiperbólica.

• Rede Neural do Modelo do Processos (Parâmetros não mensuráveis):

3 camadas (11-6-1);

Número de perturbações: 1;

Número de respostas: 1;

Horizonte passado: $\lambda=5$;

Função de transferência: Sigmoidal.

• Parâmetros do Controlador: Considerando as seguintes variáveis:

Variável controlada(1): Concentração relativa de oxigênio dissolvido;

Variável manipulada(1): velocidade de rotação do agitador

Parâmetros usados:

Penalização na ação de controle $\lambda(1)=0.75$;

Parâmetro da trajetória do *setpoint* $\alpha(1)=0.5$.

ESTRATÉGIA BASEADA NA ESTRUTURA DE CONTROLE *FEEDBACK*

- Rede Neural do Controlador:

3 camadas (10-6-2);

Número de perturbações: 2;

Número de respostas: 2;

Horizonte passado: $\lambda=2$;

Função de transferência Tangente Hiperbólica.

- Parâmetros do Controlador: Considerando as seguintes variáveis:

Variável controlada(1): Temperatura do primeiro ponto de colocação ortogonal;

Variável controlada(2): Temperatura do terceiro ponto de colocação ortogonal;

Variável manipulada(1): Vazão Mássica de alimentação do reator;

Variável manipulada(2): Relação ar/etanol.

Parâmetros usados:

Penalização na ação de controle $\lambda(1)=0.79$;

Penalização na ação de controle $\lambda(2)=0.69$;

Parâmetro da trajetória do *setpoint* $\alpha(1)=0.93$;

Parâmetro da trajetória do *setpoint* $\alpha(2)=0.93$.

- Rede Neural do Modelo do Processos (Aprendizagem usando erro global):
 - 3 camadas (22-10-2);
 - Número de perturbações: 2;
 - Número de respostas: 2;
 - Horizonte passado: $\lambda=5$;
 - Função de transferência: Sigmoidal.

- Rede Neural do Modelo Inverso (Aprendizagem usando modelo inverso):
 - 3 camadas (10-6-2);
 - Número de perturbações: 2;
 - Número de respostas: 2;
 - Horizonte passado: $\lambda=2$;
 - Função de transferência: Tangente Hiperbólica.

Capítulo 5

Os parâmetros referentes a estratégia de controle preditivo com restrições, bem como a estratégia de controle adaptativo baseada em redes neurais, são os mesmos apresentados nos capítulos 3 e 4, respectivamente. Os parâmetros usados no módulo de otimização do *setpoint*, encontram-se abaixo:

- Rede Neural do Modelo Estático:
 - 5 camadas (4-5-6-5-4);
 - Número de perturbações: 4;
 - Número de respostas: 4;
 - Horizonte passado: $\lambda=0$;
 - Função de transferência: Sigmoidal.