

UNIVERSIDADE ESTADUAL DE CAMPINAS

FACULDADE DE ENGENHARIA QUÍMICA

MARCELO DALL'AGNOL

**APLICAÇÃO DE CONTROLADORES FEEDBACK EM SISTEMA EXPERIMENTAL
DE REFRIGERAÇÃO E DESENVOLVIMENTO DE MODELO PREDITIVO BASEADO
EM REDES NEURAIIS**

Dissertação de Mestrado apresentada à Faculdade de Engenharia Química da Universidade Estadual de Campinas, como requisito para a obtenção do título de Mestre em Controle e Automação de Processos.

Orientadora: Dra. Ana Maria Frattini Fileti

Co-Orientador: Dr. Flávio Vasconcelos da Silva

Campinas

2010

FICHA CATALOGRÁFICA ELABORADA PELA

BIBLIOTECA DA ÁREA DE ENGENHARIA E ARQUITETURA - BAE - UNICAMP

Dall'Agnol, Marcelo

Aplicação de controladores feedback em sistema experimental de refrigeração e desenvolvimento de modelo preditivo baseado em redes neurais / Marcelo Dall'Agnol. -- Campinas, SP: [s.n.], 2010.

Orientadores: Ana Maria Frattini Fileti, Flávio Vasconcelos da Silva.

Dissertação de Mestrado - Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. Refrigeração. 2. Redes Neurais. 3. Controle Feedback.
I. Fileti, Ana Maria Frattini. II. Silva, Flávio Vasconcelos da.
III. Universidade Estadual de Campinas. Faculdade de Engenharia Química. IV. Título.

Título em Inglês: Application of feedback controllers in experimental system of refrigeration and development of predictive model based on neural networks

Palavras-chave em Inglês: Refrigeration, Neural Networks, Feedback Control

Área de concentração: Sistemas de Processos Químicos e Informática

Titulação: Mestre em Engenharia Química

Banca examinadora: José Vicente Hallak D'Angelo, Luis Antônio Rezende Muniz

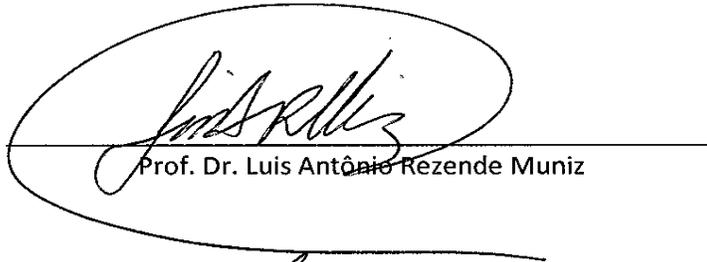
Data da defesa: 29/03/2010

Programa de Pós Graduação: Engenharia Química

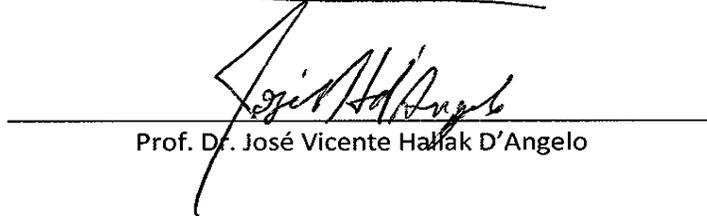
Dissertação de Mestrado defendida por Marcelo Dall'Agnol e aprovada em 29 de Março de 2010 pela banca examinadora constituída pelos doutores:



Prof. Dr. Ana Maria Frattini Fileti

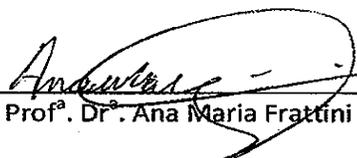


Prof. Dr. Luis Antônio Rezende Muniz



Prof. Dr. José Vicente Haak D'Angelo

Este exemplar corresponde à versão final da Dissertação de Mestrado em Engenharia Química.



Prof^a. Dr^a. Ana Maria Frattini Fileti

RESUMO

O presente trabalho teve por objetivo propor uma sistemática para identificação de um sistema de refrigeração dotado de compressor e bomba de fluido secundário de rotação variável, para a aplicação futura de alternativas de controle multivariáveis com foco na redução do consumo de energia e otimização do coeficiente de performance (COP) de tais sistemas. Uma primeira etapa de ensaios experimentais foi realizada para gerar um conhecimento prévio do sistema, caracterizando-o como não-linear. Foram utilizadas diversas arquiteturas de redes neurais artificiais (RNA) para gerar modelos não-lineares MIMO (*Multiple Input-Multiple Output*), através do software MATLAB®, para fazer a predição das temperaturas de evaporação e do fluido secundário. Para treinamento da rede neural foram utilizados dados de ensaios em malha aberta e malha fechada com um controlador PID desenvolvido especialmente para este fim. Neste sistema de controle PID foram utilizadas técnicas como *anti-windup* e *derivative-kick* na tentativa de melhorar a atuação de tal controlador. O PID, apesar de conseguir controlar o sistema em torno de 2°C, mostrou-se pouco eficiente quando eram aplicados degraus no set-point e na carga elétrica. A modelagem não-linear resultou em uma rede neural com arquitetura 11x2x8x2 que foi capaz de prever as duas temperaturas com precisão de mais ou menos 0,5°C, mesmo quando o sistema de refrigeração se manteve instável.

ABSTRACT

The main aim of the present work was to propose a systematic identification procedure for an experimental refrigeration system operating under variable rotation speed compressor and pump of secondary fluid. This work will find future applications in alternative multivariable control development with focus on reducing energy consumption and on the optimization of the coefficient of performance (COP) of such systems. The non-linear feature of the system was proved by means of imposed speed disturbances process reaction curves. Using the software MATLAB®, various artificial neural networks (ANN) architectures were used to generate non-linear models with Multiple-Input-Multiple-Output (MIMO) variables, in order to predict the evaporation and the secondary fluid temperatures. Open-loop and also closed loop assays were carried out and the experimental data used in the neural network training procedure. A specially designed PID, using anti-windup and derivative-kick techniques, was employed in the closed-loop runs. This conventional temperature controller showed an acceptable off-set of 2 °C, however some instability occurred when set point and thermal load changed. The best non-linear modeling resulted in a neural network with architecture 11x2x8x2. This neural model was able to predict the two temperatures with precision of about 0.5 °C, even when the refrigeration system remained unstable.

**A meus pais Hermes e Iria,
em quem eu me espelho, e me fizeram ser quem sou.**

AGRADECIMENTOS

Agradeço aos professores Ana Frattini Fileti e Flávio Vasconcelos da Silva pelas oportunidades e, pela orientação e paciência ao longo deste tempo. Com certeza contribuíram e muito para o meu aperfeiçoamento profissional e pessoal. Agradeço ainda pelo constante incentivo, confiança, respeito e amizade.

Gostaria de agradecer a minha futura esposa e companheira Christiane, pelo incentivo e compreensão com que aceitou todo o tempo dedicado a este trabalho. TE AMO!!

Agradeço muitíssimo a meus queridos Gi e Ivan por todo apoio desde o início deste trabalho. Se não fosse por vocês eu não teria começado e muito menos terminado este trabalho. OBRIGADO!

Aos colegas de Thiago “Araxá” e Ivan que, com o passar do tempo e depois de inúmeros cafezinhos, muitas cervejas e dias de lab. Foram promovidos à categoria de grandes e queridos amigos.

E a todos os colegas, Pinelli, Tati, Camila, Manu, Marcão e outros, pelos muitos momentos de descontração, e por toda a ajuda que deram durante o desenvolvimento de meu trabalho.

“Faça tudo valer a pena, a vida é tão imensa e ao mesmo tempo é tão pequena”

(RUB)

SUMÁRIO

LISTA DE FIGURAS.....	XII
LISTA DE TABELAS.....	XIV
1. INTRODUÇÃO.....	1
2. FUNDAMENTAÇÃO TEÓRICA.....	4
2.1. Sistemas de Controle.....	4
2.2. Modelagem de Sistemas Dinâmicos	8
2.2.1. Introdução.....	8
2.2.2. Conceitos básicos.....	8
2.2.3. Classificação dos modelos matemáticos.....	10
2.2.4. Representação de sistemas dinâmicos	12
2.2.5. Identificação de Sistemas Dinâmicos.....	13
2.3. Redes Neurais Artificiais.....	15
2.3.1. Introdução.....	15
2.3.2. Neurônio biológico	15
2.3.3. Neurônio artificial.....	16
2.3.4. Topologias de Redes Neurais Artificiais (RNA's)	17
2.3.5. Arquitetura de RNA's.....	19
2.3.6. Arquitetura do MLP	21
2.3.7. Treinamento de RNA's.....	22
2.3.8. Treinamento supervisionado	23
2.3.9. Etapas de projeto de uma rede neural	28
2.3.10. Redes Neurais na Identificação de Sistemas Dinâmicos não lineares	29
2.4. Sistema de Refrigeração.....	32
2.4.1. A importância em se controlar estes sistemas	33
2.4.2. A RNA's na modelagem de sistemas de refrigeração.....	34
3. MATERIAIS E MÉTODOS	37
3.1. Unidade experimental.....	37
3.1.1. Planta de refrigeração	37
3.1.2. Descrição do protótipo experimental.....	38
3.1.3. Instrumentação do protótipo.....	40
3.1.4. Sistema de controle digital.....	43

3.1.5.	Procedimento inicial de operação	45
3.2.	Desenvolvimento do controlador PID	46
3.3.	Sintonia dos controladores PID	49
3.4.	Redes Neurais	50
3.4.1.	Desenvolvimento do modelo neural	51
3.4.2.	Coleta de dados e normalização	53
3.4.3.	Treinamento da RNA	54
3.4.4.	Validação online do modelo neural	54
4.	RESULTADOS OBTIDOS	56
4.1.	Análise de não-linearidade do sistema	56
4.2.	Implementação das malhas de controle PID	58
4.3.	Ensaio com o controlador convencional	59
4.4.	Redes Neurais	62
4.4.1.	Modelagem do processo utilizando redes neurais	62
4.4.2.	Treinamento das redes neurais	63
4.4.3.	Predição on-line a partir do modelo desenvolvido	88
5.	CONCLUSÕES	93
5.1.	Resultados obtidos com a implementação do controlador multimalha	93
5.2.	Redes Neurais Artificiais	94
5.3.	Sugestões para trabalhos futuros	94
6.	REFERÊNCIAS BIBLIOGRÁFICAS	95

LISTA DE FIGURAS

Figura 2. 1 Esquema de modelagem de um processo real.	9
Figura 2. 2 Neurônio artificial.....	16
Figura 2. 3 Organização da rede neural em camadas. [TONSIG, 2000]	18
Figura 2. 4 Fluxograma do desenvolvimento de uma rede neural.	29
Figura 2. 5 Esquema de um sistema de refrigeração básico.....	32
Figura 3. 1 Módulo de refrigeração. [Franco, 2009]	39
Figura 3. 2 Diagrama do circuito principal. [PINELLI,2008].....	39
Figura 3. 3 Diagrama do circuito secundário [PINELLI, 2008]	40
Figura 3. 4 Diagrama de instrumentação do sistema de refrigeração.....	41
Figura 3. 5 Controlador Lógico Programável (CLP) MCI02-QC (HI Tecnologia) utilizado no projeto.	43
Figura 3. 6 Malha de controle implementada no SIMULINK/MATLAB para as duas estratégias de PID.	47
Figura 3. 7 Curva de reação para a obtenção dos parâmetros de Ziegler e Nichols.....	50
Figura 3. 8 Síntese dos ensaios realizados para definição da arquitetura da rede neural.	52
Figura 4. 1 Comportamento do desvio das variáveis controlada e manipulada frente a perturbações na carga aplicada ao sistema de refrigeração.	60
Figura 4. 2 Comportamento dos desvios da variável controlada e manipulada, sob ação do controlador PID, frente à perturbações aplicadas no set-point	60
Figura 4. 3 Gráfico ampliado do comportamento da variável manipulada frente a perturbação no set-point.....	61
Figura 4. 4 Desempenho do modelo neural 1 durante o treinamento.....	65
Figura 4. 5 Gráfico de dispersão entre saída real e predita para temperatura de evaporação (TE). 66	
Figura 4. 6 Comparação entre saída real e calculada pela rede para TE.	66
Figura 4. 7 Gráfico de dispersão entre saída real e predita para temperatura de saída do propileno (TPG).....	67
Figura 4. 8 Comparação entre saída real e calculada pela rede para TPG.....	67
Figura 4. 9 Dispersão entre a saída real e a calculada para a TE, para o modelo neural 1.....	68
Figura 4. 10 Comparação entre a saída real e a calculada pela rede, para o modelo neural 1.	69
Figura 4. 11 Dispersão entre a saída real e a calculada para a TPG, para o modelo neural 1.	69
Figura 4. 12 Comparação entre a saída real e calculada pela rede neural.	70
Figura 4. 13 Desempenho do modelo neural 2 durante o treinamento.....	71
Figura 4. 14 Dispersão entre saída real e calculada para TE.	72
Figura 4. 15 Comparação entre saída real e calculada pelo modelo neural 2 para TE.	72
Figura 4. 16 Dispersão entre saída real e calculada para TPG.	73
Figura 4. 17 Comparação entre saída real e calculada pelo modelo neural 2 para TPG.	73
Figura 4. 18 Desempenho da rede neural durante o treinamento.....	75
Figura 4. 19 Dispersão entre saída real e calculada para TE.	75
Figura 4. 20 Comparação entre a saída real e calculada para TE.....	76
Figura 4. 21 Dispersão entre a saída real e a calculada para a TPG.	76

Figura 4. 22 Comparação entre saída real e calculada para TPG.	77
Figura 4. 23 Desempenho da rede neural 4 durante o treinamento.	78
Figura 4. 24 Dispersão entre a saída real e predita para TE.	79
Figura 4. 25 Comparação entre saída real e predita pela rede para TE.	79
Figura 4. 26 Dispersão entre saída real e predita para TPG.	80
Figura 4. 27 Comparação entre a saída real e predita para TSP.	80
Figura 4. 28 Dispersão entre saída real e calculada para TE.	82
Figura 4. 29 Comparação entre saída real e predita pela rede neural 5 para TE.	82
Figura 4. 30 Dispersão entre saída real e calculada para TSP.	83
Figura 4. 31 Comparação entre saída real e predita para TSP.	83
Figura 4. 32 Dispersão entre saída real e calculada pela rede para temperatura de evaporação (TE).	85
Figura 4. 33 Comparação entre saída real e predita pela rede para a TE.	85
Figura 4. 34 Dispersão entre saída real e calculada para a temperatura de saída do propileno (TSP).	86
Figura 4. 35 Comparação entre a saída real e a predita pela rede para a TSP.	86
Figura 4. 36 Rede neural feedforward utilizada como modelo empírico.	88
Figura 4. 37 Dispersão entre saída real e calculada para TE.	89
Figura 4. 38 Comparação entre saída real e predita para TE.	89
Figura 4. 39 Erro de predição para TE.	90
Figura 4. 40 Dispersão entre saída real e predita para TSP.	90
Figura 4. 41 Comparação entre saída real e predita para TSP.	91
Figura 4. 42 Erro de predição para TSP.	91

LISTA DE TABELAS

Tabela 3. 1 Descrição dos medidores de vazão utilizados.....	41
Tabela 3. 2 Descrição dos elementos sensores.....	42
Tabela 4. 1 Parâmetros de gráficos, do ganho (K), obtidos dos ensaios com perturbações da rotação da bomba em diversas amplitudes para temperatura de saída do propileno-glicol (TPG).	56
Tabela 4. 2 Parâmetros de gráficos, do ganho (K), obtidos dos ensaios com perturbações na rotação do compressor em diversas amplitudes para temperatura de saída do propileno-glicol (TPG).....	57
Tabela 4. 3 Parâmetros de gráficos, do ganho (K), obtidos dos ensaios com perturbações da rotação do compressor em diversas amplitudes para temperatura de evaporação (TE).	57
Tabela 4. 4 Valores ajustados para PID.....	58
Tabela 4. 5 Valores ajustados PID.....	59
Tabela 4. 6 Parâmetros definitivos para os controladores PID.....	59
Tabela 4. 7 Respostas do controlador frente a perturbações.....	62
Tabela 4. 8 Melhores resultados obtidos em testes.....	62
Tabela 4. 9 Disposição dos neurônios nas camadas de entrada e saída.....	64
Tabela 4. 10 Arquitetura e funções utilizadas pela rede neural.....	65
Tabela 4. 11 Arquitetura utilizada para o modelo neural 2.....	71
Tabela 4. 12 Arquitetura utilizada para o modelo neural 3.....	74
Tabela 4. 13 Arquitetura utilizada para o modelo neural 4.....	78
Tabela 4. 14 Arquitetura da rede neural do modelo 5.....	81
Tabela 4. 15 Arquitetura de rede utilizada no teste do modelo neural 6.....	84

1. INTRODUÇÃO

Com o advento dos computadores e a rápida expansão da capacidade de processamento dos mesmos, cada vez mais, as tarefas desempenhadas por um sistema de controle estão sendo executadas de forma automática, com mais facilidade, simplicidade e segurança. Este fato permitiu tanto a ampla difusão da utilização do controle automático, como o desenvolvimento de tecnologia para tal (BALCHEN, 1999). Atualmente, os sistemas de controle automático são amplamente encontrados em todos os setores da indústria, tais como: controle de qualidade de produtos manufaturados, linha de montagem automática, controle de máquinas operatrizes, tecnologia espacial, sistemas de armamento e defesa, sistemas de transporte, e muitos outros. Dessa forma, os sistemas de controle automático têm assumido um papel progressivamente importante no desenvolvimento da moderna civilização e tecnologia. Praticamente todo aspecto de nossas atividades diárias é afetado por algum tipo de sistema de controle (FACCIN, 2004).

Para trabalhar com sistemas é necessário conhecer, estudar e compreender a sua essência. Isto pode ser feito a partir de um modelo. Um modelo contém as características principais de um sistema real possibilitando ao usuário aprender, comunicar, monitorar, controlar e fazer previsões. Segundo BASSANEZI (2002) o processo usual é selecionar, no sistema, argumentos ou parâmetros considerados essenciais e formalizá-los através de um sistema artificial, “o modelo”. Quando se transcreve o funcionamento de um sistema com linguagem matemática obtém-se um modelo matemático. Segundo OGATA, 2003, o modelo matemático de um sistema dinâmico é definido como um conjunto de equações que representa com precisão ou, pelo menos, razoavelmente bem a dinâmica do sistema. O estudo dos diferentes métodos de implementar modelos matemáticos denomina-se modelagem matemática.

A identificação de sistemas é a área que permite obter modelos matemáticos para sistemas dinâmicos reais, a partir de dados observados de entrada e saída. É uma área do conhecimento com significativas evoluções no decorrer das últimas décadas, sendo que para BILLINGS (1980) os modelos lineares são os mais desenvolvidos devido a

um maior conhecimento e facilidade de desenvolvimento, se comparados com sistemas não lineares. Avançadas ferramentas de álgebra linear são empregadas na análise de sistemas dinâmicos, quando estes podem ser aproximados linearmente. Porém, grande parte dos sistemas de ordem prática apresenta não linearidades significativas. Quando são resolvidos por aproximação linear perde-se muita informação, o que acaba por reduzir sua eficiência. Sistemas não-lineares se mostram mais complexos em seu comportamento e exigem métodos de resolução mais avançados que se adaptam às dinâmicas.

Redes Neurais Artificiais (RNA's) compõem-se de um método para solucionar problemas de inteligência artificial, construindo um sistema que possua circuitos que tentem simular em máquinas (computadores), de uma maneira simplificada, o funcionamento do cérebro humano buscando uma similaridade entre o comportamento do neurônio biológico e um modelo de neurônio computacional (TAFNER ET al. 1996). Trata-se de técnicas computacionais que apresentam um modelo inspirado na estrutura neural de organismos inteligentes e que adquirem conhecimento através de experiência (MENDES FILHO, 2000). Elas são capazes de reconhecer padrões, extrair regularidades e detectar relações subjacentes em um conjunto de dados aparentemente desconexos. Além disso, apresentam habilidades de lidar com dados ruidosos, incompletos ou imprecisos, e de prever sistemas não-lineares (MELLO, 2002).

Diante disto, o presente trabalho teve por objetivo propor uma sistemática para identificação de um sistema de refrigeração dotado de compressor e bomba de fluido secundário de rotação variável, caracterizado como não-linear, para a aplicação futura de alternativas de controle multivariáveis com foco na redução do consumo de energia e otimização do coeficiente de performance (COP) de tal sistema. Sabe-se que a eficiência energética em equipamentos de refrigeração é cada vez mais importante para indústrias, comércio e residências, devido à magnitude que tais equipamentos representam no consumo de energia elétrica.

Uma primeira etapa de ensaios experimentais foi realizada para gerar um conhecimento prévio do sistema, utilizando-se uma bancada experimental com capacidade de refrigeração máxima de aproximadamente 1800 W. Neste trabalho foram utilizadas diversas arquiteturas de redes neurais artificiais (RNA) para gerar modelos não-

lineares MIMO's (Multiple Input-Multiple Output), através do software MATLAB®, para fazer a predição das temperaturas de evaporação e do fluido secundário. Para treinamento da rede neural foram utilizados dados de ensaios em malha aberta e malha fechada (de um controlador PID desenvolvido especialmente para este fim).

Um embasamento teórico e os resultados obtidos são apresentados neste trabalho, bem como alguns trabalhos nos quais se obteve sucesso com o tipo de modelagem aqui utilizada.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo são abordados conceitos fundamentais a respeito de controle de processos, de redes neurais artificiais, modelagem de processos e sistemas de refrigeração. Também foi realizada uma pequena revisão da literatura sobre aplicações de redes neurais artificiais em controle de processos e sistemas de refrigeração.

2.1. Sistemas de Controle

O controle de processos é indispensável para o bom funcionamento de uma planta industrial. Controlar um processo significa atuar sobre ele, ou sobre as condições a que o processo está sujeito, de modo a atingir algum objetivo, por exemplo: suprimir a influência de perturbações externas, prevenindo condições instáveis no processo para a garantia da segurança das pessoas e equipamentos; otimizar o desempenho de um processo químico, mantendo os processos em seus pontos operacionais ótimos; e mostrar dados aos operadores da planta para assegurar a estabilidade de um processo químico. O controle de processo pode ser também a combinação desses fatores ou, ainda, muito mais que isso, pode depender de muitos aspectos de projeto e operação da planta.

No contexto de controladores industriais, existem duas estratégias de controle que merecem destaque: controle *feedback* (controle por realimentação) e, controle *feedforward* (controle por antecipação).

- Controle por realimentação: uma malha de controle por realimentação é composta por um dispositivo para medição da variável a ser controlada (saída do processo), um comparador para cálculo do desvio entre o valor de referência (*set-point*) e o valor medido, e um controlador que atua, de acordo com o desvio calculado, sobre uma variável manipulada (normalmente vazão) para compensar os efeitos das perturbações.
- Controle por antecipação: o controle por antecipação mede a perturbação e atua no processo antes que este sinta os efeitos desta variação. Este controlador se

baseia na predição do comportamento futuro da planta a ser controlada e atua de acordo com um modelo do processo.

Segundo FUJIKI (2009), em aplicações práticas o controle *feedforward*, é usado em conjunto com controle *feedback*. O controle *feedforward* é usado para reduzir os efeitos de distúrbios mensuráveis, enquanto o *feedback* compensa imprecisões no modelo do processo, erros de medida e distúrbios não medidos.

Os controladores *feedback* mais utilizados são: proporcional (P), proporcional + integral (PI) e proporcional + integral + derivativo (PID).

- **Controle Proporcional:**

Sua saída atuante $c(t)$ é proporcional ao erro $e(t)$:

$$c(t) = K_C \cdot e(t) + c_S \quad (2.1)$$

$$e(t) = y_{SP}(t) - y_m(t) \quad (2.2)$$

onde K_C é o ganho proporcional do controlador e c_S é o sinal “bias” do controlador para quando é igual a 0.

O sinal do controlador é descrito pelo valor de seu ganho proporcional K_C ou equivalentemente por sua banda proporcional (PB), onde $PB = 100/K_C$. A banda proporcional caracteriza a faixa sobre a qual o erro precisa mudar para conduzir um sinal atuante do controlador sobre sua faixa total. Usualmente:

$$1 \leq PB \leq 500$$

É claro que “quanto maior o ganho K_C , ou equivalentemente, quanto menor a banda proporcional, maior a sensibilidade do sinal atuante com relação ao desvio “e””.
Define-se:

$$c'(t) = c(t) - c_S \quad (2.3)$$

Então,

$$c'(t) = K_C \cdot e(t) \quad (2.4)$$

Desta equação, a Transformada de Laplace resulta na função de transferência do controlador proporcional.

$$G(S) = K_C$$

- **Controle Proporcional-Integral**

Seu sinal atuante se relaciona com o erro e pela seguinte equação.

$$c'(t) = K_C \cdot e(t) + \frac{K_C}{\tau_i} \int_0^t e(t) \cdot dt \quad (2.5)$$

onde τ_i é a constante de tempo integral ou “reset-time,” normalmente em segundos. O “reset-time” é um parâmetro ajustável e algumas vezes são referidas como “segundos por repetição”. Usualmente, ele varia na faixa de:

$$10 \leq \tau_i \leq 300 \text{ s}$$

Alguns fabricantes não calibram seus controladores em termos de τ_i , mas em termos do recíproco, $(1/\tau_i)$ - repetições por minuto - também conhecido como taxa de “reset”, ou *anti-windup*.

A função de transferência dada para este controlador é:

$$G(s) = K_C \cdot \left(1 + \frac{1}{\tau_i \cdot s}\right) \quad (2.6)$$

- **Controle Proporcional-Integral-Derivativo**

O sinal de saída do controlador é dado por:

$$c'(t) = K_C \cdot e(t) + \frac{K_C}{\tau_i} \int_0^t e(t) \cdot dt + K_C \cdot \tau_D \cdot \frac{de(t)}{dt} \quad (2.7)$$

onde τ_D é a constante de tempo derivativo, normalmente, dado em minutos.

Com a presença do termo derivativo $(de(t)/dt)$, o PID antecipa ao desvio e aplica uma ação de controle que é proporcional à taxa de mudança corrente da mudança de erro. Devido a esta propriedade, a ação de controle derivativa é referida como ação antecipatória. As maiores desvantagens da ação de controle derivativa são as seguintes:

I) para uma resposta com desvio “e” não-nulo constante ele não fornece nenhuma ação de controle;

II) para uma resposta com ruído e erro quase zero ele pode computar grandes derivadas resultando em uma ação de controle extensa sem necessidade.

A função de transferência deste controlador é dada por:

$$G(s) = K_C \cdot \left(1 + \frac{1}{\tau_i \cdot s} + \tau_D \cdot s \right) \quad (2.8)$$

O controlador PID continua sendo responsável por cerca de 95% das funções de controle dos processos industriais. No entanto, os 5% restantes podem ainda significar uma grande oportunidade de ganho, em termos de custo de produção e na qualidade dos produtos. Quando o impacto econômico de um desempenho de controle ruim é significativo e o processo é de natureza complexa, os algoritmos de controle avançado devem ser considerados na elaboração de uma estratégia de controle.

Devido à grande complexidade dos processos industriais atuais, os sistemas regulatórios convencionais não oferecem desempenho satisfatório. As dificuldades de controle usualmente encontradas nos processos industriais, tais como, tempo morto, não linearidade, interação entre as malhas de controle, fazem com que a sintonia dos controladores PID, em alguns casos, não seja adequada, comprometendo o resultado dos sistemas de controle. (FONSECA ET al., 2004)

Se uma boa condição de regulação do processo pode ser obtida através do controle regulatório básico e do emprego de algoritmos avançados, então a próxima etapa para se conseguir bons resultados globais, em termos de qualidade do produto e lucratividade, é definir quais são as condições ótimas de operação da planta. As condições operacionais de uma planta são definidas pelo mercado, pela disponibilidade e qualidade das matérias-primas e por questões econômicas. Se estas condições sofrem variações no decorrer do tempo, então as condições de processos devem ser novamente otimizadas para maximizar o retorno do investimento.

2.2. Modelagem de Sistemas Dinâmicos

2.2.1. Introdução

Segundo LJUNG (1987), nas últimas décadas diversas concepções de algoritmos para modelagem e identificação de sistemas dinâmicos complexos têm sido propostos na literatura. Uma abordagem relevante entre tantas outras para representação matemática de sistemas dinâmicos com comportamento complexo é a das redes neurais artificiais.

Nesta seção será abordada a identificação de sistemas de forma introdutória, sendo focados essencialmente a terminologia e conceitos que foram empregados neste trabalho, bem como o emprego de redes neurais para formulação de sistemas dinâmicos não lineares. A área de modelagem de sistemas dinâmicos é extremamente ampla, com extensa bibliografia voltada para a teoria, mecanismos e ferramentas para modelagem e aplicações desses sistemas. Para estudo mais detalhado, livros como (LJUNG, 1999) e (AGUIRRE, 2004) podem ser usados como referências em trabalhos na área de Identificação de Sistemas.

2.2.2. Conceitos básicos

Primeiramente, um sistema pode ser definido como uma série de objetos que atuam em conjunto e realizam certo objetivo. Os sistemas interagem com o mundo externo através de entradas, saídas e perturbações.

A característica básica dos sistemas conceituados como dinâmicos é traduzida como a de sistemas cujo estado assumido no tempo é condicionado pelas suas variáveis de entrada e pelo seu histórico, ou seja, pelo seu estado anterior, assumindo-se que estas variáveis são dependentes do tempo, onde:

$$\frac{d\vec{x}(t)}{dt} \neq 0$$

Em outras palavras, o estado do sistema assumido em t depende dos valores das variáveis de entrada neste instante bem como do estado no qual o sistema se encontrava

em um número de instantes imediatamente anteriores, ou seja, $t-1$, $t-2$ e assim por diante.

Para modelos dinâmicos as variáveis referidas acima são sinais reais que evoluem como uma função do tempo, discreto (k) ou contínuo (t).

Para o desenvolvimento das áreas a exemplo da Engenharia, Física, Economia, Medicina ou mesmo em tarefas do cotidiano é fundamental a construção de modelos matemáticos para representar os sistemas. As entradas e saídas do sistema representam as variáveis do modelo, conforme ilustra a Figura 2.1. A idéia é que os modelos dos sistemas estabeleçam relações entre essas variáveis e na medida do possível, representem o sistema o mais próximo da realidade em suas características essenciais.

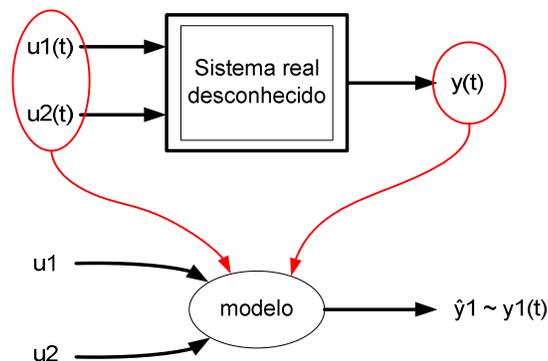


Figura 2. 1 Esquema de modelagem de um processo real.

Os tipos de modelagem utilizados na identificação de sistemas podem ser classificados pelo conhecimento dos sistemas a serem identificados. E podem ser divididos nos seguintes grupos:

- Sistemas caixa branca;
- Sistemas caixa cinza; e,
- Sistemas caixa preta;

Essencialmente todos os modelos são obtidos a partir de dados observados, entretanto alguns são construídos a partir de experiência ou conhecimento prévio do sistema e das variáveis envolvidas e não da observação direta dos dados. Modelos assim

construídos são obtidos através da modelagem físico-química do processo, utilizando-se das leis físicas, a exemplo das leis de Newton, da Teoria da Relatividade de Einstein, Leis de Continuidade (B.M., B.E. e B Momentum), entre outras e de relacionamentos bem estabelecidos que tenham raízes em trabalhos empíricos anteriores (LJUNG, 1987). Estes sistemas são conhecidos como “caixa branca”, modelagem pela natureza do processo, modelagem fenomenológica ou conceitual. Usualmente são modelos descritos por equações de diferenças (Eq. 2.9) ou equações diferenciais (algébricas) (Eq. 2.10).

$$y_{n+1} = f(y_n) \quad (2.9)$$

$$\frac{d\vec{x}}{dt} = \vec{f}(\vec{x}, t) \quad (2.10)$$

Nos sistemas do tipo caixa cinza, se possui o conhecimento das entradas e saídas, mas o processo de funcionamento é pouco conhecido, ou seja, não se sabe exatamente todos os parâmetros que influenciam o sistema.

A outra classe de modelos são os chamados modelos empíricos e são construídos diretamente a partir dos dados observados sem que seja necessário conhecimento prévio do comportamento interno do sistema e por isso conhecidos como modelos “caixa preta”. Identificação de Sistemas é área que lida com o problema de modelagem empírica.

2.2.3. Classificação dos modelos matemáticos

Para o desenvolvimento de modelos matemáticos é importante analisar algumas propriedades fundamentais dos sistemas. Estas propriedades permitem definir que tipos de ferramentas teóricas serão necessárias para modelar o problema. Em função destas propriedades os modelos matemáticos podem ser basicamente classificados da seguinte forma:

- **Linear ou não linear**

Segundo OGATA, 2003, um sistema é dito linear se o princípio da superposição se aplicar a ele. Este princípio afirma que a resposta produzida pela aplicação simultânea de duas funções diversas é a soma das duas respostas individuais. Caso este princípio não se aplique o sistema é considerado não-linear. Então, para um sistema linear, a resposta a

diversas entradas pode ser calculada tratando uma entrada de cada vez e somando os resultados. Estes modelos desprezam efeitos de segunda ordem. Na prática, as variáveis que os definem estão elevadas todas à primeira potência e não há produto entre variáveis. Considere os exemplos de modelos:

- $2\ddot{y}x(t) + x(t) = u(t)$ – é linear
- $2\ddot{y}x(t) + x^2(t) = u(t)$ – e
- $2\ddot{y}x(t) + x(t)u(t) = 0$ – não são lineares

- **Variante ou invariante no tempo**

Um sistema é dito invariante quando sua resposta independe do instante de tempo em que uma entrada é aplicada, ou seja, o comportamento do sistema que está sendo modelado não varia com o tempo. Segundo (AGUIRRE, 2004) isto não significa que as variáveis do sistema têm seus valores constantes, normalmente esses valores flutuam com o tempo. Caso haja ao menos uma entrada (parâmetro) variando ao longo do tempo o sistema é variante no tempo, ou seja, a resposta desse sistema depende do instante de tempo em que uma entrada é aplicada. Por exemplo, um painel solar é um sistema variante no tempo, pois a acumulação de pó diminui o seu rendimento.

De acordo com AGUIRRE, 2004, determinar se o sistema é ou não invariante no período em que se considera o sistema, é uma das premissas fundamentais para a modelagem matemática. O fato de ser invariante simplifica muito o problema da modelagem e implica em que apenas um modelo é requerido para representar o sistema.

Quanto ao número de entradas os modelos recebem as seguintes denominações:

- SISO — *Single Input, Single Output*: O sistema possui uma única entrada e uma única saída;
- MISO — *Multiple Input, Single Output*;
- SIMO — *Single Input, Multiple Output*;
- MIMO — *Multiple Input, Multiple Output*;

A combinação de tais atributos dá origem a diversos tipos de modelos: modelos de sistemas lineares invariantes no tempo, modelos de sistemas não lineares discretos variantes no tempo, modelos não lineares contínuos invariantes no tempo.

O foco deste estudo será o desenvolvimento de um modelo dinâmico, não linear, discreto e variante no tempo.

2.2.4. Representação de sistemas dinâmicos

As duas formas de representações matemáticas de sistemas citadas por (HAYKIN, 2004) são: representação de modelos em espaços de estados e representação entrada-saída. A representação por espaço de estados é reconhecidamente a mais completa e, portanto mais vantajosa, porque além das variáveis de entrada e saída, as informações internas do sistema estão disponíveis, ou seja, variáveis de estado. Entretanto, nem sempre é possível se obter as variáveis de estado, sendo para problemas não lineares uma tarefa de alta complexidade. Para os casos em que se pode assumir que as variáveis de entrada e saída observadas contém informação suficiente para representar os estados do sistema, modelos representados por entrada-saída são mais comumente empregados (BRAGA, ET al., 1999).

2.2.4.1. Representação de sistemas dinâmicos em entrada-saída.

Um sistema é descrito por um modelo de entrada-saída quando é modelado através de um mapeamento de um conjunto de sinais de entrada $u(t)$ para um conjunto de sinais de saída $y(t)$. Para sistemas discretos, representados por equações a diferenças, o mapeamento pode ser escrito como:

$$y(k) = f[u(k-1), u(k-2), \dots, u(k-n), y(k-1), y(k-2), \dots, y(k-n)]$$

$$y(k) \in \mathfrak{R}^P - \text{saídas do sistema}$$

$$u(k) \in \mathfrak{R}^S - \text{entradas do sistema}$$

onde n é um valor inteiro que descreve o número de amostragens anteriores de sinais u e y que são necessários para predizer a próxima saída.

2.2.5. Identificação de Sistemas Dinâmicos

A identificação de sistemas é a área da teoria de matemática de sistemas que lida com o problema da representação matemática de modelos baseados em dados experimentais, medidos ou observados da própria operação do sistema real. A idéia básica é permitir a construção de modelos matemáticos de um sistema dinâmico baseado em dados medidos, fazendo-se ajustes de parâmetros para um dado modelo de forma que as saídas se aproximem o mais próximo possível das saídas medidas (LJUNG, 1987).

Essencialmente, o procedimento de construção de modelos a partir de dados é composto das seguintes etapas (LJUNG, op. cit.):

A - Obtenção de dados

Os dados para modelagem tanto podem ser obtidos através de um projeto experimental, de um processo de monitoração, por exemplo, quanto da operação normal do sistema. A vantagem da monitoração é que o usuário pode determinar quais as variáveis que devem ser observados e por quanto tempo.

B - Escolha da representação e estrutura dos modelos

A escolha do tipo de representação dos modelos depende basicamente da finalidade destes, do conhecimento que se tem do sistema real, da sua complexidade e também da qualidade e quantidade de dados medidos. Se a representação matemática for uma Rede Neural, pode-se representá-la através de espaço de estado ou entrada-saída, por exemplo, e a escolha do número de neurônios e número de camadas de neurônios define a estrutura do modelo.

C - Validação dos modelos

A validação consiste em determinar se os modelos escolhidos representam características essenciais do sistema real em estudo, ou seja, como os modelos se comportam quando tentam reproduzir os dados medidos do sistema real.

Em função da complexidade dos modelos, do fato de serem lineares ou não lineares, do conhecimento da distribuição estatística das variáveis, os modelos empíricos podem ser paramétricos e não paramétricos.

Métodos de identificação de sistemas paramétricos são aqueles cuja estrutura do relacionamento funcional entre as variáveis dependentes e independentes é conhecida (LIMA, 2000). Estes métodos se baseiam no conhecimento de expressões analíticas da distribuição de probabilidades. Isto significa dizer que para a construção do modelo são assumidas hipóteses da distribuição que aproxima os dados do sistema, uma distribuição gaussiana, por exemplo. São também conhecidos como modelos “caixa branca”.

Para os métodos não paramétricos não existe conhecimento (ou pouco) a priori sobre a estrutura do modelo, que será determinada pelos dados medidos do sistema. São conhecidos como métodos de distribuição livre ou “caixa preta”.

Vale ressaltar que existe uma escala entre os modelos paramétricos e não paramétricos. Há modelos puramente não paramétricos numa extremidade, nenhuma suposição estatística sobre as variáveis é feita, sendo que para construí-los é necessário uma grande quantidade de dados para o treinamento. Numa outra extremidade encontram-se os modelos puramente paramétricos, para os quais a quantidade de dados necessária é mínima e conseqüentemente é exigido um conhecimento grande sobre a estrutura do modelo. É comum na literatura o termo semiparamétrico ou “caixa-cinza” para expressar modelos que se utilizam tanto da abordagem paramétrica quanto da não paramétrica.

Diversos métodos paramétricos e não-paramétricos são amplamente empregados para identificação de sistemas a exemplo de Redes Neurais, Análise Multivariada e Métodos *Box-Jenkins*. As redes neurais são consideradas ferramentas não paramétricas se a arquitetura da rede neural puder ser definida em função do problema de aproximação. Se a estrutura (estrutura de conexão e número de neurônios) da rede neural for definida previamente, independente do problema de aproximação, ela representa um modelo paramétrico (VON ZUBEN, 1996).

2.3. Redes Neurais Artificiais

Neste ítem são apresentados os fundamentos teóricos das redes neurais artificiais, descrevendo sua célula básica, sua arquitetura e os algoritmos de aprendizagem.

2.3.1. Introdução

As redes neurais foram originalmente desenvolvidas, na década de 40, por Warren McCulloch (psiquiatra) e Walter Pitts (matemático), os quais projetaram uma estrutura que é conhecida como a primeira rede neural, onde fizeram uma analogia entre células nervosas vivas e o processamento eletrônico. O trabalho consistia de um modelo de resistores variáveis e amplificadores representando conexões sinápticas (SILVA, 1998).

Desde então, mais enfaticamente a partir dos anos 80, diversos modelos de redes neurais artificiais têm surgido com o propósito de desenvolver e aplicar esta tecnologia, também conhecida como conexionismo ou sistemas de processamento paralelo e distribuído, (BRAGA ET al., 2000). Muitos desses estudos foram bem sucedidos o que comprova a eficácia das RNA's, que podem ser aplicadas a diversas áreas, assim como em controle de processos, utilizando a linha adaptativa ou preditiva como observado em FILETI (1995), TASINAFFO (2004), VIEIRA (2002).

2.3.2. Neurônio biológico

No cérebro humano, o sistema nervoso engloba bilhões de neurônios que interagem de uma maneira complexa (CARVALHO, 2006).

Os neurônios são formados por dendritos que são um conjunto de terminais entrada, pelo corpo central, e pelos axônios que são longos terminais de saída. A comunicação entre os neurônios é feita via sinapses que são as regiões onde dois neurônios entram em contato e por onde os impulsos nervosos são transmitidos entre os mesmos, (CARVALHO, 2006). Os impulsos recebidos por um neurônio são processados, e atingindo um dado limiar de ação provoca um disparo, produzindo uma substância

neurotransmissora que flui do corpo celular para o axônio, e do axônio aos dendritos de outros neurônios (PAULA, 2007). O potencial de ação, de intensidade e duração destes impulsos, é provocado em um neurônio pelo efeito combinado de todos os estímulos de excitação e inibição que chegam até ele (integração sináptica), e por despolarização que seja suficientemente significativa para atingir o limiar de secreção dos neurotransmissores (KLASSEN, 2008).

O sistema nervoso detecta estímulos externos e internos, tanto físicos quanto químicos, e enviam estas informações de entrada (“*inputs*”) através dos neurônios ao cérebro. Dentro do cérebro ocorrem reações eletroquímicas que fazem com que estes sinais de entrada se propaguem entre os neurônios, e então, baseado nestas informações o cérebro chega a uma conclusão que é enviada como resposta na forma de informação de saída (“*outputs*”) que podem ser reações musculares e glandulares (FILETI, 1995).

2.3.3. Neurônio artificial

Assim como no cérebro humano, nas redes neurais artificiais o neurônio artificial é o elemento processador básico. Estes neurônios (Figura 2.2) são estruturas lógico-matemáticas que, da mesma maneira que os neurônios biológicos, possuem entradas e saídas. São dispositivos simples capazes de calcular o somatório de N entradas ponderadas, cujo resultado será a entrada de uma função não-linear denominada função de ativação, onde o valor de saída do neurônio será calculado.

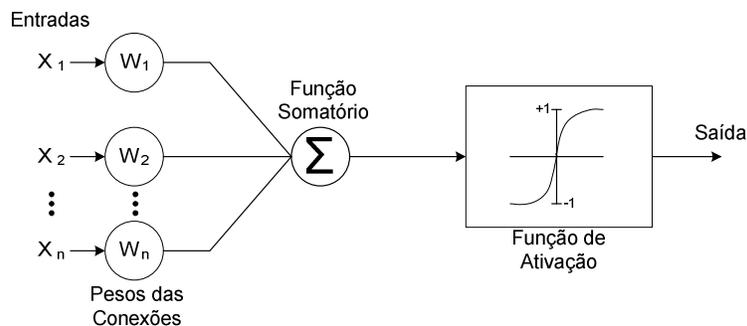


Figura 2. 2 Neurônio artificial.

A rede neural artificial é um sistema de neurônios, ligados por conexões sinápticas, e dividido em neurônios de entrada, que recebem estímulos do meio externo, neurônios internos ou ocultos e neurônios de saída, que se comunicam com o exterior.

Os neurônios internos são de suma importância na rede neural, pois provou-se que sem estes torna-se impossível a resolução de problemas linearmente não separáveis. Em outras palavras pode-se dizer que uma rede é composta por várias unidades de processamento, cujo funcionamento é bastante simples. Essas unidades, geralmente são conectadas por canais de comunicação que estão associados a determinado peso. As unidades fazem operações apenas sobre seus dados locais, que são entradas recebidas pelas suas conexões (RUSSELL & NORVIG, 1995).

Os cálculos realizados por um neurônio artificial são dados por:

$$y_j = \sum(x_i \cdot W_{ij}) + b_i \quad (2.11)$$

$$z_j = f(y_j)$$

Onde: z: é o sinal de ativação do neurônio;

x: é o valor de uma entrada do neurônio;

y: é o estímulo dos sinais de saída;

W: é o valor de cada peso do neurônio;

b: bias;

f: é normalmente uma função não linear do tipo sigmoidal ou tangente hiperbólica.

2.3.4. Topologias de Redes Neurais Artificiais (RNA's)

O comportamento inteligente de uma Rede Neural Artificial vem das interações entre as unidades de processamento da rede (neurônios). A maioria dos modelos de redes

neurais possui alguma regra de treinamento, onde os pesos de suas conexões são ajustados de acordo com os padrões apresentados; ou seja, elas aprendem através de exemplos. Arquiteturas neurais são tipicamente organizadas em camadas (Figura 2.3), com unidades que podem estar conectadas às unidades da camada posterior. Por convenção a camada que inicialmente recebe os dados é chamada de entrada, a camada intermediária ou oculta, que podem ser uma ou várias, e por último, a camada de saída. Cada camada pode ter de 1 a n neurônios artificiais (DHAR & STEIN, 1997).

Usualmente as camadas de uma rede neural são classificadas em três grupos:

- Camada de Entrada: esta na verdade não é formada por neurônios reais, aqui simplesmente os padrões são apresentados à rede, e os valores distribuídos para os neurônios da camada intermediária;
- Camadas Intermediárias ou Ocultas: onde é feita a maior parte do processamento, através das conexões ponderadas; podem ser consideradas como extratoras de características;
- Camada de Saída: onde o resultado final é concluído e apresentado.

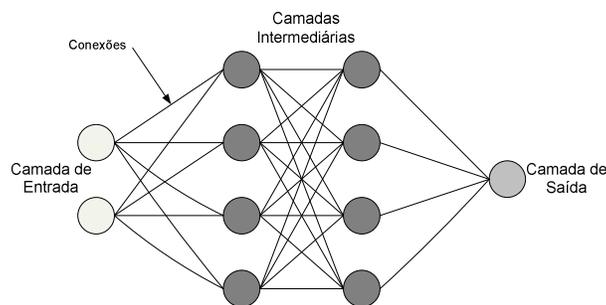


Figura 2. 3 Organização da rede neural em camadas. [TONSIG, 2000]

A rede neural passa por um processo de treinamento a partir dos casos reais conhecidos, adquirindo, a partir daí, a sistemática necessária para executar adequadamente o processo desejado dos dados fornecidos. Sendo assim, a rede neural é capaz de extrair regras básicas a partir de dados reais, diferindo da computação programada, onde é necessário um conjunto de regras rígidas pré-fixadas e algoritmos.

Redes neurais são também classificadas de acordo com a arquitetura em que foram implementadas, características de seus nós ou neurônios, regras de treinamento, e tipos de modelos.

2.3.5. Arquitetura de RNA's

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

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

2.3.5.1. Perceptron de camada única

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

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

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

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

classes devem ser linearmente separáveis para que o perceptron funcione adequadamente.

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

A topologia original do perceptron é composta por unidades de entrada, por um nível intermediário formado pelas unidades de associação e por um nível de saída formado pelas unidades de resposta. Embora esta topologia original possua três níveis, ela é conhecida como perceptron de uma única camada, já que somente o nível de saída (unidade de resposta) possui propriedades adaptativas (KOVÁCS, 2006).

2.3.5.2. Perceptron multicamadas

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

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

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

entradas linearmente separáveis. Na camada de saída, uma outra rede combina as saídas produzidas pelas redes da primeira camada, gerando, assim, a classificação final (CYBENKO, 1989).

2.3.6. Arquitetura do MLP

As redes MLP (*multi-layer perceptron*) apresentam um poder computacional maior do que aquele apresentado pelas redes sem camadas intermediárias. As RNA's do tipo MLP podem tratar com dados que não são linearmente separáveis. A precisão obtida e a implementação de uma rede neural MLP dependem do número de nodos utilizados nas camadas intermediárias, e um dos problemas é a definição de uma função de ativação.

Segundo CYBENKO(1989), nota-se que duas características desta estrutura podem ser citadas:

- rede progressiva (*feedforward*): o sinal se propaga camada a camada em sentido progressivo, conectando-se as saídas dos neurônios de uma camada às entradas dos neurônios da camada seguinte, sem laços de realimentação;
- com relação à conexão: a rede pode ser completamente conectada, onde cada nó de uma camada é conectado aos nós da camada adjacente, ou parcialmente conectado, onde algumas sinapses podem estar faltando, o que ocorre com as redes localmente conectadas.

2.3.6.1. Definição de uma topologia ideal para o MLP

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

Segundo TAFNER (1996), o projeto de uma rede MLP requer a consideração dos aspectos de:

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

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

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

- não utilizar unidades demais, o que pode levar a rede a memorizar padrões de treinamento, ao invés de extrair as características gerais que permitem a generalização ou reconhecimento de padrões não vistos durante o treinamento (este problema é denominado de *overfitting*);
- não utilizar um número muito pequeno, que possa forçar a rede a gastar tempo em excesso tentando encontrar uma representação ótima (as unidades utilizadas podem ficar sobrecarregadas, tendo que lidar com um elevado número de restrições).

2.3.7. Treinamento de RNA's

A propriedade mais importante das redes neurais é a habilidade de aprender de seu ambiente e com isso melhorar seu desempenho. Isto é feito por meio de um processo iterativo de ajustes aplicado a seus pesos, conhecido como treinamento. O aprendizado ocorre quando a RNA atinge uma solução generalizada para uma classe de problemas (PAULA, 2007). Esta etapa consiste em um processo iterativo de ajuste dos pesos sinápticos das conexões entre as unidades de processamento, durante a exposição de

exemplos, em resposta à quantidade de erros gerados pela rede. Ao final deste processo a rede guarda o conhecimento que adquiriu do ambiente em que esta operando (TAFNER, 1996).

Na modelagem da rede pode ocorrer um ou dois efeitos de resposta, conhecido como overfitting e underfitting. O overfitting é uma situação que ocorre quando há sobre parametrização da rede, ou seja, quando a rede tem mais parâmetros (pesos) do que necessário para a resolução do problema. Por sua vez, o underfitting ocorre quando a rede tem menos parâmetros do que necessário. O objetivo do treinamento deve ser então o de encontrar o ajuste ideal na fronteira entre o overfitting e o underfitting. No entanto, estimar o número de parâmetros é uma tarefa árdua que requer conhecimento sobre a complexidade do problema. Este conhecimento normalmente não está disponível, principalmente tratando-se de problemas multidimensionais, pois muitas vezes é este conhecimento que se deseja obter por meio do processo de modelagem (REZENDE, 2003).

Diversos métodos para treinamento de redes foram desenvolvidos, podendo esses ser agrupados em dois modelos principais: treinamento supervisionado e treinamento não-supervisionado (HAIKYN, 2001).

2.3.8. Treinamento supervisionado

Neste trabalho somente foi utilizado o treinamento supervisionado, então será dada ênfase a somente este modelo. Este tipo de treinamento é caracterizado pela utilização de um conjunto de dados que possua vetores de entrada, e, vetores de saída que indiquem à rede a resposta desejada para o padrão de entrada. O ajuste dos pesos ocorre quando o sistema compara a saída da rede com a resposta desejada previamente conhecida (PAULA, 2007).

Neste tipo de treinamento, os estímulos de entrada são apresentados a rede, a qual calcula uma resposta utilizando como parâmetros os valores de pesos atuais. A resposta obtida pela rede é comparada com a resposta desejada, dada anteriormente. A partir daí o erro é calculado e retropropagado através da rede e os pesos são ajustados

para cada vetor, até que possibilitem à rede classificar o mais corretamente possível as entradas de treinamento (KLASSEN, 2008).

Uma desvantagem deste tipo de treinamento consiste no fato de que a rede não aprende novas estratégias para situações particulares que não foram consideradas no conjunto de treinamento.

Um algoritmo de treinamento supervisionados muito popular é baseado no método do gradiente, comumente denominado algoritmo de retropropagação (*backpropagation*), que leva este nome devido ao fato do sinal do erro ser retropropagado através da rede, camada por camada.

O algoritmo de retropropagação de erro, ou simplesmente retropropagação, é um algoritmo utilizado no treinamento de redes neurais multicamadas com uma ou mais camadas escondidas. Segundo BALLINI, 2000, durante o treinamento com o algoritmo *backpropagation*, a rede opera em uma sequência de dois passos. Primeiro, um padrão é apresentado à camada de entrada da rede. A atividade resultante flui através da rede, camada por camada, até que a resposta seja produzida pela camada de saída. No segundo passo, a saída obtida é comparada à saída desejada para esse padrão particular. Se esta não estiver correta, o erro é calculado. O erro é propagado a partir da camada de saída até a camada de entrada, e os pesos das conexões das unidades das camadas internas vão sendo modificados conforme o erro é retropropagado.

O método *backpropagation* atualiza os pesos incrementalmente depois de analisar cada par entrada-saída, depois disso diz-se que uma época foi concluída. Este treinamento em geral requer muitas épocas (PAULA, 2007).

A correção do valor dos pesos da rede neural é feita utilizando-se o método dos mínimos quadrados, com o objetivo de encontrar um valor para o peso que minimize o erro na saída da rede. Mas, mesmo conhecendo o erro geral da rede, não é possível determinar os pesos exatos para corrigi-los. Porém, com base nesta informação, pode-se estabelecer a direção à qual os pesos devem ser ajustados para minimizar o erro quadrático total de saída da rede (SSE). Sendo conhecida esta direção é possível que os pesos sejam ajustados até que se chegue ao menor erro possível na saída da rede neural.

Segundo, Paula, 2007, o ajuste de um peso w_{ij} que define seu valor para a próxima iteração é definido pela equação 2.12.

$$w_{ij}(t + 1) = w_{ij}(t) + \eta \delta_j x_j \quad (2.12)$$

Onde: $w_{ij}(t)$: é o peso sináptico da neurônio j no tempo t;

η : é uma constante positiva que define a taxa de aprendizado;

δ_j : é o gradiente local do erro para o neurônio;

x_j : é o sinal de entrada da rede.

Segundo, HAYKIN, 1999, o gradiente local do erro pode ser calculado da seguinte forma: se o neurônio pertencer a camada de saída da rede neural temos:

$$\delta_j = y_j(1 - y_j)(d_j - y_j) \quad (2.13)$$

Onde: d_j : representa a saída desejada;

y_j : representa a saída real;

E, se o neurônio pertencer a camada intermediária temos:

$$\delta_j = x_j(1 - x_j) \sum \delta_k w_{jk} \quad (2.14)$$

Onde k representa todos os elementos processadores acima de j.

2.3.8.1. Algoritmo de Levenberg-Marquardt com regularização bayesiana

O método Levenberg-Marquardt foi desenvolvido para calcular uma aproximação da matriz hessiana. Quando a função objetivo tem a forma da soma dos quadrados, como é o caso da função erro de uma rede MLP (multi-layer perceptron), a matriz hessiana pode ser aproximada como:

$$\nabla V(x) = J^T(x) \cdot E(x) \quad (2.15)$$

$$\nabla^2 V(x) = J^T(x).J(x) + S(x) \quad (2.16)$$

Onde:

$$V(x) = \sum_{i=1}^n e_i^2(x);$$

$e(x)$ = vetor de erros da rede;

$\nabla^2 V(x)$ = matriz hessiana;

$J(x)$ = matriz jacobiana;

$$S(x) = \sum_{i=1}^n e_i \nabla^2 e_i(x);$$

O método Gauss-Newton assume que $S(x) \approx 0$, assim a equação de atualização dos pesos é descrita como:

$$\Delta_x = [J^T(x).J(x)]^{-1}.J^T(x).e(x) \quad (2.17)$$

O método Levenberg-Marquardt modifica Gauss-Newton, incluindo um fator μ responsável pelo ajuste do tamanho do incremento realizado pelo algoritmo. A equação 2.18 descreve esta modificação.

$$\Delta_x = [J^T(x).J(x) + \mu I]^{-1}.J^T(x).e(x) \quad (2.18)$$

Desta forma, quando o algoritmo reduz $V(x)$, μ é dividido por um fator $\mu' > 1$. Da mesma forma, quando $V(x)$ é incrementado, μ é multiplicado pelo fator $0 < \mu' < 1$. Uma observação importante deste algoritmo é que se μ for muito grande o algoritmo se torna gradiente descendente. Por outro lado, se for muito pequeno, o algoritmo se torna Gauss-Newton (HAGAN & MENHAJ, 1994).

Como a matriz jacobiana, descrita na equação 2.19, é composta pelas derivativas de primeira ordem dos pesos, bias e erros da rede, esta pode ser facilmente calculada através da técnica utilizada no algoritmo backpropagation, sendo muito menos complexa que a matriz hessiana.

$$J(x) = \begin{bmatrix} \frac{de_1(x)}{dx_1} & \frac{de_1(x)}{dx_2} & \dots & \frac{de_1(x)}{dx_m} \\ \frac{de_2(x)}{dx_1} & \frac{de_2(x)}{dx_2} & \dots & \frac{de_2(x)}{dx_m} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{de_n(x)}{dx_1} & \frac{de_n(x)}{dx_2} & \dots & \frac{de_n(x)}{dx_m} \end{bmatrix} \quad (2.19)$$

Durante o treinamento de uma rede neural, o objetivo principal é gerar uma rede que produza um baixo erro de aprendizado e, principalmente, que seja capaz de responder adequadamente à apresentação de novos dados, isto é, que a rede tenha uma boa capacidade de generalização.

Um método considerado eficiente para aperfeiçoar a generalização de uma rede é a regularização, que se baseia na restrição dos valores dos pesos da rede. A idéia é manter os pesos com valores pequenos, possibilitando uma suavização da saída da rede. Este método tem como premissa que a função modelada pela rede tem um certo grau de suavidade.

Tipicamente, o treinamento de uma RNA tem como meta a minimização do erro médio quadrático (SSE).

$$SSE = \sum_{i=1}^n (y_i - a_i)^2 \quad (2.20)$$

Onde:

a_i = resposta da rede;

y_i = resposta desejada;

O treinamento a ser utilizado para a avaliação proposta nesta dissertação baseia-se no algoritmo de Levenberg-Marquardt com regularização Bayesiana (função “trainbr” no MATLAB). Segundo LOPREATO (2009), tal algoritmo usa conceitos básicos da regra delta generalizada realizando uma ponte entre dois algoritmos de treinamento existentes, o método de Gauss-Newton e o método do gradiente descendente, incorporando a velocidade de convergência do primeiro método (Gauss-Newton) com a precisão fornecida pelo segundo método (gradiente decrescente) para as regiões próximas ao mínimo da função objetivo. Os métodos apresentados possuem, no entanto, o inconveniente de estarem sujeitos a sobreajustes, resultando na perda de um dos

principais trunfos das redes neurais: sua capacidade de generalização. Para minimizar este problema, os algoritmos de regularização modificam a função objetivo, de forma que esta seja dependente não somente da soma do erro quadrático (SSE) da rede, mas também da soma dos pesos quadráticos da rede (SSW). A função a ser minimizada, neste caso, proposta por FOREESE e HAGAN (1997), é expressa pela equação 2.21.

$$F = \alpha.SSE + \beta.SSW \quad (2.21)$$

Onde:

F = Função objetivo;

SSE = Soma dos erros quadráticos;

SSW = Soma dos pesos quadráticos;

α e β = Parâmetros ajustáveis da função objetivo;

$$SSW = \frac{1}{n} \sum_{i=1}^n w_i^2 \quad (2.22)$$

O comportamento do treinamento depende dos valores dos parâmetros α e β . Se $\alpha \ll \beta$, o algoritmo conduzirá a função objetivo para a minimização do erro. No entanto, se $\alpha \gg \beta$ a ênfase do algoritmo será a redução dos valores dos pesos da rede, produzindo uma saída mais suave. Sendo assim, o principal problema da implementação da regularização é a definição dos valores dos parâmetros α e β , que podem ser obtidos aplicando-se a Lei de Bayes (BROMELING, 1985). Segundo FOREESE E HAGAN (1997), a incorporação da regularização Bayesiana ao método de Levenberg-Marquardt conduz as redes à respostas mais suaves e reduzida probabilidade de sobreajuste.

2.3.9. Etapas de projeto de uma rede neural

O projeto de uma rede neural resume-se a algumas etapas que devem ser executadas todas em sequência, de forma iterativa e até mesmo com diversos ciclos de repetição. A Figura 2.4 mostra um diagrama de blocos tirado de TAFNER, 1996, que mostra as principais etapas para se desenvolver e treinar uma rede neural.

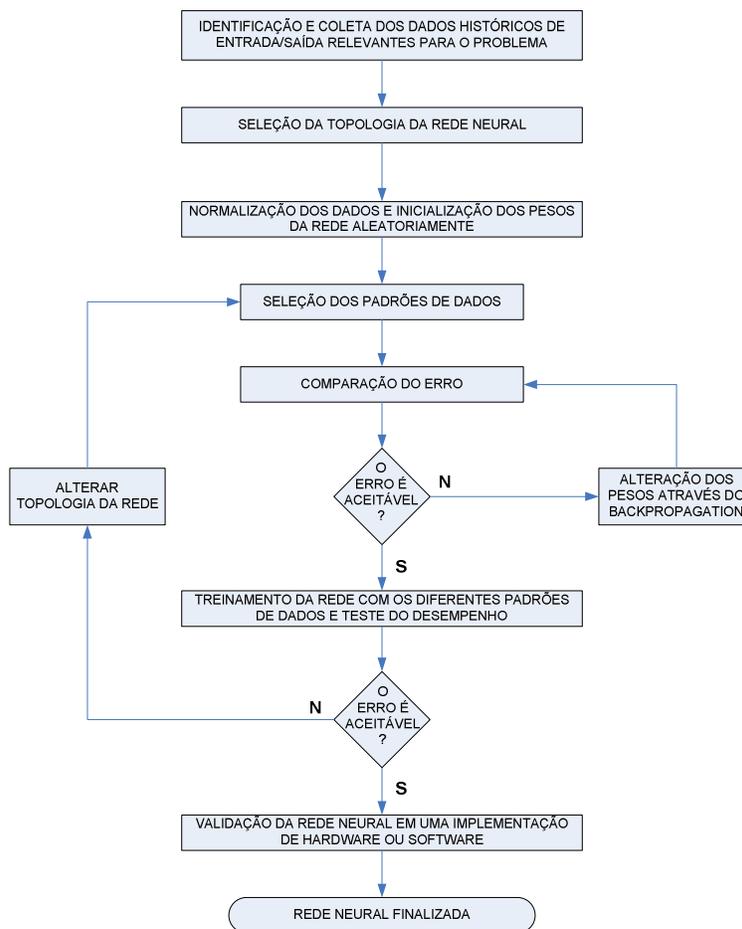


Figura 2. 4 Fluxograma do desenvolvimento de uma rede neural.

A escolha da arquitetura inicial para uma rede neural, ainda trata-se de um processo pouco compreendido e, a escolha da arquitetura ideal depende da experiência do projetista, e, na maioria das vezes ainda é feita através do método heurístico.

2.3.10. Redes Neurais na Identificação de Sistemas Dinâmicos não lineares

A identificação é uma área relevante em muitos campos da engenharia como é o caso de sistemas de refrigeração, sendo esta identificação a modelagem de um processo, para propósitos de previsão/compreensão do comportamento do processo.

Um importante passo na modelagem de sistemas é escolher uma estrutura adequada que represente o comportamento dinâmico do sistema. Algumas representações utilizadas na modelagem de sistemas não lineares são: Redes Neurais;

Funções de Base Radial; Séries de Volterra; “Wavelets”; Funções Polinomiais e Racionais; Equações Diferenciais e Polinomiais (AGUIRRE, 2004).

Devido à grande complexidade existente em muitos problemas físicos reais, desenvolver um modelo matemático que represente adequadamente a dinâmica do processo é uma tarefa praticamente impossível. Para LINHARES ET al. (2008), as RNA's, através de seu processo de aprendizagem e de sua capacidade de aproximação universal, conseguem representar a função correspondente à dinâmica do sistema com relativa simplicidade.

Segundo COELHO, 2000, 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, ou seja, modelos caixa-preta. Os modelos obtidos através de redes neurais permitem levar em consideração as não linearidades do processo, bem como as interações entre as suas variáveis (NARENDRA & PARTHASARATHY, 1990). Além disto, os modelos neurais são formados basicamente por equações algébricas, o que lhes confere rapidez para predição em tempo real.

Teorias, novas arquiteturas e aplicações de redes neurais para o problema de identificação e controle de sistemas dinâmicos lineares e não lineares têm sido reportadas por diversos autores (NARENDRA & PARTHASARATHY, 1990; CHEN AND BILLINGS, 1992; NERRAND ET al., 1994; RIVALS ET al., 1996). RNA's são citadas como poderosas ferramentas para identificação e controle principalmente quando:

- Há pouco conhecimento sobre o sistema;
- A estrutura do sistema apresenta não-linearidades significativas e,
- Somente dados de entrada-saída estão disponíveis.

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

Na identificação de sistemas a rede neural trabalha com pares de vetores $u(t)$ e $y(t)$ que são os dados de entrada e de saída do sistema respectivamente. O sinal de entrada é aplicado à rede e ela fornece uma saída $\hat{y}(t)$, o sinal erro é calculado e os pesos da rede são ajustados de tal forma que a saída da rede $\hat{y}(t)$ aproxime-se da saída real $y(t)$ para todo $u(t)$. O algoritmo de treinamento para o ajuste dos pesos da rede corresponde ao estágio de estimação de parâmetros. E a seleção da arquitetura da rede corresponde ao estágio de seleção do modelo (REZENDE, 2003).

Segundo HUNT ET al. 1992, citado por ASSIS, 2001, do ponto de vista de aplicação em controle de processos, a identificação de sistemas não lineares atende a dois propósitos específicos. A modelagem preditiva, que é o procedimento de treinar uma rede neural para representar o comportamento do sistema um ou mais passos à frente, tendo o modelo uma característica preditiva. E, a modelagem inversa, que tem o objetivo de encontrar um modelo que identifique o sistema a ser controlado de modo inverso, ou seja, dada a saída requerida do sistema (*set-point*), o modelo matemático fornecerá o valor da entrada (variável manipulada) a ser aplicada ao sistema de modo a se obter a saída desejada.

Segundo CANCELIER, 2004, a modelagem empírica baseada em redes neurais vem sendo bastante utilizada em processos químicos, principalmente na identificação de sistemas, ou no emprego de técnicas de controle preditivo. Neste contexto, modelos baseados em RNA's, mostram-se como uma poderosa ferramenta na modelagem de sistemas não lineares, devido a duas de suas qualidades, a rápida adaptabilidade e a inerente capacidade de aproximação. As informações feitas neste parágrafo são compartilhadas por diversos pesquisadores como, CHEN & NARENDRA (2002), ŁAWRYN' CZUK (2006), AL-GALAFF (2002), JALILI-KHARAAJOO & ARAABI (2004), LAZAR & PASTRAVANU (2000), RONCO & GAWTHROP (1997), GENÇAY & LIU (1997), VIEIRA (2002), ASSIS (2001), o que prova a viabilidade de se aplicar este tipo de modelagem na aplicação de sistemas não lineares, como é o objetivo deste estudo.

2.4. Sistema de Refrigeração

O ciclo de compressão de vapor possui basicamente os seguintes componentes: o evaporador, o compressor, o condensador, os aparelhos de expansão, as tubulações de interconexão e uma variedade de componentes auxiliares. A operação deste sistema frigorífico básico (Figura 1.5) é caracterizada pela retirada de calor da carga de refrigeração agregada ao evaporador, onde o refrigerante troca calor e se vaporiza (7). O vapor resultante é succionado pelo compressor (1). Então, uma quantidade de energia mecânica é adicionada ao compressor, elevando a pressão e temperatura do gás de descarga (2). Este gás flui para o condensador (3), trocando calor com o fluido de interface do trocador de calor e sai na condição de líquido saturado ou subresfriado, ainda sob alta pressão (4). Este líquido (5) passa posteriormente por um aparelho de expansão onde parte do líquido evapora instantaneamente (6), promovendo o resfriamento da fase líquida remanescente, além da redução de pressão do fluido que é utilizado novamente no evaporador (7), fechando, desta forma, o ciclo termodinâmico.

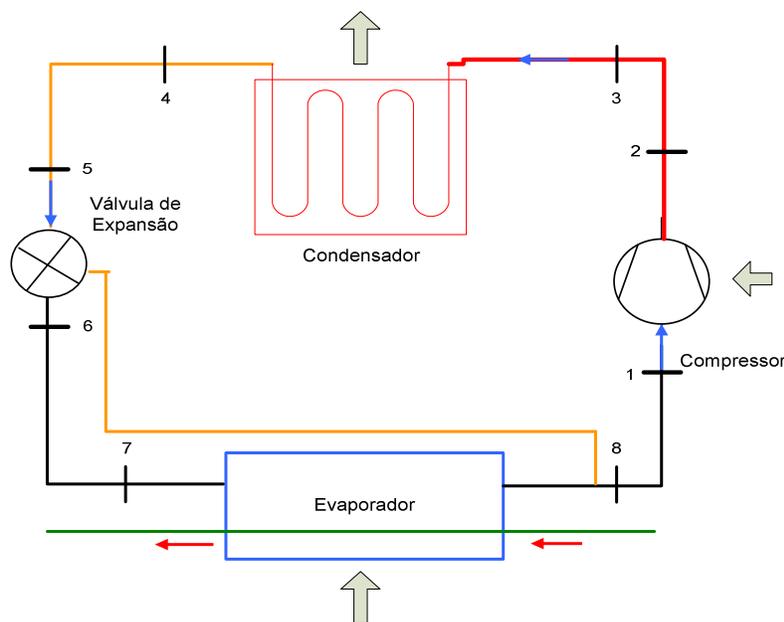


Figura 2. 5 Esquema de um sistema de refrigeração básico.

2.4.1. A importância em se controlar estes sistemas

Sabe-se que os sistemas de refrigeração são grandes consumidores de energia elétrica e, além disso, o problema da destruição da camada de ozônio pelos refrigerantes CFC's utilizados nos ciclos frigoríficos proporcionaram uma preocupação adicional quanto aos seus efeitos sobre o meio ambiente. Estes dois fatores têm motivado não só o setor produtivo como a comunidade científica a estudar formas de melhorar o desempenho energético dos sistemas de refrigeração. O processo de melhoria energética em um sistema de refrigeração inicia-se na fase de projeto, com análise de eficiência exergética de seus componentes e a implementação de controladores nas variáveis do processo, (PINELLI, 2008)

A pesquisa científica de sistemas de refrigeração, buscando economia de energia tem sido realizada e apresentada de diversas maneiras, tais como a utilização de novos processos de degelo (KRAKOW ET AL; 1992), a utilização de válvulas eletrônicas para expansão, sub-resfriamentos, projetos e dimensionamentos de tubulações, além da utilização de microcontroladores, entre outros.

Com a evolução dos inversores de frequência surge a possibilidade de se atuar diretamente na fonte de maior consumo de energia em um sistema de refrigeração, que são os compressores. A partir desta evolução tem-se estudado aplicações para a sua utilização. Uma revisão sobre dispositivos de controle de capacidade para bombas de calor foi apresentada por PARREIRA e PARISE (1993). Esse estudo mostrou que o sistema operando com um compressor de rotação variável é uma das alternativas mais promissoras em relação ao sistema "on-off" tradicional.

TASSOU & QUERSHI (1997) avaliaram o desempenho de modo dinâmico e análise de energia em compressores de refrigeração controlados por inversores de frequência. Os resultados mostram que partidas sem utilização de inversores de frequência consomem 5% a mais, do que com a utilização do inversor. Isto principalmente devido à rampa de partida em inversores. Neste estudo, também foi verificado que a redução de consumo de energia é uma função do tipo de compressor empregado no sistema de velocidade variável.

TASSOU & QUERSHI (1998) estudaram o comparativo de desempenho de compressores de deslocamento positivo em aplicações de refrigeração com variação de velocidade. Foram considerados no estudo um compressor aberto a pistão, semi-hermético a pistão e aberto tipo parafuso. Os resultados indicam que todos os compressores foram projetados para a eficiência máxima na velocidade nominal. Quando foi aplicada a variação de velocidade, todos os compressores analisados apresentaram economia de energia em relação ao seu funcionamento em velocidade fixa. A análise demonstrou uma economia de 12 % a 24 % no consumo de energia.

POTTKER (2006) desenvolveu uma simulação numérica de um sistema de refrigeração de velocidade variável. O estudo modelou um sistema de refrigeração por compressão mecânica. No estudo foram executadas simulações com o objetivo de verificar a possibilidade de controlar o sistema de refrigeração e o superaquecimento do refrigerante na saída do evaporador variando a velocidade do compressor. Os resultados indicam que o modelo proposto pode ser usado para formular um algoritmo para controlar um sistema de refrigeração.

2.4.2. A RNA's na modelagem de sistemas de refrigeração

As aplicações de redes neurais na área de refrigeração têm aumentado significativamente desde o fim da década de 90. Sendo na grande maioria das aplicações encontradas utilizadas para a modelagem do processo e como preditores, e estes modelos utilizados posteriormente em técnicas avançadas de controle.

HABTOM (1999) utilizou redes neurais para modelar um sistema de refrigeração de uma sala. Neste experimento estabeleceu-se um modelo para a predição da temperatura e da umidade relativa do ar dentro da sala, sendo manipuladas a velocidade do compressor e do ventilador do evaporador e utilizando como entradas da rede neural, as velocidades do compressor e do ventilador e a realimentação das entradas. Os resultados mostram, como o próprio autor coloca, uma predição razoável das variáveis estabelecidas, devendo-se estudar melhor as variáveis a serem utilizadas para compor as entradas da rede neural.

RESENDE (2000) apresentou a modelagem e simulação de um sistema de refrigeração aplicado ao congelamento de produtos alimentícios, onde o modelo foi responsável pela predição do perfil termodinâmico do fluido refrigerante, e também serviu de base para a simulação da planta de refrigeração, onde verificou-se que com a utilização de uma taxa de congelamento ótima, reduz-se a carga térmica total, o que permite uma redução no consumo de energia do sistema.

Já SWIDER ET al. (2000), estudaram outro exemplo de modelagem utilizando redes neurais para modelar dois diferentes *chillers*. Estes modelos foram utilizados para fazer a predição do coeficiente de performance (COP) das plantas.

Outro trabalho que emprega a utilização de redes neurais na modelagem e predição de sistemas de refrigeração é o caso proposto por YIGIT e ERTUNC (2006) que modelaram uma unidade de refrigeração através das redes neurais, e utilizaram este modelo para fazer a predição da temperatura e da umidade do ar na saída de trocador de calor do tipo aramado (bastante utilizado em geladeiras).

Um estudo mais completo nesta linha de pesquisa utilizando redes neurais na modelagem de sistemas pode ser visto em BORJA (2006), que automatizou e fez o controle online de um sistema de refrigeração utilizando redes neurais. O modelo neural foi utilizado como preditor para as temperaturas de evaporação e de condensação do sistema. BORJA também aplicou no mesmo trabalho controladores preditivos, onde manipulava a velocidade do compressor e a abertura da válvula de expansão eletrônica, com a finalidade do uso racional de energia destes equipamentos e apresentou resultados satisfatórios.

A pesquisa mais recente encontrada nesta área foi feita por TIAN, FENG e ZHU (2008), e mostra um estudo experimental de um controlador MIMO (*multi-input, multi-output*), para um sistema de refrigeração. Assim como BORJA (2006), TIAN ET AL. (2008) analisaram controladores utilizando redes neurais e a técnica *fuzzy* para o controle da planta. O modelo neural foi utilizado para fazer a predição da temperatura e da umidade do ar na saída do trocador de calor.

Sendo assim, fica demonstrado o crescente interesse e viabilidade da utilização de redes neurais artificiais para a modelagem e predição em sistemas de refrigeração. No presente estudo, o modelo neural será desenvolvido de forma a predizer as temperaturas de evaporação e de saída do propileno-glicol (temperatura do fluido para utilidade industrial), frente à variação da rotação do compressor e da bomba de propileno. Temperaturas estas escolhidas, pois para o sistema em questão há uma maior facilidade em relação ao controle das temperaturas do propileno glicol e de evaporação devido aos inversores de frequência estarem ligados a bomba de propileno-glicol e ao compressor. E também para um controle em cascata da temperatura de saída do propileno-glicol a temperatura de evaporação é mais útil.

3. MATERIAIS E MÉTODOS

Neste capítulo serão descritos os materiais e métodos utilizados no desenvolvimento do presente estudo e serão detalhados os procedimentos realizados durante cada etapa. Inicialmente será feito um detalhamento do sistema de refrigeração utilizado para implementar os controladores. Posteriormente serão abordados o desenvolvimento e a sintonia do controlador PID, seguindo-se do modelo neural.

3.1. Unidade experimental

O sistema de refrigeração utilizado para os ensaios referentes ao desenvolvimento desta dissertação foi projetado e montado por PINELLI (2008), para a realização de estudos para a sua dissertação de mestrado, onde podem ser vistos mais detalhes sobre o sistema e seus componentes. Sistema este, montado no Laboratório de Automação e Controle de Processos (LCAP) da Faculdade de Engenharia Química da Universidade Estadual de Campinas (UNICAMP).

3.1.1. Planta de refrigeração

O ciclo de compressão de vapor possui basicamente os seguintes componentes: o evaporador, o compressor, o condensador, o aparelho de expansão, as tubulações de interconexão e uma variedade de componentes auxiliares. A operação deste sistema frigorífico básico é caracterizada pela retirada de calor da carga de refrigeração agregada ao evaporador, onde o gás refrigerante (R-22) troca calor e se vaporiza. O vapor resultante é succionado pelo compressor. Então, uma quantidade de energia mecânica é adicionada ao compressor, elevando a pressão e temperatura do gás de descarga. Este gás flui para o condensador, trocando calor com o fluido de interface do trocador de calor e sai na condição de líquido saturado ou subresfriado, ainda sob alta pressão. Este líquido passa posteriormente por um aparelho de expansão onde parte do líquido evapora

instantaneamente, promovendo o resfriamento da fase líquida remanescente, além da redução de pressão do fluido que é utilizado novamente no evaporador, fechando, desta forma, o circuito termodinâmico. No sistema utilizado neste trabalho existe um circuito secundário por onde circula uma solução aquosa de propileno-glicol que consiste no fluido frio para utilidades. O propileno que está no tanque é succionado pela bomba de evaporação e passa pelo evaporador onde troca calor com o refrigerante e novamente volta ao tanque de propileno, onde há um agitador para que a temperatura do líquido seja uniforme em todo o reservatório.

3.1.2. Descrição do protótipo experimental

O protótipo experimental de ensaios utilizado neste trabalho é, conforme mostra a Figura 3.1, composto por: um tanque de estocagem de propileno-glicol (1); bomba de evaporação (2); evaporador de placas (3); válvula de expansão, manual (4); válvula de expansão, termostática (5); motor elétrico do compressor (6); compressor, tipo aberto (7); condensador de tubos (8); separador de líquidos (9); separador de óleo (10); medidor de vazão, tipo turbina (11); condensador de placas (12); medidor de vazão magnético (13) e bomba de condensação (14).

O sistema utilizado é composto por dois circuitos: um secundário de água + propileno-glicol a 50%, e um principal de refrigerante (R-22), sendo este responsável pelo resfriamento da solução aquosa do circuito secundário.

Este sistema possui também duas opções de condensação, a água e a ar. Neste trabalho foi somente utilizada a condensação a ar, por isso iremos nos ater somente aos equipamentos que fazem parte do circuito que utiliza este tipo de condensação.

O diagrama do circuito principal é mostrado na Figura 3.2. A circulação do fluido refrigerante é feita por um compressor aberto que faz parte da unidade condensadora UC165/2-III da Bitzer Compressores Ltda. O compressor possui um deslocamento volumétrico de $4,5 \text{ m}^3/\text{h}$ a uma frequência de 60 Hz. Esta unidade condensadora possui um condensador a ar constituído por tubos de cobre e aletas de alumínio. No

condensador a ar foi instalado um ventilador axial (Separis NW/4-350) dotado de hélices de alumínio e é acionado por um motor elétrico blindado, bifásico, quatro pólos, 220 V e 60 Hz.

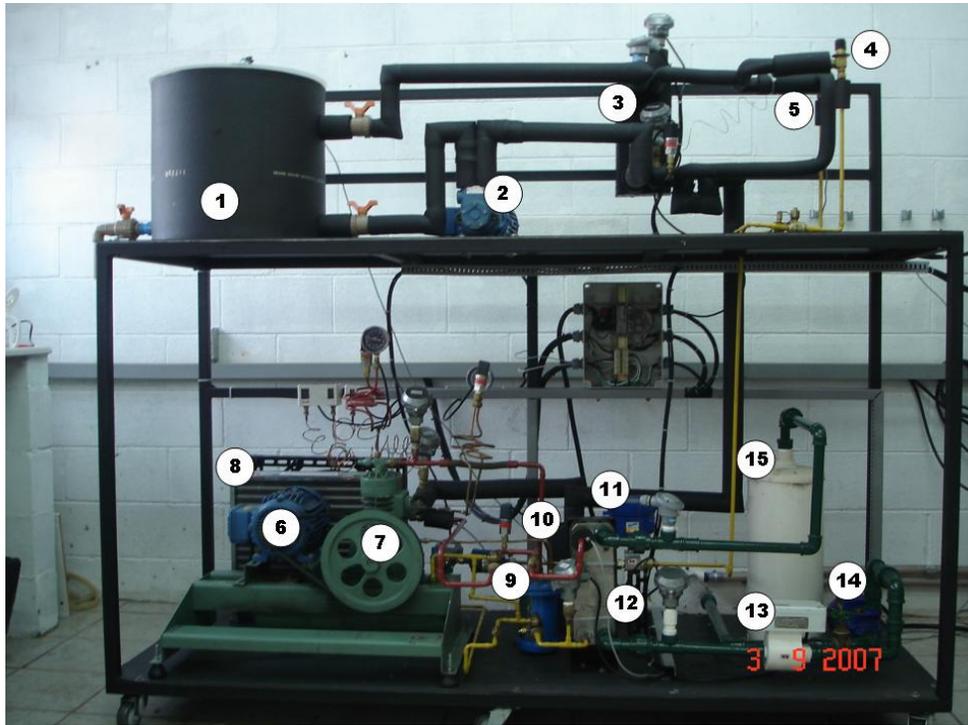


Figura 3. 1 Módulo de refrigeração. [Franco, 2009]

Antes do compressor, tem-se um trocador de calor (evaporador) do tipo placas brasado modelo CB 26M da Alfa Laval.

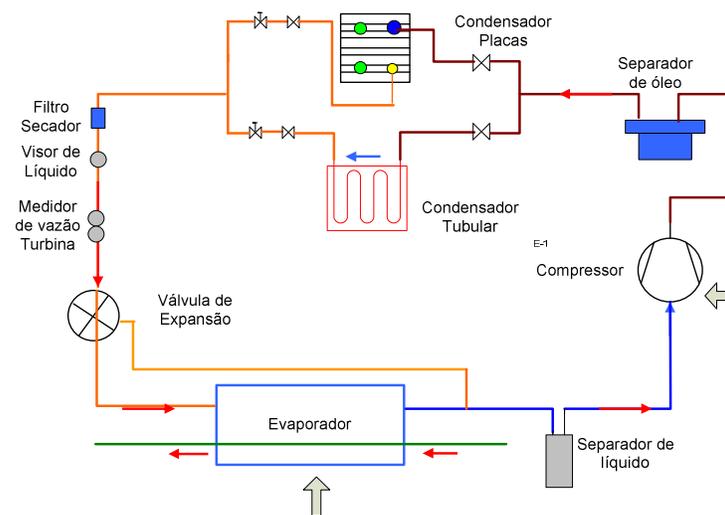


Figura 3. 2 Diagrama do circuito principal. [PINELLI,2008]

O protótipo apresenta ainda uma válvula de expansão termostática e uma válvula de expansão pressostática, em paralelo. Sendo que, a única utilizada neste trabalho foi a termostática.

O diagrama do circuito secundário do protótipo é mostrado na Figura 3.3.

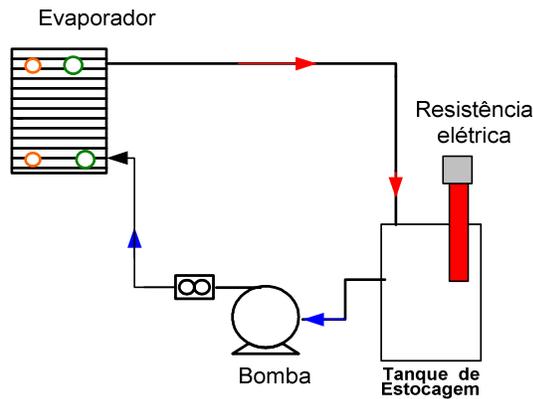


Figura 3. 3 Diagrama do circuito secundário [PINELLI, 2008]

No circuito secundário a solução aquosa de propileno-glicol resfriada no evaporador circula com a ajuda de uma bomba de deslocamento positivo RZR modelo RZR500, acionada por um motor de 0,75 HP de potência. Depois de passar pelo evaporador, o propileno segue para um tanque de aço inoxidável onde recebe uma carga térmica artificial (perturbação externa), proveniente de uma resistência elétrica de imersão de 220 V e 2000 W, que pode ter sua carga variada através de um variador de potência, o que possibilita simulações de variações de carga que ocorrem nos processos reais de refrigeração além de suprir as oscilações elétricas da rede, permitindo repetibilidade aos ensaios.

3.1.3. Instrumentação do protótipo

O sistema de resfriamento de líquido em questão também conta com um vasto sistema de instrumentação, feito com o intuito de se realizar medições de grande número

de variáveis do sistema para que se pudesse fazer uma melhor modelagem e um melhor controle do processo a fim de se obter uma melhor performance.

Todo o sistema de instrumentação da planta de refrigeração utilizada, foi desenvolvida e implementada por PINELLI (2008).

Para as medidas de temperatura no sistema de resfriamento de líquido foram utilizadas termo resistências Pt100 a três fios com cabeçote. Para a medição de vazão foram utilizados dois medidores do tipo turbina da marca Incontrol, como visto na tabela 3.1.

Tabela 3. 1 Descrição dos medidores de vazão utilizados.

MODELO	VAZÃO (m ³ /h)	MEDIÇÃO
VTPL-110A020211RAAC20	entre 1,13 a 10,3	R-22
VTL-009A020211BCC20	entre 0,17 a 1,70	PROPILENO

Ambos os medidores possuem sinal de saída de 4 a 20 mA e foram aferidos pelo fabricante, para as vazões de R-22 e da linha de propileno glicol.

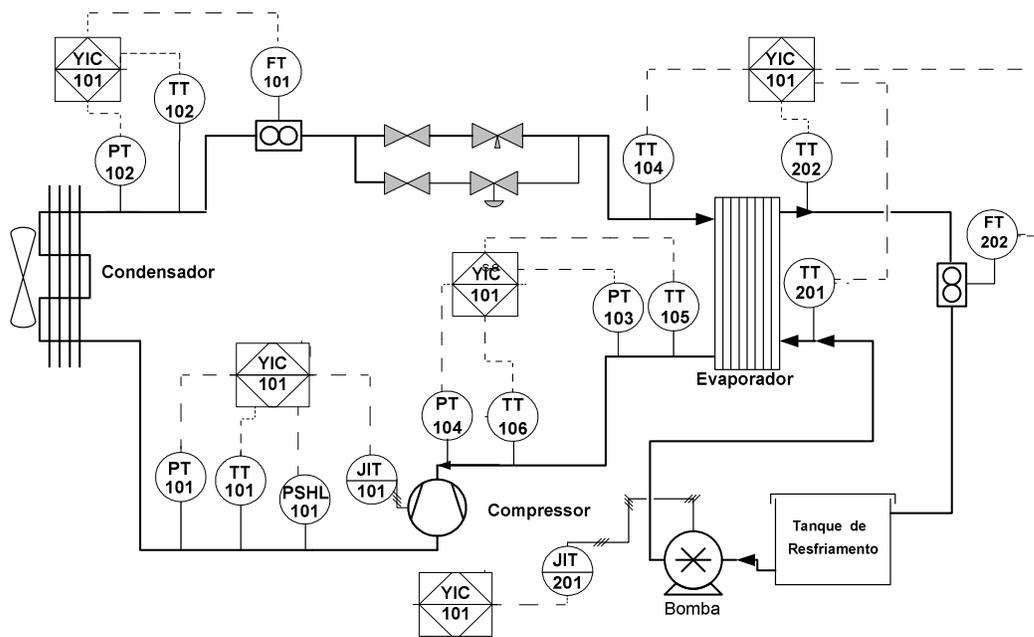


Figura 3. 4 Diagrama de instrumentação do sistema de refrigeração.

E, para as medições das pressões de descarga, condensação, evaporação e sucção, foram utilizados sensores piezo resistivos Danfoss modelos 06011G25 para pressões entre 0 e 10 bar, e 06011G05, para pressões entre 0 e 40 bar, com sinal de saída de 4 a 20 mA e precisão de $\pm 0,3\%$ do fundo de escala, já aferidos pelo fabricante. A descrição dos sensores conforme o diagrama pode ser vista na tabela 3.2, e o fluxograma da instrumentação pode ser visto na Figura 3.4.

Tabela 3. 2 Descrição dos elementos sensores.

SENSOR	Descrição	Nomenclatura Adotada
TT 101	Temperatura de Descarga	TD
TT 102	Temperatura de Condensação	TC
TT 104	Temperatura de Evaporação	TE
TT 105	Temperatura de Saída do Evaporador (R-22)	TSE
TT 106	Temperatura de Sucção	TS
TT 201	Temperatura de Entrada do Evaporador (propileno)	TEP
TT 202	Temperatura de Saída do Evaporador (propileno)	TPG
FT 101	Vazão de R-22	
FT 201	Vazão de Propileno	
PT 101	Pressão de Descarga	
PT 102	Pressão de Condensação	
PT 103	Pressão de Evaporação	
PT 104	Pressão de Sucção	
PSHL	Dispositivo de Alarme para pressões alta e baixa	
JIT 101	Inversor de Frequência para Compressor	
JIT 201	Inversor de Frequência para Bomba de Propileno	

A metodologia empregada neste trabalho buscou variar dois parâmetros do sistema de refrigeração. Um referente à variação da frequência de trabalho do compressor, e outro relacionado à frequência de rotação da bomba de solução de propileno-glicol. Para a modulação da velocidade de rotação do motor do compressor foi utilizado um variador de frequência Danfoss VLT 2815, para através da variação da frequência controlar-se a capacidade de refrigeração do sistema. E com a finalidade de se

realizar variações na vazão da linha de propileno foi utilizado um inversor de frequência Danfoss VLT 2822.

3.1.4. Sistema de controle digital

O CLP utilizado no trabalho é de fabricação da empresa HI Tecnologia, modelo MCI02-QC (Figura 3.5)



Figura 3. 5 Controlador Lógico Programável (CLP) MCI02-QC (HI Tecnologia) utilizado no projeto.

A disposição modular do MCI02-QC confere bastante flexibilidade às aplicações. Possui a capacidade de gerenciar pontos discretos e variáveis analógicas, incluindo blocos funcionais de controle. A arquitetura do MCI02-QC é formada por racks com quatro módulos, podendo ter no máximo 15 racks.

O CLP do protótipo possui a finalidade de gerenciar funções de intertravamentos, acionar motores e leitura de temperatura das termoresistências, além da possibilidade de efetuar controle nas variáveis do processo.

O programa SPDSW é o configurador em ambiente Windows para o MCI02-QC que possui a finalidade de definir o hardware para o sistema, criar lógicas de controle através de diagrama LADDER, criar blocos de função, testar as aplicações e otimizá-las.

Como o diagrama LADDER do CLP não possibilita a implementação de controladores PID com técnicas de anti-windup e derivative-kick, foi necessário o emprego de outro software que possibilitasse o desenvolvimento deste tipo de PID. Para isso baseado em uma interface de supervisão criada no software MATLAB/Simulink no trabalho de FRANCO (2009), foi possível desenvolver a estratégia de controle pretendida (detalhada na seção 2.3).

O MATLAB possui inúmeras “*toolboxes*” com algoritmos pré-programados para conhecimentos específicos. Na comunicação MATLAB/SIMULINK-Sistema de Refrigeração para que o controlador desenvolvido em MATLAB atue no sistema de controle foi utilizado a “*OPC toolbox*”. A “*OPC toolbox*” é uma coleção de funções que estendem a potencialidade do MATLAB podendo adquirir dados do “*Server OPC*” e escrever dados diretamente ao “*Client OPC*”. Uma vez estabelecida a conexão entre o cliente e o servidor OPC pode-se monitorar e adicionar valores ao servidor OPC (FRANCO, 2009).

A obtenção de dados em tempo real proveniente de sensores (temperatura, pressão, vazão, etc), comandos de controle (abrir, fechar, ligar, desligar, etc), status de comunicação, dados de desempenho, estatística do sistema e as ações do controlador são realizadas através do padrão OPC.

OPC é um acrônimo para “*OLE for Process Control*” ou “OLE para Controle de Processos”. Baseado nas tecnologias Microsoft OLE COM (*Component Object Model*) e DCOM (*Distributed Component Object Model*), o OPC é um conjunto comum de interfaces, métodos e propriedades de comunicação, agregados dentro de uma especificação padronizada e aberta para acesso público.

O OPC é um padrão de comunicação de dados onde várias informações localizadas em diversos pontos (controladores, dispositivos de E/S, banco de dados, etc) podem ser conectadas diretamente a um “*Cliente*”, o qual pode executar funções de IHM (Interface Homem-Máquina), supervisão, tratamento gráfico, etc. Para realizar essa comunicação, o OPC utiliza-se de um protocolo universal para envio de dados entre “*Servers*” (fontes de dados) e “*Clients*” (aplicações), eliminando dentro do possível o uso de “*drivers*” e conversores, na maioria das vezes proprietários. Esses “*drivers*” proprietários criam uma

dependência, muitas vezes custosa e ineficiente, ente o usuário e o fornecedor daquele sistema proprietário. Quando se utiliza de um padrão OPC é necessário somente um “driver” padrão que se torna o Servidor (*Server*). Os dados são convertidos para uma forma que qualquer Cliente (*Client*) que tenha uma disposição OPC possa se conectar sem qualquer tipo de preparo prévio. Resumindo, o OPC proporciona o verdadeiro *plug-and-play* no campo de supervisão dentro do contexto da moderna automação industrial (FRANCO, 2009).

3.1.5. Procedimento inicial de operação

Todos os ensaios realizados, para a obtenção das curvas de reação para o teste e sintonia fina dos controladores como também para coleta de dados e teste das redes neurais, foram realizados da mesma maneira, para que as condições sempre fossem, na medida do possível, idênticas de um ensaio para o outro. A forma como eram realizados os ensaios está descrita a seguir.

Inicialmente ligava-se a planta de refrigeração (compressor, bomba de propileno glicol e ventilador do condensador a ar), quando a temperatura de evaporação atingia -10 °C, se acionava a resistência elétrica a uma potência de 1800 W e esperava-se o sistema entrar em regime permanente, ou seja, todas as temperaturas monitoradas chegassem ao estado estacionário.

Optou-se por usar este tipo de ensaio para que as condições fossem parecidas com as obtidas em trabalho realizado por PINELLI (2008) na mesma planta de refrigeração para que pudessem ser feitas comparações.

A temperatura do líquido de resfriamento foi submetida a uma carga artificial através do acionamento de uma resistência monofásica de 2000 W instalada dentro do tanque de propileno glicol.

3.2. Desenvolvimento do controlador PID

O controlador PID é a estratégia de controle mais genérica e provavelmente uma das mais utilizadas. Fornece resposta rápida, bom controle de estabilidade do sistema e baixo erro em regime permanente para sistemas contínuos e lineares.

Para o sistema de refrigeração em questão foram configuradas duas malhas de controle, onde a variável controlada é a temperatura de saída do propileno glicol do evaporador (TPG), e as variáveis manipuladas são as frequências do compressor (FC) e da bomba de propileno glicol (FB).

A fim de se realizar as ações de controle, funções devem ser implementadas eletronicamente, através de um controle discreto ou analógico, baseado em modelos matemáticos, que se iniciam com o sinal do erro, calculado através da equação 3.1, onde é calculada a diferença entre a temperatura de saída do propileno (T_{PG}) e a temperatura de referência (T_{SP}) que corresponde ao *set-point*.

$$e(t) = T_{SP}(t) - T_{PG}(t) \quad (3.1)$$

O funcionamento consistiu em se estabelecer um nível desejado de temperatura para o líquido controlado (*set-point*). No estudo foi considerada a temperatura de *set-point* inicialmente 4 °C, por ser a temperatura em regime permanente para TPG, e depois aplicados degraus para 5 °C, 6 °C e então novamente para 4°C para abranger condições diferentes de controle, além de serem feitas perturbações na carga térmica onde a carga era perturbada de 1800 W para 1000 W. O objetivo da ação de controle era de variar as rotações do compressor e da bomba de evaporação a fim de se fazer com que a temperatura de *set-point* fosse atingida o mais rapidamente possível e evitar oscilações indesejáveis na temperatura do propileno após o seu *set-point* ser atingido. A temperatura do propileno glicol possui uma grande influência quanto ao desempenho geral do sistema frigorífico, sendo altamente influenciada por perturbações externas. O controle desta temperatura está diretamente ligado à qualidade do produto a ser resfriado, por essas razões essa temperatura foi definida como a variável a ser controlada.

As estratégias de controle foram implementadas através do software MATLAB/Simulink, devido a facilidade de sua implementação e da implementação de técnicas como *reset windup* e *derivative kick* (SEBORG, 2002) para as malhas de controle, Figura 2.6.

Para a aplicação das estratégias de controle, os controladores foram previamente configurados de forma que se obtivessem duas malhas SISO/feedback, em paralelo, das variáveis selecionadas, que foram implementadas e sintonizadas no protótipo experimental para a avaliação de desempenho.

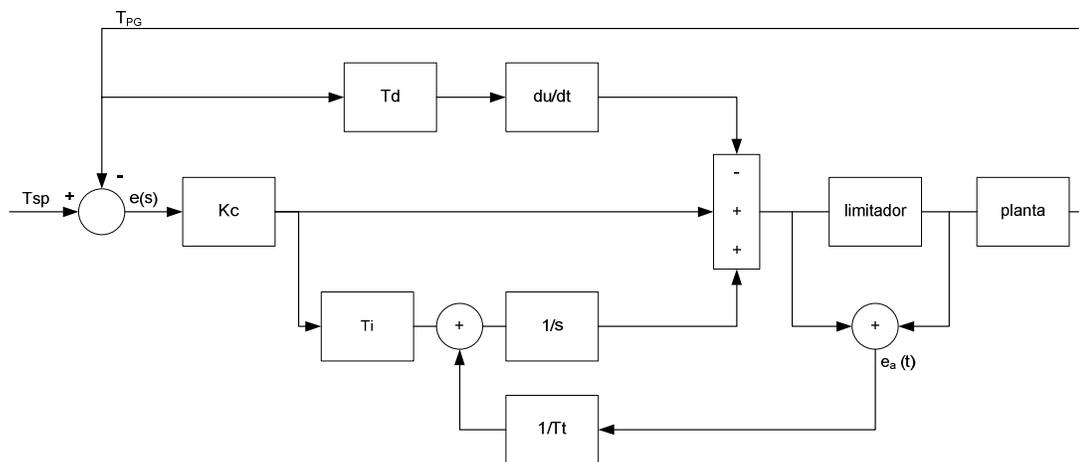


Figura 3. 6 Malha de controle implementada no SIMULINK/MATLAB para as duas estratégias de PID.

Neste tipo de controle, a estratégia consiste em aplicar na planta um sinal de excitação proporcional ao erro, mais a função integral deste, que é somada ao *anti-windup*, menos sua função derivativa que é aplicada à realimentação. Desta maneira, a equação de controle resulta na equação 3.2 (PAZOS, 2002).

$$p(t) = \left[Kc \cdot e(t) + \left(\frac{1}{\tau_t} \cdot e_s(t) + \frac{Kc}{\tau_i} \int e(t) \cdot dt \right) - Kc \cdot \tau_d \cdot \frac{du(t)}{dt} \right] \quad (3.2)$$

O sistema de *anti-windup* consiste em se impor condições limite de operação ao sistema, e quando a saída do atuador saturar, o termo integral é novamente calculado de forma que seu valor permaneça no limite do atuador. É vantajoso fazer esta correção não instantaneamente, mas dinamicamente com uma constante de tempo Tt.

O sistema apresenta uma malha de realimentação adicional. A diferença entre a entrada e saída do atuador constitui um erro e_s que é realimentado à entrada do integrador com ganho $1/T_t$. Note que quando não existe saturação o erro e_s é nulo e, portanto, a malha não tem nenhum efeito quando o controlador está operando linearmente, ou seja, quando a saída $p(t)$ não está saturada. Se existe a saturação e_s é diferente de zero. O tempo para que a entrada do integrador chegue a zero é determinado pelo ganho $1/T_t$, onde T_t pode ser interpretado como a constante de tempo que determina o quão rápido a entrada do integrador é levada a zero. Assim, a seleção de valores pequenos para T_t pode parecer vantajosa à primeira vista. Entretanto, deve-se ter cuidado na seleção de T_t especialmente em sistemas com ação derivativa. O que pode acontecer é que ruídos espúrios podem levar a saída do controlador à saturação provocando a atuação rápida da malha anti-windup e tornando a entrada do integrador indesejavelmente zero. Na prática deve-se ter T_t maior que T_d e menor que T_i . Uma regra empírica sugerida é selecionar de acordo com $T_t = \sqrt{T_i * T_d}$.

Restrições de operação foram impostas tanto ao compressor quanto a bomba de evaporação. Para o compressor houve a necessidade de não trabalhar com freqüências abaixo de 25 Hz no inversor de freqüência, para evitar problemas de lubrificação no compressor, e acima de 70 Hz por razões de consumo de energia e desgaste prematuro da máquina. E, para a bomba de evaporação foram impostas condições de mínimo 15 Hz para evitar o aquecimento do motor, já que a baixas rotações a ventilação do mesmo fica comprometida, e máximo de 70 Hz pelas mesmas razões do compressor.

E o *derivative kick*, segundo Seborg (1989), é um fenômeno que acontece quando há uma variação brusca no valor de referência. Ou seja, como inicialmente a diferença entre o Erro e a referência passa de zero até ao seu valor máximo, o termo derivativo vai conseqüentemente ser muito elevado, provocando esse efeito indesejado. Assim, a solução é colocar a ação derivativa no anel de realimentação passando a denominar-se PI-D. A variação dos seus parâmetros faz com que a resposta do sistema varie da seguinte maneira: o aumento do parâmetro T_d origina uma resposta mais lenta, enquanto que o aumento de K_c diminui o Erro Estacionário.

3.3. Sintonia dos controladores PID

Como a sintonia adequada dos parâmetros dos controladores é de extrema importância, foram realizados vários ensaios e várias sintonias esperando-se chegar aos melhores resultados de controle.

Para a sintonia inicial do controlador PID, foi aplicado o Método de Ziegler-Nichols 1, descrito pela Equação 3.3 à Equação 3.5 (SEBORG, EDGAR e MELLICHAMP, 1989).

Para a realização da sintonia das malhas de controle do sistema de refrigeração, foi utilizado o método da curva de reação para determinação de um modelo FOPDT (do inglês, primeira ordem com tempo morto) e aplicação de Ziegler e Nichols, descrito pelas equações 3.3, 3.4 e 3.5 (SEBORG, EDGAR e MELLICHAMP, 1989). Método este, escolhido devido à facilidade de execução dos ensaios e pelos resultados satisfatórios.

$$K_c = 1,2 \cdot \frac{\tau}{K \cdot t_d} \quad (3.3)$$

$$\tau_i = 2 * t_d \quad (3.4)$$

$$\tau_d = 0.5 * t_d \quad (3.5)$$

Para a aplicação deste método foram realizados ensaios em malha aberta, com perturbações em degrau, de 25% na variável manipulada para o compressor e 20% para a bomba de propileno glicol, com observação da variável controlada, com a finalidade de se obter os parâmetros do processo (K_p , t_d , T).

Os parâmetros do processo, ganho (K_p), tempo morto (t_d) e a constante de tempo (τ), foram determinados através da Figura 3.7 que representa o modelo de curva de reação:

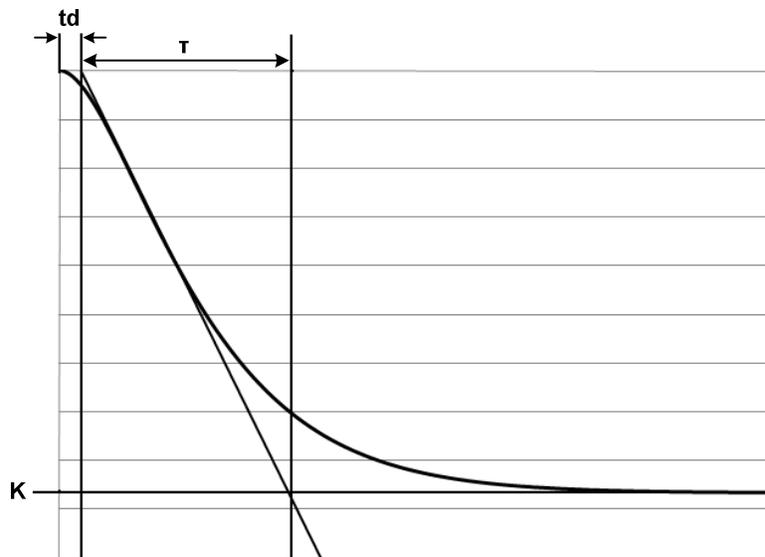


Figura 3. 7 Curva de reação para a obtenção dos parâmetros de Ziegler e Nichols.

O ganho K_p foi obtido através da equação 3.6

$$K_p = K/A \quad (3.6)$$

Onde K é o valor do novo estado estacionário e A é a amplitude do degrau dado a perturbação.

Após serem obtidos os parâmetros do processo através do método descrito, e a implementação dos controladores, procedeu-se a sintonia fina dos controladores através de métodos heurísticos e análise dos seguintes critérios: overshoot e tempo de subida.

3.4. Redes Neurais

Como foi possível observar na revisão apresentada anteriormente, a modelagem empírica baseada em redes neurais artificiais (RNA's) vem sendo bastante utilizada em processos de refrigeração, principalmente para a identificação dos sistemas ou para o emprego de técnicas de controle preditivo. Neste contexto, os modelos baseados em RNA's apresentam-se como uma poderosa ferramenta na modelagem de sistemas estáticos e dinâmicos, com grandes não-linearidades e tempo morto elevado,

principalmente devido a duas fundamentais qualidades: a rapidez de cálculos e sua inerente capacidade de aproximação.

Neste trabalho, como modelo empírico para o processo de refrigeração de líquido secundário (solução aquosa de propileno-glicol à 50%), foi utilizada uma rede neural do tipo *feedforward*. Este tipo de rede está entre as mais estudadas em processos de engenharia, principalmente por ser de fácil compreensão e implementação, com uma simples definição matemática e com grande capacidade para aplicações matemáticas em identificação/modelagem e controle de processos. (NARENDRA e PARTASARATY, 1990; FILETI, 1995; FUJIKI, 2009)

3.4.1. Desenvolvimento do modelo neural

Para uma RNA realizar de forma satisfatória seu trabalho, suas entradas devem conter informações que descrevam as principais características do problema a ser resolvido. A quantidade de informação (número de entradas) é uma questão a ser observada pelo especialista, e deve atender a dois critérios principais:

1. Se o número de informações for pequeno em relação à complexidade do problema, a rede dificilmente conseguirá resultados satisfatórios;
2. Se o número de informações for muito grande, a rede poderá ser muito complexa (grande número de conexões), o que pode torná-la instável, ou ainda, provocar overfitting;

As redes neurais utilizadas neste trabalho foram desenvolvidas com a utilização do software MATLAB v. 7.0.1 (*The Math Works Inc.*), que contém funções específicas e pacotes matemáticos para elaboração de redes neurais (*Toolbox Neural Networks*).

Diversas estruturas foram testadas para obter a configuração que fornecesse os melhores resultados ao tipo de sinal empregado.

Todas as redes criadas possuem três ou quatro camadas sendo, uma camada de entrada, uma ou duas camadas ocultas e uma camada de saída. Os neurônios das camadas ocultas variam e possuem como funções de ativação a tangente hiperbólica Eq. 3.4). E como função de ativação da camada de saída foi utilizada a função linear (Eq 3.5).

Com relação ao número de neurônios nas camadas ocultas, este foi definido empiricamente e dependeu de diversos fatores, como: números de exemplos de treinamento, complexidade da função a ser aprendida, quantidade de ruído presente nos exemplos. Uma limitação do número de conexões empregadas reduz a incidência de overfitting, ou seja, evita que a rede memorize os padrões de treinamento, em vez de extrair as características que permitirão o seu poder de generalização. Por outro lado, pode ocorrer underfitting, ou seja, a rede não converge durante o seu treinamento e/ou apresenta uma baixa capacidade de generalização.

Para a determinação dos parâmetros das redes neurais foram construídas várias redes com configurações diferentes. Para cada estrutura foram geradas redes com parâmetros distintos, visando a determinação da configuração que apresentasse a melhor generalização do problema. Os principais parâmetros variados foram: número de entradas da rede; número de camadas intermediárias e o número de neurônios das camadas intermediárias;

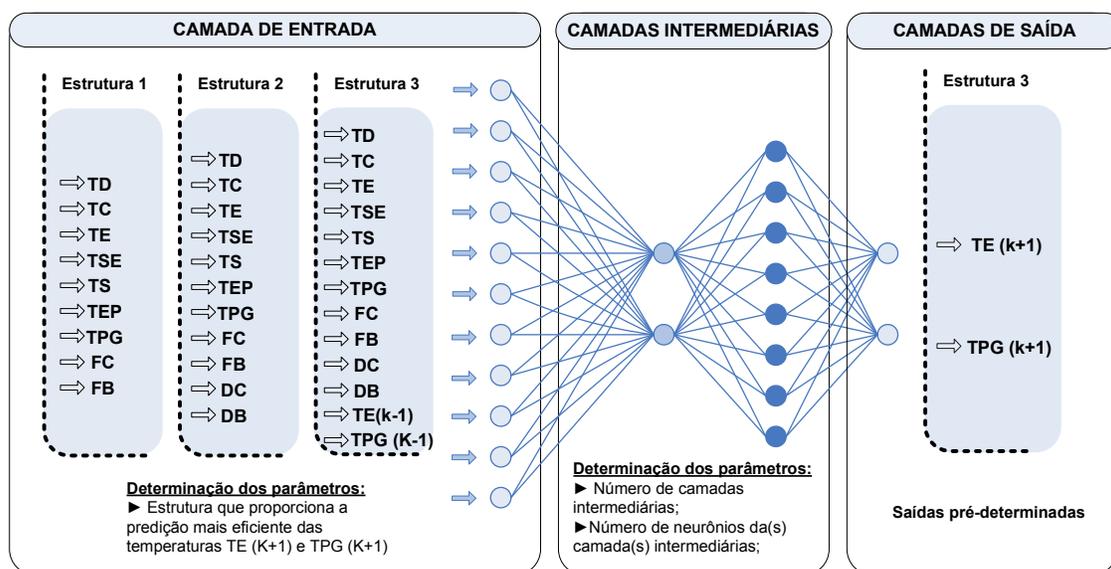


Figura 3. 8 Síntese dos ensaios realizados para definição da arquitetura da redes neurais.

A Figura 3.8 apresenta uma síntese dos ensaios elaborados na busca da arquitetura da rede que apresentasse o melhor desempenho na predição das temperaturas de evaporação (TE) e do propileno-glicol (TPG). A nomenclatura das variáveis está em conformidade com a Tabela do item 3.1.3

3.4.2. Coleta de dados e normalização

A construção do banco de dados foi feita através de ensaios com o sistema de refrigeração funcionando em malha aberta, com intervalo de amostragem de 10 s. Estes ensaios ocorreram da seguinte forma, foram feitos quatro ensaios para a coleta dos dados, sendo dois em que a frequência de rotação da bomba de propileno-glicol foi mantida fixa e a frequência de rotação do motor do compressor foi alterada em forma de degraus de 10 em 10 Hz na faixa de 30 Hz até 70 Hz (faixa esta escolhida por condições de consumo energético e vida útil do conjunto compressor/motor). Primeiramente os degraus foram aplicados na forma ascendente e depois descendente, sempre após o sistema entrar em regime permanente. Nos outros dois ensaios, a frequência de rotação do motor do compressor foi mantida fixa, e a frequência da bomba de propileno-glicol foi alterada em forma de degraus também de 10 em 10 Hz na faixa de 20 à 70 Hz, sendo feitos os ensaios da mesma forma como para o compressor.

Também para compor o banco de dados foi desenvolvido um controlador PID, no qual foram realizados vários ensaios, que mais tarde foram utilizados tanto para o treinamento como para o teste da rede neural.

Para que os dados coletados pudessem ser utilizados no treinamento, teste e validação das redes neurais, todos foram inicialmente normalizados para o intervalo [-1; 1], e depois, para a visualização dos resultados eram desnormalizados para este mesmo intervalo. Este intervalo foi escolhido, dado o emprego da tangente hiperbólica como função de ativação, por ser o intervalo em que a função tem seus valores mais significativos, contribuindo melhor para o treinamento.

3.4.3. Treinamento da RNA

Como foram obtidos valores de saídas para cada condição operacional, utilizou-se o treinamento do tipo supervisionado e, como algoritmo de treinamento, o de Levenberg-Marquardt, por apresentar, dentre todas as variações do algoritmo *backpropagation*, o menor tempo de treinamento.

As redes neurais foram criadas a partir da técnica treinar-e-testar, e neste caso o banco de dados foi dividido em dois grupos:

- O conjunto de treinamento utilizado no ajuste dos pesos e *bias*;
- O conjunto de testes.

Todas as redes utilizadas neste trabalho possuem a arquitetura *feedforward*, e foram treinadas utilizando o software MATLAB v. 7.0.1 (*The Math Works Inc.*) tendo como método de aprendizagem o algoritmo de Levenberg-Marquardt, descrito no item 2.3.8.

3.4.4. Validação online do modelo neural

Após a rede ser treinada, foi avaliada com testes *offline* com dados resultantes do controlador PID, em ensaios diferentes dos utilizados no treinamento. E, somente os modelos que apresentaram resultados satisfatórios foram validados online no sistema.

Para esta validação online foi criada no do software MATLAB v. 7.0.1 (*The Math Works Inc.*) através da ferramenta Simulink, uma interface para reproduzir a rede neural, onde os dados de entrada eram lidos através de interface de comunicação OPC (*OLE for Process Control*) e normalizados através da função *mapminmax*, de acordo com a equação 3.7:

$$pn = 2 * \frac{(p-pmin)}{(pmax-pmin)} - 1 \quad (3.7)$$

Onde:

pn= valor normalizado da variável p;

p = valor real da variável p ;

p_{min} e p_{max} = são respectivamente os valores mínimo e máximo de p ;

Para fazer uso da rede neural ,gerada a partir do treinamento, no Simulink, foi utilizado o comando *gensim (net)* do MATLAB, que gerou um bloco onde a rede neural é simulada. A função de ativação aplicada em cada uma das camadas intermediárias foi a tangente hiperbólica (terminologia do MATLAB: *tansig*); sendo assim, o sinal proveniente da camada de entrada é recebido pela 1ª camada intermediária, e a saída da 1ª camada intermediária é recebido pela 2ª camada intermediária, estes sinais são ponderados pelos pesos e bias, sendo processados segundo a Equação 3.8:

$$a = \text{tansig}(n)$$
$$a = \frac{2}{(1 + \exp(-2*n)) - 1} \quad (3.8)$$

Então, a camada de saída processa o sinal dos neurônios da 2ª camada intermediária segundo a função linear (terminologia do MATLAB: *purelin*) vista na Equação 3.9:

$$\text{purelin}(n) = a \quad (3.9)$$

Depois de a rede neural fazer os cálculos a saída da rede deve ser desnormalizada. Esta desnormalização é agora feita através da função *mapminmax* (reverse), conforme a equação 3.10:

$$p = 0.5 * (pn + 1) * (p_{max} - p_{min}) + p_{min} \quad (3.10)$$

Para a realização dos ensaios online, foi utilizado o controlador PID descrito na seção 3.2 em paralelo com a rede neural, com o objetivo de testar a capacidade de predição do modelo em malha fechada.

4. RESULTADOS OBTIDOS

4.1. Análise de não-linearidade do sistema

Após a realização dos ensaios citados no item 2.4.2, foram obtidas as curvas de reação, apresentadas pelas variáveis do processo, para cada uma das perturbações aplicadas ao sistema para verificação da não-linearidade do sistema. Para os ensaios em que a rotação do motor do compressor (FC) recebia as perturbações, foram obtidas curvas de reação a partir da temperatura de evaporação (TE) e da temperatura de saída do propileno-glicol (TPG), mantendo-se a rotação da bomba constante em 40 Hz. Quando a frequência de rotação da bomba de propileno-glicol era perturbada foram obtidas as curvas de reação da temperatura de saída do propileno-glicol (TPG), mantendo-se a rotação do compressor constante em 50 Hz. A carga na potência de aquecimento do propileno-glicol foi mantida sempre em 1800 W.

Utilizando as curvas de reação ajustadas, citadas anteriormente foram obtidos os ganhos do processo conforme mostram as Tabelas 4.1 a e 4.2, sendo a primeira para as perturbações realizadas na bomba de propileno e a segunda para o motor do compressor.

Tabela 4. 1 Parâmetros de gráficos, do ganho (K), obtidos dos ensaios com perturbações da rotação da bomba em diversas amplitudes para temperatura de saída do propileno-glicol (TPG).

Ponto Operacional	Ganhos para perturbações de 10Hz (°C/Hz)	
	Negativas	Positivas
Freq. de rotação da bomba (Hz)		
20	-	0,40
30	0,25	0,14
40	0,14	0
50	0	0
60	0	0,06
70	0,03	-

Tabela 4. 2 Parâmetros de gráficos, do ganho (K), obtidos dos ensaios com perturbações na rotação do compressor em diversas amplitudes para temperatura de saída do propileno-glicol (TPG).

Ponto operacional	Ganhos para perturbações de 10Hz (°C/Hz)	
	Negativas	Positivas
30	-	-0,70
40	-0,71	-0,27
50	-0,31	0,10
60	-0,09	0,05
70	0	-

Tabela 4. 3 Parâmetros de gráficos, do ganho (K), obtidos dos ensaios com perturbações da rotação do compressor em diversas amplitudes para temperatura de evaporação (TE).

Ponto Operacional	Ganhos para perturbações de 10Hz (°C/Hz)	
	Negativas	Positivas
30	-	-0,63
40	-0,64	-0,45
50	-0,51	-0,47
60	-0,47	-0,34
70	-0,4	-

Analisando-se as tabelas acima, verifica-se que os ganhos obtidos para perturbações negativas na rotação do motor do compressor foram sempre maiores do que os ganhos obtidos para as perturbações positivas. Isto significa que a redução da rotação do compressor é mais impactante sobre a temperatura de evaporação do que o aumento da mesma. Tomando-se por base o ponto operacional de 40 Hz (na rotação do compressor), uma variação positiva de 30 Hz causaria uma modificação de 13,5 °C na

temperatura e, no caso de uma variação negativa de 30 Hz, a alteração causada na temperatura de evaporação seria de 19,2 °C. Além disso, verifica-se também pelas tabelas que para cada ponto operacional do compressor obtém-se um valor diferente de ganho.

Este comportamento é típico de processos não-lineares e não é compatível com o projeto de controladores PID convencionais, que operam com o ganho do controlador constante. Sabe-se que a sintonia do ganho do controlador se baseia no valor do ganho do processo (eq. 4.7), portanto, não há um valor constante apropriado para o PID em toda a faixa de operação do processo. Isto justifica a procura por modelos não-lineares para serem usados internamente em esquema de controle preditivo multivariável.

4.2. Implementação das malhas de controle PID

Todas as malhas de controle do sistema de refrigeração foram sintonizadas inicialmente através do método original de Ziegler e Nichols 1. Para a aplicação do método foi obtida a curva de reação do processo para as duas malhas, sendo que para a malha de propileno foi aplicado um degrau na bomba de propileno e, para a malha de R-22 foi aplicado um degrau no compressor. Os parâmetros encontrados foram substituídos nas equações do método Ziegler e Nichols (descrito na seção anterior), e assim obtidos os valores da Tabela 4.4.

Onde fala-se de malha de propileno refere-se a malha onde a bomba de propileno-glicol atua sobre a temperatura de saída do propileno. E, onde fala-se de malha de R-22 refere-se a malha onde o compressor atua sobre a temperatura de saída do propileno.

Tabela 4. 4 Valores ajustados para PID

	Kc (Hz/°C)	τ_i (s)	τ_d (s)
Malha de Propileno	245,31	35,19	5,34
Malha de R-22	47,78	108,3	16,4

Como se percebeu durante os ensaios que o controlador ficou instável com os valores obtidos inicialmente, optou-se por utilizar inicialmente valores dos parâmetros que correspondessem a cerca de 10% dos valores obtidos inicialmente. Assim apresenta-se na Tabela 4.5 os parâmetros que foram utilizados para os ensaios.

Tabela 4. 5 Valores ajustados PID

	Kc (Hz/°C)	τ_i (s)	τ_d(s)
Malha de Propileno	24,53	3,59	0,53
Malha de R-22	4,78	10,83	1,64

Após os resultados do método de Ziegler e Nichols terem se mostrado insatisfatórios, optou-se por um posterior ajuste fino dos parâmetros. E, após varias tentativas de tentar melhorar a curva de resposta do controladores, chegou-se aos resultados mostrados na figura 4.1, e os ganhos mostrados na Tabela 4.6.

Tabela 4. 6 Parâmetros definitivos para os controladores PID.

	Kc (Hz/°C)	τ_i (s)	τ_d(s)
Malha de Propileno	0,7224	95,24061	0,2
Malha de R-22	0,504	9,90099	0,4

4.3. Ensaios com o controlador convencional

Abaixo nas Figuras de 4.1 à 4.3 estão ilustrados os comportamentos das variáveis controlada e manipulada sob a ação do controlador PID utilizando a estratégia de controle multimalha para perturbações de $\pm 30\%$ positiva e negativa na carga térmica do sistema de refrigeração e também para várias perturbações aplicadas no *set-point* do controlador.

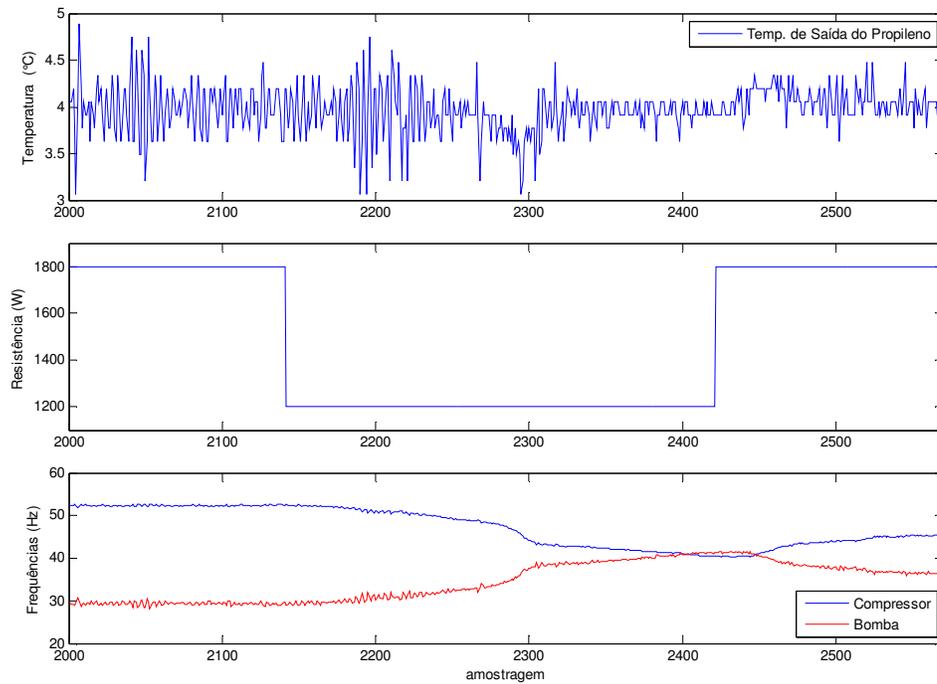


Figura 4. 1 Comportamento do desvio das variáveis controlada e manipulada frente a perturbações na carga aplicada ao sistema de refrigeração.

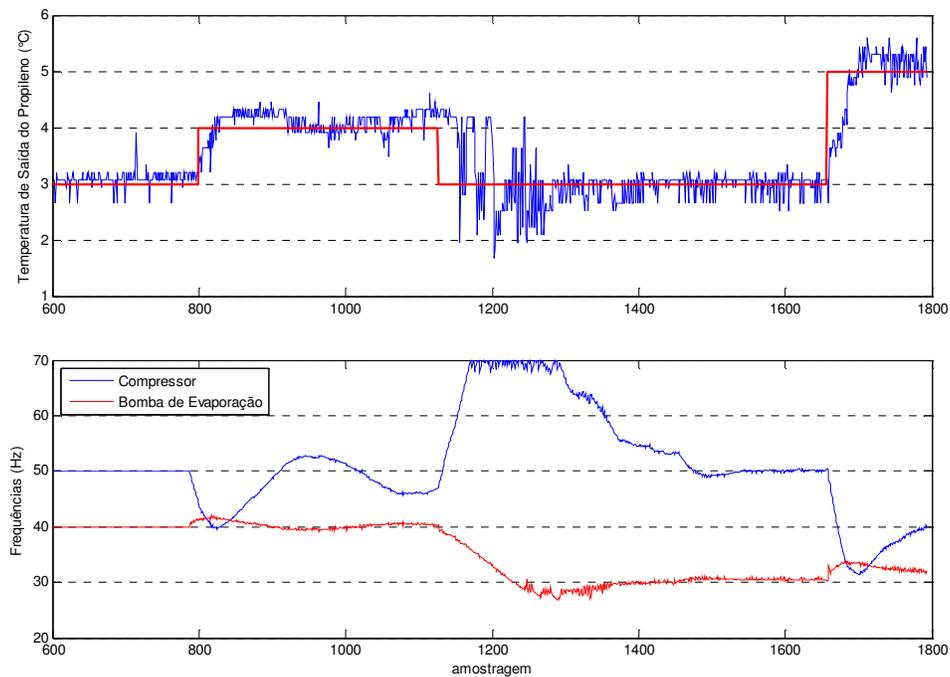


Figura 4. 2 Comportamento dos desvios da variável controlada e manipulada, sob ação do controlador PID, frente à perturbações aplicadas no set-point

Quanto a perturbações na carga, pode ser observado que as variáveis manipuladas demoram um pouco a atuar, mas isso devido à resistência elétrica, na qual é aplicada a perturbação, estar instalada dentro do tanque de propileno e demora um pouco até que haja uma diferença na temperatura. Nota-se que as variáveis de controle atuam de forma a minimizar as variações na variável controlada.

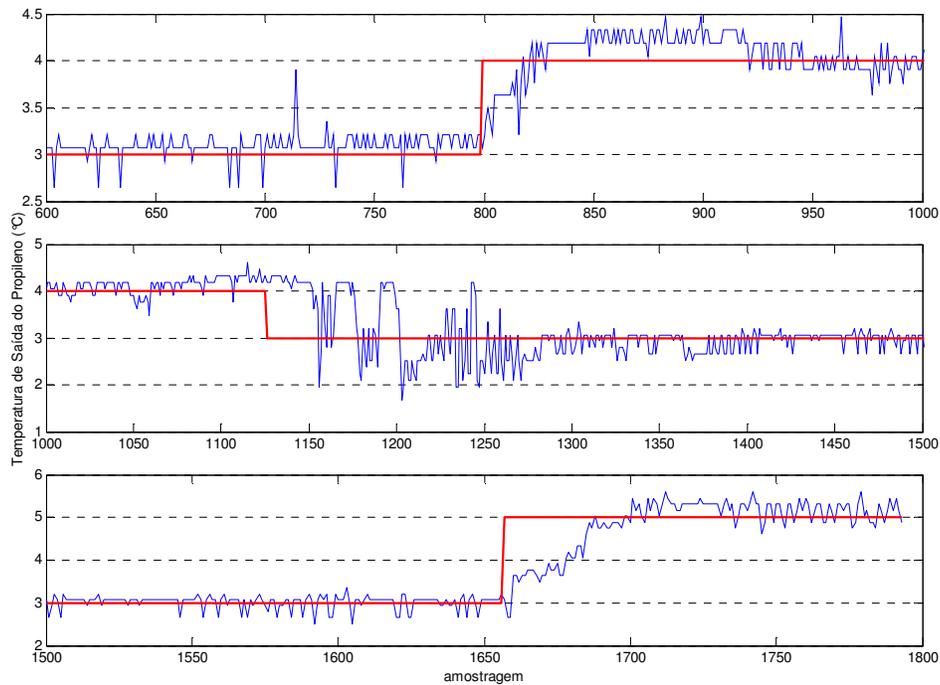


Figura 4. 3 Gráfico ampliado do comportamento da variável manipulada frente a perturbação no set-point

Como se pode observar principalmente na Figura 4.3, o controlador apresenta um tempo de atraso e um aumento na oscilação, principalmente no degrau negativo aplicado ao set-point, o que acontece devido aos baixos valores utilizados para K_p , K_i e K_d (Tabela 4.6), o que foi necessário devido ao controlador não ser capaz de estabilizar o sistema quando usados valores altos para estes parâmetros.

Tabela 4. 7 Respostas do controlador frente a perturbações.

Parâmetros	Controlador Multimalha			
	Set - Point		Carga	
	Negativo	Positivo	Negativo	Positivo
Ensaio	1		2	
Overshoot (°C)	1,0	0,3	0,8	0,3
Tempo de subida (amostras)	40	40	0	0

Como pode ser observado, o controlador com os parâmetros:

Tabela 4. 8 Melhores resultados obtidos em testes.

	Kp (Hz/s)	Ki (s)	Kd (s)
Compressor	0,504	0,050904	0,2016
Bomba	0,7224	0,007585	0,14448

apresentou resultados satisfatórios, vistos através do parâmetros da Tabela 4.7 e das figuras acima, já que se esperava um controle em torno de 1°C para a temperatura de saída do propileno, uma vez que sabia-se através do item 3.1 que o PID não era adequado para o processo em questão.

Os melhores resultados possíveis apresentadas pelo controle multimalha são ratificadas pelos valores apresentados na Tabela 4.8, principalmente na perturbação da carga.

4.4. Redes Neurais

4.4.1. Modelagem do processo utilizando redes neurais

Nesta seção serão apresentados os resultados de treinamento e testes obtidos pelo modelo de predição neural aplicado ao problema de predição das temperaturas de

evaporação e de saída do propileno. Com relação à representação da dinâmica destes problemas pela rede neural, dois diferentes tipos de entradas foram utilizados: entradas atrasadas no tempo, e o delta das frequências tanto da bomba quanto do compressor.

Quando se propõe trabalhar com redes neurais artificiais, o primeiro passo deve ser a definição das variáveis de entrada da rede, a etapa de treinamento e definição da arquitetura da rede e ajuste dos pesos e finalmente a etapa de validação online da rede neural.

4.4.2. Treinamento das redes neurais

A etapa de treinamento consiste no ajuste da arquitetura e dos pesos da rede neural de modo a se representar da melhor forma o sistema que se deseja modelar, portanto, é uma das etapas mais importantes de sua implementação. O algoritmo a ser utilizado depende do tipo de rede que se está utilizando.

Esta etapa envolveu diversas tentativas, isto é, diferentes combinações de variáveis de entrada foram utilizadas, de forma que o treinamento fosse suficiente para a rede responder adequadamente. Observa-se que as diferentes arquiteturas testadas influenciam muito pouco no erro durante o treinamento da rede, porém há grande diferença nos testes de validação.

Aqui serão apresentadas as arquiteturas de seis redes neurais de todas as dezenas de tentativas obtidas durante a fase de treinamento, com o objetivo mostrar um pouco do trabalho feito até se chegar a resultados satisfatórios. Na Tabela 4.9. é mostrada a disposição dos neurônios da camada de entrada e saída para cada modelo neural.

O treinamento da rede neural inicialmente foi feito apenas utilizando os ensaios do sistema funcionando em malha aberta, os testes foram realizados com dados do sistema funcionando em malha fechada (pois com dados de malha aberta o modelo convergia bem, mas quando testado em malha fechada o modelo apresentava resultados insatisfatórios), e a validação dos modelos foi feita on-line, mas apenas para os modelos que apresentaram bons resultados nesta fase. Como já foi dito anteriormente, em muitos

treinamentos o modelo mostrava resultados adequados, mas durante a validação do modelo os resultados não eram os esperados.

Tabela 4. 9 Disposição dos neurônios nas camadas de entrada e saída.

Modelo	Entradas													Saídas	
	TD	TC	TE	TSE	TS	TEP	TPG	FC	FB	DC	DB	TE (K-1)	TPG (K-1)	TE (K+1)	TPG (K+1)
1	X	X	X	X	X	X	X	X	X	X	X			X	X
2	X	X	X	X	X	X	X	X	X	X	X			X	X
3	X	X	X	X	X	X	X	X	X					X	X
4	X		X	X	X	X	X	X	X			X	X	X	X
5	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
6	X	X	X	X	X	X	X	X	X	X	X			X	X

Foram utilizados como critérios de avaliação os parâmetros do ajuste linear (coeficientes angular, linear e de determinação) dos gráficos de temperatura real versus temperatura calculada pelo modelo neural. Uma avaliação visual também foi feita para verificar a concordância entre os pontos experimentais e os calculados.

Modelo Neural 1

Esta rede foi treinada utilizando uma determinada variável de entrada (Temperatura de entrada da válvula de expansão) que foi descartada nos outros treinamentos devido a erros de leitura do sensor, o que causava inconsistências no banco de dados.

Na Tabela 4.10, é mostrada a arquitetura e as funções utilizadas neste modelo 1, e nas Figuras 4.4, 4.5, 4.6, 4.7 e 4.8, são mostrados os resultados obtido com o treinamento da rede e teste. O eixo chamado “performance” corresponde aos valores de SSE do conjunto de treinamento, sendo determinado pela seguinte equação:

$$SSE = \sum_{i=1}^n (y_i - a_i)^2 \quad (4.1)$$

Tabela 4. 10 Arquitetura e funções utilizadas pela rede neural.

Camadas	Número de neurônios	Funções de ativação
Entrada	12	
1ª Intermediária	24	<i>tansig</i>
2ª Intermediária	X	X
Saída	2	<i>purelin</i>

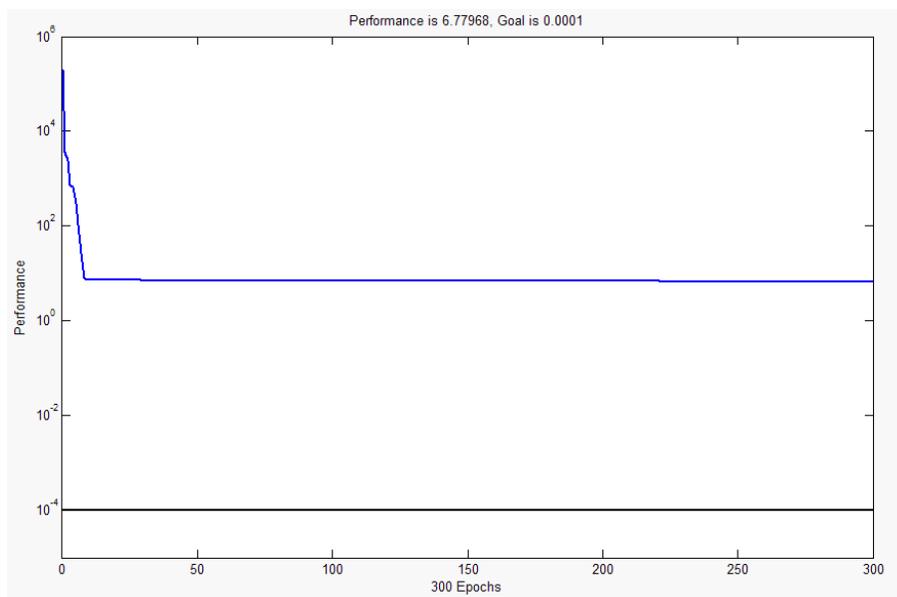


Figura 4. 4 Desempenho do modelo neural 1 durante o treinamento

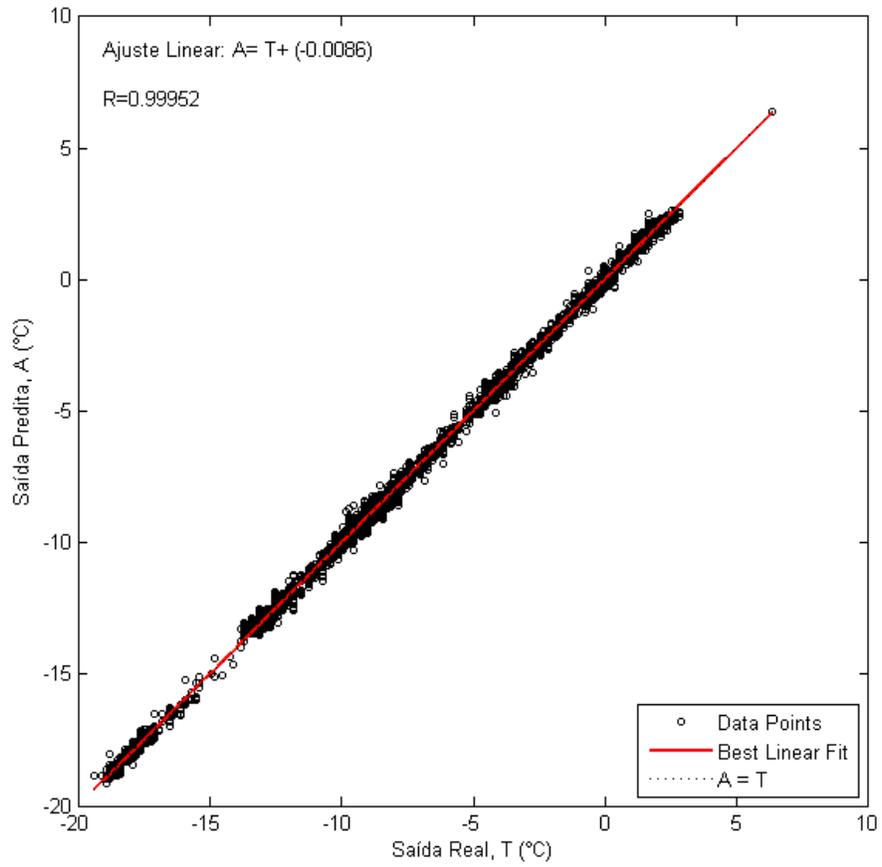


Figura 4. 5 Gráfico de dispersão entre saída real e predita para temperatura de evaporação (TE).

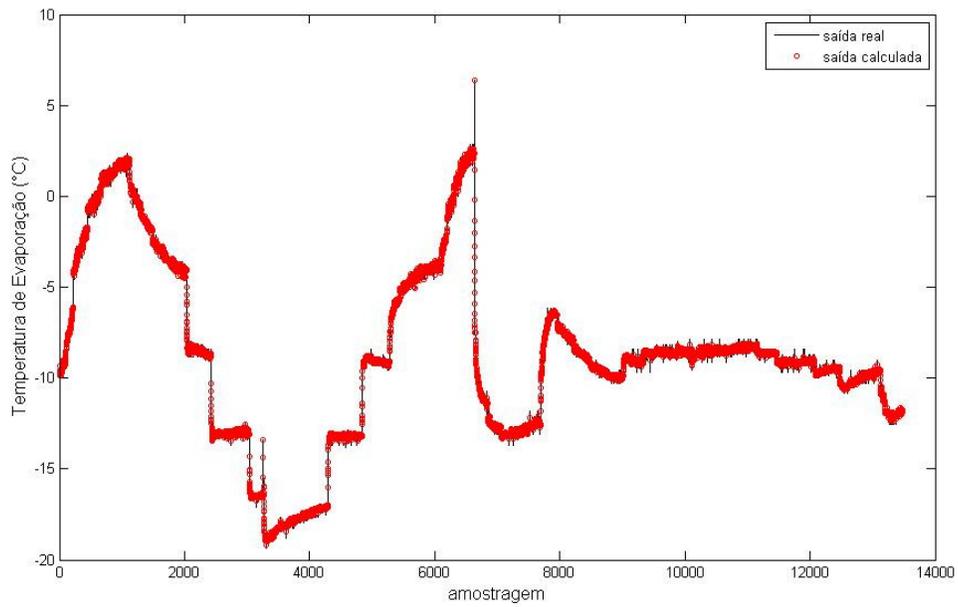


Figura 4. 6 Comparação entre saída real e calculada pela rede para TE.

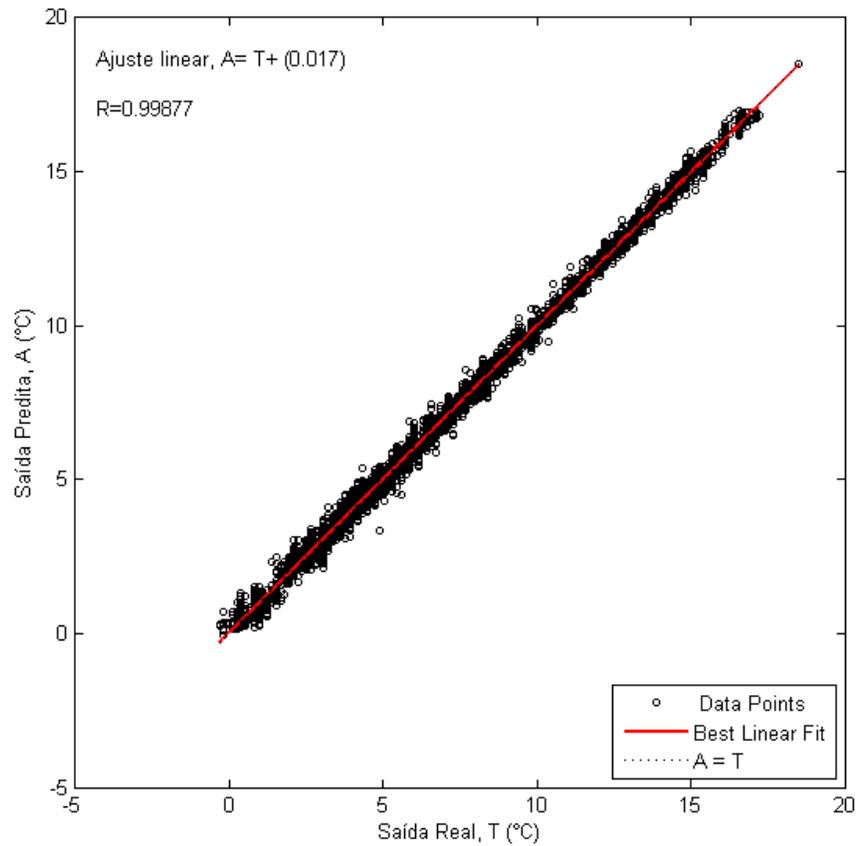


Figura 4. 7 Gráfico de dispersão entre saída real e predita para temperatura de saída do propileno (TPG).

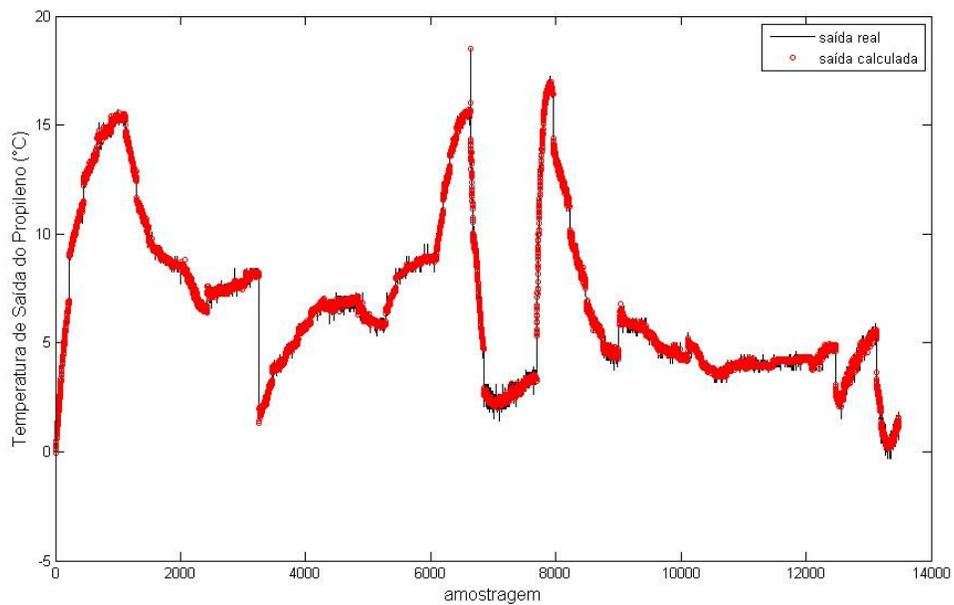


Figura 4. 8 Comparação entre saída real e calculada pela rede para TPG.

Como já era esperado, a rede neural calculou bons resultados durante o treinamento, mas na fase de validação os resultados foram bem diferentes dos esperados. Por este motivo os resultados mostrados daqui pra frente serão dos testes dos modelos neurais, pois através destes resultados é que foram definidos quais modelos seriam mais apropriados para serem testados on-line.

Nas Figura 4.9, 4.10, 4.11 e 4.12abaixo são mostrado os resultados dos testes do modelo neural 1, com os dados obtidos em malha fechada com PID, não usados no treinamento.

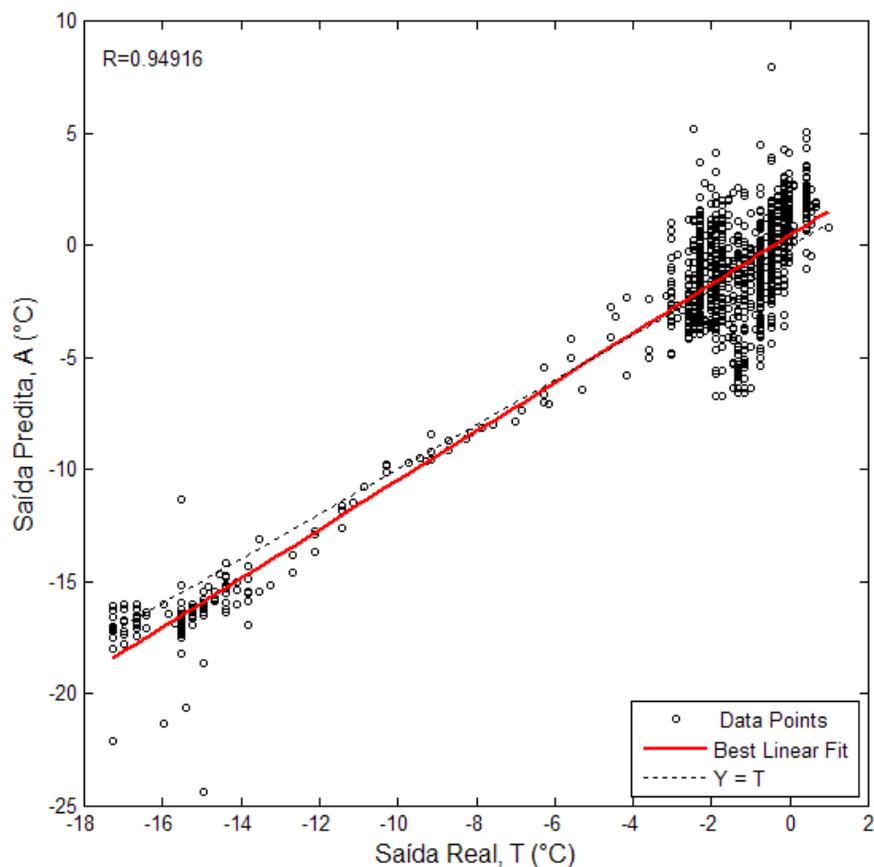


Figura 4. 9 Dispersão entre a saída real e a calculada para a TE, para o modelo neural 1.

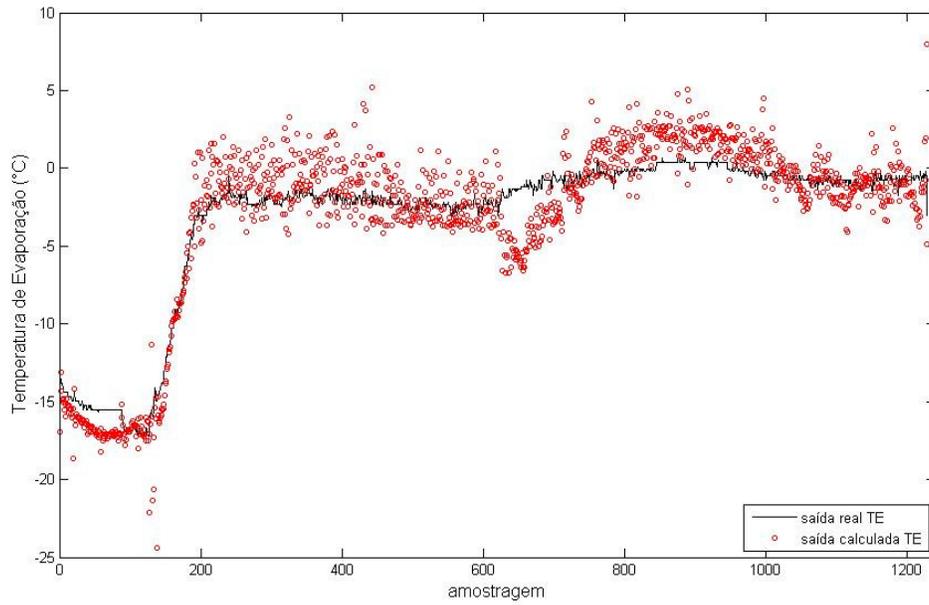


Figura 4. 10 Comparação entre a saída real e a calculada pela rede, para o modelo neural 1.

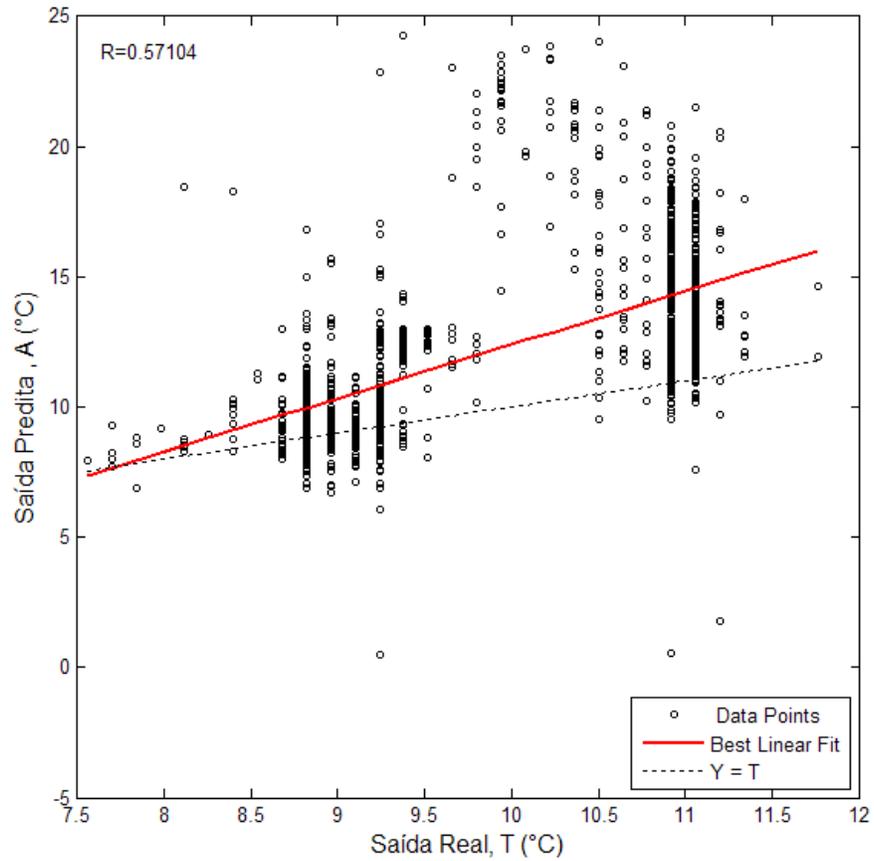


Figura 4. 11 Dispersão entre a saída real e a calculada para a TPG, para o modelo neural 1.

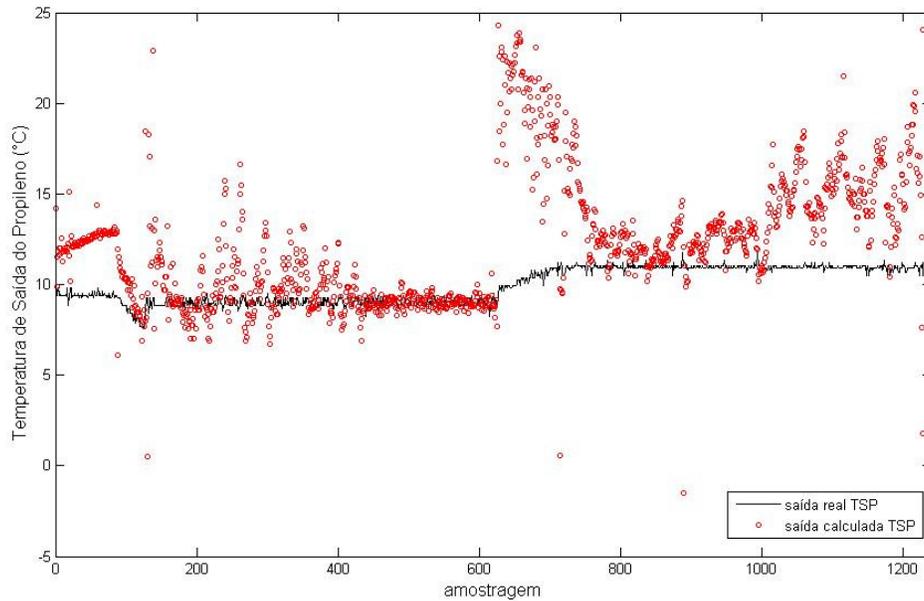


Figura 4. 12 Comparação entre a saída real e calculada pela rede neural.

Como se pode observar, durante o treinamento a rede, apesar de apresentado um SSE da ordem de 10^1 , conseguiu prever com certa precisão as variáveis de saída mas durante o teste a predição ficou fora da meta. O que mostrou que dados do sistema em malha fechada deveriam ser incluídos no treinamento e que as variáveis utilizadas na entrada da rede talvez não fossem capazes de fornecer informações suficientes para que a rede capturasse a dinâmica do processo.

Modelo Neural 2

Portanto, realizou-se um novo treinamento utilizando dados do sistema funcionando em malha fechada com o controlador PID e, excluiu-se a variável da temperatura de entrada da válvula de expansão.

A arquitetura utilizada para esta rede é mostrada na Tabela 4.11, e os resultados correspondentes a esta etapa são mostrados da Figura 4.13 à Figura 4.17.

Tabela 4. 11 Arquitetura utilizada para o modelo neural 2.

Camadas	Número de neurônios	Funções de ativação
Entrada	11	
1ª Intermediária	22	<i>tansig</i>
2ª Intermediária	X	X
Saída	2	<i>purelin</i>

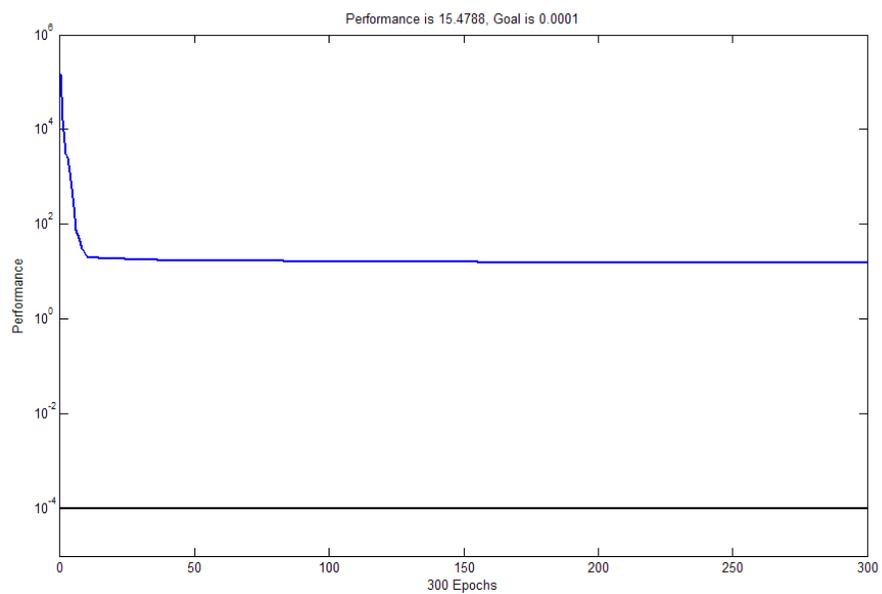


Figura 4. 13 Desempenho do modelo neural 2 durante o treinamento.

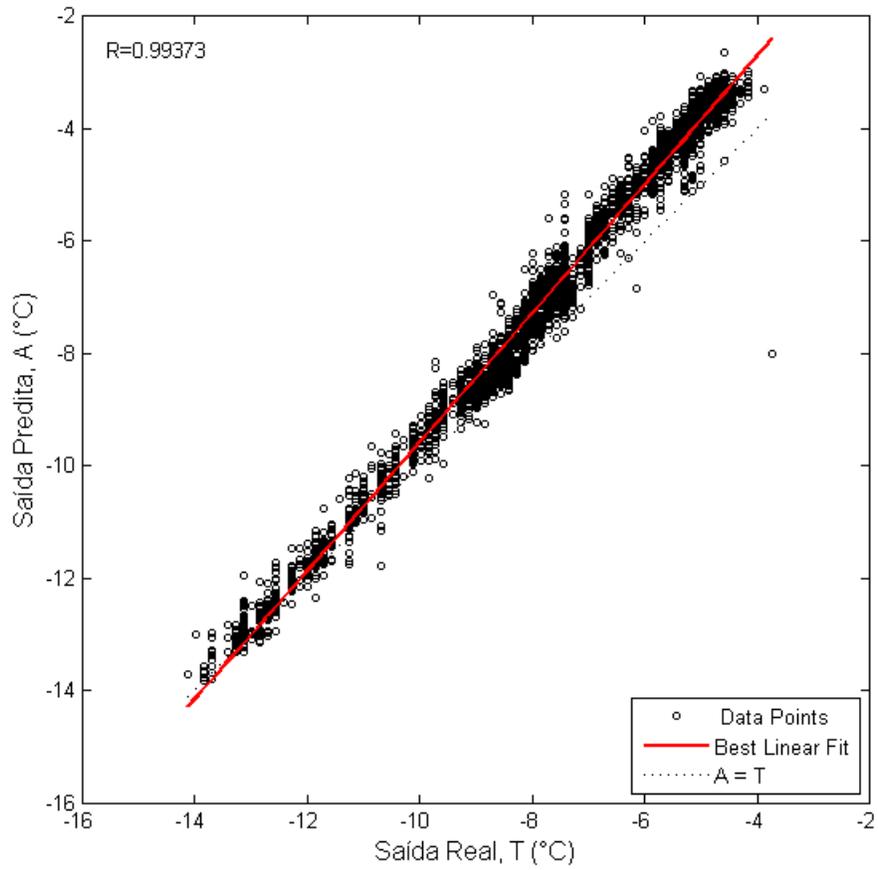


Figura 4. 14 Dispersão entre saída real e calculada para TE.

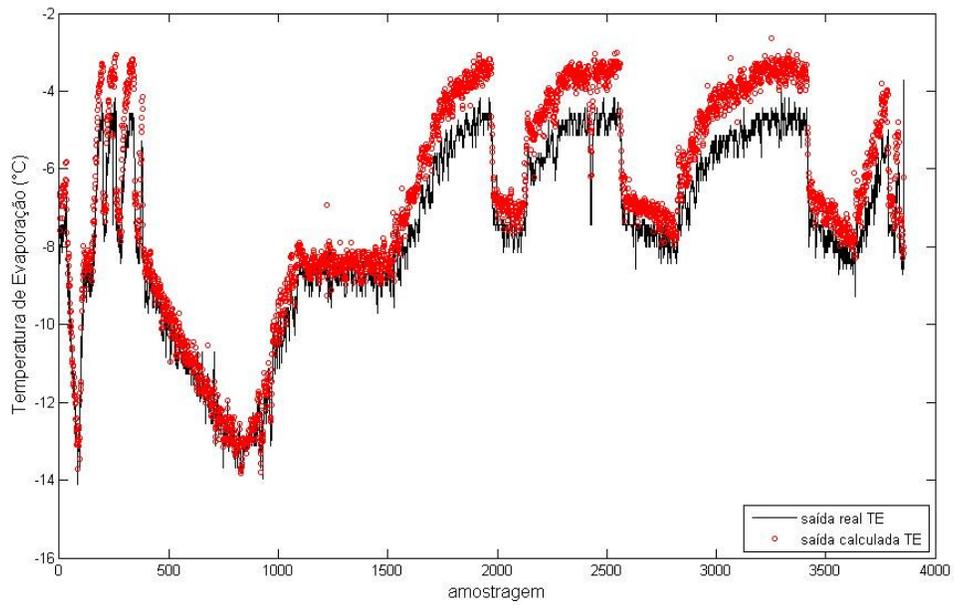


Figura 4. 15 Comparação entre saída real e calculada pelo modelo neural 2 para TE.

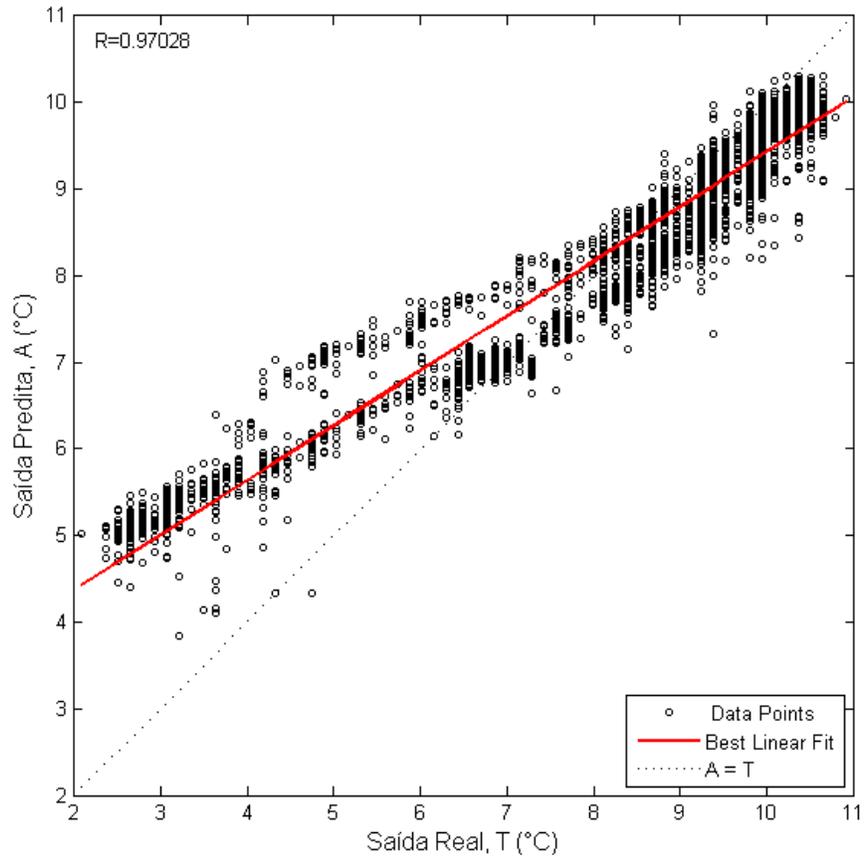


Figura 4. 16 Dispersão entre saída real e calculada para TPG.

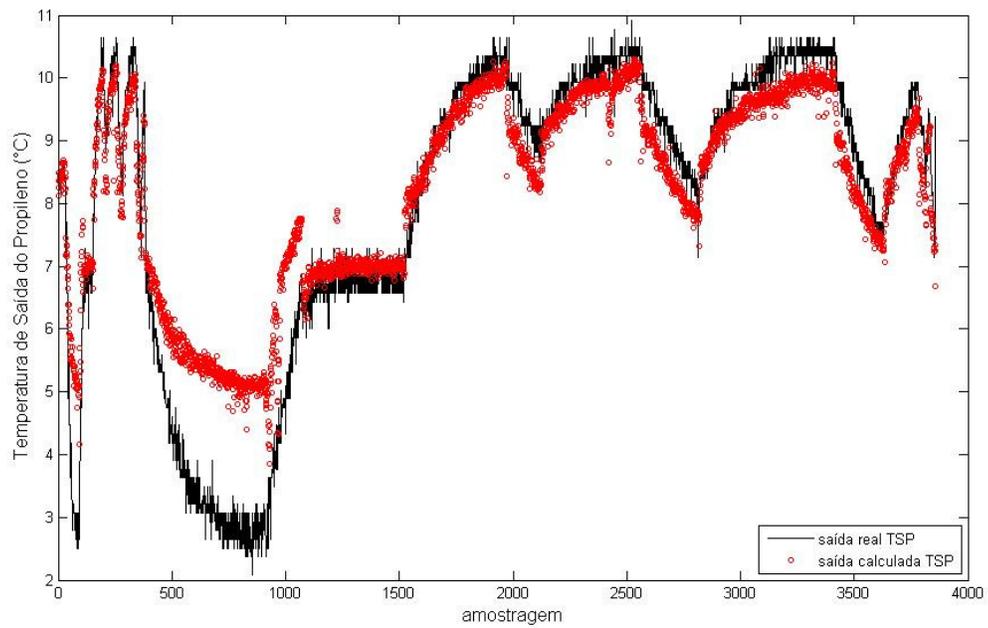


Figura 4. 17 Comparação entre saída real e calculada pelo modelo neural 2 para TPG.

Nota-se nos resultados apresentados por este modelo que o uso de dados do sistema funcionado em malha fechada e a exclusão da variável da temperatura que causava inconsistência no banco de dados melhoraram o desempenho da rede neural.

Modelo Neural 3

Com a expressiva melhora nos resultados obtidos a partir do uso de dados do sistema em malha fechada, no treinamento definiu-se que o banco de dados utilizados estava adequado, e só seria alterada a arquitetura da rede (entradas e camada intermediária) a partir deste momento.

Assim, a partir dos mesmos dados realizou-se novo treinamento, desta vez excluindo os deltas das frequência tanto do compressor como da bomba, com o objetivo de entender a influência que estas variáveis estavam tendo sobre o modelo neural. Estas variáveis foram incluídas com o objetivo de fornecer a rede informações a respeito das variações das frequências de rotação frente a variações nas temperaturas.

Na Tabela 4.12 é mostrada a arquitetura utilizada e, da Figura 4.18 à Figura 4.22 são mostrados os resultados apresentados por este modelo neural.

Tabela 4. 12 Arquitetura utilizada para o modelo neural 3.

Camadas	Número de neurônios	Funções de ativação
Entrada	9	
1ª Intermediária	18	<i>tansig</i>
2ª Intermediária	X	X
Saída	2	<i>purelin</i>

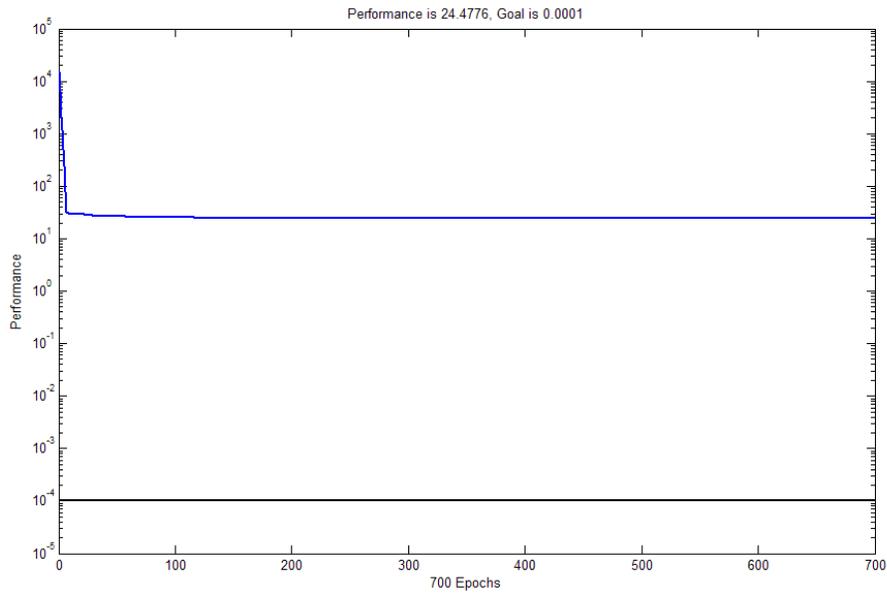


Figura 4. 18 Desempenho da rede neural durante o treinamento.

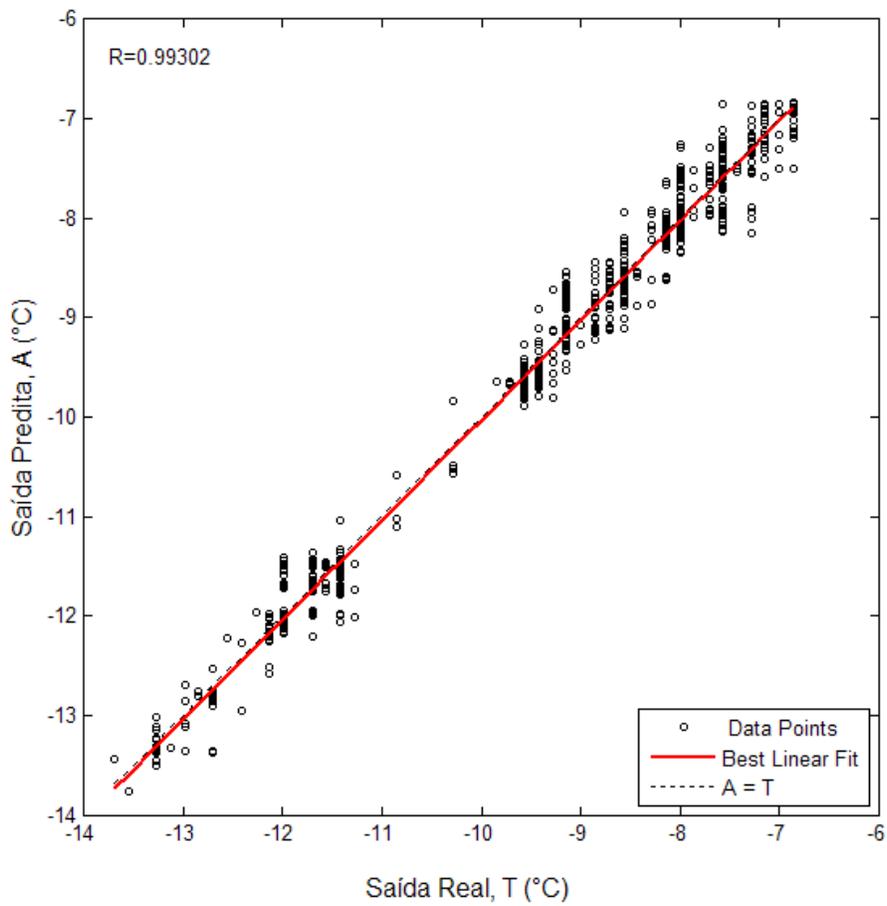


Figura 4. 19 Dispersão entre saída real e calculada para TE.

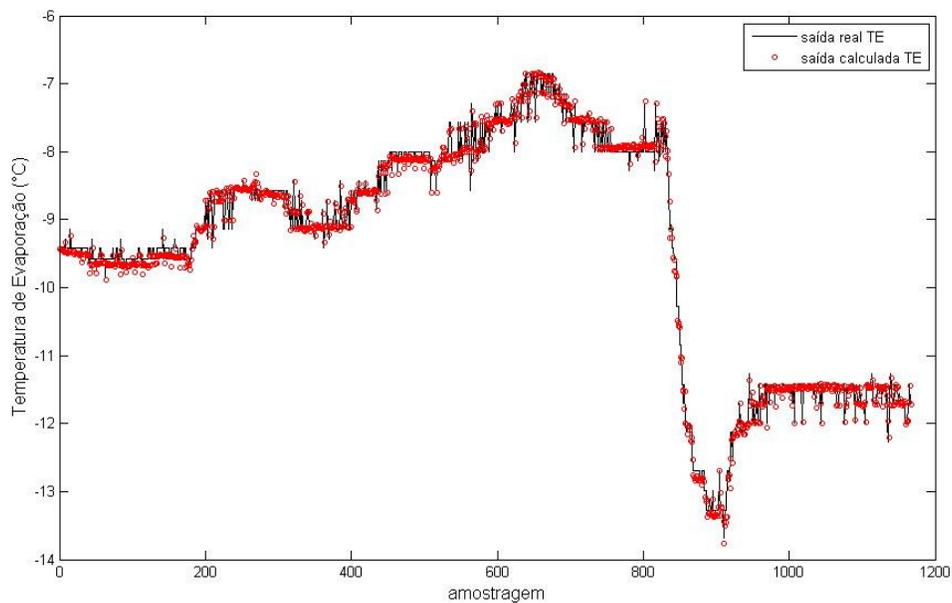


Figura 4. 20 Comparação entre a saída real e calculada para TE.

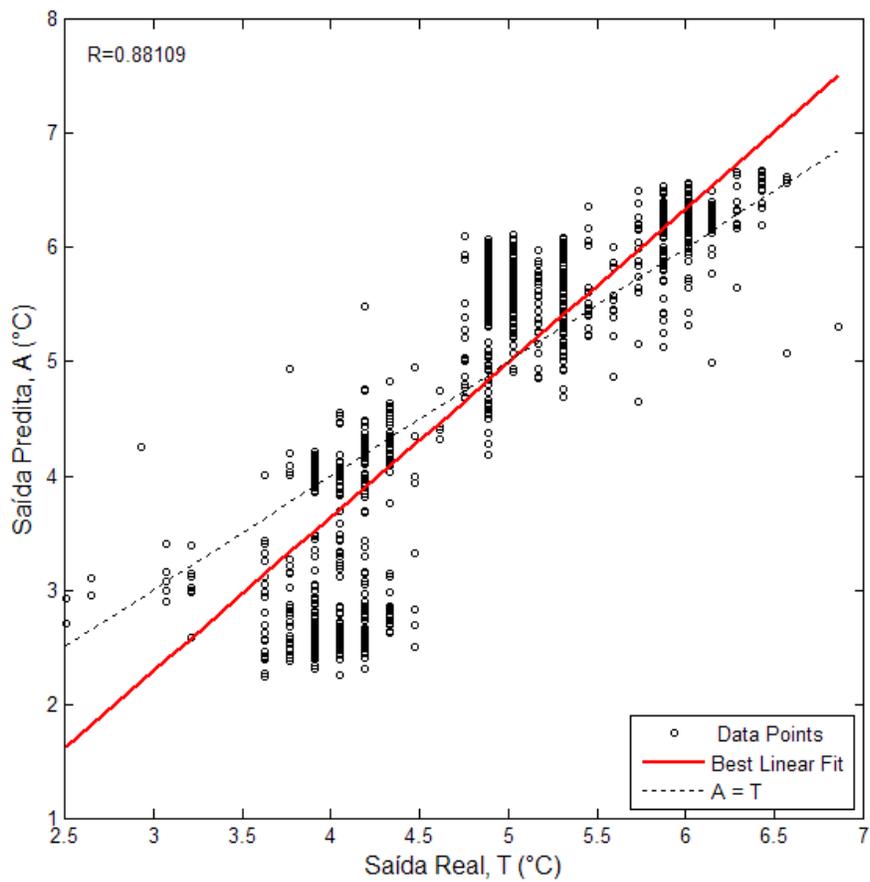


Figura 4. 21 Dispersão entre a saída real e a calculada para a TPG.

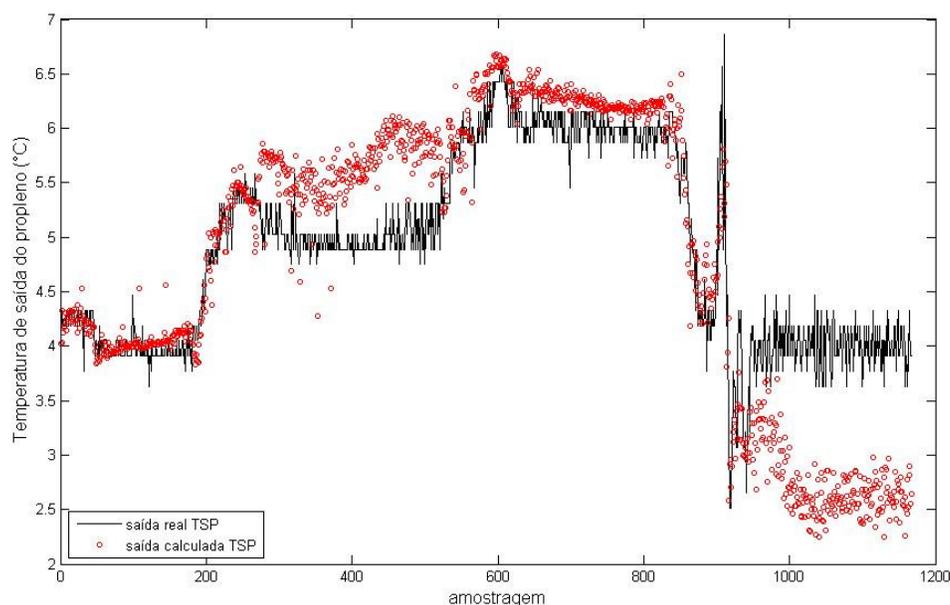


Figura 4. 22 Comparação entre saída real e calculada para TPG.

Após outros testes serem realizados neste modelo decidiu-se alterar o conjunto de dados de teste da rede neural, pois com os dados utilizados anteriormente a rede neural não estava convergindo adequadamente, talvez pela grande quantidade de variações aplicadas no ensaio em malha fechada em questão, e estas variações também poderiam se mais úteis sendo utilizadas no conjunto de dados de entradas da rede.

Mas, como pode ser observado nas figuras o erro durante o treinamento diminui pouco com a ausência das duas variáveis excluídas, e a rede neural foi capaz de prever com precisão a TE, mas não foi capaz de prever a TPG.

Modelo Neural 4

Em alguns modelos foram feitas tentativas de utilizar as variáveis que seriam preditas, na entrada com até três atrasos no tempo como é o caso deste modelo em que a temperatura de evaporação (TE) e de saída do propileno (TSP) foram utilizadas desta forma. E, também durante esta etapa de treinamentos optou-se por aumentar o numero

de camadas intermediárias, já que com apenas uma a rede neural não estava conseguindo prever as duas temperaturas ao mesmo tempo.

Na tabela 4.13 é apresentada a arquitetura da Rede 4, e da Figura 3.23 à Figura 3.27 são apresentados os resultados para este modelo neural.

Tabela 4. 13 Arquitetura utilizada para o modelo neural 4.

Camadas	Número de neurônios	Funções de ativação
Entrada	15	
1ª Intermediária	15	<i>tansig</i>
2ª Intermediária	7	<i>tansig</i>
Saída	2	<i>purelin</i>

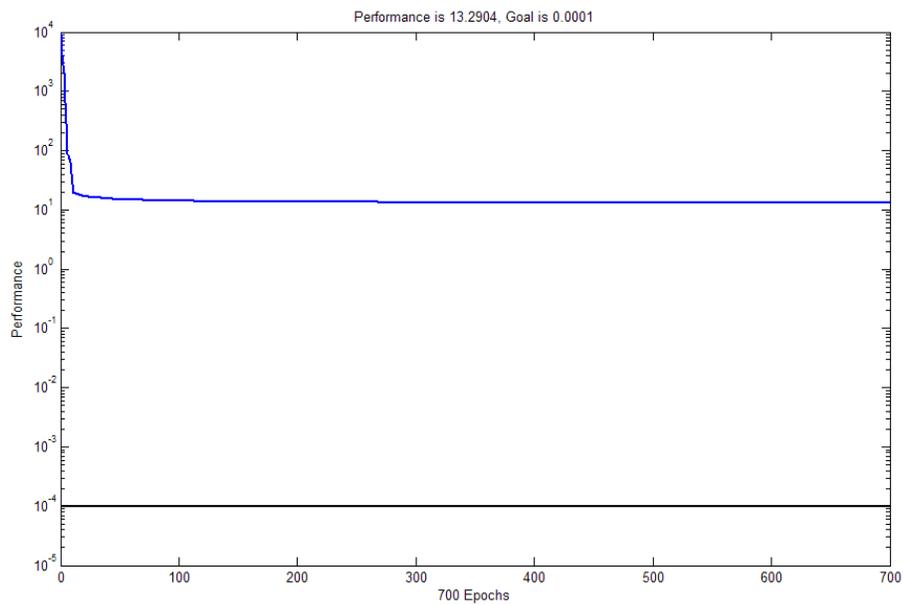


Figura 4. 23 Desempenho da rede neural 4 durante o treinamento.

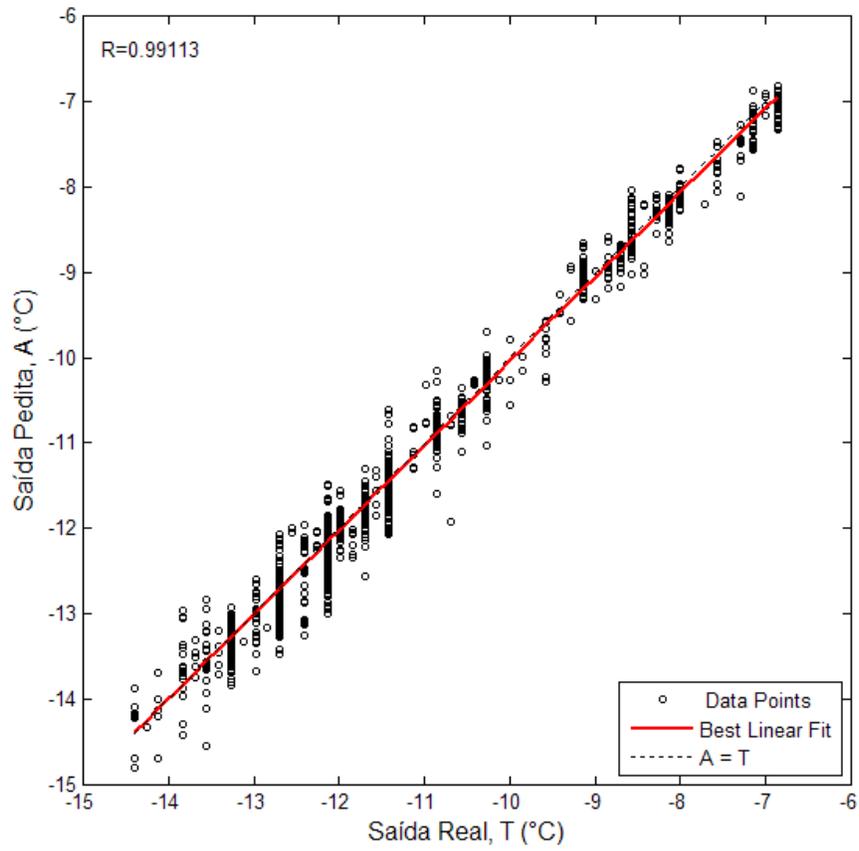


Figura 4. 24 Dispersão entre a saída real e predita para TE.

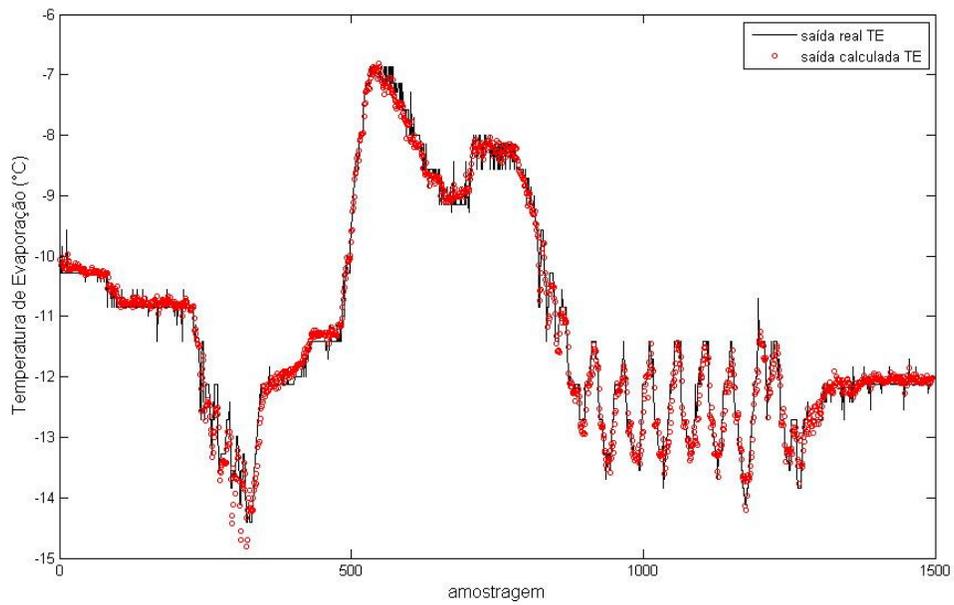


Figura 4. 25 Comparação entre saída real e predita pela rede para TE.

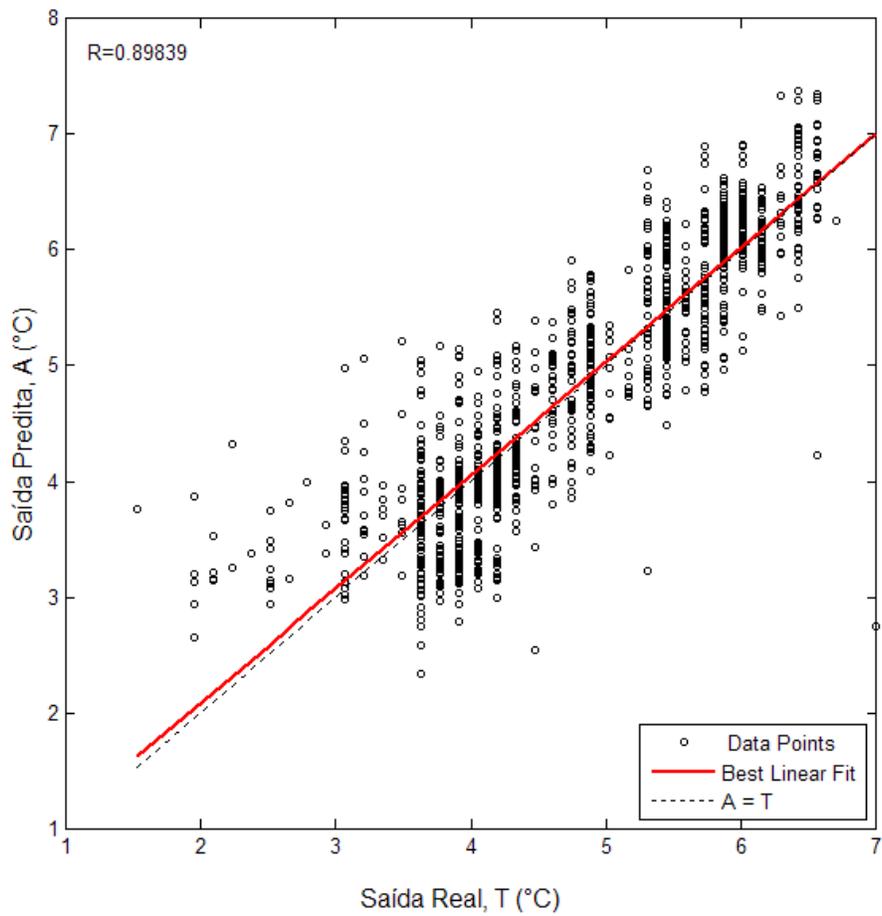


Figura 4. 26 Dispersão entre saída real e predita para TPG.

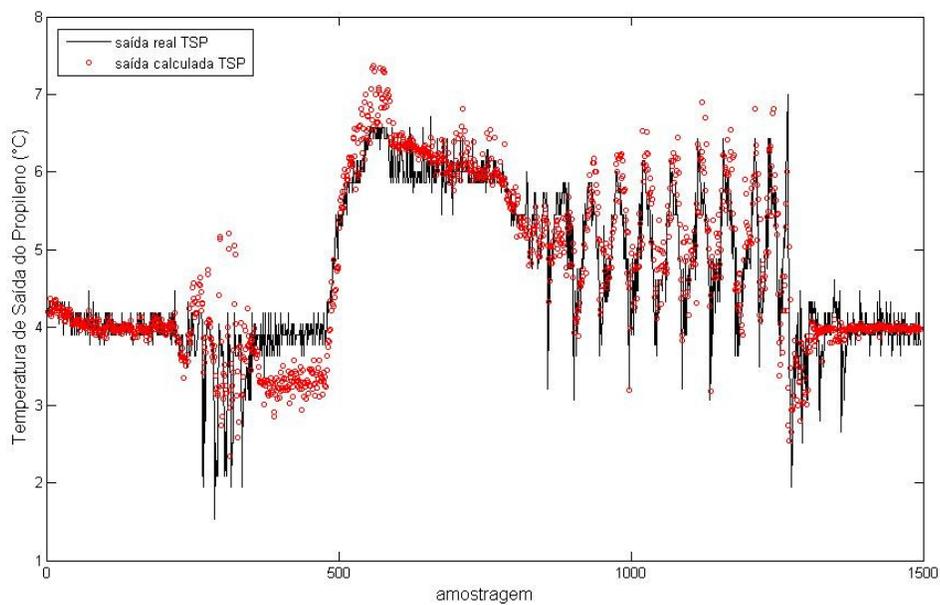


Figura 4. 27 Comparação entre a saída real e predita para TSP.

Os resultados são apresentados em diferentes conjuntos de saída pois para provar que realmente a rede neural era capaz de prever as temperaturas de evaporação e de saída do propileno, quando apresentavam um resultado razoável com o conjunto usado, eram testadas em outro para realmente testar sua eficiência.

Mesmo com a inclusão das variáveis de saída, na entrada da rede neural com atrasos de até três amostras, a rede não foi capaz de prever novamente a temperatura de saída do propileno.

Modelo Neural 5

Como pôde ser observado, a rede ainda não foi capaz de prever com exatidão as saídas desejadas, então esperando uma melhor convergência da rede neural foi inclusa mais uma camada intermediária. Na tentativa de melhorar o desempenho da rede neural foi utilizado como entrada de alguns modelos neurais o valor da potência aplicada à resistência que esta instalada no tanque de propileno, de forma a tentar mostrar para a rede neural os momentos que eram feitas perturbações no sistema.

Os resultados podem ser vistos nas Figuras, 4.28, 4.29, 4.30 e 4.31, e na tabela 4.14 é apresentada a arquitetura da Rede 5.

Tabela 4. 14 Arquitetura da rede neural do modelo 5

Camadas	Número de neurônios	Funções de ativação
Entrada	14	
1ª Intermediária	6	<i>tansig</i>
2ª Intermediária	8	<i>tansig</i>
3ª Intermediária	3	<i>tansig</i>
Saída	2	<i>purelin</i>

O erro durante o treinamento permaneceu praticamente o mesmo para todos os modelos treinados, em torno de 10^1 .

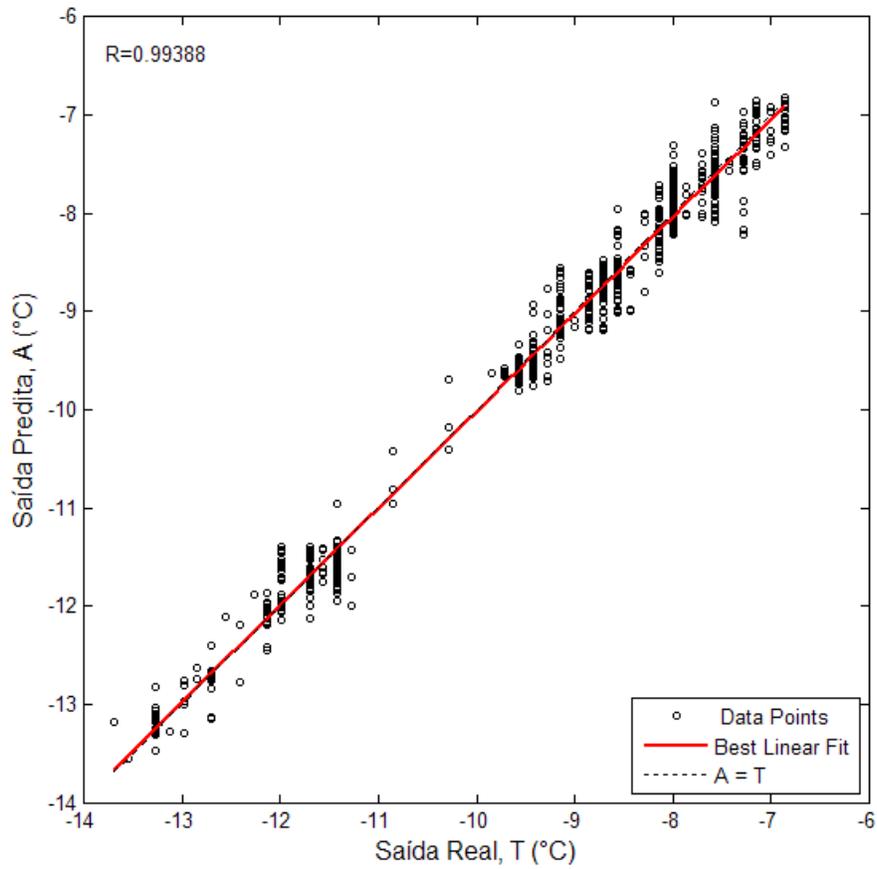


Figura 4. 28 Dispersão entre saída real e calculada para TE.

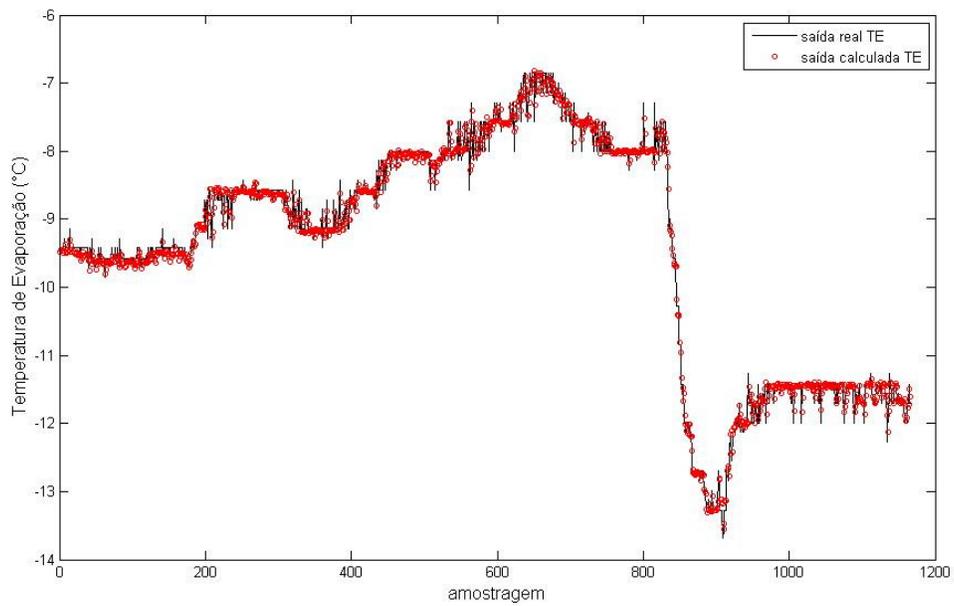


Figura 4. 29 Comparação entre saída real e predita pela rede neural 5 para TE.

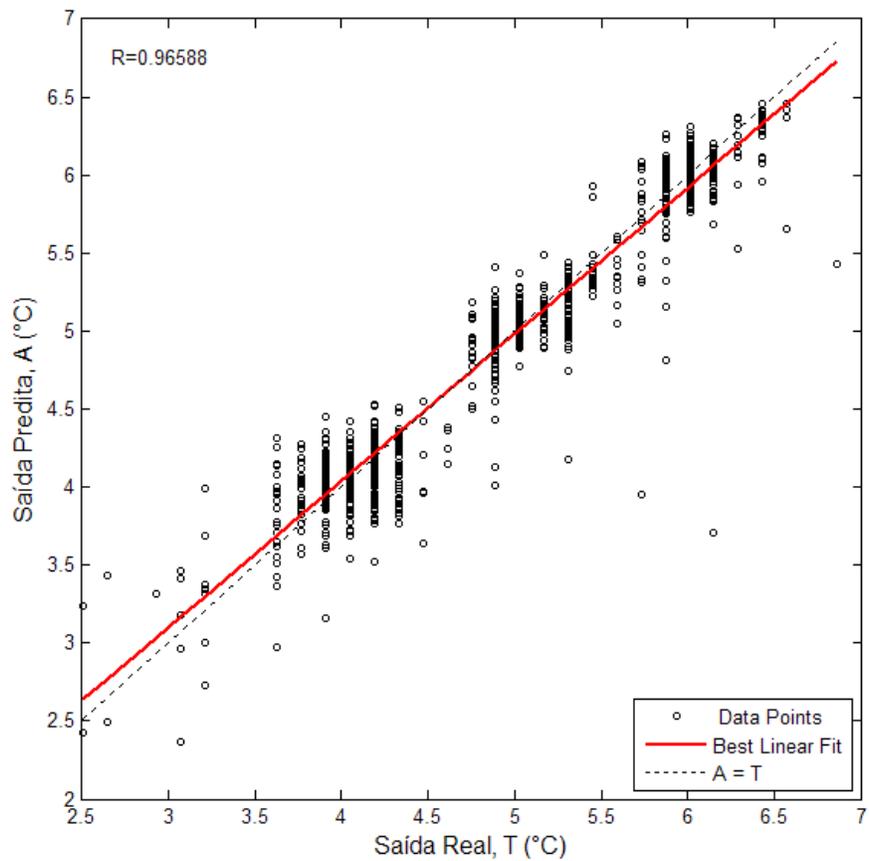


Figura 4. 30 Dispersão entre saída real e calculada para TSP.

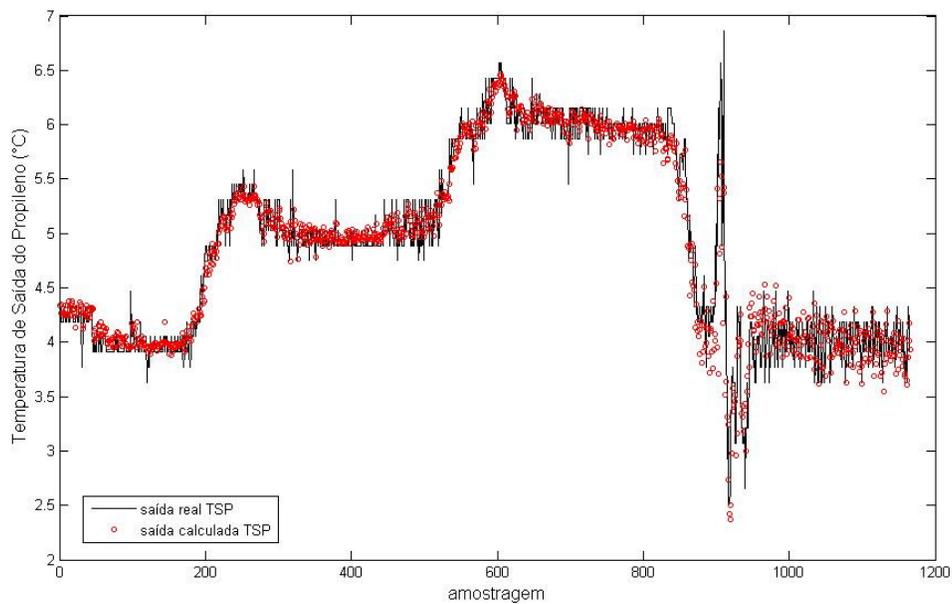


Figura 4. 31 Comparação entre saída real e predita para TSP.

Analisando os resultados apresentados por este modelo (Figura 4.28 à 4.31) percebe-se que com a inclusão de uma camada oculta a mais na arquitetura da rede, o número de iterações foi maior e com isso a rede conseguiu prever de forma mais satisfatória as saídas.

Modelo Neural 6

Com toda a base de conhecimento adquirido até agora, e após serem realizados dezenas de treinamentos testando diversas arquiteturas para o modelo neural, chegou-se a um conjunto de dados de treinamento e uma arquitetura ideal que foi capaz de prever com precisão as saídas, mesmo sendo alterado o conjunto de teste.

Para este modelo neural foram usadas como entradas apenas as variáveis coletadas através dos instrumentos instalados na planta, e os deltas das frequências que dão à rede informações sobre o quanto frequências devem ser alteradas devido à variações nas temperaturas.

Na tabela 4.15 é apresentada a arquitetura da Rede 6, e da Figura 4.32 à Figura 4.35 são apresentados os resultados obtidos com os testes deste modelo neural, que foi o modelo que apresentou os melhores resultados.

Tabela 4. 15 Arquitetura de rede utilizada no teste do modelo neural 6.

Camadas	Número de neurônios	Funções de ativação
Entrada	11	
1ª Intermediária	2	<i>tansig</i>
2ª Intermediária	8	<i>tansig</i>
Saída	2	<i>purelin</i>

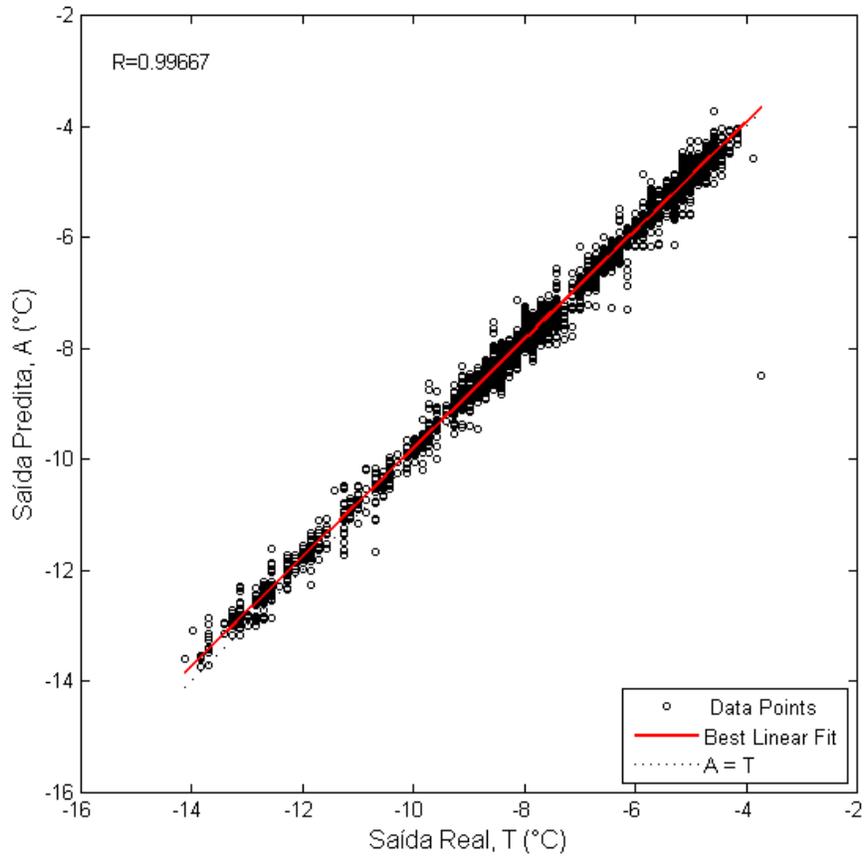


Figura 4. 32 Dispersão entre saída real e calculada pela rede para temperatura de evaporação (TE).

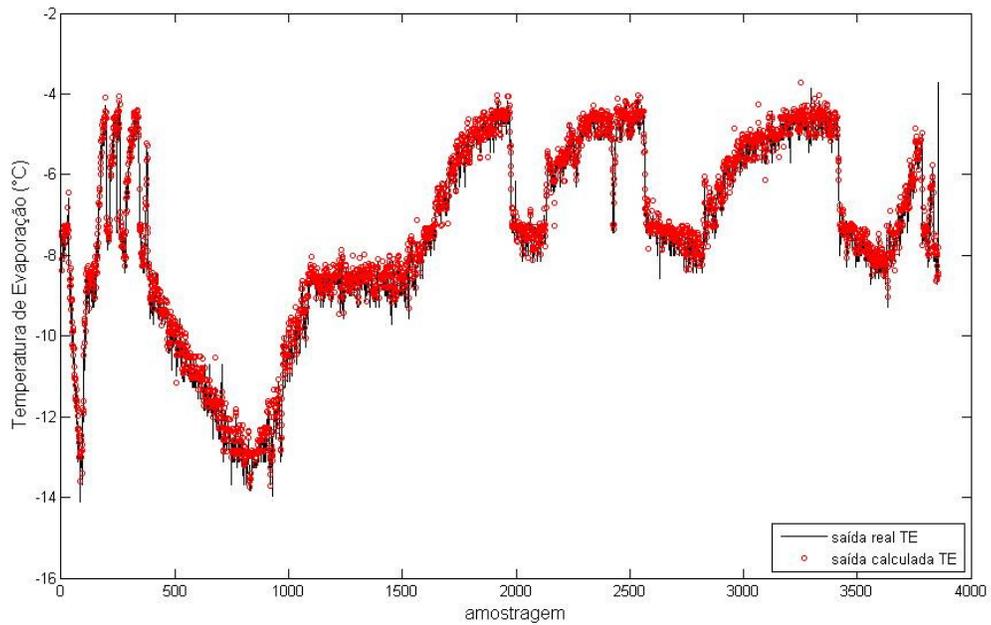


Figura 4. 33 Comparação entre saída real e predita pela rede para a TE.

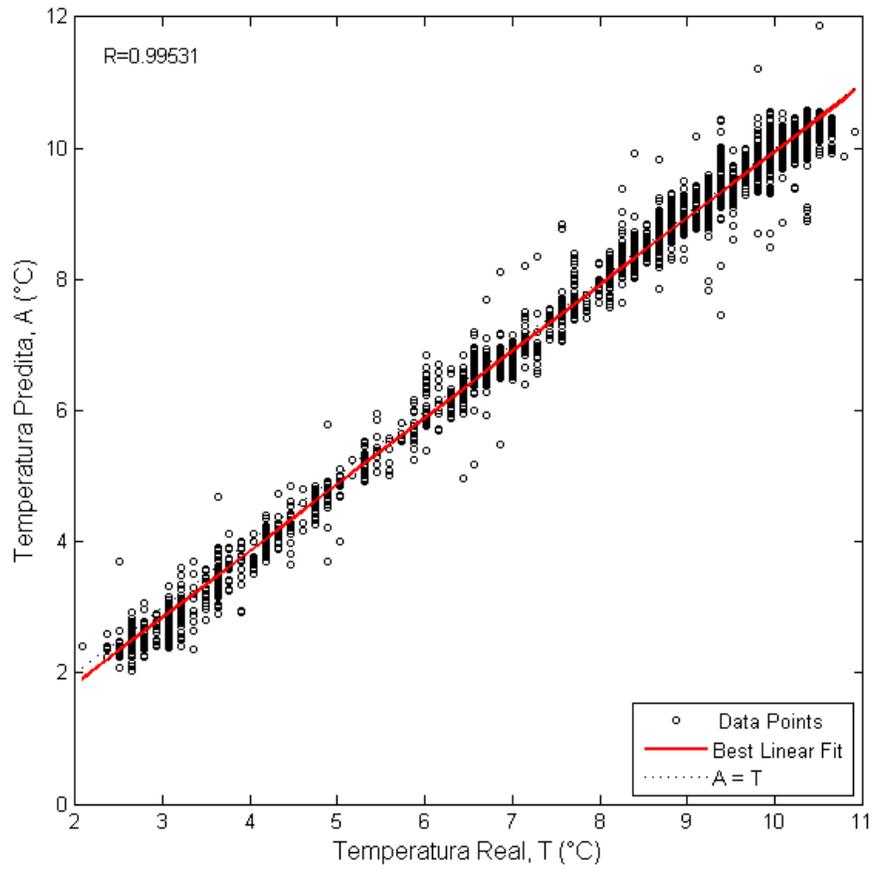


Figura 4. 34 Dispersão entre saída real e calculada para a temperatura de saída do propileno (TSP).

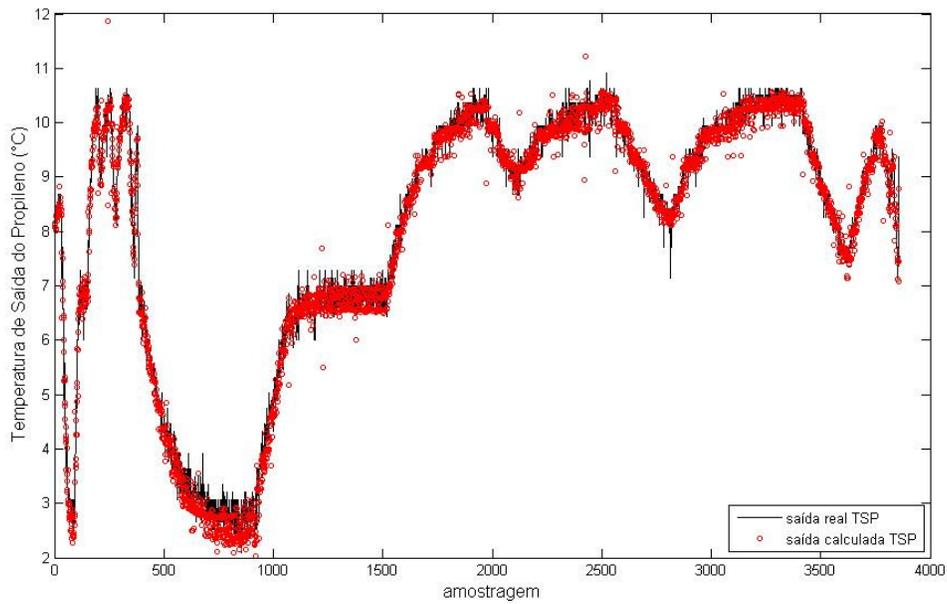


Figura 4. 35 Comparação entre a saída real e a predita pela rede para a TSP.

Os resultados alcançados até aqui mostram que para a modelagem do sistema de refrigeração em questão através de redes neurais foi necessário antes testar diversas arquiteturas e diversos conjuntos de dados até se chegar em um modelo ideal, já que o uso de algumas variáveis de entrada e alguns conjuntos de dados podem causar inconsistências no modelo neural o que faz com que o modelo não tenha a convergência necessária para prever as saídas desejadas.

Para a identificação do sistema, uma rede neural do tipo *feedforward* completamente conectada com quatro camadas foi empregada como modelo do sistema. Foram adotadas como entradas da rede as seguintes variáveis:

- Temperatura de descarga (TD);
- Temperatura de condensação (TC);
- Temperatura de evaporação (TE);
- Temperatura de saída do evaporador (R-22) (TSE);
- Temperatura de sucção (TS);
- Temperatura de entrada do propileno no evaporador (TEP);
- Temperatura de saída do propileno do evaporador (TPG);
- Frequência de rotação do motor do compressor (FC);
- Frequência e da bomba de propileno-glicol (FB);
- Delta da frequência do compressor (k-(k-1)) (DC);
- Delta da frequência da bomba de propileno (k-(k-1)) (DB);

E como saídas (as duas sendo preditas um instante à frente):

- Temperaturas de evaporação (k+1) (TE);
- Temperatura de saída do propileno do evaporador (k+1) (TPG);

A arquitetura final da rede (modelo neural 6) é mostrada na figura 4.36.

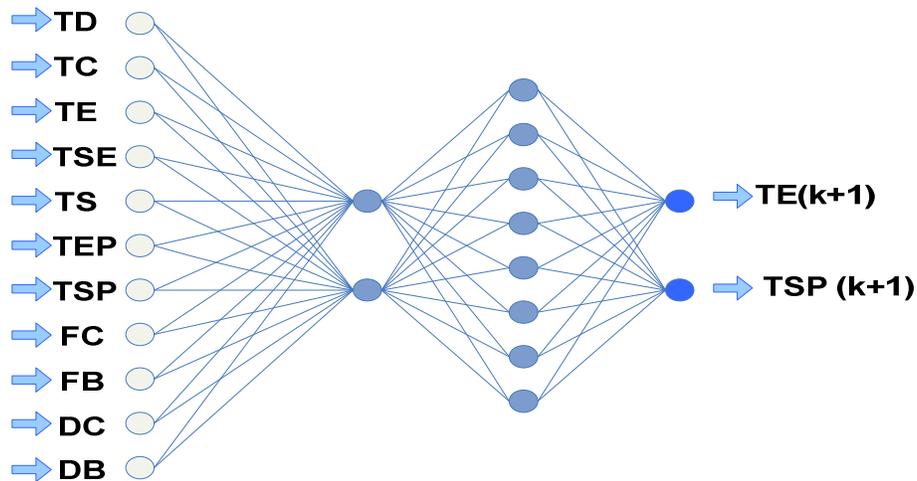


Figura 4. 36 Rede neural feedforward utilizada como modelo empírico.

Tendo sido alcançados os objetivos de predição esperados, considerou-se o modelo 6 adequado para a realização de testes online onde a confiabilidade do modelo seria realmente testada, para que o mesmo futuramente possa ser implementado em sistemas de controle preditivo.

4.4.3. Predição on-line a partir do modelo desenvolvido

Após o treinamento da rede neural, testes *on-line* também foram realizados, somente para o modelo considerado bom nos testes *off-line*, ou seja, apenas para o último modelo apresentado (Figura 4.36), pois foi o único capaz de prever com exatidão as variáveis de saída da rede.

Os ensaios de validação on-line do modelo neural foram realizados com o controlador PID atuando no sistema de refrigeração a fim de se testar várias condições de trabalho onde o modelo deveria fazer corretamente as predições das temperaturas de evaporação (TE) e de saída do propileno do evaporador (TSP).

Com o modelo neural definido partiu-se para os testes on-line. Os resultados destes teste podem ser vistos das figuras 4.37 à 4.42.

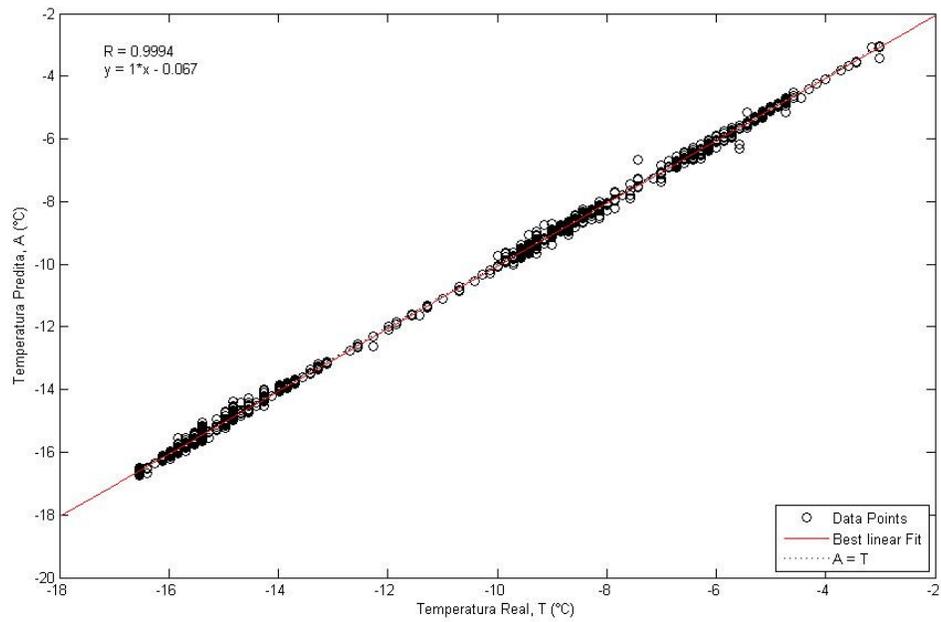


Figura 4. 37 Dispersão entre saída real e calculada para TE.

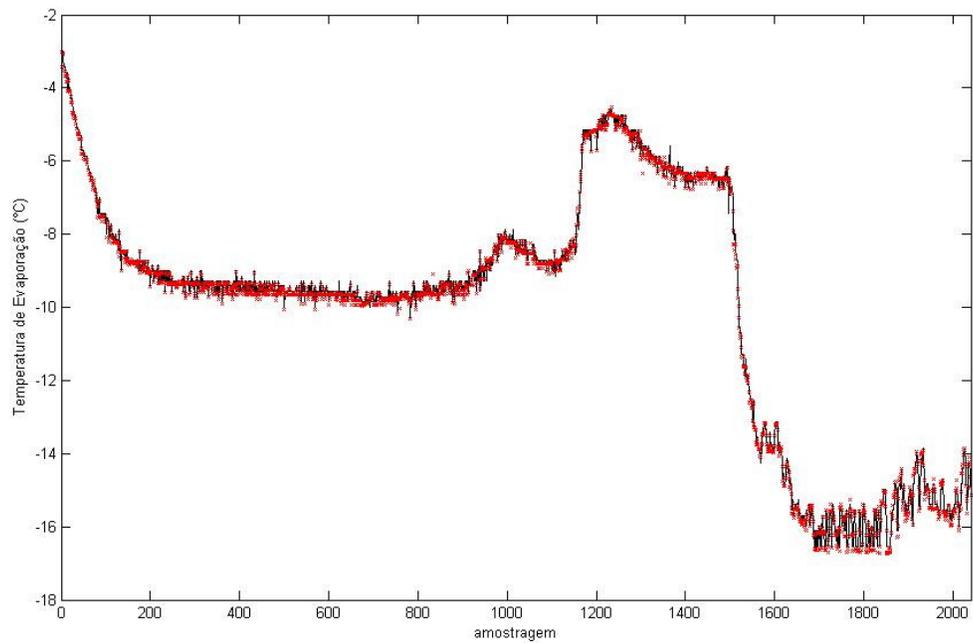


Figura 4. 38 Comparação entre saída real e predita para TE.

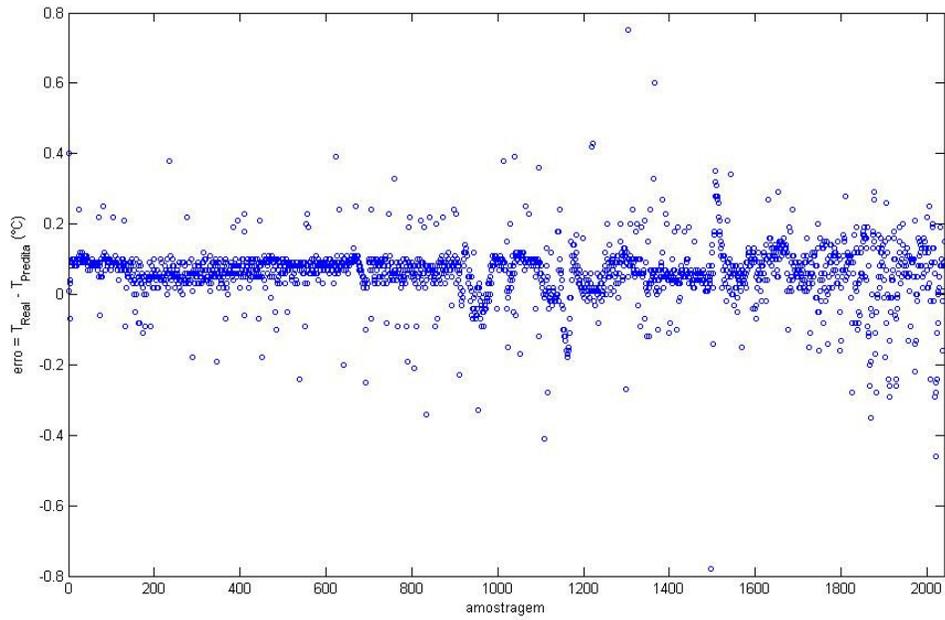


Figura 4. 39 Erro de predição para TE.

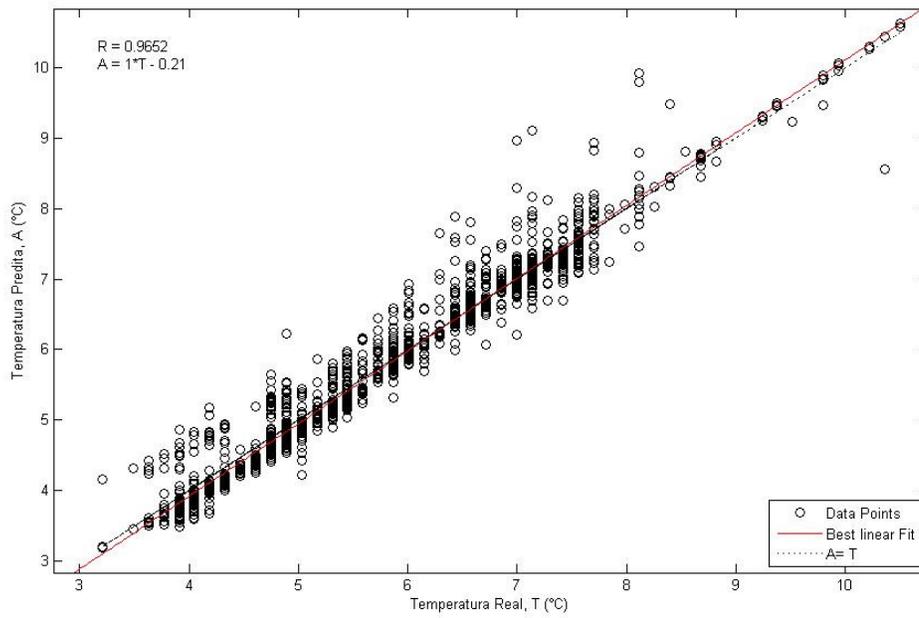


Figura 4. 40 Dispersão entre saída real e predita para TSP.

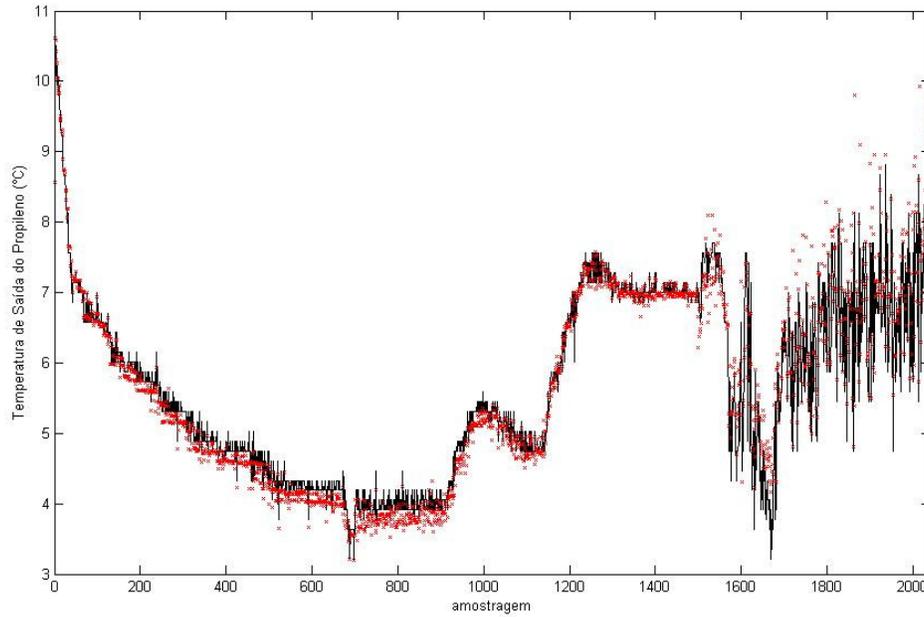


Figura 4. 41 Comparação entre saída real e predita para TSP.

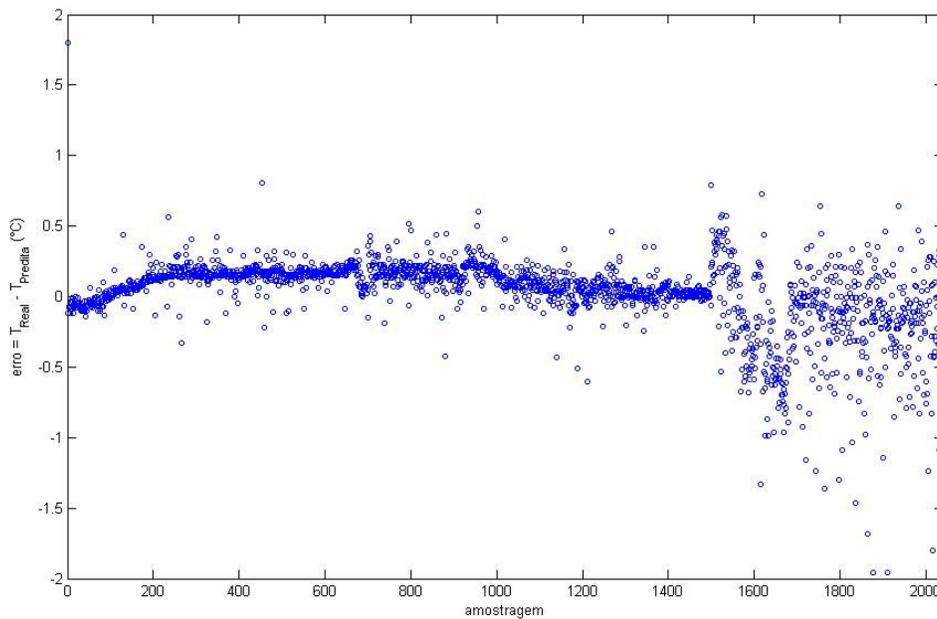


Figura 4. 42 Erro de predição para TSP.

Olhando para os resultados apresentados acima se pode ver que durante os testes on-line a rede neural foi capaz de prever a temperatura de evaporação com precisão de mais ou menos 0,5°C mesmo quando o controlador PID não conseguiu manter

a estabilidade do processo devido a uma perturbação em seu *set-point*. Mas, para a temperatura de saída do propileno, enquanto o controlador PID permaneceu estável a rede neural fez a predição da temperatura com erro também em torno de mais ou menos 0,5°C. Durante o período de instabilidade do controlador a rede perdeu um pouco da capacidade de predição da temperatura devido à grande variação desta.

5. CONCLUSÕES

Neste capítulo serão apresentadas as conclusões obtidas com o desenvolvimento deste trabalho, e está dividido em três seções: uma onde serão apresentados os resultados obtidos com a implementação do sistema de controle PID e outra onde serão apresentados os resultados da modelagem do sistema de refrigeração através de redes neurais artificiais. Uma terceira seção apresenta as sugestões para trabalhos futuros.

A proposta inicial do trabalho era a de desenvolver e implementar um controlador convencional (PID), e também modelar o sistema através de redes neurais e com o modelo obtido, implementar um controlador preditivo utilizando uma técnica de otimização utilizando o *solver* do Excel. O modelo neural foi desenvolvido e testado com sucesso e apresentou resultados satisfatórios, mas devido a problemas relacionados a sinais ruidosos no sistema de refrigeração, e a tentativa de eliminação destes mesmos, para o sucesso da implementação dos controladores, o tempo para o estudo e desenvolvimento de um controlador preditivo foi minimizado e o trabalho acabou reduzido a apenas a modelagem do sistema utilizando preditores neurais.

5.1. Resultados obtidos com a implementação do controlador multimalha

A utilização do MATLAB para a implementação das estratégias de controle facilitou a versatilidade dos ensaios e a aquisição dos dados em tempo real, agilizando a execução e análise dos parâmetros de desempenho.

As malhas de controle PID, utilizando a manipulação da frequência de rotação do compressor e da bomba de propileno, mostraram que para este tipo de controle o mais apropriado seria um tipo de controle adaptativo, já que o para variações positivas, controlador apresentou uma performance diferente das variações negativas, tanto para as perturbações na carga quanto nas perturbações do *set-point*.

O controlador multimalha, representado pela ação conjunta de duas variáveis manipuladas, apresentou resultados satisfatórios no controle da temperatura de saída do propileno glicol. A ação conjunta das malhas e os valores minimizados dos parâmetros dos

controladores, ainda apresentou um comportamento amortecido das variáveis manipuladas, viabilizando uma utilização mais adequada da variação da frequência do compressor e variação da frequência da bomba de propileno glicol.

5.2. Redes Neurais Artificiais

Na ausência de modelos matemáticos de processos, as redes neurais podem utilizar um banco de dados para construir modelos preditivos, ou seja, um modelo que pode prever as reações do processo a novas condições.

Com base nos resultados apresentados nota-se que as RNA's são eficientes no processo de identificação de sistemas não lineares multivariáveis. Além disso, as RNA's são capazes de prever o comportamento do sistema com um passo a frente sem erros significativos. No entanto estes resultados dependem do projeto da rede, ou seja, do número de entradas e saídas, do número de neurônios e de camadas ocultas, do método de treinamento usado, e também do conjunto de dados que é utilizado para o treinamento da rede neural.

Neste caso foram necessárias várias tentativas até chegar a resultados satisfatórios. Todas as variáveis que apresentaram maior influência sobre o processo tiveram que ser inseridas como entradas da rede neural.

5.3. Sugestões para trabalhos futuros

- Com o modelo neural obtido, implementar um controlador preditivo multivariável, por meio do uso de técnica de otimização, utilizando o solver do Excel. Com este objetivo, os cálculos algébricos da rede neural treinada deverão ser realizados na própria planilha eletrônica a partir dos pesos e bias obtidos do treinamento em MATLAB;
- O controlador PID poderia ser melhorado implementando metodologias como *gain scheduling* ou sintonia automática (*self-tuning*);

6. REFERÊNCIAS BIBLIOGRÁFICAS

AGUIRRE, L. A. **Introdução à Identificação de Sistemas: Técnicas Lineares e Não-Lineares Aplicadas a Sistemas Reais**, 2 ed, Editora UFMG, Belo Horizonte, BH. 2004.

AGUIRRE, L. A., RODRIGUES, G. G., JÁCOME, C. R. F. **Identificação de sistemas não-lineares utilizando modelos narmax polinomiais – uma revisão e novos resultados**. SBA Controle e Automação, v.9, n.2, agosto 1998.

AL-GALLAF, E. A., **Artificial Neural Network Based Nonlinear Model Predictive Control Strategy**, Department of Electrical and Electronics Engineering, College of Engineering University of Bahrain, P. O. Box:13184, Kingdom of Bahrain.

ASSIS, A. J. de, **Identificação e controle de processos não-lineares utilizando redes neurais artificiais**. Campinas: Faculdade de Engenharia Química, Universidade Estadual de Campinas, 2001, Tese (Doutorado)

BALLINI, R., VON ZUBEN, F. J. **Application of neural networks to adaptative control of nonlinear systems**. Automatica, v.36, 2000.

BORJA, T.J.A., **Automatização e controle inteligente on-line de sistemas de refrigeração utilizando redes neurais artificiais**. Tese de Doutorado, Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia, Uberlândia – MG, 2006.

BRAGA, A. de P., CARVALHO, A. P. de L, LUDERMIR, T. B. **Fundamentos de Redes Neurais Artificiais**; 11° Escola de Computação - Rio de Janeiro. DCC/IM, COOPE/Sistemas, NCE/UFRJ, 1999.

BRAGA, A. P., LUDEMIR, T. B., and CARVALHO, A. C. P. L., **Redes Neurais Artificiais: Teoria e Aplicações**. Rio de Janeiro LTC, 2000.

BROMELING, L., **Bayesian Analysis of Linear Models**. New York, 1985, Marcel Dekker.

- CANCELIER, A. **Otimização e controle do processo de produção do poliestileno em suspensão. Doutorado em Engenharia de produção**, Universidade Federal de Santa Catarina – UFSC, Florianópolis - SC , 2004
- CARVALHO, A. P. L. F., home Page, **Redes neurais artificiais**, <<http://www.icmc.usp.br/~andre/research/neural/>>.Consulta em 10/10/2006.
- CHEN, L., NAREDA, K. S., **Nonlinear Adaptive Control Using Neural Networks and Multiple Models**. Scientific Systems Company Inc., 500 West Cummings Park, Suite 3000, Woburn, Massachusetts, 01801, USA.
- CHEN,S., BILLINGS, S. A., and GRANT, P. M., **Non-linear systems identification sing neural networks**. International Journal of Control, 51, 1191-1214; 1992, Recursive hybrid algorithm for non-linear system identification using radial basis function networks. Ibid., 55, 1051-1070. 1990a
- COELHO, L. S. **Identificação e controle de processos multivariáveis via metodologias avançadas e inteligência computacional**, Tese Doutorado, Universidade Federal de Santa Catarina, Florianópolis. 2000.
- CYBENKO C., **Approximation by superpositions of a sigmoid function**, **Mathematics of Control Signals and Systems**, vol. 2, pp. 303–314, 1989.
- CZUK, M. L., **A family of model predictive control algorithmswith artificial neural networks**. **Institute of Control and Computation Engineering**, Faculty of Electronics and Information Technology, Warsaw University of Technology, ul. Nowowiejska 15/19, 00–665 Warsaw, Poland,2007.
- DHAR, V. e STEIN, R. **Seven methods for transforming corporate data into business intelligence**. Prentice-Hall, New Jersey, 1997. P-77-106.
- FILETI, A. M. F., **Controle em destilação batelada: controle adaptativo e controle preditivo com modelo baseado em redes neurais artificiais**. Tese de Doutorado,

- Universidade Estadual de Campinas, Faculdade de Engenharia Química, Campinas, SP,1995.
- FILHO, M., FELIX, E., home page “**Redes Neurais Artificiais**”. Universidade de São Paulo, SP, [HTTP://www.icmsc.sc.usp.br/~prico/neural1.html](http://www.icmsc.sc.usp.br/~prico/neural1.html), consulta em 07/04/2000
- FORESEE, D., F. & HAGAN, M. T.(1997). **Gauss-newton approximation to bayesian learning**. International Conference on Neural Networks 3, 1930–1935.
- FRANCO,I. C., SILVA,F.V. **Desenvolvimento de um sistema de controle e automação baseado em comunicação OPC em sistema de refrigeração**. Qualificação de Doutorado, FEQ, UNICAMP, 2008, pp. 26.
- FUJIKI, T. L. **Desenvolvimento de um controlador digital baseado em um modelo neural e um otimizador de uma planilha eletrônica**. Campinas: Faculdade de Engenharia Química, Universidade Estadual de Campinas, 2009. Dissertação (Mestrado).
- HABTOM, R., **Modeling a Refrigeration System Using Recurrent**, Neural Networks Resson Institute of Process Automation, University of Kaiserslautern, Postfach 3049, D-67653
- HAGAN, M. T. & MENHAJ, M. B. (1994). **Training feedforward networks with the marquardt algorithm**. IEEE Transactions on Neural Networks 5(6), 989–993. Haykin, S. (1999). **Neural Networks: A comprehensive Foundation** (2nd ed.). New Jersey : Prentice-Hall Inc.
- Haykin, S. **Redes neurais: princípios e pratica**. Trad. Paulo Martins Engel. 2ª. Ed Porto Alegre: Bookman, RS. 2001
- HUNT, K. J., SBARBADO, D., ZBIKOWSKI, R., GAWTHROP, P. J. **Neural networks for control systems – a survey**. Automatica, v. 28, p.1083-1113, 1992.
- JALILI-KHARAAJOO1, M., ARAABI, B. N., **Neural network based predictive control of a heat exchanger nonlinear process**.

- KLASSEN, T., MARTINS, T. D., FILHO, L. C. e SILVA, E. A., **Modelagem do sistema de resfriamento por imersão de carcaças de frangos utilizando redes neurais artificiais.** Centro de Engenharias e Ciências Exatas, Universidade Estadual do Oeste do Paraná, Toledo-PR. ²Departamento de Engenharia Química, Universidade Estadual de Maringá, Maringá- PR, 2006.
- KLASSEN, T., **Uso de redes neurais artificiais para modelagem da temperatura e da retenção de água no processo de resfriamento de carcaças de frangos por imersão.** Dissertação de Mestrado, Universidade Estadual do Oeste do Paraná, Campus Toledo, Centro de Engenharias e Ciências Exatas, 2008.
- KOVACS, Z. L. **Redes neurais artificiais: fundamentos e aplicações.** Texto Básico Escola Politécnica da Universidade de São Paulo, 2. ed., São Paulo, 1996.
- KRAKOW, K. I., YAN, L., LIN, S., 1992, **A model of hot-gas defrosting of evaporators. Heat and mass transfer theory.** ASHRAE Transactions 98(1)
- LAZAR, M., and PASTRAVANU, O., **A neural predictive controller for non-linear systems, Mathematics and Computers in Simulation,** 2002, 60, (2), pp. 315-324
- LINHARES, L. L. e SILVA, D. L. R. & ARAÚJO, F. M. U. e MAITELLI, A. L., **Estimação da fração molar de etanol no produto de fundo de uma coluna deetanizadora a partir de um sistema neural de inferência.**, Salvador, Bahia. 2008.
- LJUNG, L. (1996). **Development of system identification,** Technical Report LiTH-ISY-R-1910, Dept of EE. Linköping University, S-581 83 Linköping, Sweden.
- LJUNG, L. **System identification – Theory for the user.** 2ed., Prentice Hall, USA, 1999.
- LJUNG, L. **System Identification: Theory for the User,** Prentice-Hall, Englewood Cliffs, NJ. 1987.
- LOPREATO, L.G. R., **Desenvolvimento de modelo neural a partir de sistemas binários para predição de equilíbrio líquido vapor de sistemas ternários.**-Campinas, SP: [s.n.], 2009.

- MELO JR, W. S., EVAGELISTA, M. A., ARRUDA, L. V. R., NEVES JR, F., RODRIGUES, L.C.A., **Controle preditivo multivariável baseado em modelo adaptativo aplicado a um fermentador alcoólico**. Programa de Pós-Graduação em Engenharia Elétrica e Informática Industrial, Centro Federal de Educação Tecnológica do Paraná, Curitiba, PR, 2002.
- NARENDRA, K. & PARTHASARATHY, K. **Identification and Control of Dynamical Systems using Neural Networks**, IEEE Transactions on Neural Networks 1(1): 4-27, 1990.
- NERRAND, O., ROUSSEL-GAGOT, P., URBANI, D., PERSONNAZ, L., and DREYFUS, G. **Training recurrent neural networks: Why and how? An illustration in dynamical process modeling**. IEEE Transactions on Neural Networks, 5(2):178-184.1994.
- PARREIRA, E. P., PARISE J. A. R., 1993, **Performance analysis of capacity control devices for heat pump reciprocating compressors**, Heat Recovery Syst CHP Vol. 13(5) 451-562
- PAULA, W. C. F. **Análise de superfícies de peças retificadas com o uso de redes neurais artificiais.**, 2007. 151 f.
- PAZOS, F. **Automação de Sistemas e Robotica**, Rio de Janeiro TJ, 2002, Editora Axcel.
- PINELLI, T. G., **Automação e análise do consumo de energia de um sistema de refrigeração para resfriamento de líquido**. Dissertação de Mestrado, Universidade Estadual de Campinas, Faculdade de Engenharia Química, Campinas, SP: [s.n.], 2008.
- POTTKER, G., **Análise do efeito combinado de compressores e expansores de ação variável sobre desempenho de sistemas de refrigeração**. 2006. Dissertação (Mestrado em Engenharia Mecânica) – Universidade Federal de Santa Catarina – UFSC, Florianópolis – SC.
- RESENDE, E. D., **Técnicas de modelagem, simulação e otimização de processos de refrigeração e de congelamento**. Tese de Doutorado, Universidade Estadual de Campinas, Faculdade de Engenharia Química, Campinas, SP, 2000.

- REZENDE, S. O. **Sistemas Inteligentes: fundamentos e aplicações**. 1.ed. Barueri: Editora Manole Ltda, 2003, p. 141-168.
- RIVALS, I., CANAS, D., PERSONNAZ, L. e DREYFUS, G., **Modeling and Control of Robots and Intelligent Vehicles by Neural Networks**. In: Proc of the IEEE Conference on Intelligent Vehicles. Paris. France, 1994.
- RONCO, E. AND GAWTHROP, P. J., **Neural networks for modeling and control. Centre for system and control**, Department of Mechanical Engineering, University of Glasgow, 1997.
- RONCO, E. GAWTHROP, P. J. **Neural networks for modeling and control**. Technical Report: SCS 97008, Centre for System and Control, Department of Mechanical Engineering, University of Glasgow, Glasgow, UK, 1997.
- ROSENBLATT, F., **The perception: a probabilistic model for information storage and organization in the brain**, Psychological Review, vol. 65, no. 6, pp. 386–408, Nov 1958.
- RUSSEL, S. e NORVIG, P. **Artificial Intelligence – A modern approach**. Prentice-Hall, New Jersey, 1995, p. 563-597.
- SEBORG, D. E., EDGAR, T. F., MELLICHAMP, D. A. **Process Dynamics and Control**. New York: John Wiley and Sons, Inc, 1989, 717 p.
- SILVA, F. V. S., **Comparação do desempenho de um sistema de refrigeração para resfriamento líquido, controlado a diferentes modos de controle.**, Universidade Estadual de Campinas, Campinas, SP: [s.n.], 2003.
- SILVA, L. N., **Análise e Síntese de Estratégias de Aprendizado para Redes Neurais Artificiais**, Dissertação de Mestrado, Universidade Estadual de Campinas, 248 p., 1998.
- STEPHANOPOULOS, G., **Chemical Process Control**. Englewood Cliffs: Prentice Hall (1984).

- SWIDER, D.J., BROWNE, M.W., BANSAL, P.K., KECCMAN, V., **Modelling of vapour-compression liquid chillers with neural networks**, Department of Mechanical Engineering, The University of Auckland, Private Bag 92019, Auckland, New Zealand, 1999.
- TASINAFFO, P. M., **Estrutura de integração neural feedforward testadas em problemas de controle preditivo**, São José dos Campos: INPE, 2003. 230p.- (INPE-10475-TDI/945).
- TASSOU, S. A., QURESHI, T. Q., 1997, **Dynamic mode performance evaluation and energy analysis of mains and inverter driven refrigeration compressor**. *Journal of Power and Energy*, 211(4): 339-346
- TASSOU, S. A., QURESHI, T. Q., 1998, **Comparative performance evaluation of positive displacement compressors in variable-speed refrigeration applications**. *International Journal of Refrigeration*, 21(1): 29-41
- TIAN, J., FENG, Q., ZHU, R., **Analysis and experimental study of MIMO control in refrigeration system**. School of Energy and Power Engineering, Xi'an Jiao Tong University, Xi'an 710049, PR China, 2007.
- TONSIG, S. L. **Simulando o Cérebro: Redes Neurais**, PUC Campinas 2000
- VIEIRA, W. G. **FCC: Controle preditivo e identificação via redes neurais**, Universidade Estadual de Campinas, Campinas, SP, Tese de Doutorado Faculdade de Engenharia Química, 2002.
- YIGIT, S. K., ERTUNC, H. M., **Prediction of the air temperature and humidity at the outlet of a cooling coil using neural networks**. Mechanical Engineering Department, Kocaeli University, 41040 Veziroglu Campus, Kocaeli, Turkey, 2006.