

CONTROLE NEURAL E FUZZY EMBARCADO APLICADO A UM SISTEMA TÉRMICO

ELIOMAR DO CARMO*, FULTON RIKER*, IVO MEDEIROS*, ALEX OLIVEIRA*, JOSÉ BARREIROS*, ORLANDO SILVA*,
CLEISON SILVA*, MARCUS GOMES*

*Laboratório de Engenharia Elétrica, Departamento de Engenharia Elétrica e da Computação, Universidade Federal do Pará, Caixa Postal 479, CEP 66075-110 Belém, PA, BRASIL

Emails: eliomar@gmail.com, www.fulton@gmail.com, ivo701@gmail.com,
alex.ufpa@gmail.com, barreiro@ufpa.br, orfosi@ufpa.br, cleisond@gmail.com,
marcuscmg@gmail.com

Abstract — This work presents a methodology to embed intelligent controllers in microcontrollers, conceived from the application of Neural and Fuzzy control to a didactic plant for temperature control. The plant under test is presented, and the implementations of the controllers, they are first made in a personal computer and later in a microcontroller. The sequence of steps shows used to embed the controllers is presented and finally, the work is concluded showing the success obtained in the task to embed them, which validates the methodology employed.

Keywords — Embedded Control, Temperature Control, Intelligent Control

Resumo — Este trabalho apresenta uma metodologia para embarcar controladores inteligentes em microcontroladores, concebida a partir da aplicação de controle Neural e Fuzzy a um protótipo didático para controle de temperatura. Apresenta-se resumidamente a planta de teste e, em seguida, são mostradas as implementações dos controladores, que primeiramente são feitas num computador pessoal e depois num microcontrolador. É apresentada a seqüência de passos utilizada para se embarcar os controladores e, por fim, conclui-se o trabalho mostrando o êxito obtido na tarefa de embarcá-los, o que valida a metodologia empregada.

Palavras-chave — Controle Embarcado, Controle de Temperatura, Controle Inteligente.

1 Introdução

Este trabalho trata da aplicação de controle neural e fuzzy embarcados, aplicado a um protótipo didático para controle de temperatura, denominado Sistema Térmico (ST). Além disso, apresenta-se a metodologia empregada para embarcar os controladores inteligentes em um microcontrolador. O ST foi construído na Universidade Federal do Pará (UFPA) e descrito em (Carmo *et al.*, 2006). Os dois controladores são, primeiramente, implementados em um Computador Pessoal (PC) para que sejam sintonizados e, em seguida, embarcados em um microcontrolador, que já era parte do ST, mas antes tinha função limitada à aquisição de dados.

São apresentados também, a resposta da planta e o sinal de controle, tanto para os controladores não embarcados quanto para os embarcados equivalentes, em testes de desempenho realizados.

2 O Protótipo utilizado como planta de teste

O protótipo didático ST foi concebido como um trabalho de conclusão de curso e publicado no Congresso Brasileiro de Automática (CBA), (Carmo *et al.*, 2006). Trata-se de um equipamento construído para ser utilizado no ensino e na pesquisa. Esta seção descreverá sucintamente o protótipo.

O ST, ilustrado na figura 1, consiste em um recipiente cilíndrico, de vidro, com capacidade para 400ml; no interior do recipiente existe um sensor de temperatura e uma resistência elétrica para aquecer o líquido. Também compõe o protótipo, uma caixa



Figura 1. Fotografia do Protótipo

plástica, que abriga um circuito de condicionamento do sinal de temperatura advindo do sensor; um circuito de acionamento da resistência elétrica e um circuito que faz interface entre o PC e um microcontrolador PIC16F877, conforme mostra o diagrama da figura 2.

A atuação no sistema é realizada por meio da variação da potência média da resistência, cujos dados de placa são: 1000W/110V AC. A técnica empregada para variar a potência é o controle de ciclos inteiros proposta em (Ahmed, 2000).

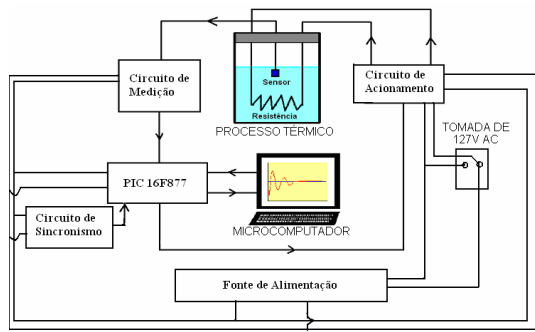


Figura 2. Diagrama Esquemático dos Componentes Físicos do Protótipo.

A programação envolvida no equipamento é constituída por duas partes distintas. A primeira, é destinada à utilização do PIC16F877, que no ST original limitava-se à aquisição de dados e à comunicação com o PC, e nesse trabalho é empregada também, na implementação dos controladores embarcados. A segunda parte refere-se a uma programação de alto nível, Linguagem gráfica LabVIEW, (Silva et al., 2005), destinada à implementação dos controladores no PC e ao desenvolvimento da interface gráfica com o usuário. A figura 3 contribui para um melhor en-

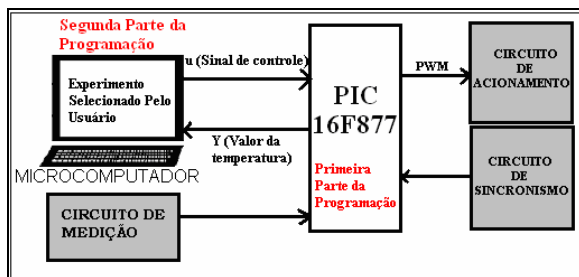


Figura 3. Programação Envolvida no Protótipo tendimento da programação envolvida no protótipo.

Em (Carmo *et al.*, 2006), levantou um modelo matemático do ST, que consiste na função de transferência de primeira ordem apresentada na equação 1. O modelo foi utilizado em simulações e no projeto de um controlador Proporcional Integral (PI) clássico, cuja saída da planta para uma referência de 50°C aplicada ao sistema (a partir da temperatura ambiente) é mostrada na figura 4. Na figura 4 estão mostrados dois testes com valores diferentes dos ganhos proporcional (K_p) e integral (K_i) (foram aplicados também perturbações de carga, após o sistema ter entrado em regime).

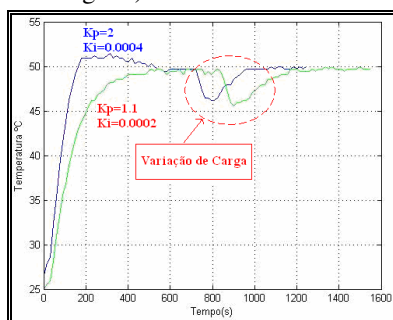


Figura 4. Resposta do sistema Controlado por um Controlador PI Clássico.

$$G(s) = \frac{27,72}{3683s + 1} \quad (1)$$

O comportamento dinâmico do sistema em malha fechada (MF), com o controlador PI clássico, foi utilizado para determinar os conjuntos fuzzy de entrada e de saída, e também, para a construção da base de regras, além de fornecer os dados para treinamento do controlador neural.

Os teste de variação de carga presente nas figuras 4, 10, 12, 15 e 17, consistiu em se adicionar 80ml de água, à temperatura ambiente, ao conteúdo existente no recipiente.

3 Controlador Fuzzy

A Teoria de Controle Fuzzy, foi concebida por L. A. Zadeh com o objetivo de fornecer ferramentas matemáticas para o tratamento de informações de caráter vago ou impreciso, (Spooner, 2002). Ela baseia-se na Lógica Fuzzy, a qual foi construída a partir de conceitos e fundamentos já presentes na teoria clássica. Os fundamentos da lógica fuzzy podem ser utilizados para traduzir, em termos matemáticos, a informação imprecisa expressa por um conjunto de regras linguísticas.

Empregou-se este recurso para intervir na dinâmica do Protótipo ST, a partir da constituição de um sistema de inferência, utilizando um conjunto de regras "SE...ENTÃO" como estratégia de controle.

No caso do Sistema Térmico, por ser um sistema SISO (*Single Input Single Output*) relativamente simples, as técnicas convencionais de controle conseguem atender às especificações desejadas de projeto de maneira satisfatória, como visto na seção 2. No entanto justifica-se o emprego do controle fuzzy, por se tratar de uma pesquisa que visa estudar a possibilidade de implementar um controlador inteligente do tipo fuzzy em um pequeno *chip* de processamento limitado, como é o microcontrolador PIC utilizado no equipamento.

3.1 Definição do controlador fuzzy a ser implementado

Nesta subseção serão apresentados a base de regras e os conjuntos de entrada e de saída do controlador fuzzy.

Segundo (Paraskevopoulos, 1996) e (Shaw and Simões, 1999), para a implementação de um controlador fuzzy PI, define-se como variáveis de entrada o erro (e) e a variação do erro (ce), e como saída a variação do sinal de controle (du). Portanto, é necessária a integração de du antes de se aplicar à planta, como mostrado na figura 5.

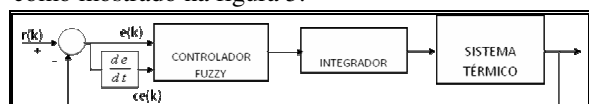


Figura 5. Controlador PI fuzzy.

As equações 2 e 3 mostram o erro $e(k)$ e a variação do erro $ce(k)$ (que podem ser filtrados), ambos em domínio discreto:

$$e(k) = r(k) - y(k) \quad (2)$$

$$ce(k) = \frac{e(k) - e(k-1)}{T} \quad (3)$$

Onde $r(k)$ é o sinal de referência, $y(k)$ é a saída do sistema, $ce(k)$ é correspondente à derivada do erro de/dt no domínio de tempo discreto e T é o período de amostragem, que neste trabalho é de 15s.

Todas as funções de pertinência foram definidas como trapezoidais e foram ajustadas por tentativas. As variáveis de entrada erro (e) e variação do erro (ce) assumem os rótulos: Negativo (N), Zero (Z) e

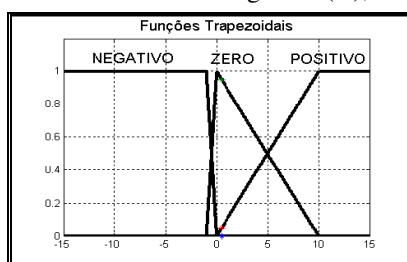


Figura 6. Apresentação dos conjuntos referentes à variável de entrada “erro”.

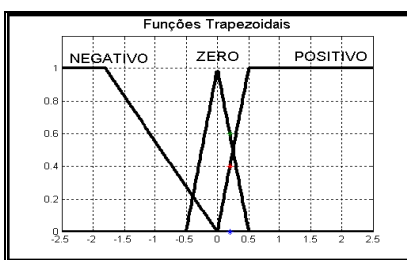


Figura 7. Apresentação dos conjuntos referentes à variável de entrada “variação do erro”.

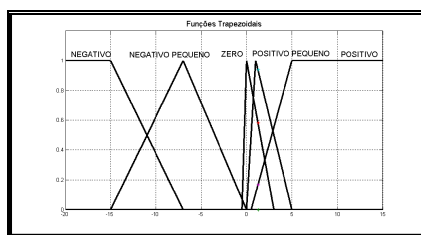


Figura 8. Apresentação dos conjuntos referentes à variável de saída “variação de potência”.

Tabela 1. Base de Regras.

- Regra 1: **SE** N(e) **e** N(ce) **ENTÃO** N(du);
 Regra 2: **SE** N(e) **e** Z(ce) **ENTÃO** N(du);
 Regra 3: **SE** N(e) **e** P(ce) **ENTÃO** N(du);
- Regra 4: **SE** Z(e) **e** N(ce) **ENTÃO** Z(du);
 Regra 5: **SE** Z(e) **e** Z(ce) **ENTÃO** Z(du);
 Regra 6: **SE** Z(e) **e** P(ce) **ENTÃO** PP(du);
- Regra 7: **SE** P(e) **e** N(ce) **ENTÃO** Z(du);
 Regra 8: **SE** P(e) **e** Z(ce) **ENTÃO** PP(du);
 Regra 9: **SE** P(e) **e** P(ce) **ENTÃO** P(du).

Positivo (P). A variável de saída variação da potência, assume os rótulos: negativo (N), negativo pequeno (NP), zero (Z), positivo pequeno (PP) e positivo (P), conforme ilustram as figuras 6, 7 e 8.

A definição da base de regras é mostrada na tabela 1.

3.2 Resultados obtidos com o controlador PI fuzzy

Após o estágio de definição das funções de pertinência e da base de regras, implementou-se o controlador fuzzy em um PC e realizou-se ensaios com a planta real. Estes ensaios foram do tipo variação degrau na temperatura da água, a partir de seu valor em temperatura inferior (em alguns casos, aplicou-se posteriormente perturbações de carga, como explicado no final da seção 2). Os gráficos referentes à saída da planta e ao sinal de controle são mostrados nas figuras 9 e 10.

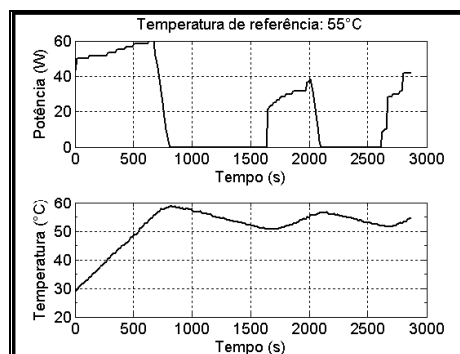


Figura 9. Sinal de controle e resposta do sistema, para o controlador Fuzzy.

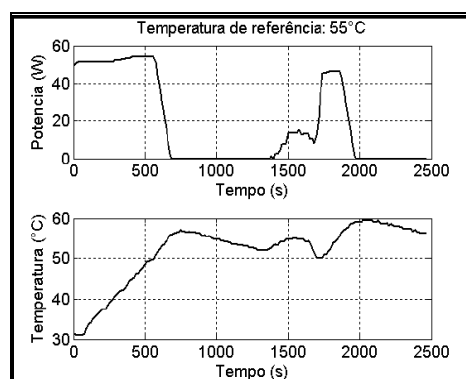


Figura 10. Sinal de controle e resposta do sistema, para o controlador Fuzzy, sujeito a uma variação de carga.

A saída apresenta oscilações e um grande sobre-sinal. De acordo com os resultados obtidos em simulação consegue-se uma melhora significativa aumentando o número de rótulos para as variáveis linguísticas de entrada. No entanto, isso não foi realizado na prática devido à intenção de embarcar o controlador em um microcontrolador, visto que o esforço computacional despendido por um controlador fuzzy, depende dentre outros fatores, do número de regras a serem analisadas.

Considerando, então como sendo satisfatório para os fins desse trabalho, o desempenho do controlador fuzzy implementado, passou-se à etapa de embarcar o controlador no microcontrolador. A metodologia adotada para embarcar o controlador é apresentada na seção 5. As figuras 11 e 12 trazem os gráficos referentes a ensaios realizados com o controlador embarcado, para a mesma sintonia do controlador usado nos ensaios mostrados nas figuras 9 e 10.

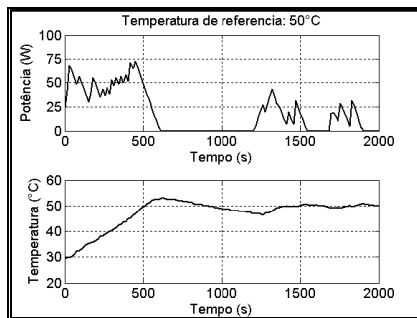


Figura 11. Sinal de controle e resposta do sistema, para o controlador Fuzzy embarcado.

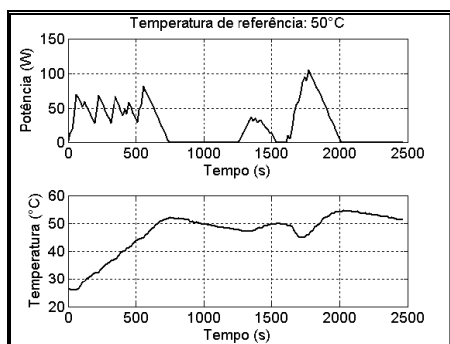


Figura 12. Sinal de controle e resposta do sistema, para o controlador Fuzzy embarcado, sujeitado a uma variação de carga.

Nota-se que o desempenho do controlador embarcado, não é inferior ao implementado no PC, sendo que o embarcado apresenta inclusive um sobre sinal menor, porque ele reage mais rapidamente à variação na saída da planta, pois não há atrasos decorrentes da transmissão de dados do PC para o PIC e vice-versa.

4 Controlador Neural

As Redes Neurais Artificiais (RNAs) são modelos matemáticos que se assemelham às estruturas neurais biológicas e que têm capacidade computacional adquirida por meio de aprendizado e generalização, (Haykin, 2001).

Neste trabalho, foi utilizada uma topologia de controle neural supervisionado, a qual consiste em uma rede neural artificial (RNA) Perceptron Multicamada, (Haykin, 2001), que foi treinada a partir dos dados de entrada e saída do controlador clássico PI. A RNA, depois de treinada com um conjunto de da-

dos experimentais, referentes a um determinado valor de temperatura arbitrário, foi utilizada como controlador, substituindo o controlador PI existente, como pode ser visto na figura 13. Isso foi feito para se observar algumas características da RNA, tais como capacidade de generalização e robustez.

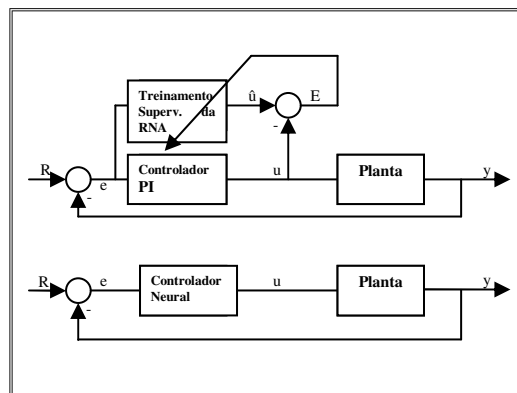


Figura 13: Controlador Neural

Nas seções 2 e 3, já se comentou sobre o fato de o controlador PI clássico atender aos requisitos deste sistema; porém, a utilização do neurocontrole, também é justificada, pela intenção de se embarcar o controlador neural em um microcontrolador.

4.1 Definição do controlador neural supervisionado a ser implementado.

Inicialmente, definiu-se uma arquitetura para a RNA: uma entrada (sinal de erro); uma camada intermediária com dois neurônios, que possuem função de ativação tangente hiperbólica; e uma saída (sinal de controle), com função de ativação linear. Em seguida, coletou-se um conjunto de dados, que foram obtidos realizando-se experimentos com variação do tipo degrau, com o controlador PI clássico, para as temperaturas 50°C e 60°C. Os dados foram divididos em dois grupos: um grupo de treinamento (50°C) e outro grupo de validação (60°C).

Então, após o treinamento em lote, utilizando o algoritmo de Levenberg-Marquadt, (Haykin, 2001), analisou-se o desempenho da RNA, simulando-a com os dados de validação, a fim de ratificar a arquitetura. Confirmada a topologia proposta e fixados os parâmetros livres - pesos, bias e taxa de aprendizagem (que foi adotada como fixa e definida por tentativa e erro), que renderam melhor desempenho - a rede neural foi implementada no sistema real substituindo o controlador PI. Em seguida, por meio da interface gráfica com o usuário, realizaram-se novos experimentos, alterando-se os valores de temperatura, a fim de verificar se o controlador neural aprendera realmente a dinâmica do controlador PI clássico e, além disso, se consegue repetir a lei de controle para valores diferentes dos de treinamento e validação, e também observar o comportamento da RNA para experimento de variação de carga.

4.2 Resultados obtidos com o controlador neural supervisionado

Os gráficos referentes aos ensaios na planta, considerando a RNA implementada no PC, são apresentados nas figuras 14 e 15.

Considerados satisfatórios, para os fins deste trabalho, o desempenho do controlador Neural implementado no PC, passou-se à etapa de implantação da RNA no microcontrolador, o que é detalhado na

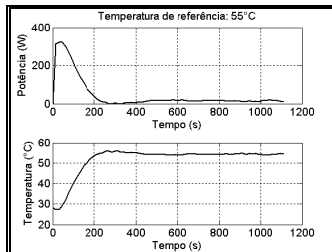


Figura 14: Sinal de controle e resposta do sistema, para o controlador Neural.

seção 5. As figuras 16 e 17 apresentam os gráficos referentes aos ensaios realizados para as mesmas especificações dos ensaios presentes nas figuras 14 e 15, agora usando controlador embarcado.

O desempenho do controlador embarcado é similar ao implementado no PC, comparando-se as figuras 14 e 15 com as figuras 16 e 17 percebe-se que no controlador embarcado, a atuação no sistema é realizada mais rapidamente, pois não há atrasos decorrentes da transmissão de dados do PC para o PIC e vice-

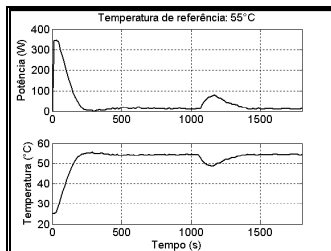


Figura 15: Sinal de controle e resposta do sistema, para o controlador Neural, sujeito a uma variação de carga.

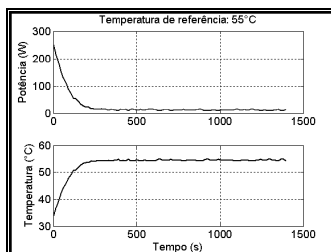


Figura 16: Sinal de controle e resposta do sistema, para o controlador Neural embarcado.

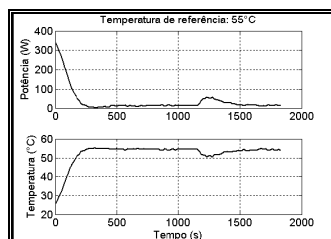


Figura 17: Sinal de controle e resposta do sistema, para o controlador Neural embarcado, sujeito a uma variação de carga.

versa.

5 Metodologia utilizada para embarcar os controladores

A metodologia utilizada neste trabalho para embarcar os controladores inteligentes do tipo fuzzy e neural está dividida em seis etapas:

- A primeira consiste em implementar os controladores no PC e sintonizá-los. Para o controlador fuzzy, nesta etapa, os conjuntos de entrada e saída são ajustados e é definida a base de regras. Para o controlador neural, nesta etapa, os pesos e bias são determinados através do algoritmo de treinamento.
- Na segunda etapa, fluxogramas ou esquemas equivalentes, são criados para auxiliar a programação.
- Na terceira etapa os controladores foram implementados, a partir dos esquemas ou fluxogramas, utilizando linguagem C adaptada para programação de microcontroladores PIC, (Pereira, 2003).
- Na quarta etapa foi realizada a otimização do programa em linguagem C para PIC, com o objetivo de reduzir o consumo de memória e o tempo de execução do programa. A otimização foi realizada por meio da depuração do programa, pois a memória e a velocidade de processamento são fatores críticos quando se trata de microcontroladores.
- Na quinta etapa, os controladores são simulados em ambiente virtual que emula o microcontrolador utilizado. A simulação foi feita através da inserção de estímulos nos pinos de entrada do microcontrolador emulado, para verificação dos valores de saída e comparação com a resposta, para a mesma entrada aplicada aos controladores desenvolvidos no PC.
- Por fim, na sexta etapa o programa é embarcado no microcontrolador. Também são realizados ensaios com a planta real e verificado o desempenho do controlador embarcado em relação ao controlador implementado no PC.

No decorrer das etapas listadas nessa seção, não se deve passar adiante sem que se tenha obtido êxito na etapa anterior. Se necessário, deve-se refazer etapas anteriores.

6 Conclusão

Os resultados comprovam ser válida a metodologia adotada para se embarcar os controladores implementados, em vista do êxito na tarefa de produzir sinais de controle adequados para atuar na planta ST.

O fato de implementá-los primeiramente em um computador pessoal, onde se dispõe de grande recursos computacionais, foi essencial, pois tornou-se possível a avaliação dos algoritmos de controle inteligente através de simulações, considerando as limitações inerentes ao microcontrolador, tais como: frequência de operação e consumo de memória, para então embarcar os controladores.

Embora os dois controladores, fuzzy e neural, tenham sido projetados em função do desempenho do controlador PI clássico, para a configuração adotada, apenas o Neural conseguiu reproduzir satisfatoriamente seu comportamento. O controlador fuzzy provavelmente só o conseguiria se aumentássemos os rótulos das variáveis linguísticas de entrada para cinco, elevando assim o número de regras para vinte e cinco ao invés das nove implementadas. Isso aumentaria drasticamente o custo computacional desse controlador, que mesmo com nove regras já era maior que o exigido pelo Neural.

Para esta aplicação em particular, por se ter um controlador clássico bem ajustado para fornecimento dos dados para treinamento, uma topologia da RNA simples e pouco complexa, apresentou melhor desempenho que o controlador fuzzy de nove regras, obtendo-se respostas coerentes para todos os valores de temperatura e reagindo bem à variação de carga.

Os controladores embarcados apresentaram-se como uma alternativa de baixo custo, tendo seu uso justificado em aplicações que exijam portabilidade, já que dispensam o uso de computadores para realização do controle. A pesquisa realizada nesse trabalho propiciou o desenvolvimento de uma metodologia simples para se embarcar controladores inteligentes.

Agradecimentos

Agradecemos à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pelos recursos financeiros para a realização deste trabalho. Agradecemos também ao Programa de Educação Tutorial de Engenharia Elétrica (PET-EE) da UFPA pela infra-estrutura cedida.

Referências Bibliográficas

- Ahmed, A. (2000). *Eletrônica de potência*, São Paulo: Prentice Hall.
- Carmo, E. A., Silva, O. F., Barreiros, J. A. L., Ferreira, A. O. (2006). Construção de Protótipo Didático: Uma Abordagem Diferenciada no Processo de Ensino/Aprendizagem de Engenharia, *Anais do XVI Congresso Brasileiro de Automática (CBA)*, Salvador-BA, pp. 3445 – 3450

Haykin, S. (2001). *Redes Neurais: Princípios e prática*. Porto Alegre: Bookman.

Paraskevopoulos, P. N. (1996). *Digital Control Systems*. Londres: Prentice Hall.

Pereira, F. (2003). *Microcontroladores PIC: Programação em C*, São Paulo: Érica

Shaw, I. S., Simões, M. G. (1999). *Controle e Modelagem Fuzzy*, São Paulo: Editora Edgard Blücher.

Silva, M. F., Pereira, P. S. and Regazzi, R. D. (2005). *Soluções Práticas de Instrumentação e Automação – Utilizando a Linguagem Labview*, Rio de Janeiro: Rogério Dias Regazzi.

Spooner, J. T., Maggiore, M., Ordóñez, R., Passino, K.M. (2002). *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques*, EUA: John Wiley & Sons.