
SISTEMA FUZZY COMPACTO EMBARCADO EM DSP E SUA APLICAÇÃO PARA CONTROLE V/F DE MOTORES DE INDUÇÃO

Marcelo Suetake*

Ivan Nunes da Silva*

Alessandro Goedel†

*USP/EESC/SEL
CP 359, CEP 13566-590, São Carlos - SP

†UTFPR
Av. Alberto Carazzai, 1640, Cornélio Procópio - PR

RESUMO

Embedded DSP-Based Compact Fuzzy System and its Application for Induction Motor V/F Control

Este trabalho apresenta uma metodologia de implementação de algoritmos com estratégias *fuzzy* para sistemas embarcados em processadores digitais de sinais, cujo propósito de aplicação consiste no controle escalar de motores de indução trifásicos. A estratégia de controle adotada reside no ajuste da amplitude e frequência (V/f) do sinal fundamental da tensão de alimentação do motor de indução, que foi modulado por largura de pulso aplicado a um inversor trifásico. Para tanto, o sistema de controle *fuzzy* foi integralmente embarcado em um processador digital de sinal empregando-se técnicas de simplificação que visam à redução dos requisitos de memória e custo computacional. O desempenho do controlador foi avaliado experimentalmente sob condições de variação de torque de carga aplicado ao eixo do motor de indução trifásico e referência de velocidade. Análises comparativas com as técnicas de controle PI e PID foram também realizadas com o propósito de validação da metodologia proposta.

PALAVRAS-CHAVE: controle *fuzzy*, motores de indução, sistema de tempo real, processador digital de sinal.

ABSTRACT

This paper presents a methodology for the implementation of embedded fuzzy system algorithms to be built in digital signal processors. For the purpose of this application, the technique was applied for induction motor scalar speed control. The adopted control strategy is to adjust the amplitude and frequency (V/f) of fundamental supply voltage signal of induction motor, which is achieved by a three-phase pulse width modulation inverter. The fuzzy control system was therefore entirely embedded in DSP by applying simplification techniques, which aim at computational cost and memory requirements reduction. The controller performance in relation to load torque and speed reference variations was evaluated by experimental tests. A comparative analysis with conventional PI and PID controllers was also achieved.

KEYWORDS: fuzzy control, induction motor, real time system, digital signal processor.

1 INTRODUÇÃO

Os Motores de Indução Trifásicos (MIT) são amplamente empregados em diversos setores industriais. Suas principais virtudes residem na simplicidade construtiva, robustez na operação, e baixo custo de aquisição e manutenção.

Diversas pesquisas têm sido publicadas no campo do con-

Artigo submetido em 31/05/2009 (Id.: 01010)

Revisado em 27/07/2009, 16/10/2009, 19/10/2009

Aceito sob recomendação do Editor Associado Prof. Enes Gonçalves Marra

trole vetorial e controle direto de torque, já que estes produzem melhores respostas dinâmicas (Bim, 2001; Cruz e Paredes, 2003; Karanayil *et alii*, 2007; Paladugu e Chowdhury, 2007; Singh *et alii*, 2007; Uddin *et alii*, 2007; Maiti *et alii*, 2008; Zidani *et alii*, 2008). Em contrapartida, o controle escalar (El-Saady *et alii*, 1994; Tae-Chon *et alii*, 2001; Islam *et alii*, 2005) apresenta uma estrutura mais simples, cujas principais características consistem no seu baixo custo de implementação e no erro de regime permanente reduzido. Portanto, aborda-se neste artigo o controle escalar V/f constante em virtude de sua vasta aplicação industrial, além de sua sedimentação conceitual na literatura (Krause *et alii*, 1995; Krishnan, 2001; Trzynadlowski, 2001).

A metodologia de controle PI é usualmente empregada na estratégia de controle escalar V/f de motores de indução trifásicos (Krishnan, 2001; Trzynadlowski, 2001). Entretanto, além do projeto de um controlador convencional solicitar o modelo matemático da planta do sistema, a dificuldade de se identificar os parâmetros precisos de um comportamento complexo, não-linear e variante no tempo, as quais são inerentes a uma planta real, pode tornar o processo de ajuste fino dos parâmetros dos controladores muito dispendioso (Dazhi *et alii*, 2004; Callai *et alii*, 2007). Adicionalmente a tais fatores, técnicas complementares como sistemas de identificação online e controle adaptativo são alternativas a serem exploradas com o propósito de se alcançar patamares superiores de eficiências, uma vez que os métodos convencionais são altamente sensíveis às variações paramétricas inerentes às operações de uma planta real e, em especial, dos motores elétricos.

Por outro lado, as metodologias de controle fundamentadas nos conceitos da teoria de conjuntos *fuzzy* apresentam habilidades de lidar com as não-linearidades, sendo que o desempenho do controle é menos afetado pela variação paramétrica do sistema. Além disso, as técnicas *fuzzy* se constituem por uma base de regras lingüísticas, a qual é composta mediante a exploração de aspectos qualitativos e de conhecimento especialista do problema (Lee, 1990; El-Saady *et alii*, 1994; Nobre e Palhares, 1997; Dos Santos Coelho *et alii*, 2003; Islam, Haider e Bashir Uddin, 2005; Ustun e Demirtas, 2008). Tais características dispensam a necessidade de um modelo matemático da planta de controle, proporcionando então um procedimento de projeto e dimensionamento do controlador mais simples de ser implementado, mesmo quando metodologias empíricas de ajuste fino de parâmetros são empregadas. Métodos alternativos fundamentados em sistemas inteligentes, tais como redes neurais artificiais, algoritmos genéticos e sistemas híbridos, os quais são também caracterizados pela habilidade de tratar tais problemas, têm também proporcionado resultados promissores (Bim, 2001; Deng e Tu, 2006; Goedel *et alii*, 2006; Oh *et alii*, 2006; Goedel *et alii*, 2007).

Diversas abordagens têm sido propostas recentemente no campo da aplicação de técnicas *fuzzy* em controle de motores de indução trifásicos. Entretanto, uma parcela considerável de trabalhos publicados, que fornecem resultados experimentais (El-Saady *et alii*, 1994; Islam, Haider e Bashir Uddin, 2005; Chun-Jung e Tien-Chi, 2007; Ustun e Demirtas, 2008), realizam a execução do processo de inferência *fuzzy* em computadores pessoais (PC). Em Ustun e Demirtas (2008), embora ambos o sistema de controle convencional PI e o sinal PWM que aciona o inversor sejam embarcados em um dsPIC da *Microchip*, o sistema *fuzzy* responsável pelo ajuste dos ganhos do controlador foi desenvolvido em uma plataforma PC.

De forma adversa ao desenvolvimento em plataforma PC, o *hardware* envolvido em sistemas embarcados usualmente apresenta um baixo poder computacional em termos de processamento e espaço de memória, visto que o propósito destes consiste justamente em realizar uma determinada tarefa em tempo real, utilizando-se para tanto de dispositivos de baixo custo. Por conseguinte, diferentes técnicas devem ser empregadas com o intuito de embarcar o mesmo sistema implementado na plataforma PC em um *hardware* de desempenho e recursos limitados. Adicionalmente, o sistema ainda dependerá dos requisitos de tempo real da aplicação e das características do *hardware* envolvido.

Geralmente, um sistema embarcado pode ser constituído em um microcontrolador, Processador Digital de Sinal (*Digital Signal Processor - DSP*), ou *hardware* reconfigurável (*Field Programmable Gate Array - FPGA*), conforme (Baturone *et alii*, 2008). Um circuito integrado dedicado a uma aplicação particular também pode ser empregado, conforme o trabalho proposto em Cao *et alii* (2006), cujo processamento fora realizado por um chip dedicado ao processo de inferência *fuzzy*.

Em Uddin e Hao (2007) um controlador neuro-*fuzzy* auto-ajustável foi integralmente desenvolvido em um DSP. Contudo, o algoritmo foi implementado utilizando a ferramenta computacional *Matlab/Simulink*[®], juntamente com o *toolbox real-time*, enquanto que a codificação do programa para o DSP foi gerado automaticamente por meio do *software Control-desk*, desprovendo de técnicas de otimização de códigos. Outro sistema *fuzzy* totalmente embarcado foi aplicado em Deng e Tu (2006) visando ajustar os parâmetros de um controlador PI de um Controle Direto de Torque (DTC) de motores de indução. Em Seul e Sung-su (2007) foi apresentada uma solução em *hardware* para uma Rede Neural Artificial (RNA) embarcada com um controlador Proporcional-Integral-Derivativo (PID). Em tal trabalho, os autores combinaram o DSP com FPGA com o propósito de distribuir as tarefas do algoritmo de controle.

Assim sendo, este trabalho propõe um método alternativo

para simplificação de um algoritmo de inferência *fuzzy* para ser embarcado em *hardwares* dedicados, cujo espaço de memória e recursos computacionais são limitados quando comparados a uma plataforma PC.

O método proposto explora as características simétricas das funções de pertinência com o intuito de armazenar sua parcela mínima em um vetor discreto. Desta forma, a partir de um simples e rápido cálculo de indexação, a função de pertinência inteira, contendo todos os seus termos lingüísticos, pode ser restaurada. Tal procedimento reduz significativamente o espaço de memória exigido para alocação das funções de pertinência. A metodologia proposta foi empregada para embarcar um algoritmo *fuzzy* para o controle em tempo real de velocidade de motores de indução trifásicos a partir da abordagem de manutenção da relação V/f constante (Sutake, 2007).

De forma complementar, o artigo também discute a análise de desempenho do sistema de controle *fuzzy* em relação à variação de torque de carga aplicado ao eixo do MIT e à referência de velocidade. Análises comparativas de desempenho dinâmico entre controladores PI e PID convencionais são tratadas com o propósito de validar o controlador *fuzzy* compacto. Cabe salientar que a avaliação de performance foi realizada com base nos resultados experimentais.

O algoritmo de controle *fuzzy* foi implementado no DSP TMS320F2812 da *Texas Instruments*. O dispositivo foi responsável por mensurar a velocidade angular do eixo do MIT por intermédio de um *encoder* óptico, executar o algoritmo de controle *fuzzy* e, finalmente, gerar os sinais PWM senoidais para o acionamento dos seis *Insulated-Gate Bipolar Transistors* (IGBTs) de um inversor trifásico.

A organização do presente artigo está estruturada como se segue. Na Seção 2, são apresentados os aspectos da estrutura do sistema de controle *fuzzy*, assim como os principais elementos da bancada de ensaios experimentais. Na Seção 3, descreve-se o sistema *fuzzy* compacto voltado às aplicações para sistemas embarcados. Na Seção 4, são discutidos os resultados experimentais relativos ao desempenho do controlador *fuzzy* de velocidade, sendo que análises comparativas com controladores PI e PID convencionais são também apresentadas. Finalmente, na Seção 5, as principais conclusões do trabalho são apresentadas.

2 ESTRUTURA DO SISTEMA DE CONTROLE V/f FUZZY DE VELOCIDADE

A metodologia proposta empregada para o controle de velocidade do MIT consistiu em manter a relação V/f constante, cujo objetivo reside na manutenção do fluxo no entreferro, o que garante a operação adequada da máquina. Caso a tensão

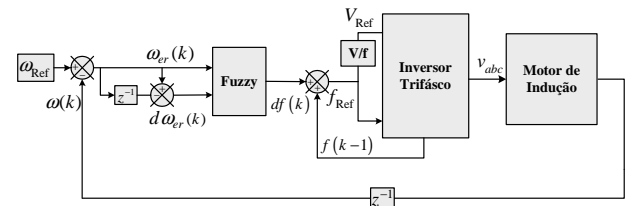


Figura 1: Diagrama de blocos do sistema de controle V/f *fuzzy* de velocidade.

de alimentação do MIT seja alterada sem o devido ajuste de frequência, ou vice-versa, a máquina pode operar na região de saturação de fluxo ou com o campo enfraquecido. O diagrama de blocos ilustrado na Figura 1 representa o esquema do sistema de controle *fuzzy* proposto.

Os pulsos gerados pelo *encoder* óptico incremental foram fornecidos aos canais do DSP TMS320F2812, o qual calcula a velocidade angular mediante a aferição da frequência dos pulsos de entrada. O sinal de velocidade do MIT foi comparado com a velocidade de referência ω_{Ref} , fornecendo ao bloco do controlador *fuzzy* as variáveis de entrada, ou seja, o sinal de erro $w_{er}(k)$ e a variação de erro de velocidade $d\omega_{er}(k)$, sendo esta determinada por meio da diferença entre a amostra atual e aquela anterior.

O algoritmo do sistema *fuzzy* processa as entradas de acordo com sua base de regras lingüísticas, produzindo-se em sua saída a variação de frequência $df(k)$ fundamental do sinal PWM senoidal. Em seqüência, tal sinal é adicionado ao valor da frequência anterior $f(k-1)$, resultando-se na frequência fundamental atual. A lógica de geração de sinais com modulação PWM senoidais, cuja frequência fundamental é proporcional à sua amplitude tal que se mantenha a relação V/f constante, foi codificada no DSP de maneira que os seis IGBTs do inversor trifásico sejam acionados. O sistema *fuzzy* embarcado foi desenvolvido levando-se em consideração a redução de memória e a otimização de código para um DSP de ponto fixo.

A estrutura do laboratório de pesquisa, ilustrada na Figura 2, consistiu de uma bancada de ensaios constituída basicamente por um MIT, um gerador de corrente contínua (GCC), um torquímetro e um *encoder* óptico incremental com resolução de 2000 pulsos por revolução.

A implementação do sistema de controle *fuzzy* foi desenvolvida no DSP de modelo TMS320F2812 (*Texas Instruments*), cujas principais características residem no elevado clock de 150 MHz, 16 canais de PWM, 12 conversores A/D e canais QEP (*Quadrature Encoder Pulse*) específicos para aplicações de *encoder*, tal como a medição de velocidade.

O gerador CC de 2 kW (Varimot) foi responsável pela im-

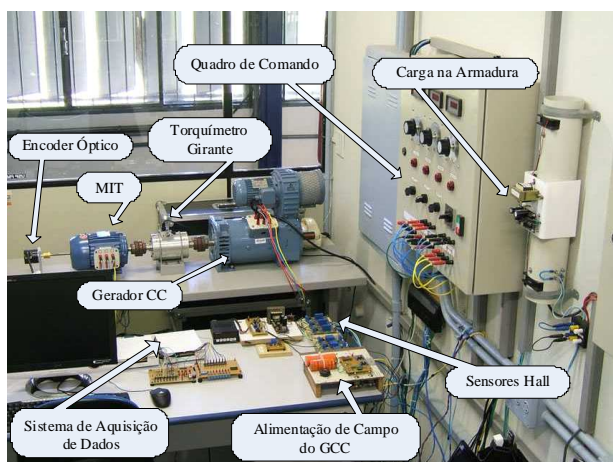


Figura 2: Estrutura da bancada de ensaios experimentais.

posição de carga no eixo do MIT, a qual pôde ser controlada mediante o ajuste do autotransformador conectado à fonte de alimentação do enrolamento de campo.

O MIT empregado neste trabalho consistiu no motor da WEG, linha *Standard* do tipo N de 4 pólos, 1 cv, 60Hz, 220/380V e conjugado nominal de 4,1 Nm. O acionamento do MIT foi realizado por inversor trifásico da *Semikron* (modelo SKS/26F/B6U+B6CI+E1CIF*2/10V6). A frequência de chaveamento foi ajustada para 10 kHz.

Embora o DSP utilizado (TMS320F2812) tenha um considerável espaço de memória *FLASH* de 128 KB, o método proposto procurou minimizar tanto o esforço computacional como a exigência de memória, que é requerida na alocação do código visando propósitos de aplicações em microcontroladores de baixo custo.

Na unidade QEP os pulsos do *encoder* óptico foram contados e armazenados em seu contador interno, a partir do qual a velocidade foi determinada conforme a seguinte expressão:

$$\omega_k = \left(\frac{x_k - x_{k-1}}{\Delta t} \right) \cdot \left(\frac{1}{4 \cdot R} \right) \cdot 60 \quad (rpm) \quad (1)$$

em que ω_k é a velocidade angular em rpm (revolução por minuto), x_k é o valor do contador de pulsos no instante k , Δt é o intervalo de tempo fixo pré-definido pelo projetista e R é a resolução do enconder (2000 pulsos por revolução), o qual foi multiplicado por 4 em virtude da propriedade da unidade QEP do DSP. Neste estudo, o valor de Δt foi definido como 20 ms, considerando-se a dinâmica de velocidade do motor. Portanto, a ação de controle do sistema *fuzzy* foi realizado a cada intervalo de 20 ms.

A geração dos sinais PWM, responsáveis pelo acionamento dos IGBTs do inversor trifásico da *Semikron*, foi realizado

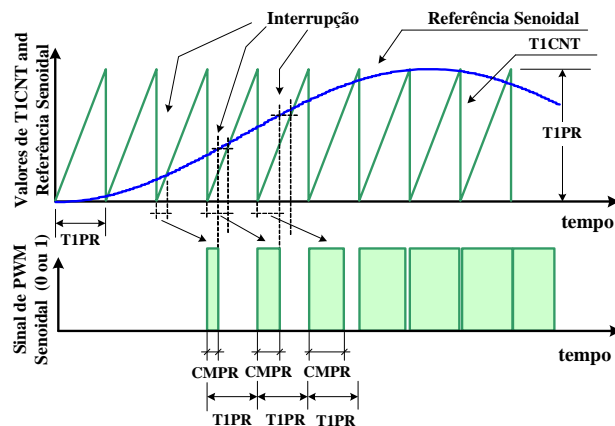


Figura 3: Geração de sinais PWM.

mediante a configuração de alguns registradores contidos no periférico da unidade *Event Manager* do DSP, conforme a Figura 3.

Uma portadora triangular foi definida por meio de um timer, cujo contador TICNT é incrementado com base na frequência de um oscilador interno. O contador é reinicializado no instante que seu valor atinge àquele do registrador TIPR, o qual define a frequência da portadora. O ciclo de trabalho do sinal PWM foi determinado a partir do valor contido no registrador CMPR. Enquanto o valor de TICNT for menor que de CMPR, a saída apresentará nível alto.

A técnica adotada para desenvolver a modulação PWM senoidal consistiu em aproveitar o sinal de interrupção, gerado no instante que TICNT se iguala a CMPR para executar a rotina de alteração do valor de CMPR para aquele proporcional à amplitude da senóide no próximo período, conforme ilustra a Figura 3. A senóide de referência foi armazenada na estrutura de lookup table no espaço de memória do DSP. Em virtude de sua frequência e amplitude serem proporcionais à amplitude e frequência fundamental da tensão de alimentação do motor, a relação V/f pôde ser adquirida mediante o controle desta variável (amplitude e frequência da senóide de referência).

3 O SISTEMA FUZZY EMBARCADO

O sistema *fuzzy* embarcado proposto neste artigo foi desenvolvido considerando-se a redução de exigência de memória e de esforço computacional visando aplicações de tempo real e de baixo custo. Entretanto, tais variáveis freqüentemente são inversamente relacionadas.

Para aplicações de tempo real em *hardware* que exigem o mínimo custo computacional, uma possível solução seria armazenar no espaço de memória uma matriz de relação entrada-

saída na estrutura de *lookup table*. Desta forma, não haveria necessidade de se efetuar nenhuma computação envolvida no sistema de inferência *fuzzy*, uma vez que tal processo é realizado por meio de uma simples indexação de tabela. Entretanto, este método demandaria um elevado espaço de memória à medida que se exige precisão na saída, o que pode proporcionar um aumento significativo do custo envolvido no projeto.

Por outro lado, em aplicações nas quais se desejam a mínima exigência de memória, o processo de inferência *fuzzy* pode ser realizado integralmente de forma *online*, evitando-se o armazenamento das funções de pertinência em memória. Contudo, o custo computacional tornaria elevado, deteriorando-se o desempenho de tempo real.

Conseqüentemente, em diversas aplicações, recomenda-se que os requisitos de memória e de esforço computacional sejam equilibrados a fim de visar uma estrutura que satisfaça os requisitos envolvidos com o custo de projeto e com o desempenho em tempo real frente a uma determinada aplicação.

Para o projeto de um sistema embarcado de inferência *fuzzy* compacto para o controle de motores de indução consideraram-se funções de pertinências triangulares simétricas, conforme ilustradas na Figura 4.

Todas as variáveis lingüísticas do sistema de controle *fuzzy* (erro de velocidade, variação do erro e variação de frequência) foram enquadradas em um universo de discurso comum com valores entre $[-1, 1]$. Conseqüentemente, por meio de um único conjunto de funções de pertinência, foi possível mapear simultaneamente as três variáveis do sistema *fuzzy*.

Os termos lingüísticos são descritos como se segue: “NG” é “Negativo e Grande”, “NM” é “Negativo e Médio”, “NP” é “Negativo e Pequeno”, “ZZ” é “Zero”, “PP” é “Positivo e Pequeno”, “PM” é “Positivo e Médio” e, finalmente, “PG” é “Positivo e Grande”.

O universo de discurso da variável lingüística “erro de velocidade” foi ajustado para o intervalo de $[-200, 200]$ rpm. Na seqüência, o sinal de erro de velocidade foi dividido por 200 na etapa que antecede o processo de fuzzificação com

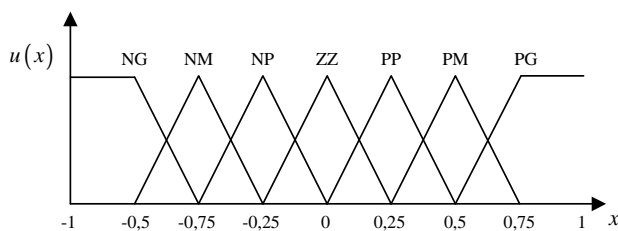


Figura 4: Funções de pertinências do sistema *fuzzy*.

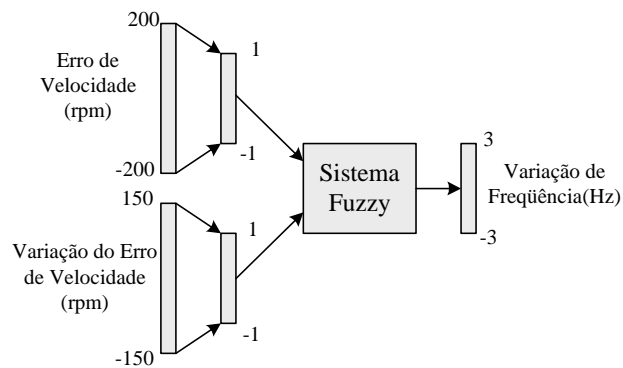


Figura 5: Normalização do universo de discurso.

o intuito de normalizar o intervalo para $[-1, 1]$, conforme ilustrado na Figura 5. De forma similar, o sinal da variável lingüística “variação do erro de velocidade”, cujo universo de discurso fora ajustado para o intervalo de $[-150, 150]$ rpm, foi dividido por 150 para enquadrá-lo ao intervalo de $[-1, 1]$. Por outro lado, a variável lingüística de saída “variação de frequência”, cujos valores de defuzzificação se enquadram no intervalo de $[-1, 1]$, foram multiplicados por 3 com o intuito de considerar o universo de discurso de $[-3, 3]$ Hz. Nos casos em que o valor da variável de entrada ser (em módulo) maior que o universo de discurso, o mesmo será saturado ao limite inferior ou superior, ou seja, -1 ou 1 .

Assim, a estratégia adotada neste trabalho procurou aproveitar as tais características qualitativas e intuitivas proporcionadas pelo uso da abordagem *fuzzy* com o intuito de modelar um sistema de controle simples e rápido de forma que o mesmo possa ser embarcado em microprocessadores de baixo custo.

As faixas de valores definidos neste processo foram determinadas por meio de análises do comportamento da planta mediante sucessivos ensaios experimentais, possibilitando, desta forma, a extração de informações relevantes para a modelagem do sistema *fuzzy*.

O ganho da variável lingüística “variação do erro de velocidade” foi inicialmente ajustado de acordo com a resposta dinâmica do MIT sob condições nominais. A partir das observações realizadas nos resultados experimentais, notou-se que o MIT partiu em aproximadamente 200 ms e atingiu a velocidade de regime por volta de 1800 rpm, sendo o mesmo acionado com tensão (220 V) e frequência (60 Hz) nominais. Neste contexto, considerando que a ação de controle foi realizada a cada 20 ms, espera-se uma variação máxima de velocidade de 180 rpm para tensão e frequência nominal. Embora esta informação seja imprecisa, visto que a variação de velocidade é dependente da carga, a mesma foi empregada para a definição do valor inicial da parametrização do

sistema *fuzzy*.

A variável lingüística “erro de velocidade” foi inicialmente ajustada considerando-se os aspectos qualitativos da planta, visto que o processo do ajuste de seu ganho é intuitivo. Por exemplo, o ganho de 200 rpm ou 300 rpm em seu universo de discurso forneceria um intervalo de $[-50, 50]$ rpm ou $[-75, 75]$ rpm para o termo lingüístico “ZZ” (Verificar Figura 4), os quais são valores razoáveis para serem considerados como “baixo erro de velocidade”.

Finalmente, a variável lingüística “variação de freqüência” também foi ajustada levando-se em consideração seus aspectos qualitativos. Quando o valor do ganho é configurado para 1 Hz, o sistema *fuzzy* variará a freqüência da tensão fundamental de alimentação para produzir a abrangência máxima de $[-0,5 \text{ Hz} (-30 \text{ rpm}); 0,5 \text{ Hz} (30 \text{ rpm})]$ na velocidade mecânica do MIT (de 4 pólos) a cada 20 ms. Cabe salientar que as considerações supracitadas consistem de cálculos aproximados que não consideram a dinâmica do processo e o carregamento da máquina. Neste aspecto, o principal objetivo do cálculo seria quantificar os valores e definir os aspectos qualitativos para o sistema *fuzzy*. Nota-se ainda que este parâmetro está relacionado com a resposta dinâmica do motor e com as oscilações de velocidade em torno da referência. Quanto maior o valor, mais rápido será a resposta do controlador; entretanto, maiores oscilações ou possíveis *overshoots* tendem também a aparecer na velocidade do MIT.

Deve-se ainda ser mencionado que todas as informações adquiridas no processo de ajuste dos parâmetros do controlador *fuzzy* foram baseadas em aspectos qualitativos e intuitivos, ao invés de valores precisos e quantitativos. Desta forma, a habilidade do controlador *fuzzy* para lidar com informações imprecisas simplifica o procedimento de ajuste. Mediante os sucessivos ensaios experimentais, o ajuste fino dos parâmetros iniciais do controlador *fuzzy* foram realizados visando à obtenção de melhores resultados.

Considerando que o DSP TMS320F2812 é um processador de ponto fixo, projetou-se o sistema de inferência *fuzzy* se baseando apenas em variáveis de tipo inteiro de 16 e 32 bits. Neste contexto, o universo de discurso, cujo intervalo abrange valores entre $[-1, 1]$, foi então enquadrado ao intervalo entre $[0, 4096]$ representado agora por valores de tipo inteiro.

Portanto, a conversão de valores em ponto flutuante no intervalo de $[-1, 1]$ das funções de pertinência para o intervalo de $[0, 4096]$ em ponto fixo pode ser realizado conforme a seguinte expressão:

$$x_{integer} = 4096 \cdot (x_{float} + 1)/2 \quad (2)$$

em que x_{float} é o valor definido no formato ponto flutuante e $x_{integer}$ é o valor convertido no formato ponto fixo, a ser

utilizado no sistema de inferência *fuzzy* embarcado.

A utilização de funções de pertinência triangulares e simétricas para todas as variáveis lingüísticas de entrada e de saída possibilitou o armazenamento de somente uma função rampa linear no espaço de memória, a qual se constitui de um vetor de 512 posições com valores de tipo inteiro e que variam entre 0 a 1000. Tais valores foram definidos levando-se em consideração a melhor relação de precisão/complexidade. A partir desta função rampa linear, possibilitou-se, portanto, a determinação de quaisquer retas paralelas ou antiparalelas pertencentes à função de pertinência, conforme ilustrado na Figura 6.

Para propósitos de elucidação, seja um valor arbitrário x_1 pertencente ao intervalo $[d, d + 511]$. O valor da função de pertinência pode ser obtido mediante o acesso do índice (*ind*) do vetor discreto armazenado em memória, o qual pode ser determinado por uma simples operação de subtração de x_1 em d posições, conforme sugere a seguinte expressão:

$$ind = x_1 - d \quad (3)$$

Procedendo-se de forma similar aos passos adotados no esquema anterior, é possível também determinar o valor da reta antiparalela ilustrada na Figura 7.

Para tanto, seja um valor x_1 pertencente ao intervalo $[d, d + 511]$. O cálculo do índice do vetor discreto, que retorna o valor equivalente ao de pertinência da reta antiparalela deslocada, pode ser efetuado como segue:

$$ind = (d + 511) - x_1 \quad (4)$$

Estendendo este raciocínio, torna-se então possível a obtenção do grau de pertinência de qualquer variável lingüística, conforme a ilustração da Figura 8. Nota-se que todas as retas podem ser representadas por um único vetor discretizado, mediante um simples cálculo de readequação de índice denotado pelas expressões (3) e (4).

Em relação à Figura 8, observa-se que os termos lingüísticos

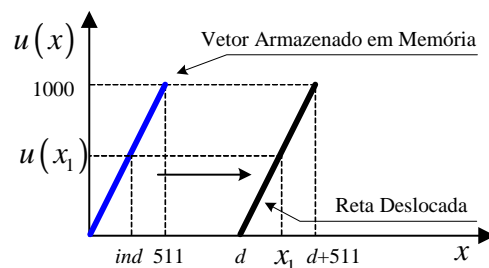


Figura 6: Determinação da reta paralela ao vetor discreto armazenado em memória.

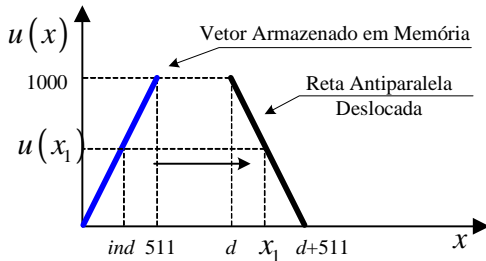


Figura 7: Determinação da reta antiparalela ao vetor discreto armazenado em memória.

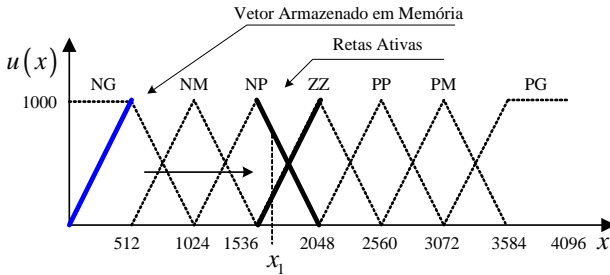


Figura 8: Determinação das retas ativas dos conjuntos fuzzy.

“NG” e “PG” consistem de funções trapezoidais. Em virtude de a base superior ser uma constante e igual a 1000, basta-se aplicar o seguinte teste de condição: “se x_1 for menor que 512 ou maior que 3584, então $u(x_1) = 1000$ ”.

Nota-se que as computações envolvidas na readequação de índices descritas em (3) e (4) consistem de simples instruções de adição e subtração, os quais podem ser facilmente realizados por microprocessadores.

As regras fuzzy podem ser implementadas com base no conhecimento especialista do processo, as quais são tratadas de forma lingüística na estrutura “se-então”. Em decorrência a este fato, dispensa-se de conhecimentos detalhados, precisos e até mesmo do modelo matemático representativo da planta de controle. As regras compostas neste trabalho foram baseadas nos estudos conduzidos em Lee (1990) e El-Saady *et alii* (1994), cujas primeiras três regras fuzzy são representadas como segue:

- Se (erro é NG) e (variação do erro é NG)
Então (variação da frequência é NG)
- Se (erro é NM) e (variação do erro é NG)
Então (variação da frequência é NG)
- Se (erro é PM) e (variação do erro é NG)
Então (variação da frequência é NP)
- (...)

A base de conhecimento de todas as regras aplicadas é descrita pela Tabela 1.

Tabela 1: Base de regras do sistema de controle fuzzy.

		Erro de Velocidade						
		NG	NM	NP	ZZ	PP	PM	PG
Variação do erro	NG	NG	NG	NG	NM	NM	NP	ZZ
	NM	NG	NM	NM	NP	NP	ZZ	PP
	NP	NM	NM	NP	NP	ZZ	PP	PP
	ZZ	NM	NP	NP	ZZ	PP	PP	PM
	PP	NP	NP	ZZ	PP	PP	PM	PM
	PM	NP	ZZ	PP	PP	PM	PM	PG
	PG	ZZ	PP	PM	PM	PG	PG	PG

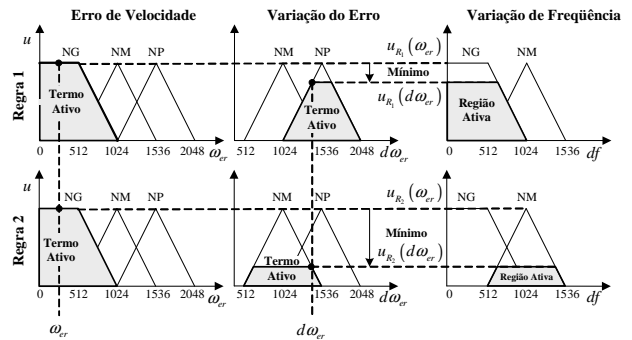


Figura 9: Implícção de Mamdani para duas regras ativas.

O operador de implícção de Mamdani, representado pela Figura 9, foi escolhido devido à sua maior simplicidade computacional. Neste exemplo genérico, ilustram-se as duas regras ativadas pelos valores de erro (ω_{er}) e variação do erro de velocidade ($d\omega_{er}$). Para a “Regra 1”, os termos lingüísticos ativados por ω_{er} e $d\omega_{er}$ são “NG” e “NP”, respectivamente. Aos valores de ω_{er} e $d\omega_{er}$ estão associados os seus graus de pertinência $u_{R_1}(\omega_{er})$ e $u_{R_1}(d\omega_{er})$. No processo de implícção de Mamdani, basta-se efetuar o mínimo entre tais valores e realizar o corte no respectivo termo lingüístico de saída associado à “Regra 1” (“NG”), resultando-se na região ativa destacada na Figura 9.

Em virtude da variação de erro ($d\omega_{er}$) ativar dois termos lingüísticos, tem-se então duas regras ativadas. A “Regra 2” consiste nos termos lingüísticos “NG” e “NM” para ω_{er} e $d\omega_{er}$, respectivamente. Pelo mesmo processo descrito no parágrafo anterior, a região ativa (“NM”) para a “Regra 2” é ilustrada em destaque pela Figura 9.

Após determinar as regiões ativas de todas as regras ativadas pelas variáveis de entrada ω_{er} e $d\omega_{er}$, efetuou-se o processo de agregação que consiste na união de todas as contribuições individuais de cada regra ativa, conforme ilustrado na Figura 10.

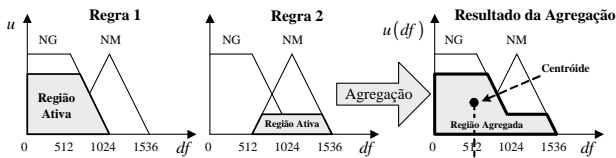


Figura 10: Agregação e defuzzificação.

Portanto, o valor de saída (Δf), o qual representa a variação da frequência fundamental da tensão de alimentação do MIT, pode ser obtido a partir da seguinte expressão:

$$\Delta f = \frac{\sum_{k=1}^N u(df_k) \cdot df_k}{\sum_{k=1}^N u(df_k)} \quad (5)$$

em que df_k representa o valor discreto de df e $u(df_k)$ é o grau de pertinência associado a cada df_k pertencente à região ativada.

Na seqüência, a variação de frequência foi convertida do formato de ponto fixo para aquele de ponto flutuante mediante a seguinte expressão:

$$x_{float} = \frac{2 \cdot x_{integer}}{4096} - 1 \quad (6)$$

A técnica aqui proposta para a simplificação do sistema *fuzzy* embarcado é conveniente para todas aquelas aplicações que não exijam funções de pertinências assimétricas. Complementarmente, o método pode ser generalizado para funções de pertinências Gaussianas sem nenhum custo computacional adicional, bastando-se apenas o armazenamento da metade de sua curva em um vetor discretizado em memória. Nota-se que tal característica torna a técnica vantajosa em virtude das funções Gaussianas requererem muito mais instruções para serem processadas que as funções lineares, devido à presença de funções exponenciais.

Outra vantagem do método proposto reside na capacidade de expandir o número de termos lingüísticos de cada variável, visto que todas as funções de pertinências são calculadas por indexação a partir de um único vetor alocado. A única exigência adicional de memória consiste apenas na codificação para identificar quais termos estarão ativos pela linha x_1 (Figura 8), a qual é insignificante comparando-se com a estrutura *fuzzy* restante, visto que as mesmas são dotadas por estruturas lógicas de programação simples do tipo “*if-then-else*”.

Por outro lado, uma quantidade adicional de memória será requerida para o armazenamento da tabela de regras, visto

que esta aumenta em concordância com o número de termos lingüísticos. Entretanto, esta propriedade é inerente a qualquer sistema *fuzzy*, a menos que regras desnecessárias sejam descartadas.

4 RESULTADOS EXPERIMENTAIS

Embora a estratégia de acionamento com modulação PWM senoidal na tensão propõe o fornecimento de correntes senoidais com frequência e amplitude fundamental controladas, a tensão eficaz (RMS) aplicada é reduzida em aproximadamente 86,3% do valor nominal. A informação supracitada foi comprovada experimentalmente, na qual foi observada que a tensão eficaz se reduziu a cerca de 190 Vrms em 60 Hz quando o inversor fora alimentada com 220 Vrms de tensão de linha, o qual fornecera 310 Vdc no barramento CC junto ao filtro capacitivo. Neste contexto, procurou-se ajustar a relação V/f para regiões de operação do motor de indução limitada de 20% a 90% da velocidade nominal.

A performance do sistema de controle *fuzzy* proposto foi então avaliada mediante os seguintes experimentos: i) resposta ao degrau de referência, ii) resposta à rampa de referência e, finalmente, iii) resposta ao degrau de torque aplicado ao eixo do motor de indução. Análises comparativas entre controladores PI e PID também fizeram parte do estudo apresentado neste artigo.

Os controladores convencionais PI e PID discretos foram implementados no DSP de acordo com as seguintes expressões:

$$f(k) - f(k-1) = q_0 \omega_{er}(k) + q_1 \omega_{er}(k-1) + q_2 \omega_{er}(k-2) \quad (7)$$

$$q_0 = K_p \left(1 + \frac{T_D}{T_0} \right) \quad (8)$$

$$q_1 = -K_p \left(1 + 2 \frac{T_D}{T_0} - \frac{T_0}{T_I} \right) \quad (9)$$

$$q_2 = K_p \frac{T_D}{T_0} \quad (10)$$

em que K_p é o ganho proporcional, T_I é a constante de integração, T_D é a constante de tempo derivativa, e T_0 é o período de amostragem (20 ms). O controlador PI foi ajustado para $K_p = 0,001$, $T_I = 0,002$ e $T_D = 0$; enquanto que os parâmetros do PID foram configurados para $K_p = 0,020$, $T_I = 0,031$ e $T_D = 0,001$. Considerando-se que a dinâmica mecânica do motor predomina sobre as influências elétricas, a equação que descreve o comportamento mecânico foi levada em consideração nos procedimentos de sintonia do PI e PID, os quais foram realizados pelo método de *root locus* empregando-se a ferramenta computacional *Matlab*[®]. Em seguida, o ajuste fino dos parâmetros foi executado por meio do comportamento observado durante os resultados experimentais.

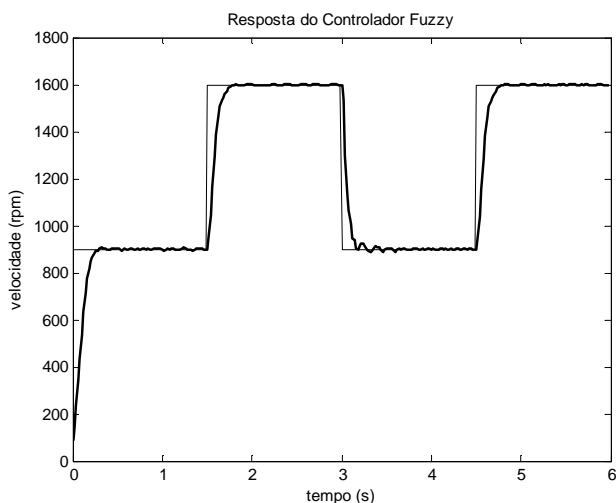


Figura 11: Resposta do controlador *fuzzy* frente aos degraus de referência de velocidade.

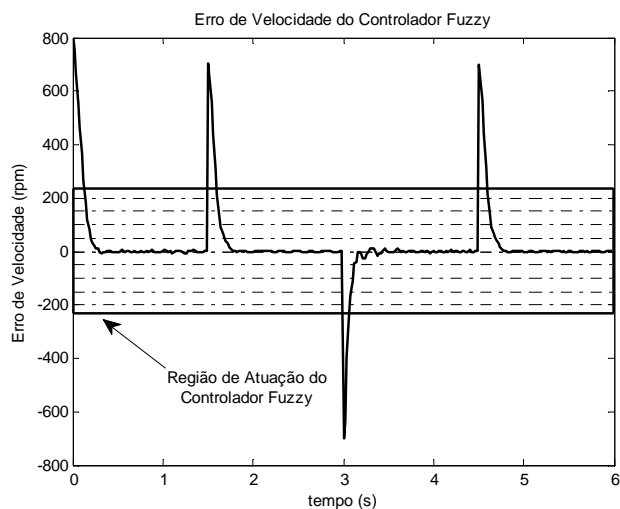


Figura 12: Erro de velocidade do controlador *fuzzy* para os ensaios de degraus de subida e de descida.

4.1 Resposta ao Degrau de Referência

Neste ensaio experimental, partiu-se o motor de indução com a referência de velocidade de 900 rpm, a qual foi alterada instantaneamente para 1600 rpm em 1,5 segundos. Visando a análise do desempenho do controlador *fuzzy* frente ao degrau de descida, a referência foi novamente reduzida para 900 rpm em 3 segundos. O torque de carga aplicado no eixo do motor de indução varia de aproximadamente 2,3 Nm à velocidade de 900 rpm para 4,0 Nm à velocidade 1600 rpm. Tais variações ocorrem em virtude do torque resistente produzido pelo gerador CC ser linear com a velocidade.

A resposta do controlador *fuzzy* frente ao degrau de subida e degrau de descida do valor da referência é ilustrada na Figura 11. A velocidade do motor de indução foi adquirida a cada intervalo de 20 ms. O erro encontrado entre a velocidade do MIT e a referência de velocidade é detalhado na Figura 12. As linhas horizontais se referem aos centros dos termos lingüísticos, as quais representam os valores máximos relativos às funções de pertinência triangulares e trapezoidais. A primeira e a última linha horizontal se referem aos limites do universo de discurso.

A Figura 13 apresenta em detalhes as oscilações de erro de velocidade proporcionado pelo sistema de controle *fuzzy* proposto. Notam-se amplitudes inferiores a 10 rpm (pico-a-pico) tanto no degrau de subida como no degrau de descida da referência.

A frequência de alimentação do MIT é ilustrada na Figura 14. Observa-se que há um incremento repentino no valor da frequência no instante do degrau de subida e de descida frente à referência de velocidade.

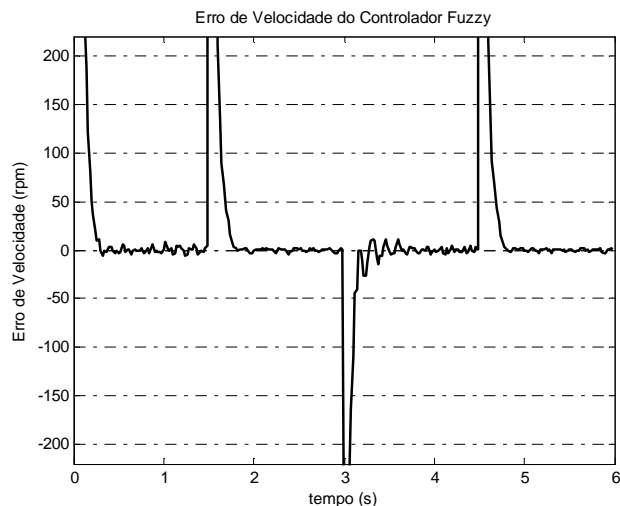


Figura 13: Erro de velocidade do controlador *fuzzy* para os ensaios de degraus de subida e de descida (detalhes).

No sentido de proporcionar uma resposta dinâmica mais rápida, a frequência de acionamento foi ajustada para o valor de referência (traduzido de rpm para Hz) enquanto o erro for maior que o universo de discurso. Neste contexto, o sistema de controle *fuzzy* foi responsável apenas pelo ajuste fino do erro. A mesma técnica foi empregada para os controladores PI e PID, os quais serão avaliados nas seções seguintes.

4.2 Resposta à Rampa de Referência

Neste experimento, partiu-se o MIT com uma referência de velocidade de 900 rpm, aumentando-a linearmente à 1600 rpm em 1,5 segundos. Na seqüência, a mesma foi diminuída

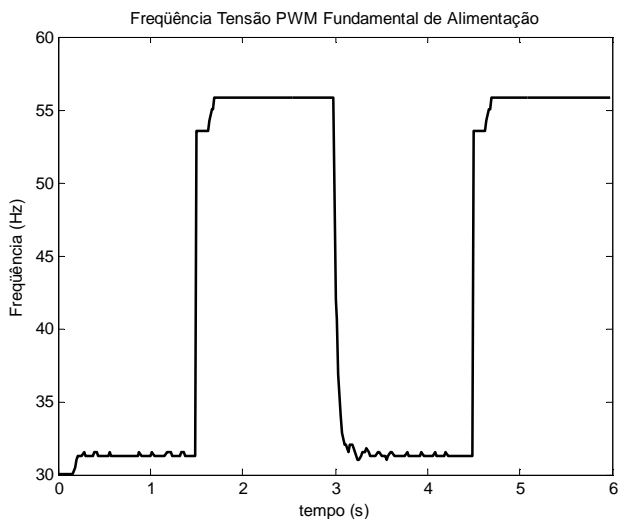


Figura 14: Frequência da tensão PWM fundamental de alimentação.

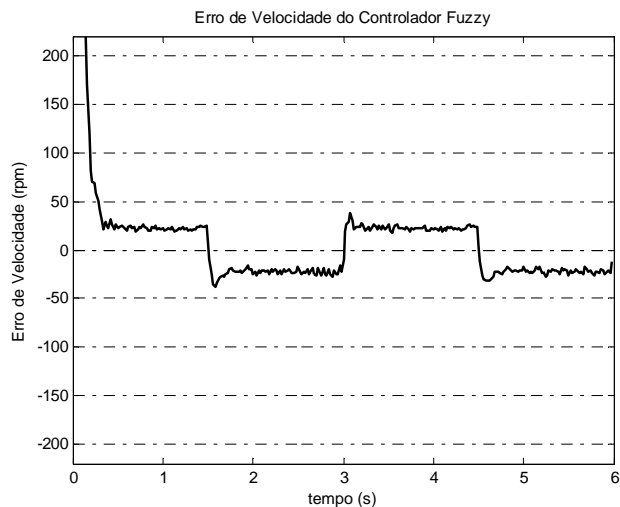


Figura 16: Erro de velocidade do controlador *fuzzy* para os ensaios de rampa de subida e de descida.

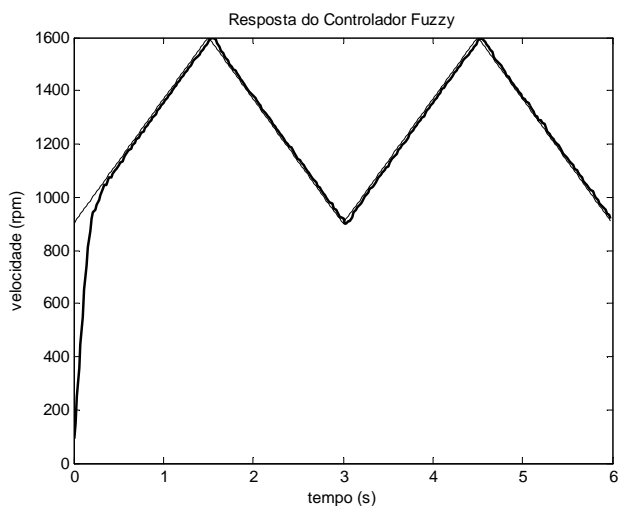


Figura 15: Resposta do controlador *fuzzy* frente às rampas de referência de velocidade.

para 900 rpm em 3 segundos. O torque de carga aplicado ao eixo do motor de indução variou linearmente de 2,3 Nm em 900 rpm à 4,0 Nm em 1600 rpm. A Figura 15 ilustra o desempenho do controlador *fuzzy* com relação à rampa de referência.

O erro entre a velocidade no eixo do MIT e a sua referência é detalhado na Figura 16. Observa-se que o erro correspondente durante todo o ensaio foi de aproximadamente 20 rpm.

Na Figura 17 ilustra-se a frequência de alimentação do MIT neste experimento.

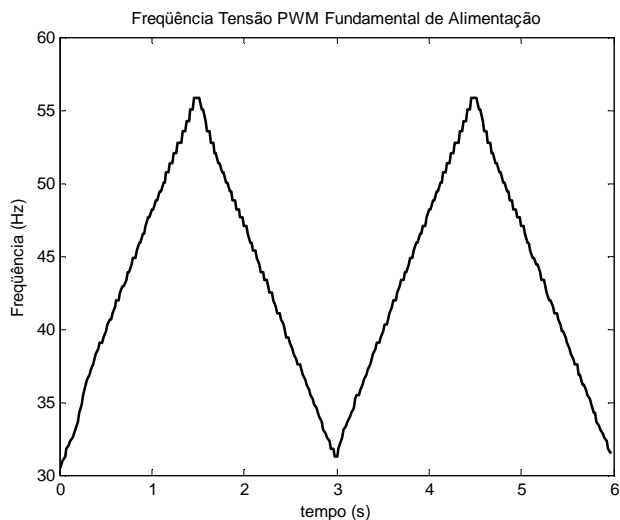


Figura 17: Frequência da tensão PWM fundamental de alimentação.

4.3 Resposta aos Degraus de Torque

O experimento para a avaliação do sistema de controle *fuzzy* foi agora realizado partindo-se o MIT com a referência de velocidade de 1600 rpm e, após alguns segundos, aplicou-se manualmente um torque de carga de 0,5 Nm à 4,5 Nm por meio do ajuste do auto-transformador conectado na fonte de alimentação do enrolamento de campo. A Figura 18 ilustra o torque de carga aplicado no eixo do MIT ao longo do ensaio experimental.

A resposta do controlador *fuzzy* em relação a este torque de carga aplicado ao eixo do motor é apresentada na Figura 19.

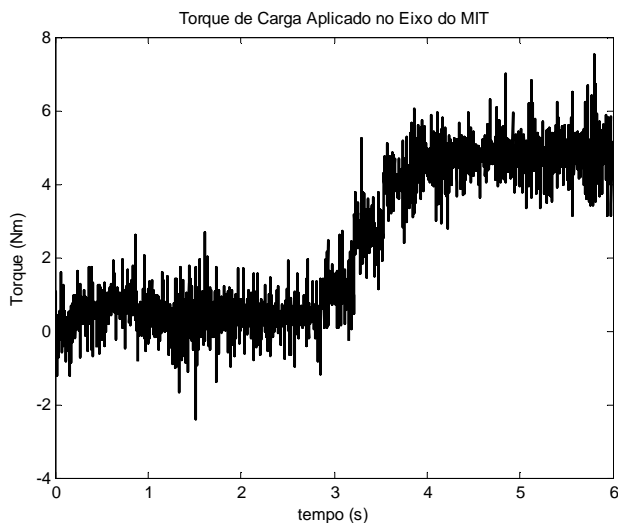


Figura 18: Torque de carga aplicado ao eixo do MIT.

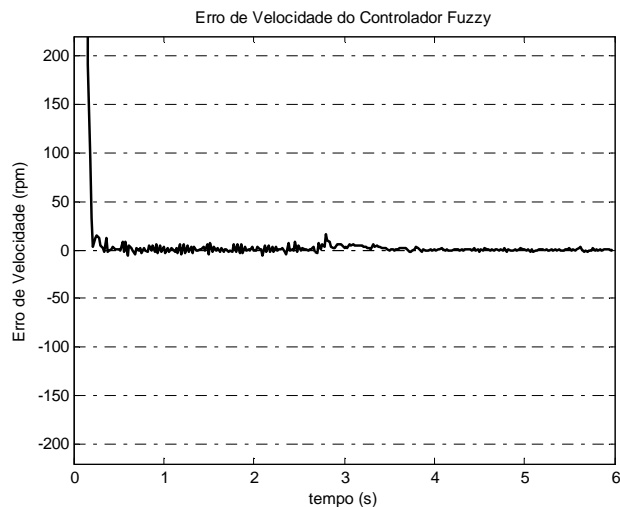


Figura 20: Erro de velocidade do controlador *fuzzy* para o ensaio de degrau de torque de carga.

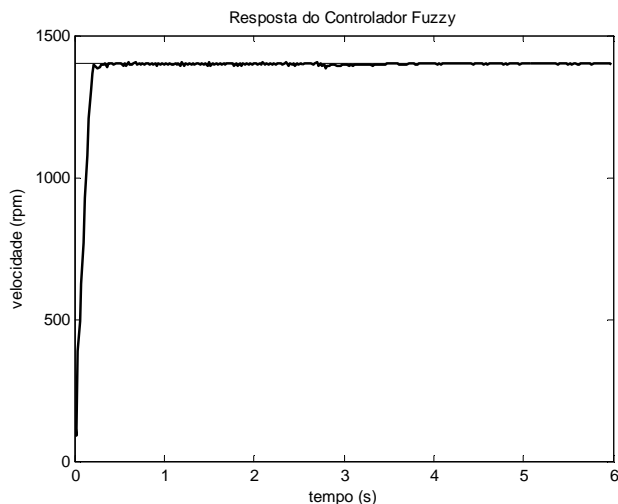


Figura 19: Resposta do controlador *fuzzy* frente ao degrau de torque de carga.

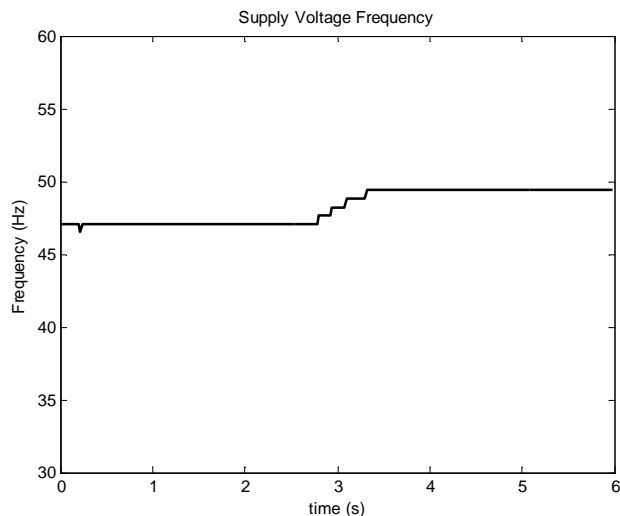


Figura 21: Freqüência da tensão PWM fundamental de alimentação.

Cabe-se salientar que a tendência do MIT foi reduzir a velocidade no instante em que o torque de carga fora aplicado. Entretanto, o sistema *fuzzy* embarcado foi capaz de corrigir o erro de velocidade mediante o incremento na freqüência da tensão fundamental de acionamento. O gráfico que detalha o desempenho da ação de correção do controlador *fuzzy* é ilustrado na Figura 20. Observa-se que, durante todo o ensaio experimental, o erro de velocidade apresentou valores em módulo menores que 10 rpm.

A Figura 21 ilustra a freqüência da componente fundamental da tensão de alimentação do MIT.

4.4 Análise Comparativa com Controles Convencionais PI e PID

Nesta seção serão apresentados os resultados de análises comparativas entre o desempenho do sistema de controle *fuzzy* proposto e os controladores convencionais PI e PID a fim de efetuar uma validação da técnica. Para tanto, experimentos que constituem de variações de degrau e rampa de referência de velocidade, assim como da inserção de degrau de torque de carga, foram realizados visando análises de performance do método proposto.

A Figura 22 ilustra a resposta de velocidade de cada contro-

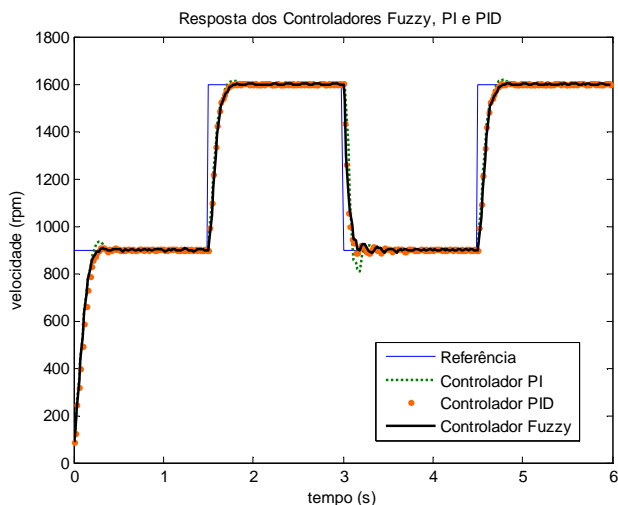


Figura 22: Respostas dos controladores *fuzzy*, PI e PID frente aos degraus de referência de velocidade.

lador. De acordo com os resultados, o controlador *fuzzy* e o controlador PID eliminaram o *overshoot* e o *ripple* de velocidade com relação àquela resposta produzida pelo controlador PI.

Cabe-se também mencionar que quando o ganho do controlador PI fora ajustado para uma melhor performance em ensaios de degraus de subida e descida (procurando-se respostas mais rápidas com menores *overshoots*), o desempenho em ensaios de rampa de referência se tornou deteriorado, apresentando-se maiores desvios. Neste contexto, os parâmetros do controlador PI foram ajustados empiricamente de maneira que a performance pudesse apresentar resultados com baixo erro para ambos os experimentos. O torque de carga aplicado no eixo do MIT varia entre aproximadamente 2,3 Nm (quando a velocidade é 900 rpm) e 4,0 Nm (a 1600 rpm).

O desempenho dos controladores *fuzzy*, PI e PID com relação ao ensaio de rampa de referência são ilustrados na Figura 23.

Nota-se que os resultados experimentais de todos os controladores refletem graficamente desempenhos dinâmicos bem similares. Neste sentido, com o intuito de avaliar a performance com valores precisos, o erro relativo médio (Erro) e o desvio padrão (STD) com relação à velocidade de referência foi portanto calculado, conforme destacado na Tabela 2.

Os resultados experimentais ilustrados na Figura 24 foram realizados partindo-se o MIT à referência de velocidade de 1400 rpm e, em seguida, aplicou-se um torque de carga de 0,5 Nm e a 4,5 Nm. A Figura 25 ilustra em detalhes as respostas dos controladores *fuzzy*, PI e PID.

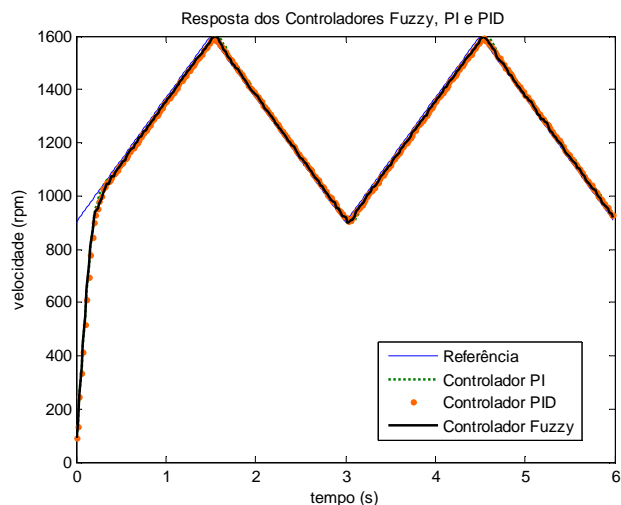


Figura 23: Respostas dos controladores *fuzzy*, PI e PID frente às rampas de referência de velocidade.

Tabela 2: Tabela de resultados comparativos entre os desempenhos dos controladores *fuzzy*, PI e PID.

	Referência Degrau		Referência Rampa	
	Erro (%)	STD (%)	Erro (%)	STD (%)
PI	4,18	13,6	3,45	9,72
PID	3,90	13,0	3,00	10,1
<i>fuzzy</i>	4,07	13,2	2,76	9,82

Ressalta-se, no entanto, que a variação do torque de carga aplicado no eixo do MIT foi realizada de maneira manual,

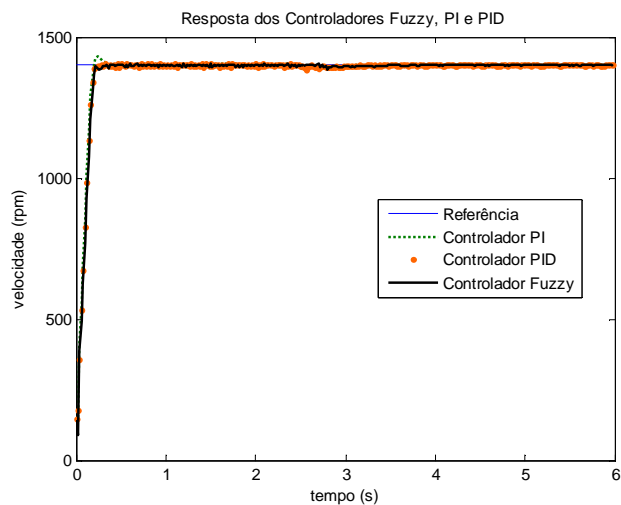


Figura 24: Respostas dos controladores *fuzzy*, PI e PID frente ao degrau de torque de carga.

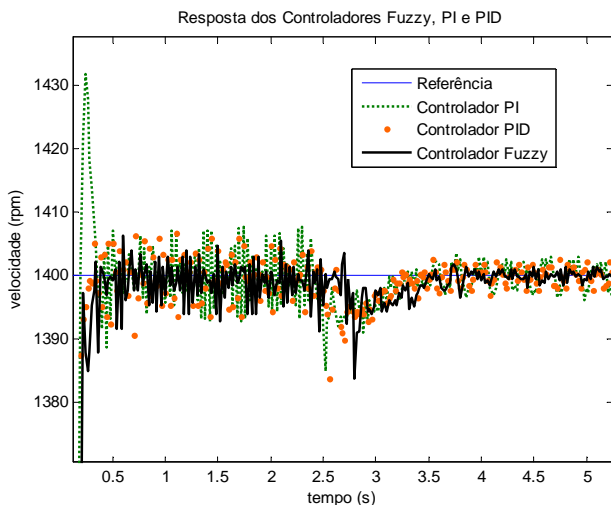


Figura 25: Respostas dos controladores *fuzzy*, PI e PID frente ao degrau de torque de carga (detalhes).

por meio da variação da tensão do retificador monofásico conectado ao enrolamento de campo do gerador CC. Neste contexto, não foi possível aplicar perfis de degrau de carga exatamente iguais (mesmo instante de aplicação, magnitude e taxa de variação) para cada experimento. Conseqüentemente, a análise comparativa com valores precisos deixou de ser realizada, pois os mesmos poderiam estar expressando essas nuances ocasionadas pela aplicação manual de carga. Entretanto, graficamente, foi possível observar que todos os controladores foram capazes de corrigir o erro de velocidade decorrente da variação de torque de carga.

5 CONCLUSÃO

Os resultados experimentais relativos ao desempenho do sistema de controle *fuzzy* de velocidade foram considerados satisfatórios, os quais apresentaram robustez na variação de carga e acompanhamento às referências de velocidades impostas no conjunto de experimento deste trabalho. Considera-se para tanto as aplicações direcionadas ao escopo da metodologia de controle V/f com acionamento PWM senoidal, na qual não se exige uma elevada resposta dinâmica e precisa. Entretanto, ressalta-se que em função das diferentes condições de operações, característica da carga acoplada ao eixo, dinâmica do motor e tipos de motores, o reajuste dos parâmetros do controlador *fuzzy*, incluindo o formato e quantidade de funções de pertinências, pode ser necessário para garantir o desempenho desejado. Portanto, a técnica proposta consiste de um método alternativo, de propósito genérico, para embarcar um sistema de inferência *fuzzy* em *hardware*, cujo poder de processamento e de espaço de memória são limitados quando comparados com uma plataforma PC.

A análise comparativa com o controlador PI convencional demonstrou que o método *fuzzy* proposto proporcionou melhores resultados no quesito da supressão de *overshoots* e *ripples* de velocidade. Por outro lado, comparando-se com o controlador PID, nota-se que o sistema *fuzzy* apresentou comportamento bem similar.

Finalmente, o sistema de inferência *fuzzy* embarcado constituiu-se de uma arquitetura compacta que exige um espaço reduzido de memória (aproximadamente $6 \text{ KB} \times 16$), sem que o custo computacional adicional interfira nos resultados práticos. Adicionalmente, a mesma metodologia pode ser empregada para funções de pertinência Gaussianas simétricas, exigindo-se esforço computacional semelhante, sendo ideal para aplicações com superfície de decisão *fuzzy* mais suaves. Além disso, uma vez que o método proposto determina as funções de pertinência por indexação, possibilita-se portanto a implementação de uma maior quantidade de termos lingüísticos.

APÊNDICE

O *hardware* utilizado neste trabalho pode ser também usado em diversas outras aplicações práticas. Relatam-se neste breve apêndice algumas notas de cunho experimental a fim contribuir em seus procedimentos de implementação.

Em suma, a documentação técnica da *Texas Instruments* é bem abrangente e apresenta uma grande diversidade de *applications notes*. Entretanto, para realizar a programação em linguagem C, algumas bibliotecas de definições de cabeçalhos são essenciais. Neste contexto, recomendam-se as seguintes documentações técnicas: SPRC097, SPRS174N, SPRU065, SPRU078 e SPRU509C, as quais podem ser diretamente obtidas a partir da página da *Texas Instruments* (<http://www.ti.com/>). Adicionalmente, em Suetake (2007), apresentam-se os detalhes e os códigos fontes da programação embarcada para o DSP TMS320F2812.

Outra documentação relevante é a SPRC087, que fornece as bibliotecas de funções (IQmath) para manipulação de ponto flutuante voltada aos processadores de ponto fixo. Como não se utiliza codificação binária definida pelo padrão IEEE, as operações básicas de ponto fixo são suficientemente capazes de trabalhar no formato de ponto flutuante IQ, melhorando o desempenho do processador.

AGRADECIMENTOS

Os autores agradecem à FAPESP (Processos 03/11353-0, 05/58404-3 e 06/56093-3) e CNPq (Processos 142326/2005-4 e 474290/2008-5) pelos auxílios financeiros concedidos. Agradecimentos especiais também aos revisores pelas suas valiosas sugestões e recomendações.

REFERÊNCIAS

- Baturone, I., Moreno-Velo, F. J., Blanco, V., *et alii* (2008). Design of Embedded DSP-Based Fuzzy Controllers for Autonomous Mobile Robots. *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 928-936.
- Bim, E. (2001). Fuzzy optimization for rotor constant identification of an indirect FOC induction motor drive. *IEEE Transactions on Industrial Electronics*, vol. 48, pp. 1293-1295.
- Callai, T. C., Dos Santos Coelho, L. e Coelho, A. A. R. (2007). Controle nebuloso adaptativo por modelo de referência: Projeto e aplicação em sistemas não-lineares. *Controle e Automação*, vol. 18, pp. 479-489.
- Cao, Q., Lim, M. H., Li, J. H., *et alii* (2006). A context switchable fuzzy inference chip. *IEEE Transactions on Fuzzy Systems*, vol. 14, pp. 552-567.
- Chun-Jung, C. e Tien-Chi, C. (2007). Speed Sensorless of an Induction Motor Using Self-tuning Fuzzy Identification. *Second International Conference on Innovative Computing, Information and Control (ICICIC)*, pp. 398-398.
- Cruz, P. P. e Paredes, J. P. S. (2003). Artificial intelligence applications in direct torque control. *The Fifth International Conference on Power Electronics and Drive System (PEDS)*, vol. 2, pp. 1208-1212.
- Dazhi, W., Renyuan, T., Hui, J., *et alii* (2004). Sensorless-speed control strategy of induction motor based on artificial neural networks. *Fifth World Congress on Intelligent Control and Automation (WCICA)*, vol. 5, pp. 4467-4471.
- Deng, J. e Tu, L. (2006). Improvement of Direct Torque Control Low-speed Performance by Using Fuzzy Logic Technique. *IEEE International Conference on Mechatronics and Automation*, pp. 2481-2485.
- Dos Santos Coelho, L., Da M. Almeida, O. e Coelho, A. A. R. (2003). Projeto e estudo de caso da implementação de um sistema de controle nebuloso. *Controle e Automação*, vol. 14, pp. 20-29.
- El-Saady, G., Sharaf, A. M., Makky, A., *et alii* (1994). A high performance induction motor drive system using fuzzy logic controller. *7th Mediterranean Electrotechnical Conference*, vol. 3, pp. 1058-1061.
- Goedtel, A., Da Silva, I. N. e Serni, P. J. A. (2006). Uma abordagem neural para estimativa de conjugado em motores de indução. *Controle e Automação*, vol. 17, pp. 364-380.
- Goedtel, A., Silva, I. N. d. e Serni, P. J. A. (2007). Load torque identification in induction motor using neural networks technique. *Electric Power Systems Research*, vol. 77, pp. 35-45.
- Islam, N., Haider, M. e Bashir Uddin, M. (2005). Fuzzy logic enhanced speed control system of a VSI-fed three phase induction motor. *2nd International Conference on Electrical and Electronics Engineering*, pp. 296-301.
- Karanayil, B., Rahman, M. F. e Grantham, C. (2007). Online stator and rotor resistance estimation scheme using artificial neural networks for vector controlled speed sensorless induction motor drive. *IEEE Transactions on Industrial Electronics*, vol. 54, pp. 167-176.
- Krause, P. C., Wasynczuk, O. e Sudhoff, S. D. (1995). *Analysis of Electric Machinery*. New York, IEEE Press.
- Krishnan, R. (2001). *Electric Motor Drives - Modeling, Analysis, and Control*. Upper Saddle River, New Jersey, Prentice Hall, inc.
- Lee, C. C. (1990). Fuzzy logic in control systems: fuzzy logic controllers - Part I and II. *IEEE Transactions on Systems, Man and Cybernetics*, vol. 20, pp. 404-435.
- Maiti, S., Chakraborty, C., Hori, Y., *et alii* (2008). Model reference adaptive controller-based rotor resistance and speed estimation techniques for vector controlled induction motor drive utilizing reactive power. *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 594-601.
- Nobre, F. S. M. e Palhares, A. G. B. (1997). Informações Quantitativas e Qualitativas no Projeto de Controladores Nebulosos. *Controle e Automação*, vol. 8, pp. 77-93.
- Oh, W. S., Cho, K. M., Kim, S., *et alii* (2006). Optimized neural network speed control of induction motor using genetic algorithm. *International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*, pp. 1377-1380.
- Paladugu, A. e Chowdhury, B. H. (2007). Sensorless control of inverter-fed induction motor drives. *Electric Power Systems Research*, vol. 77, pp. 619-629.
- Seul, J. e Sung-su, K. (2007). Hardware Implementation of a Real-Time Neural Network Controller With a DSP and an FPGA for Nonlinear Systems. *IEEE Transactions on Industrial Electronics*, vol. 54, pp. 265-271.
- Singh, B., Bhuvaneswari, G. e Garg, B. (2007). Harmonic mitigation in AC-DC converters for vector controlled induction motor drives. *IEEE Transactions on Energy Conversion*, vol. 22, pp. 637-646.

- Suetake, M. (2007). *Implementação de Sistemas Inteligentes em Processadores Digitais para Controle de Máquinas Elétricas*, Dissertação de Mestrado, Universidade de São Paulo, São Carlos
- Tae-Chon, A., Yang-Won, K., Hyung-Soo, H., *et alii* (2001). Design of neuro-fuzzy controller on DSP for real-time control of induction motors. *IFSA World Congress and 20th NAFIPS International Conference*, vol. 5, pp. 3038-3043.
- Trzynadlowski, A. M. (2001). *Control of Induction Motors*, Academic Press.
- Uddin, M. N. e Hao, W. (2007). Development of a Self-Tuned Neuro-Fuzzy Controller for Induction Motor Drives. *IEEE Transactions on Industry Applications*, vol. 43, pp. 1108-1116.
- Uddin, M. N., Huang, Z. R. e Chy, M. I. (2007). A simplified self-tuned neurofuzzy controller based speed control of an induction motor drive. *IEEE Power Engineering Society – General Meeting*, pp. 1-8.
- Ustun, S. V. e Demirtas, M. (2008). Optimal tuning of PI coefficients by using fuzzy-genetic for V/f controlled induction motor. *Expert System with Application*, vol. 34, pp. 2714-2720.
- Zidani, F., Diallo, D., Benbouzid, M. E. H., *et alii* (2008). A fuzzy-based approach for the diagnosis of fault modes in a voltage-fed PWM inverter induction motor drive. *IEEE Transactions on Industrial Electronics*, vol. 55, pp. 586-593.