

Boletim Técnico da Escola Politécnica da USP
Departamento de Engenharia de Computação e
Sistemas Digitais

ISSN 1413-215X

BT/PCS/0211

Alternativas de Implementação de
Sistemas Nebulosos em Hardware

Marcos Alves Predebon
Marco Túlio Carvalho de Andrade

1275674

São Paulo - 2002

O presente trabalho é parte da dissertação de mestrado apresentada por Marcos Alves Predebon, sob a orientação do Prof. Dr. Marco Túlio Carvalho de Andrade: "Estudo Comparativo de Implementação de Sistemas Nebulosos em Hardware", defendida em 22/02/02, na EPUSP.

A íntegra da dissertação encontra-se à disposição com o autor e na Biblioteca de Engenharia Elétrica da Escola Politécnica da USP.

FICHA CATALOGRÁFICA

Predebon, Marcos Alves

Alternativas de implementação de sistemas nebulosos em hardware / M.A. Predebon, M.T.C. de Andrade. – São Paulo : EPUSP, 2002.

10 p. – (Boletim Técnico da Escola Politécnica da USP, Departamento de Engenharia de Computação e Sistemas Digitais, BT/PCS/0211)

1. Fuzzy 2. Terapia por exercício I. Andrade, Marco Túlio Carvalho de II. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Computação e Sistemas Digitais

tais

III. Título IV. Série

ISSN 1413-215X

511.322

CDD

615.824

Alternativas de Implementação de Sistemas Nebulosos em Hardware

Marcos Alves Predebon & Marco Túlio Carvalho de Andrade

Laboratório de Técnicas Inteligentes – LTI
Depto. de Engenharia de Computação e Sistemas Digitais
Escola Politécnica da Universidade de São Paulo
Av. Professor Luciano Gualberto, travessa 3, n. 158, sala C2-48
CEP: 05508-900; São Paulo – SP - Brasil
marco.andrade@poli.usp.br

Resumo: neste trabalho são resumidas e analisadas algumas das principais técnicas para se implementar sistemas nebulosos em hardware. São abordados problemas de implementação de circuitos em vários níveis de abstração dentro da teoria nebulosa desde a implementação de funções de pertinência em hardware até processadores dedicados para computação nebulosa.

Palavras chaves: Sistemas Nebulosos, Teoria Nebulosa, Circuitos Nebulosos, Circuitos Embarcados.

Abstract: in this work some of the main Fuzzy hardware implementing techniques are summarized and analyzed. Problems of implementing circuits in several abstraction levels using the fuzzy set theory are focused.

Key words: Fuzzy Systems, Fuzzy Theory, Fuzzy Hardware, Hardware Design, Hardware.

1. INTRODUÇÃO

Em 1996 foi realizado o desenvolvimento de um controle de movimentação para uma bicicleta ergométrica, ilustrada na figura 1.1, equipada com motor de corrente contínua, usada na execução de exercícios passivos de musculatura (muitas vezes atrofiada), com a finalidade de manutenção ou recuperação da mesma.

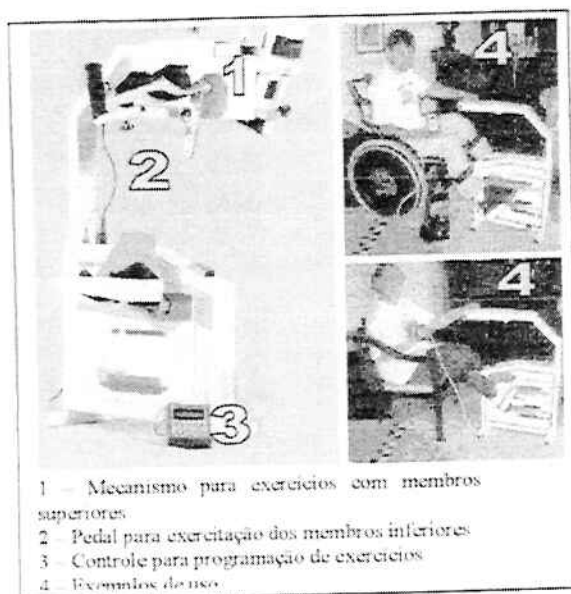


Figura 1.1 – Fotos da Bicicleta

Na ocasião, pôde-se vislumbrar uma possível aplicação da Teoria de Conjuntos Nebulosos. Tal aplicação poderia ser feita na definição dos parâmetros dos exercícios já que seria bom se os mesmos tivessem relação direta com o comportamento do paciente durante os exercícios.

Assim, essa avaliação subjetiva seria realizada por um modelo nebuloso aplicado ao circuito de controle.

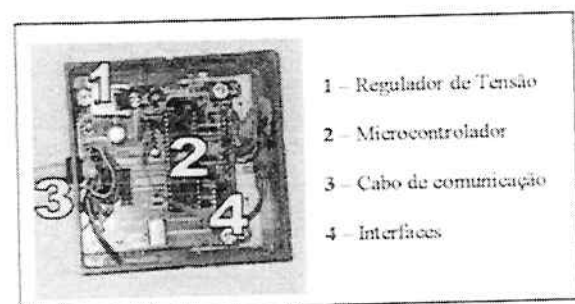


Figura 1.2 – Detalhe do Controle da Bicicleta

Os menus interativos permitem a programação dos parâmetros de: tempo total de exercício, velocidade, número máximo de espasmos permitidos e intervalo entre espasmos (acima do qual o evento é desconsiderado).

Como o circuito atual da bicicleta é composto por um microcontrolador (detalhe da figura 1.2), surgiu a idéia da realização de um estudo sobre as diversas possibilidades de desenvolvimento de um controle nebuloso baseado em hardware.

2. IMPLEMENTAÇÃO DE SISTEMAS NEBULOSOS EM HARDWARE

Atualmente há vários processadores nebulosos, há sistemas que geram programas para processadores convencionais e há também outros sistemas que possibilitam a geração de códigos de descrição de circuitos, permitindo a síntese de circuitos nebulosos projetados para aplicações específicas.

Normalmente algoritmos nebulosos são mais facilmente implementados em software pois não há limitação do número de regras entre outros parâmetros. Porém, quando há restrições de dimensão e consumo de potência, a solução tende à utilização de um hardware nebuloso.

Esse hardware pode ser implementado por meio de:

- circuitos integrados dedicados;
- processadores nebulosos;
- microcontroladores com co-processadores nebulosos;
- microcontroladores com algumas instruções nebulosas;
- ou até mesmo pelo uso de microcontroladores convencionais com programas que executam as funções nebulosas.

Apesar dos circuitos ou hardware apresentados aqui serem discretos (não nebulosos), os sistemas exemplificados aqui são nebulosos, portanto estão sendo utilizadas neste texto as expressões "hardware nebuloso" ou "circuito nebuloso" para identificar circuitos que executam algoritmos nebulosos.

Diferentes níveis de abstração podem ser utilizados para descrever um circuito nebuloso ou sistema nebuloso como pode ser visto na tabela 1.1 [Gajisk-97]:

Nível de abstração	Comportamental	Estrutural	Físico
Transistores	Equações diferenciais, diagramas	Transistores, resistências, capacitâncias	Células analógicas e digitais
Portas	Equações booleanas, Máquinas de estado finitas	Portas, circuitos lógicos, flip-flops	Módulos, unidades
Registradores, via de dados	Algoritmos, Fluxogramas, conjunto de instruções	somadores, registradores, comparadores, memórias	microcircuitos
Processadores	Especificação executável, algoritmos, programas	Processadores, memórias, ASICs, controladores	placas, módulos

Tabela 1.1 – Níveis de Abstração

Os principais métodos de desenvolvimento de hardware nebuloso foram classificados neste trabalho como os seguintes tipos:

2.1 Processadores Convencionais

Como citado anteriormente os algoritmos nebulosos são mais facilmente implementados em software, porém essa técnica limita a performance do sistema, além de, comumente, exigir muita energia ou espaço.

Os sistemas equipados com processadores de sinais digitais (DSPs), que possuem maior capacidade computacional, podem alcançar resultados da ordem de 1,5 μ s para uma inferência nebulosa de 6 entradas (DSP de 32 bits da Texas com clock de 200MHz).

Para aplicações que utilizam processadores genéricos, várias ferramentas de auxílio ao desenvolvimento foram criadas como pode ser verificado mais adiante neste texto. Essas ferramentas auxiliam o desenvolvimento e reduzem o tempo de desenvolvimento e ajuste de sistemas nebulosos, pois permitem a simulação dos sistemas e possuem interfaces gráficas para criação dos parâmetros e definição de regras.

2.2 Processadores Nebulosos

Para melhorar o desempenho, vários microcontroladores receberam expansões de suas funcionalidades internamente, por meio de adição de instruções utilizadas em processamentos nebulosos como, por exemplo, as de mínimo e máximo.

Essa expansão interna das instruções pode também ser efetuada incluindo-se blocos de processamento nebuloso na arquitetura do processador original. Dessa maneira, a comunicação entre os novos blocos nebulosos (ou ALU Nebulosa) é interna e portanto menos problemática, porém esses dispositivos consomem muita área o que dificulta e aumenta o custo de sua aplicação comercial.

Também pode ser verificado em [Hennessy-96] que ao se introduzir instruções nebulosas, o desempenho do modelo de processador avaliado (DLX – modelo teórico) foi multiplicado por um fator de 1.7 ao estimar-se a execução de um processo nebuloso.

2.3 Co-processadores Nebulosos

Unindo as duas técnicas anteriores, o uso de expansões externas para o processamento das funções nebulosas se assemelha aos circuitos co-processadores matemáticos utilizados nos computadores pessoais para aumentar o desempenho, porém neste caso para a resolução de requisições nebulosas.

Como exemplo pode ser citado o co-processador WARP 2.0 (weight-associative rule processor) fabricado pela SGS-Thomson. Trabalha com 8 bits e inclui funções de fuzificação, defuzificação e inferência. Possui memória para programação dos

anteriores e consequentes das regras, clock de 40MHz e limites de 8 entradas, 4 saídas e 256 regras e tempo de inferência de 200 microssegundos.

2.4 Circuitos Dedicados

O desenvolvimento de circuitos nebulosos dedicados é aplicado quando há necessidades de alto desempenho (baixos tempos de resposta) em sistemas específicos ou quando há restrições de espaço ou consumo de energia.

Esses circuitos integrados dedicados (ASIC – Application Specific Integrated Circuit) são normalmente desenvolvidos em linguagem de descrição de Hardware (como Verilog ou VHDL) e posteriormente gravados em matrizes programáveis de portas lógicas (FPGA) ou sintetizados em pastilhas.

Essas aplicações normalmente só se justificam caso haja previsão de produção em larga escala do circuito em questão ou da presença de parâmetros rigorosos que impossibilitem o uso de processadores convencionais ou processadores de sinais (DSPs).

O uso de ferramentas de descrição de comportamento como o de máquinas de estado aliado a linguagens para síntese de circuitos como VHDL, facilita o desenvolvimento de circuitos dedicados conforme mostrado em [Salpura-96] e exposto mais adiante nesse texto.

```
Proc_fuzzifier: PROCESS (in_var)
BEGIN
IF reset = '1' THEN
  memb_var => (Others <= '0');
ELSEIF clock'EVENT AND (clock='1') THEN
  IF (in_var < tm1m) THEN
    tmp:=conv_std_logic_vector(unsigned(unsigned(in_var)-unsigned(start)) * unsigned(
    memb_var(7 downto 0) <= tmp;
    memb_var(set_nr '8-1 downto 8) <= (Others => '0');
  ELSEIF (in_var < tm2m) THEN
    tmp:=conv_std_logic_vector(unsigned(unsigned(in_var)-unsigned(tm1m)) * unsigned(
    memb_var(7 downto 0) <= conv_std_logic_vector(1-unsigned(tmp), 8);
    memb_var(15 downto 8) <= tmp;
    memb_var(set_nr '8-1 downto 16) <= (Others => '0');
  ELSEIF (in_var < tm3m) THEN
    tmp:=conv_std_logic_vector(unsigned(unsigned(in_var)-unsigned(tm2m)) * unsigned(
    memb_var(7 downto 0) <= (Others => '0');
    memb_var(15 downto 8) <= conv_std_logic_vector(1-unsigned(tmp), 8);
    memb_var(set_nr '8-1 downto 16) <= tmp;
  ELSE
    ***
  END IF;
END PROCESS;
```

Figura 2.1 – Código em VHDL

Em [Salpura-96] pode ser encontrado um exemplo detalhado de implementação de Sistemas de controle utilizando VHDL. Abaixo é apresentado um resumo desse método, aplicado no controle de um aquecedor a gás: para o desenvolvimento, foram adotadas duas etapas separadas, sendo uma delas a descrição da base de conhecimento utilizando o método de máquinas de estado com uma ferramenta que gera código VHDL. Já os códigos do fuzzificador e defuzzificador foram desenvolvidos diretamente em VHDL, utilizando comparadores, somadores e um multiplicador.

Na figura 2.1 pode-se observar um trecho do código VHDL de descrição do fuzzificador.

O uso de linguagem de alto nível com uma ferramenta de descrição de máquinas de estado como apresentado nesse caso [Salpura-96], permite agilizar bastante o tempo de desenvolvimento.

Uma etapa que consome quase tanto tempo quanto o desenvolvimento do circuito é a criação do ambiente de teste para os códigos VHDL.

Em [Hollstein-96] são apresentadas ferramentas de auxílio ao desenvolvimento de sistema nebuloso e duas variações de sistemas nebulosos com redes neurais, todos em VHDL.

Os circuitos dedicados podem ser implementados utilizando-se uma de três técnicas (Off-line, semi-off-line e on-line) [Baturone-00] e, além dessas estratégias de processamento no tempo de projeto ou execução, pode-se optar também por circuitos analógicos ou digitais conforme descrito a seguir.

2.5 Circuitos Analógicos

O primeiro circuito nebuloso analógico foi apresentado por Yamakawa em 1987 [Yamakawa-87] e possuía a arquitetura apresentada na figura 2.2 que é totalmente paralela. O sistema consistia em dois circuitos integrados, um deles de três entradas, regra única com saída única em 25 linhas que transferiam a informação de saída para outro, encarregado de realizar a defuzzificação.

A figura 2.2 também inclui um exemplo de condição de funcionamento. No gráfico A (nessa figura), pode-se observar como seria representada uma função de consequente da regra, que conforme os valores de entrada dos antecedentes, tem os valores limitados pela função “mínimo” (min) gerando 25 saídas representadas no gráfico B (também na figura 2.2).

O circuito defuzzificador agrega os resultados de outras regras gerando um conjunto exemplificado no gráfico C (também na figura 2.2), no qual é baseado o valor da saída que corresponde ao centro de gravidade ou centro de massa desse conjunto.

Os circuitos analógicos são mais complexos e possuem uma gama muito maior de possibilidades de soluções para a mesma aplicação.

Por introduzir o conceito de chaveamento e possuir 3 fases (MFC, MFG e divisão), os conceitos de latência e processamento paralelo (pipeline) podem ser encontrados aplicados a esse tipo de solução. Esse tipo de circuito apresenta uma certa latência, que é por volta de três vezes maior que o tempo de processamento de uma inferência.

Essa técnica também exige a utilização de células de memória capazes de armazenar valores digitais para a conexão entre os módulos (pipeline).

transistores e por meio de circuitos diferenciais, produzem uma função de saída com propriedades desejadas para um circuito de cálculo de pertinência.

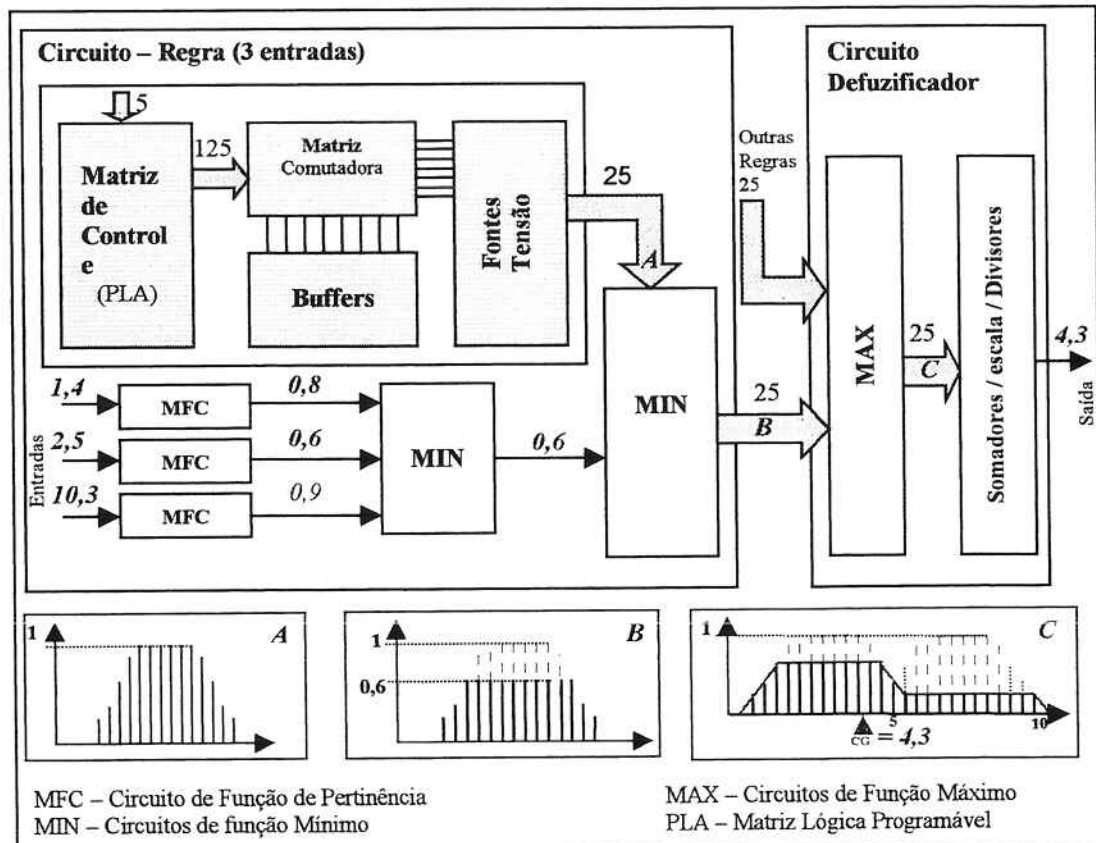


Figura 2.2 – Arquitetura do Circuito Nebuloso Analógico

Normalmente os circuitos nebulosos analógicos são implementados baseados em corrente elétrica pois isso permite a execução de operações necessárias para os processos nebulosos com poucos componentes (*min*, *max*), além de serem menos sensíveis às variações de temperatura e de poderem trabalhar com baixos valores de tensão.

Outra característica comum dos circuitos nebulosos analógicos é a de possuírem entradas e saídas em modo de tensão [Baturone-97] pois é o método mais utilizado nos circuitos discretos como interface entre componentes.

Este método foi também discutido em [Huertas-96] que apresenta a implementação de controladores programáveis (Sugeno), usando circuitos CMOS. Essa implementação também faz uso de tensões nas interfaces e correntes nos circuitos diferenciais internos.

Os circuitos baseados em transistores MOS usam as propriedades lineares da curva de resposta dos

Técnicas de otimização do uso de circuitos foram desenvolvidas como a arquitetura voltada a regras ativas, ou a de compartilhamento de circuitos de entrada, para reduzir a área necessária para sua implementação [Baturone-00] e [Jimenez-95].

Outra técnica em discussão é a de arquitetura reconfigurável, que permite circuitos mais flexíveis. Isto pode ser encontrado em [Hartenstein-01].

2.6 - Circuitos Digitais

O primeiro circuito digital, proposto por Togai em 1985, possuía caminhos distintos para cada regra, ficando assim com sua área diretamente influenciada pelo número de regras.

O primeiro protótipo foi constituído por uma memória de regras, uma unidade de inferência e um circuito de controle, porém não havia unidade de defuzificação.

Os circuitos sucessores utilizavam várias técnicas, sendo as principais as de processamento paralelo de regras, processamento serial de regras e processamento de regras ativas. Essas três maneiras são variações entre o tema de paralelismo de processamento das regras que pode ser total, reproduzindo-se o hardware necessário, consumindo assim mais área.

Já na solução seqüencial o uso de área é menor (com número grande de regras), porém o tempo torna-se o fator crítico.

A solução baseada em regras ativas tenta otimizar esses parâmetros. Detalhes dessas soluções podem ser vistos em [Ascia-97], [Kandel-98] e [Baturone-00].

Um exemplo de controlador genérico usando essa técnica pode ser encontrado em [Jimenez-95], que descreve uma implementação de um circuito flexível e otimizado em área utilizada e tempo de execução.

Normalmente chamada de PS ou "Pulse Stream", a modulação por pulsos tem sido utilizada e citada em diversas publicações por apresentar algumas vantagens sobre outras técnicas. Várias são as possibilidades para implementações de hardware nebuloso ou de redes neurais. Utilizando-se da modulação de pulsos, como pode ser encontrado em [Reyneri-94], são criados mecanismos para transmissão de valores analógicos por meio de modulação de pulsos.

As principais aplicações utilizam-se de modulação de: largura de pulsos (PWM – Pulse Width Modulation), fase (PPM Pulse Phase Modulation), frequência (PFM Pulse Frequency Modulation ou ainda PRM de rate), relação entre pulsos e pausas (Duty Cycle Modulation DCM), rajadas de pulsos (PBM Pulse Burst Modulation), amplitude (PAM Pulse Amplitude Modulation) e por codificação digital no tempo (PCM – Pulse Code Modulation).

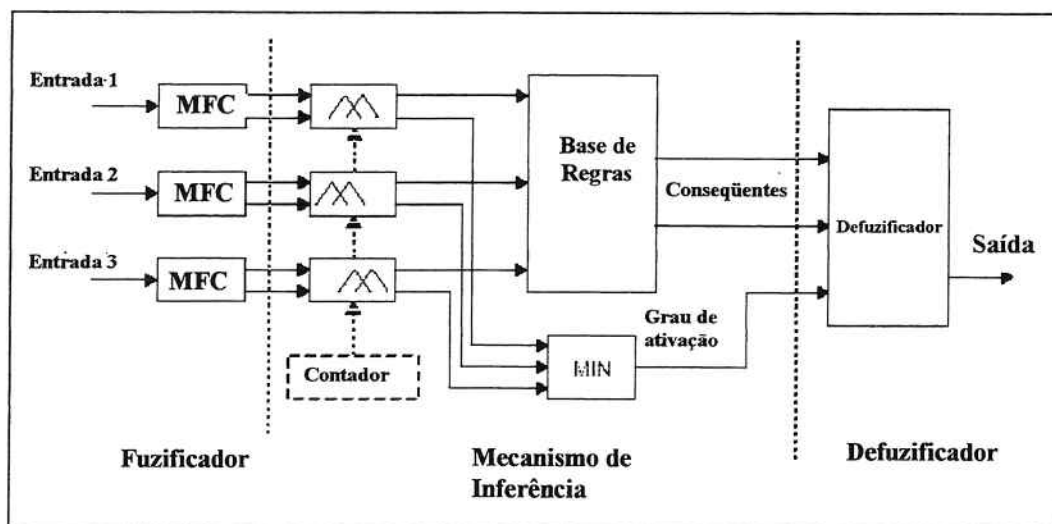


Figura 2.3 – Arquitetura do Circuito Nebuloso

Como pode ser observado na figura 2.4, o processamento dos cálculos de pertinência é realizado em paralelo e o processamento das regras é seqüencial. Nesse exemplo ressalta-se a necessidade de limitar o número de superposições dos modificadores linguísticos já que é uma das definições dessa arquitetura.

O mecanismo de inferência usa somente os pares ativos (modificador linguístico, pertinência) e transmite ao defuzificador os consequentes ativos e seus respectivos graus de atuação.

2.7 Modulação por pulsos

Visando melhoria da precisão, surgiram outras técnicas baseadas em chaveamento de capacitores ou de correntes que por meio de variação do ciclo de chaveamento controlam um determinado valor de corrente ou tensão.

Além dessas técnicas, é citado em [Reyneri-94] que a modulação sincronizada de pulsos (CPWM Coherent Pulse Width Modulation) em todas entradas apresenta

uma série de vantagens, entre elas a melhor distribuição do consumo de energia evitando a geração de ruído na fonte de alimentação.

2.8 Ferramentas de Auxílio para Desenvolvimento

Para os métodos de implementação que utilizam processadores ou microcontroladores convencionais, com co-processadores dedicados ou mesmo para os processadores nebulosos, existem atualmente várias ferramentas de auxílio ao desenvolvimento de sistemas nebulosos em hardware. Podem ser citadas como exemplo:

Xfuzzy – Trata-se de um ambiente de geração de sistemas nebulosos que possui ferramenta de edição de regras, descrição de comportamento, simulação além de bibliotecas que permitem a geração de ASICs via VHDL ou mesmo gerar tabelas para geração de FPGAs. Em [Barriga-98] pode-se ver um exemplo de uso dessa ferramenta.

Introl-CODE – Ambiente integrado de desenvolvimento para geração de código para circuitos integrados Motorola. Gera códigos para os microcontroladores 68HC05, 68HC08, 6809, 68HC11, 68HC12, 68HC16 e 68332.

FLASH – Gerador de código montado (MCS51). Por meio de uma descrição simples em linguagem natural e definição de variáveis e termos lingüísticos como apresentada na figura 2.5. Esse programa é capaz de gerar o código montado para execução em microcontrolador da família 8031, bem como as subrotinas necessárias [Yeralan-96].

Flash é uma ferramenta simples executada em DOS, porém prática. Por exemplo, além do código objeto (montado) também é gerado um relatório de variáveis, regras e termos criados.

```

: FLASH – Exemplo

: Controle de Velocidade de uma ventoinha
: Opções
#define centroid
#define p8032

: --- Entradas ---
#input byte Temperatura register 0E0h
: --- Saida ---
#output byte Vel_Vent register 0E1h

: Todos termos são trapezoides com 4 pontos {a, b, c, d}

: --- termos para Temperatura ---
#term FRIA (Temperatura) { 0, 0, 30, 45 }
#term MORNA (Temperatura) { 30, 45, 90, 100 }
#term QUENTE (Temperatura) { 90, 100, 120, 120 }

: --- termos para VEL_VENT (saida singleton) ---
#term DESL (VEL_VENT) { 0 }
#term BAIXA (VEL_VENT) { 60 }
#term ALTA (VEL_VENT) { 120 }
: --- definições das Regras ---
#rule if Temperatura is FRIA then Vel_Vent is DESL
#rule if Temperatura is MORNA then Vel_Vent is BAIXA
#rule if Temperatura is QUENTE then Vel_Vent is ALTA

```

Figura 2.4 – Código Fonte para ferramenta Flash

FuzzyTech – É uma das ferramentas mais utilizadas, que tem versões para aplicações técnicas e financeiras. Há suporte para diversos processadores nebulosos, incluindo Motorola, Siemens, Intel e Microchip, entre outros.

Em [Altrock-96] é encontrada a descrição de como as ferramentas de auxílio ao desenvolvimento podem melhorar o desempenho de um programa nebuloso sendo executado por um microcontrolador convencional. Basicamente a ferramenta tenta

maximizar as etapas do processo na fase de compilação, para que o processo seja o mais simples possível em tempo de execução.

Cinco fases normalmente são necessárias para um desenvolvimento: especificação das variáveis lingüísticas, definição da estrutura de inferência, formulação das regras, verificação do algoritmo e simulação em tempo real.

Micro FPL – Programa capaz de gerar códigos otimizados para microcontroladores Intel, Motorola e Oki. Gera códigos com tempos de resposta abaixo de milissegundos, mesmo com sistemas de média complexidade [Baturone-00].

FuzzyStudio - Sistema de desenvolvimento dedicado aos microcontroladores nebulosos da família ST52 da STMicroelectronics. Esse programa permite a geração de sistemas nebulosos por meio de interfaces gráficas. Possui ferramenta de depuração de código e simulação de execução do algoritmo, entre outras funções comuns a essas ferramentas.

A ferramenta também possui mecanismos de validação (debug) de sistemas que permitem a realização de testes simulados com diversas condições de entrada. A curva de controle pode ser visualizada como uma superfície, facilitando assim sua verificação.

FLDE – (Fuzzy Logic Development Environment) mais uma ferramenta de auxílio ao desenvolvimento de sistemas nebulosos. Pela descrição, em linguagem natural, do comportamento requerido, esse sistema gera o código de programação para diversos circuitos integrados (196/296, C991, C166 e HC12), bem como em linguagem C.

FIDE (Fuzzy Inference Development Environment) – Baseada em plataforma PC-Windows, permite o desenvolvimento depuração e simulação de aplicações Nebulosas para microcontroladores Motorola.

TIL - TILShell é uma ferramenta de desenvolvimento de sistemas nebulosos, simulação, testes e compilação de programas em C ou montadores específicos para co-processadores ou microcontroladores nebulosos.

Essa ferramenta sugere o uso de uma linguagem orientada a objetos, especialmente definida para sistemas nebulosos.

Uma comparação entre as diversas ferramentas pode ser encontrada em [Legg-94] onde estão relacionadas as funcionalidades das ferramentas e um comparativo entre custos das mesmas.

3. COMENTÁRIOS FINAIS

Este trabalho descreveu métodos e estratégias de desenvolvimento de algoritmos nebulosos em hardware.

Para isso foram resumidas e comentadas algumas das principais alternativas de implementação de sistemas nebulosos baseados em circuitos.

Constatou-se que os principais itens a serem analisados para a realização do projeto de um sistema nebuloso em hardware são:

- o desempenho exigido do sistema;
- as restrições físicas (espaço e energia).

Frente às muitas alternativas para o desenvolvimento de hardware, esses itens devem ser analisados caso a caso.

Para os circuitos embarcados, com exigência de tempos de resposta reduzidos, as possibilidades são menores, normalmente partindo de DSPs ou processadores nebulosos chegando até a circuitos específicos (ASICs), nos casos extremos.

Os sistemas específicos são melhores em relação aos parâmetros de desempenho e restrições físicas. Porém, sua aplicação é mais dispendiosa pois exige maior tempo de desenvolvimento e maior complexidade de implementação. Estes problemas são atenuados com o uso de ferramentas de auxílio ao desenvolvimento (CAD). Por outro lado, o maior investimento se justifica quando há previsão de reprodução do circuito em larga escala.

Verificou-se que, para o problema específico da bicicleta, devido às suas condições de contorno e estado de desenvolvimento do hardware, a melhor maneira de implementar o algoritmo nebuloso proposto é por meio de software aplicado ao hardware já instalado na bicicleta.

4. NOTAS BIBLIOGRÁFICAS

Neste artigo diversas técnicas de implementação de circuitos nebulosos foram abordadas.

Em [Baturone-00] pode ser encontrada uma descrição mais detalhada dessas técnicas e algumas comparações entre elas.

Também em [Hennessy-96] pode-se encontrar, em abordagem mais detalhada, as técnicas de microprogramação, análise de otimização de arquiteturas e códigos, bem como estudo sobre os métodos de programação paralela e suas limitações.

Em [Ascia-97] também é estudado em maiores detalhes o paralelismo de circuitos nebulosos e suas implicações.

Em [Salpura-96] e [Hollstein-96] podem ser encontrados detalhes de implementação de circuitos nebulosos em VHDL. Também em [Cobo-00] é

apresentada uma ferramenta para prototipagem de hardware.

Já em [Arbex-96] há uma aproximação mais genérica sobre controladores nebulosos, apresentando uma ferramenta para auxílio ao desenvolvimento de sistemas, como as que neste artigo foram relacionadas. Outra relação similar a apresentada aqui pode ser vista em [Legg-94].

Em [Pellerin-97] pode ser encontrada a apresentação detalhada da sintaxe da linguagem VHDL.

Pode-se também encontrar mais referências de processadores nebulosos em [Ascia-99], [Salpura-00] e [Gschwind-01].

Tanto em [Baturone-00] quanto em [Catania-94] e [Baturone-97] podem ser vistas as vantagens e dificuldades de implementações analógicas de circuitos nebulosos. Já a aproximação digital para esse problema possui diversas possibilidades de resolução, que são discutidas em [Ascia-98], [Kandel-98] e [Baturone-00].

A Técnica de chaveamento ou modulação de pulsos aplicada a implementação de circuitos pode ser vista em [Reyneri-94], [Reyneri-95] e também em [Baturone-00].

6. BIBLIOGRAFIA

[Altrock-96] Altrock, Constantin von; "Fuzzy Logic Design: Methodology, Standards, and Tools"; Electronic Engineering Times; Julho de 1996.

[Arbex-96] Arbex, R.T.; "Um ambiente para desenvolvimento de controladores Fuzzy"; Dissertação para obtenção de título de Mestre em Engenharia; Escola Politécnica da Universidade de São Paulo; 1996.

[Ascia-97] Ascia, Giuseppe; Vincenzo Catania, Giuseppe Ficili, Sergio Palazzo and Daniela Panno; "A VLSI Fuzzy Expert System for Real-Time Traffic Control in ATM Networks"; IEE Transactions on Fuzzy Systems, Vol. 5, no 1, p. 20-31, Fevereiro de 1997.

[Ascia-98] Ascia, Giuseppe; Davide Sarta; Dario Trifone; "A Data Dependent Approach to Instruction Level Power Estimation"; Proceedings of the IEEE; 1998.

[Barriga-98] Barriga, A.; R. Senhadji; C. J. Jiménez; I. Baturone; S. Sánchez-Solano; "A Design Methodology for Application Specific Fuzzy Integrated Circuits", Proc. 5th IEEE International Conference on Electronics, Circuits and Systems (ICECS'98), Vol. 1, p. 431-434, Lisboa, Setembro de 1998.

- [Baturone-00]** Baturone, I.; Barriga A.; C. J. Jiménez; S. Sánchez-Solano; D. R. López; "Microeletronic Design of Fuzzy Logic-Based Systems"; CRC Press LCC; 2000.
- [Baturone-97]** Baturone, I.; S. Sánchez-Solano; J.L. Huertas; Barriga A.; "Implementation of CMOS Fuzzy Controllers as Mixed-Signal Integrated Circuits"; IEE Transactions on Fuzzy Systems; vol. 5, no 1, p. 1-19, Fevereiro de 1997.
- [Catania-94]** Catania, V.; A. Puliafito; M. Russo; L. Vita; "A VLSI Fuzzy Inference Processor Based on a Discrete Analog Approach"; IEE Transactions on Fuzzy Systems; vol. 2, no 2, p. 93-106, Maio de 1994.
- [Cobo-00]** Cobo, José Edinson Aedo; "Considerações e Modelos Arquiteturais para Prototipagem Rápida em Hardware de Modelos Nebulosos"; Tese para obtenção de título de Doutor em Engenharia; Escola Politécnica da Universidade de São Paulo; 2000.
- [Gajisk-97]** Gajisk, D. D.; "Principle of Digital Design"; Prentice Hall; 1997.
- [Gschwind-01]** Gschwind, Michael; Valentina Salapura; Dietmar Maurer; "FPGA Prototyping of a RIS Processor Core for Embedded Applications"; IEEE Transactions on Very Large Scale Integration (VLSI) Systems; Vol 9, no 2, p. 241-250; Abril de 2001.
- [Hartenstein-01]** Hartenstein, Reiner; "A Decade of Reconfigurable Computing: a Visionary Retrospective"; Artigo sobre arquiteturas reconfiguráveis; disponível em: <http://www.fpl.uni-kl.de> ; acessado em 09/12/2001.
- [Hennessy-96]** Hennessy, J. L.; Patterson, D.^a; "Computer Architecture, a Quantitative Approach"; 2a Edição; Morgan Kaufmann Publishers; Inc; 1996.
- [Hollstein-96]** Hollstein, Thomas; Saman K. Halgamuge, Manfred Glesner; "Computer Aided Design of Fuzzy Systems based on generic VHDL Specifications"; IEE Transactions on Fuzzy Systems, Vol. 4, no 4, p 403-416; Novembro de 1996
- [Huertas-96]** Huertas, J. L.; S. Sánchez Solano, I. Baturone, A. Barriga; "Integrated Circuit Implementation of Fuzzy Controllers"; IEEE Journal of Solid-State Circuits, Vol. 31, nº 7, p. 1051-1058, Julho 1996.
- [Jimenez-95]** Jimenez, C. J.; Solano, S. S.; Barriga, A.; "Hardware Implementation of a General Purpose Fuzzy Controller"; Sexto Congresso – International - Fuzzy Systems Association World Congress (IFSA'95), Vol. 2, p. 185-188, São Paulo; Julho de 1995.
- [Kandel-98]** Kandel, Abraham, Gideon Langholz; "Fuzzy Hardware – Architectures and Applications"; Kluwer Academic Publishers; 1998.
- [Legg-94]** Legg, Gary; "Fuzzy-Logic design tools help build embedded systems"; EDN magazine; p. 10-23; Julho de 1994.
- [Pellerin-97]** Pellerin, David; Douglas Taylor; "VHDL Made Easy"; Prentice Hall; 1997.
- [Reyneri-94]** Reyneri; L.M.; H.C.A.M. Withagen; J.A. Hegt; M. Chiaberge; "A Comparison between Analog and Pulse Stream VLSI Hardware for Neural Networks and Fuzzy Systems"; Proc. of MICRONEURO 94, International Conference on Microelectronics for Neural Networks and Fuzzy Systems, IEEE Computer Society Press, p. 77-86; Setembro de 1994.
- [Reyneri-95]** Reyneri, L.M. "Performance Analysis of Pulse Stream Neural and Fuzzy Computing System"; IEEE Transactions on Circuits and Systems – II; Vol. 42; no 10, p. 642-660; Outubro de 1995.
- [Salapura-96]** Salapura, Valentina; Hamann, "Implementing Fuzzy Control Systems Using VHDL and Statecharts"; Proceedings of the European Design and Automation Conference EURO-DAC'96, Geneva, Setembro de 1996.
- [Yamakawa-87]** Yamakawa, T. "Fuzzy controller hardware system", Proc. 2nd IFSA Congress, Japan, p.827-830; 1987.
- [Yeralan-96]** Yeralan, Sencer; Baris Tan; "Fuzzy Logic Control With Microcontrollers"; Rigel Corporation; 1996.

6. REFERÊNCIAS ADICIONAIS

- [Andrade-95] Andrade, Marco Túlio Carvalho de; "Controlador Borroso Universal Distribuído: Base de Conocimientos, Modelo Computacional y Herramientas de Software"; Tese de Doutorado, Universidade Politécnica de Madri, Faculdade de Informática, Madri, Espanha; 1995.
- [Andrade-98] Andrade, Marco Tulio C.; "Computação Fuzzy"; Notas de aula da disciplina PCS711 do curso de pós-graduação do Departamento de Engenharia de Computação e Sistemas Digitais da Escola Politécnica da Universidade de São Paulo; 1997.
- [Ascia-99] Ascia, Giuseppe; Vincenzo Catania, and Marco Russo, "VLSI Hardware Architecture"; IEE Transactions on Fuzzy Systems, Vol. 7, no. 5, p. 553-570, Outubro de 1999.
- [Barriga-98] Barriga, A.; R. Senhadji; C. J. Jiménez; I. Baturone; S. Sánchez-Solano; "A Design Methodology for Application Specific Fuzzy Integrated Circuits", Proc. 5th IEEE International Conference on Electronics, Circuits and Systems (ICECS'98), Vol. 1, p. 431-434, Lisboa, Setembro de 1998.
- [Baturone-00b] Baturone, I.; S. Sánchez Solano; "Microelectronic Design Of Universal Fuzzy Controllers"; X Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF 2000); p. 247-25; Setembro de 2000.
- [Bermudez-01] Bermudez, A.; A. Barriga; I. Baturone; S. Sánchez-Solano; "FLEB: A Fuzzy Logic E-Book"; European Symposium on Intelligent Technologies, Hybrid Systems and their implementation on Smart Adaptive Systems (EUNITE-2001); Dezembro de 2001.
- [Bona-97] Bona, B.; S. Carabelli; M. Chiaberge; E. Miranda; L.M. Reyneri; "Neuro-Fuzzy Hardware and DSPs: a Promising Marriage for Control of Complex Systems"; Proc. MICRONEURO 97, International Conference on Microelectronics for Neural Networks and Fuzzy Systems; p. 113-120; Setembro de 1997.
- [Bonde-94] Bonde, Allen R., Sumit Ghosh, "A Comparative Study of Fuzzy Versus 'Fixed' Thresholds for Robust Queue Management in Cell-Switching Networks", IEE Transactions on Networking, Vol. 2, no 4, p. 337-344, Agosto de 1994.
- [Chianberge-95] Chiaberge, M.; L.M. Reyneri; L. Zocca; "A Neuro-Fuzzy Real Time Controller for Low Power Embedded Systems"; IEEE MICRO, Hardware for Artificial Neural Networks; Edição Especial, p. 40-47; Junho de 1995.
- [Delva-99] Delva, Justin G. R.; "FPGA Implementation of a Nonlinear Two Dimensional Fuzzy Filter"; Departamento de Engenharia Elétrica e Ciencias da Computação, UWM Milwaukee, Estados Unidos da América; 1999.
- [Dillen-00] Dillen, Steve; Farrah Rashid; "Fuzzy Logic Controller in VHDL"; Artigo sobre implementação de um controlador simples em VHDL; disponível em: <http://www.ee.ualberta.ca/~elliott/ee552/studentAppNotes/2000f/vhdl/fuzzyController>; acessado em 09/12/2001.
- [Driankov-96] Driankov D., Hellendoorn H. e Reinfrank M.; "An Introduction to Fuzzy Control"; Prentice Hall; 1996.
- [Eichfeld-96] Eichfeld, Herbert; Manfred Menke; Thomas Kuemund; Martin Klimke; Jürgen Noll; "A General-Purpose Fuzzy Inference Processor"; IEE Transactions on Fuzzy Systems; Vol. 4, no 4, p. 460-475; Novembro de 1996.
- [Fattarus-94] Fattarus, John W., Mahant-Shetti, Shivaling S. e Barton, J.Brock; "A Fuzzy Logic Inference Processor"; IEE Journal of Solid State Circuits; vol 29, no 4, p. 397-402, Abril de 1994.
- [Garms-01] Garms, Marco Antonio; "Teoria Nebulosa aplicada em Sistemas de Auxílio Motor"; Dissertação para obtenção de título de Mestre de Engenharia; Escola Politécnica da Universidade de São Paulo; 2001.
- [Garms-96] Garms, Marco A.; "Implementação de uma linguagem para tempo real não preemptivo"; Monografia do curso Pesquisa a Sistemas Digitais; Escola Politécnica da Universidade de São Paulo, 1996.
- [Ghosh-98] Ghosh, Sumit; Razouqi, Qutaiba; Schumacher, H.Jerry e Celmins, Aivars; "A Survey of Recent Advances in Fuzzy Logic in Telecommunications Networks and New Challenges"; IEE Transactions on Fuzzy Systems, Vol 6, no 3, p. 443-447; Agosto de 1998.
- [Kim-00] Kim, Daijin; "An Implementation of Fuzzy Logic Controller on the Reconfigurable FPGA System"; IEEE Transactions on Industrial Electronics; Vol 47, no 3, p. 703-715; Junho de 2000.
- [Klir-95] Klir, George J.; Bo Yuan; "Fuzzy Sets and Fuzzy Logic – Theory and Applications"; Prentice Hall; 1995.
- [Klir-97] Klir, George J.; Ute St.Clair, Bo Yuan; "Fuzzy Set Theory – Foundations and Applications"; Prentice Hall; 1997.

- [Kosko-97] Kosko B.; "Fuzzy Engineering"; Prentice Hall; 1997
- [Lago-98] Lago, E.; C. J. Jiménez; D. R. López; S. Sánchez-Solano; A. Barriga; "XFVHDL: A Tool for the Synthesis of Fuzzy Logic Controllers"; Proc. Design, Automation and Test in Europe (DATE'98), p. 102-107, Paris; Fevereiro de 1998.
- [Lee-90] Lee, C. C.; "Fuzzy Logic in control Systems - Part I"; IEEE Transactions on Systems, Man and Cybernetics, Vol.20, no 2, p. 404-418; Abril de 1990.
- [Lee-90b] Lee, C. C.; "Fuzzy Logic in control Systems - Part II"; IEEE Transactions on Systems, Man and Cybernetics, Vol. 20, no 2, p. 419-435; Abril de 1990.
- [Linkens-00] D. A. Linkens; "Fuzzy Control for Patient Muscle Relaxation"; Artigo da Universidade de Sheffield UK sobre aplicação de um controlador nebuloso para uso em tratamento médico; disponível em <http://shef.ac.uk>; acessado em 20/10/2001.
- [Mayor-94] Mayor, Antonio Ruiz; "Hardware Básico de motores de inferência borrosa utilizando trapezios codificados"; Tese de Doutorado – Departamento de tecnologia fotônica – ETSI Telecomunicación; 1994.
- [Menke-95] Menke, Manfred; Eichfeld, Herbert; Thomas Kuemund; Martin Klimke; Jürgen Noll; "A General-Purpose Fuzzy Inference Processor"; IEEE Micro, Vol. 15, no 3; Junho de 1995.
- [Mols-99] Molz, Rolf F.; Paulo M. Engel2; Fernando G. Moraes; "Uso de um Ambiente Codesign para a Implementação de Redes Neurais"; Proceedings do IV Congresso Brasileiro de Redes Neurais; p. 13-18; 1999.
- [Nakanishi-93] Nakanishi H., I.B. Turksen e M. Sugeno; "A Review and Comparison of Six Reasoning Methods"; Fuzzy Sets and Systems, no 57; p. 257-293; 1993.
- [Nascimento-00] Nascimento Jr., C.L.; Yoneyama T.; "Inteligência Artificial em controle e automação"; Ed. Edgard Blucher; 2000.
- [Neto-87] Neto J. J.; "Introdução à Compilação"; Livros Técnicos e Científicos; 1987.
- [Predebon-01] Predebon, Marcos A.; "Aplicação da Teoria Fuzzy em Telecomunicações"; Boletim Técnico de Redes e Sistemas Distribuídos, Laboratório de Arquitetura de Redes de Computadores, Universidade de São Paulo; Vol. 2, p. 7-16; Janeiro de 2001.
- [Predebon-96] Predebon, Marcos A.; "Multiplicador VHDL"; Monografia do curso Projetos de Sistemas Digitais Avançados; Escola Politécnica da Universidade de São Paulo, 1996.
- [Ruspini-98] Ruspini, E.H., Bonissone, P.P., Pedrycz, W.; "Handbook of Fuzzy Computation", Institute of Physics, 1998.
- [Salapura-00] Salapura, Valentina; "A Fuzzy RISC Processor"; IEEE Transactions on Fuzzy Systems; Vol 8, no 6, p. 781-790, Dezembro de 2000.
- [Setzer-83] Setzer, V.W.; Melo, I.S.H.; "A construção de um Compilador"; Ed. Campus; 1983.
- [Sumit-98] Sumit Ghosh, Qutaiba Razouqi, H. Jerry Schumacher, and Aivars Celmins; "A Survey of Recent Advances in Fuzzy Logic in Telecommunications Networks and New Challenges", IEE Transactions on Fuzzy Systems, Vol. 6, no 3, p. 443-447; Agosto de 1998.
- [Tsang-98] Tsang, Danny H.K.; Brahim Benasou and Shirley T.C. Lam; "Fuzzy-Based Rate Control for Real-Time MPEG Video", IEE Transactions on Fuzzy Systems, Vol. 6, no 4, p. 504-516; Novembro de 1998.
- [Velo-01] Velo, F. J. Moreno; I. Baturone; S. Sánchez Solano; A. Barriga; "XFSL: A Tool for Supervised Learning of Fuzzy Systems"; European Symposium on Intelligent Technologies. Hybrid Systems and their implementation on Smart Adaptive Systems (EUNITE-2001); Dezembro de 2001.
- [Velo-01b] Velo, F. J. Moreno; I. Baturone; S. Sánchez Solano; A. Barriga; "XFUZZY 3.0: A Development Environment for Fuzzy Systems"; International Conference in Fuzzy Logic and Technology (EUSFLAT 2001); p. 93-96; Setembro de 2001.
- [Yager-94] Yager, R.R.; Filev, D.P.; "Essentials of Fuzzy Modeling and Control"; Wiley-Interscience; 1994.
- [Zadeh-65] Zadeh, L.A.; "Fuzzy Sets", Information and Control, Vol. 8, no 3, p. 338-371; 1965.
- [Zadeh-73] Zadeh, L.A.; "Outline of a New Approach to the Analysis of Complex Systems and Decision Processes", IEEE Transactions on Systems, Man and Cybernetics, Vol. 3, p. 28-44, 1973.
- [Zadeh-73b] Zadeh, L.A.; "Fuzzy Sets Information and Control"; vol. 8; p.338-353; 1965.

BOLETINS TÉCNICOS - TEXTOS PUBLICADOS

- BT/PCS/9301 - Interligação de Processadores através de Chaves Ômicron - GERALDO LINO DE CAMPOS, DEMI GETSCHKO
- BT/PCS/9302 - Implementação de Transparência em Sistema Distribuído - LUÍSA YUMIKO AKAO, JOÃO JOSÉ NETO
- BT/PCS/9303 - Desenvolvimento de Sistemas Especificados em SDL - SIDNEI H. TANO, SELMA S. S. MELNIKOFF
- BT/PCS/9304 - Um Modelo Formal para Sistemas Digitais à Nível de Transferência de Registradores - JOSÉ EDUARDO MOREIRA, WILSON VICENTE RUGGIERO
- BT/PCS/9305 - Uma Ferramenta para o Desenvolvimento de Protótipos de Programas Concorrentes - JORGE KINOSHITA, JOÃO JOSÉ NETO
- BT/PCS/9306 - Uma Ferramenta de Monitoração para um Núcleo de Resolução Distribuída de Problemas Orientado a Objetos - JAIME SIMÃO SICHMAN, ELERI CARDOSO
- BT/PCS/9307 - Uma Análise das Técnicas Reversíveis de Compressão de Dados - MÁRIO CESAR GOMES SEGURA, EDIT GRASSIANI LINO DE CAMPOS
- BT/PCS/9308 - Proposta de Rede Digital de Sistemas Integrados para Navio - CESAR DE ALVARENGA JACOBY, MOACYR MARTUCCI JR.
- BT/PCS/9309 - Sistemas UNIX para Tempo Real - PAULO CESAR CORIGLIANO, JOÃO JOSÉ NETO
- BT/PCS/9310 - Projeto de uma Unidade de Matching Store baseada em Memória Paginada para uma Máquina Fluxo de Dados Distribuído - EDUARDO MARQUES, CLAUDIO KIRNER
- BT/PCS/9401 - Implementação de Arquiteturas Abertas: Uma Aplicação na Automação da Manufatura - JORGE LUIS RISCO BECERRA, MOACYR MARTUCCI JR.
- BT/PCS/9402 - Modelamento Geométrico usando do Operadores Topológicos de Euler - GERALDO MACIEL DA FONSECA, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9403 - Segmentação de Imagens aplicada a Reconhecimento Automático de Alvos - LEONCIO CLARO DE BARROS NETO, ANTONIO MARCOS DE AGUIRRA MASSOLA
- BT/PCS/9404 - Metodologia e Ambiente para Reutilização de Software Baseado em Composição - LEONARDO PUJATTI, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9405 - Desenvolvimento de uma Solução para a Supervisão e Integração de Células de Manufatura Discreta - JOSÉ BENEDITO DE ALMEIDA, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/9406 - Método de Teste de Sincronização para Programas em ADA - EDUARDO T. MATSUDA, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/9407 - Um Compilador Paralelizante com Detecção de Paralelismo na Linguagem Intermediária - HSUEH TSUNG HSIANG, LÍRIA MATSUMOTO SAITO
- BT/PCS/9408 - Modelamento de Sistemas com Redes de Petri Interpretadas - CARLOS ALBERTO SANGIORGIO, WILSON V. RUGGIERO
- BT/PCS/9501 - Síntese de Voz com Qualidade - EVANDRO BACCI GOUVÊA, GERALDO LINO DE CAMPOS
- BT/PCS/9502 - Um Simulador de Arquiteturas de Computadores "A Computer Architecture Simulator" - CLAUDIO A. PRADO, WILSON V. RUGGIERO
- BT/PCS/9503 - Simulador para Avaliação da Confiabilidade de Sistemas Redundantes com Reparo - ANDRÉA LUCIA BRAGA, FRANCISCO JOSÉ DE OLIVEIRA DIAS
- BT/PCS/9504 - Projeto Conceitual e Projeto Básico do Nível de Coordenação de um Sistema Aberto de Automação, Utilizando Conceitos de Orientação a Objetos - NELSON TANOMARU, MOACYR MARTUCCI JUNIOR
- BT/PCS/9505 - Uma Experiência no Gerenciamento da Produção de Software - RICARDO LUIS DE AZEVEDO DA ROCHA, JOÃO JOSÉ NETO
- BT/PCS/9506 - MetodOO - Método de Desenvolvimento de Sistemas Orientado a Objetos: Uma Abordagem Integrada à Análise Estruturada e Redes de Petri - KECHI HIRAMA, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/9601 - MOOPP: Uma Metodologia Orientada a Objetos para Desenvolvimento de Software para Processamento Paralelo - ELISA HATSUE MORIYA HUZITA, LÍRIA MATSUMOTO SATO
- BT/PCS/9602 - Estudo do Espalhamento Brillouin Estimulado em Fibras Ópticas Monomodo - LUIS MEREGE SANCHES, CHARLES ARTUR SANTOS DE OLIVEIRA
- BT/PCS/9603 - Programação Paralela com Variáveis Compartilhadas para Sistemas Distribuídos - LUCIANA BEZERRA ARANTES, LÍRIA MATSUMOTO SATO
- BT/PCS/9604 - Uma Metodologia de Projeto de Redes Locais - TEREZA CRISTINA MELO DE BRITO CARVALHO, WILSON VICENTE RUGGIERO

- BT/PCS/9605 - Desenvolvimento de Sistema para Conversão de Textos em Fonemas no Idioma Português - DIMAS TREVIZAN CHBANE, GERALDO LINO DE CAMPOS
- BT/PCS/9606 - Sincronização de Fluxos Multimídia em um Sistema de Videoconferência - EDUARDO S. C. TAKAHASHI, STEFANIA STIUBIENER
- BT/PCS/9607 - A importância da Completeza na Especificação de Sistemas de Segurança - JOÃO BATISTA CAMARGO JÚNIOR, BENÍCIO JOSÉ DE SOUZA
- BT/PCS/9608 - Uma Abordagem Paraconsistente Baseada em Lógica Evidencial para Tratar Exceções em Sistemas de Frames com Múltipla Herança - BRÁULIO COELHO ÁVILA, MÁRCIO RILLO
- BT/PCS/9609 - Implementação de Engenharia Simultânea - MARCIO MOREIRA DA SILVA, MOACYR MARTUCCI JÚNIOR
- BT/PCS/9610 - Statecharts Adaptativos - Um Exemplo de Aplicação do STAD - JORGE RADY DE ALMEIDA JUNIOR, JOÃO JOSÉ NETO
- BT/PCS/9611 - Um Meta-Editor Dirigido por Sintaxe - MARGARETE KEIKO IWAI, JOÃO JOSÉ NETO
- BT/PCS/9612 - Reutilização em Software Orientado a Objetos: Um Estudo Empírico para Analisar a Dificuldade de Localização e Entendimento de Classes - SELMA SHIN SHIMIZU MELNIKOFF, PEDRO ALEXANDRE DE OLIVEIRA GIOVANI
- BT/PCS/9613 - Representação de Estruturas de Conhecimento em Sistemas de Banco de Dados - JUDITH PAVÓN MENDONZA, EDIT GRASSIANI LINO DE CAMPOS
- BT/PCS/9701 - Uma Experiência na Construção de um Tradutor Inglês - Português - JORGE KINOSHITA, JOÃO JOSÉ NETO
- BT/PCS/9702 - Combinando Análise de "Wavelet" e Análise Entrópica para Avaliar os Fenômenos de Difusão e Correlação - RUI CHUO HUEI CHIOU, MARIA ALICE G. V. FERREIRA
- BT/PCS/9703 - Um Método para Desenvolvimento de Sistemas de Computacionais de Apoio a Projetos de Engenharia - JOSÉ EDUARDO ZINDEL DEBONI, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/9704 - O Sistema de Posicionamento Global (GPS) e suas Aplicações - SÉRGIO MIRANDA PAZ, CARLOS EDUARDO CUGNASCA
- BT/PCS/9705 - METAMBI-OO - Um Ambiente de Apoio ao Aprendizado da Técnica Orientada a Objetos - JOÃO UMBERTO FURQUIM DE SOUZA, SELMA S. S. MELNIKOFF
- BT/PCS/9706 - Um Ambiente Interativo para Visualização do Comportamento Dinâmico de Algoritmos - IZAURA CRISTINA ARAÚJO, JOÃO JOSÉ NETO
- BT/PCS/9707 - Metodologia Orientada a Objetos e sua Aplicação em Sistemas de CAD Baseado em "Features" - CARLOS CÉSAR TANAKA, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9708 - Um Tutor Inteligente para Análise Orientada a Objetos - MARIA EMÍLIA GOMES SOBRAL, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/9709 - Metodologia para Seleção de Solução de Sistema de Aquisição de Dados para Aplicações de Pequeno Porte - MARCELO FINGUERMAN, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/9801 - Conexões Virtuais em Redes ATM e Escalabilidade de Sistemas de Transmissão de Dados sem Conexão - WAGNER LUIZ ZUCCHI, WILSON VICENTE RUGGIERO
- BT/PCS/9802 - Estudo Comparativo dos Sistemas da Qualidade - EDISON SPINA, MOACYR MARTUCCI JR.
- BT/PCS/9803 - The VIBRA Multi-Agent Architecture: Integrating Purposive Vision With Deliberative and Reactive Planning - REINALDO A. C. BIANCHI, ANNA H. REALI C. RILLO, LELIANE N. BARROS
- BT/PCS/9901 - Metodologia ODP para o Desenvolvimento de Sistemas Abertos de Automação - JORGE LUIS RISCO BECCERRA, MOACYR MARTUCCI JUNIOR
- BT/PCS/9902 - Especificação de Um Modelo de Dados Bitemporal Orientado a Objetos - SOLANGE NICE ALVES DE SOUZA, EDIT GRASSIANI LINO DE CAMPOS
- BT/PCS/9903 - Implementação Paralela Distribuída da Dissecção Cartesiana Aninhada - HILTON GARCIA FERNANDES, LIRIA MATSUMOTO SATO
- BT/PCS/9904 - Metodologia para Especificação e Implementação de Solução de Gerenciamento - SÉRGIO CLEMENTE, TEREZA CRISTINA MELO DE BRITO CARVALHO
- BT/PCS/9905 - Modelagem de Ferramenta Hipermídia Aberta para a Produção de Tutoriais Interativos - LEILA HYODO, ROMERO TORI
- BT/PCS/9906 - Métodos de Aplicações da Lógica Paraconsistente Anotada de Anotação com Dois Valores-LPA2v com Construção de Algoritmo e Implementação de Circuitos Eletrônicos - JOÃO I. DA SILVA FILHO, JAIR MINORO ABE
- BT/PCS/9907 - Modelo Nebuloso de Confiabilidade Baseado no Modelo de Markov - PAULO SÉRGIO CUGNASCA, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/9908 - Uma Análise Comparativa do Fluxo de Mensagens entre os Modelos da Rede Contractual (RC) e Colisões Baseada em Dependências (CBD) - MÁRCIA ITO, JAIME SIMÃO SICHMAN

- BT/PCS/9909 – Otimização de Processo de Inserção Automática de Componentes Eletrônicos Empregando a Técnica de Times Assíncronos – CESAR SCARPINI RABAK, JAIME SIMÃO SICHMAN
- BT/PCS/9910 – MIISA – Uma Metodologia para Integração da Informação em Sistemas Abertos – HILDA CARVALHO DE OLIVEIRA, SELMA S. S. MELNIKOFF
- BT/PCS/9911 – Metodologia para Utilização de Componentes de Software: um estudo de Caso – KAZUTOSI TAKATA, SELMA S. S. MELNIKOFF
- BT/PCS/0001 – Método para Engenharia de Requisitos Norteado por Necessidades de Informação – ARISTIDES NOVELLI FILHO, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/0002 – Um Método de Escolha Automática de Soluções Usando Tecnologia Adaptativa – RICARDO LUIS DE AZEVEDO DA ROCHA, JOÃO JOSÉ NETO
- BT/PCS/0101 – Gerenciamento Hierárquico de Falhas – JAMIL KALIL NAUFAL JR., JOÃO BATISTA CAMARGO JR.
- BT/PCS/0102 – Um Método para a Construção de Analisadores Morfológicos, Aplicado à Língua Portuguesa, Baseado em Autômatos Adaptativos – CARLOS EDUARDO DANTAS DE MENEZES, JOÃO JOSÉ NETO
- BT/PCS/0103 – Educação pela Web: Metodologia e Ferramenta de Elaboração de Cursos com Navegação Dinâmica – LUISA ALEYDA GARCIA GONZÁLEZ, WILSON VICENTE RUGGIERO
- BT/PCS/0104 – O Desenvolvimento de Sistemas Baseados em Componentes a Partir da Visão de Objetos – RENATA EVANGELISTA ROMARIZ RECCO, JOÃO BATISTA CAMARGO JÚNIOR
- BT/PCS/0105 – Introdução às Gramáticas Adaptativas – MARGARETE KEIKO IWAI, JOÃO JOSÉ NETO
- BT/PCS/0106 – Automação dos Processos de Controle de Qualidade da Água e Esgoto em Laboratório de Controle Sanitário – JOSÉ BENEDITO DE ALMEIDA, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/0107 – Um Mecanismo para Distribuição Segura de Vídeo MPEG – CÍNTIA BORGES MARGI, GRAÇA BESSAN, WILSON VICENTE RUGGIERO
- BT/PCS/0108 – A Dependence-Based Model for Social Reasoning in Multi-Agent Systems – JAIME SIMÃO SICHMAN
- BT/PCS/0109 – Ambiente Multilinguagem de Programação – Aspectos do Projeto e Implementação – APARECIDO VALDEMIRO DE FREITAS, JOÃO JOSÉ NETO
- BT/PCS/0110 – LETAC: Técnica para Análise de Tarefas e Especificação de Fluxo de Trabalho Cooperativo – MARCOS ROBERTO GREINER, LUCIA VILELA LEITE FILGUEIRAS
- BT/PCS/0111 – Modelagem ODP para o Planejamento de Sistemas de Potência – ANIRIO SALLES FILHO, JOSÉ SIDNEI COLOMBO MARTINI
- BT/PCS/0112 – Técnica para Ajuste dos Coeficientes de Quantização do Padrão MPEG em Tempo Real – REGINA M. SILVEIRA, WILSON V. RUGGIERO
- BT/PCS/0113 – Segmentação de Imagens por Classificação de Cores: Uma Abordagem Neural – ALEXANDRE S. SIMÕES, ANNA REALI COSTA
- BT/PCS/0114 – Uma Avaliação do Sistema DSM Nautilus – MARIO DONATO MARINO, GERALDO LINO DE CAMPOS
- BT/PCS/0115 – Utilização de Redes Neurais Artificiais para Construção de Imagem em Câmara de Cintilação – LUIZ SÉRGIO DE SOUZA, EDITH RANZINI
- BT/PCS/0116 – Simulação de Redes ATM – HSU CHIH WANG CHANG, WILSON VICENTE RUGGIERO
- BT/PCS/0117 – Application of Monoprocessed Architecture for Safety Critical Control Systems – JOSÉ ANTONIO FONSECA, JORGE RADY DE ALMEIDA JR.
- BT/PCS/0118 – WebBee – Um Sistema de Informação via WEB para Pesquisa de Abelhas sem Ferrão – RENATO SOUSA DA CUNHA, ANTONIO MOURA SARAIVA
- BT/PCS/0119 – Parallel Processing Applied to Robot Manipulator Trajectory Planning – DENIS HAMILTON NOMIYAMA, LÍRIA MATSUMOTO SATO, ANDRÉ RIYUITI HIRAKAWA
- BT/PCS/0120 – Utilização de Padrão de Arquitetura de Software para a Fase de Projeto Orientado a Objetos – CRISITINA MARIA FERREIRA DA SILVA, SELMA SHIN SHIMIZU MELNIKOFF
- BT/PCS/0121 – Agilizando Aprendizagem por Reforço Através do uso de Conhecimento sobre o Domínio – RENÉ PEGORARO, ANNA H. REALI COSTA
- BT/PCS/0122 – Modelo de Segurança da Linguagem Java Problemas e Soluções – CLAUDIO MASSANORI MATAYOSHI, WILSON VICENTE RUGGIERO
- BT/PCS/0123 – Proposta de um Agente CNM para o Gerenciamento Web de um Backbone ATM – FERNANDO FROTA REDÍGOLO, TEREZA CRISTINA MELO DE BRITO CARVALHO
- BT/PCS/0124 – Um Método de Teste de software Baseado em Casos Teste – SÉRGIO RICARDO ROTTA, KECHI HIRAMA
- BT/PCS/0201 – A Teoria Nebulosa Aplicada a uma Bicicleta Ergométrica para Fisioterapia – MARCO ANTONIO GARMS, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/0202 – Synchronization Constraints in a Concurrent Object Oriented Programming Model – LAÍS DO NASCIMENTO SALVADOR, LÍRIA MATSUMOTO SATO

- BT/PCS/0203 – Construção de um Ambiente de Dados sobre um Sistema de Arquivos Paralelos – JOSÉ CRAVEIRO DA COSTA NETO, LIRIA MATSUMOTO SATO
- BT/PCS/0204 – Maestro: Um Middleware para Suporte a Aplicações Distribuídas Baseadas em Componentes de Software – CLÁUDIO LUÍS PEREIRA FERREIRA, JORGE LUÍS RISCO BECERRA
- BT/PCS/0205 – Sistemas de Automação dos Transportes (ITS) Descritos Através das Técnicas de Modelagem RM-OPD (ITU-T) e UML (OMG) – CLÁUDIO LUIZ MARTE, JORGE LUÍS RISCO BECERRA, JOSÉ SIDNEI COLOMBO
- BT/PCS/0206 – Comparação de Perfis de Usuários Coletados Através do Agente de Interface PersonalSearcher – GUSTAVO A. GIMÉNEZ LUGO, ANALÍA AMANDI, JAIME SIMÃO SICHMAN
- BT/PCS/0207 – Arquitetura Reutilizáveis para a Criação de Sistemas de Tutorização Inteligentes – MARCO ANTONIO FURLAN DE SOUZA, MARIA ALICE GRIGAS VARELLA FERREIRA
- BT/PCS/0208 – Análise e Predição de Desempenho de Programas Paralelos em Redes de Estações de Trabalho – LIN KUAN CHING, LIRIA MATSUMOTO SATO
- BT/PCS/0209 – Previsões Financeiras Através de Sistemas Neuronebulosos – DANIEL DE SOUZA GOMES, MARCO TÚLIO CARVALHO DE ANDRADE
- BT/PCS/0210 – Proposta de Arquitetura Aberta de Central de Atendimento – ANA PAULA GONÇALVES SERRA, MOACYR MARTUCCI JÚNIOR