

UM CLASSIFICADOR DE SINAIS PROJETADO COM PRINCÍPIOS DAS REDES NEURAIS ARTIFICIAIS PARA CONSISTENTES

Maurício Conceição Mário¹ e João Inácio da Silva Filho²

Resumo - Neste artigo, com base nos conceitos teóricos da Lógica Paraconsistente Anotada, apresentamos um Sistema capaz de classificar sinais. A lógica paraconsistente Anotada LPA, conforme apresentada em [1] é uma classe de Lógica Não Clássica que permite manipular sinais contraditórios sem trivialização. Em [5] foram apresentadas Células Neurais Artificiais Paraconsistentes construídas com Algoritmos baseados na LPA onde ficou demonstrado sua capacidade de aprender e desaprender determinados sinais em forma de funções aplicados em suas entradas. A partir dos algoritmos das Células foram feitas conexões e agrupamentos entre os algoritmos para se criar um Sistema Paraconsistente Classificador de Sinais – SPCS. O SPCS é capaz de aprender quatro tipos de sinais diferentes. Depois do aprendizado dos quatro sinais o SPCS pode reconhecer se o sinal apresentado na entrada pertence ou não ao grupo de sinais aprendido anteriormente. Os resultados obtidos com o SPCS demonstram que os algoritmos das Células Neurais Artificiais Paraconsistentes interligados desta forma se transformam em um Sistema Paraconsistente Classificador de Sinais robusto indicado para utilização em vários campos da área de Inteligência Artificial como os Sistemas Especialistas e de Reconhecimento de palavras e imagens.

Palavras chaves: lógica paraconsistente, lógica paraconsistente anotada, redes neurais, redes neurais paraconsistentes, neurocomputação.

1 – INTRODUÇÃO

Do ponto de vista de engenharia, sinais são funções ou seqüências que transportam informações de uma fonte de mensagens a um destinatário. As características específicas dos sinais dependem do canal de comunicações utilizado para o transporte do sinal. Um canal de comunicações é definido pelo tipo de distorção que introduz nos sinais, podendo esta ser do tipo: a) determinística linear (limitação da banda de frequência dos sinais); b) determinística não linear (existência de saturações); c) aleatória (presença de ruídos) [9].

Em sistemas eletrônicos de comunicação a fonte geradora de informação, o canal de comunicação, o canal de

comunicação e o destinatário são elementos pré-definidos, geralmente com características bem especificadas.

Em outras situações, como nos processos de medição em investigação científica, a fonte de mensagens e o canal de comunicações poderão estar apenas parcialmente caracterizados. Sinais bioelétricos como o eletrocardiograma (EKG), o eletroencefalograma (EEG), o eletromiograma (EMG) ou o eletroneurograma (ENG) são estudados há décadas com a finalidade de se extrair informação sobre estados patológicos de órgãos, sem que se tenha muitas vezes a certeza de que tal informação é de fato transportada por estes sinais.

Os sinais que constituem a voz humana codificam uma variedade de informações: sobre a semântico que está sendo dito, sobre a identidade do locutor, etc [9].

1.1. CLASSIFICAÇÃO DE SINAIS

Dado o sinal representado pela equação:

$$y(t) = f(t, A) + n(t)$$

Supondo que o sinal $y(t)$ possa pertencer a categorias $\{C_j\}^{M, j=1}$, conforme o valor do parâmetro A e particularizando para o caso, por exemplo, da tentativa de se diagnosticar condições cardíacas patológicas a partir de traçados de EKG, então para cada categoria diagnóstica j é escolhida uma coleção de traçados representativos:

$$\{y_{ij}(t)\}^{N_j, i=1} \quad j=1, 2, \dots, M$$

Conforme exposto em [9] uma tentativa de classificação de sinais pode ser feita fazendo-se uma abordagem direta com técnicas de Redes Neurais. Neste caso é feito o treinamento em uma rede neural multicamada $N(W)$ que terá como entradas todos os N pontos das séries temporais associadas a cada um dos sinais, e como saídas as M categorias em que se pretende classificá-los, como mostrado na figura 1:

Um problema para este tipo de abordagem é que cada série temporal é representada por cerca de 500 pontos amostrais para cada ciclo de EKG, o que impõe uma dimensão grande à rede, mesmo havendo poucas categorias de classificação.

Uma abordagem mais econômica é procurar reduzir a dimensão dos sinais a uma coleção menor de parâmetros que representem adequadamente os sinais originais e, implementar uma classificação neural para estes parâmetros.

¹ Maurício Conceição Mário, UNISANTA - Universidade Santa Cecília -Rua Osvaldo Cruz, 266 CEP-110045- Santos – SP - Brazil cmario@stceecilia.br

² João Inácio da Silva Filho IEA - Instituto de Estudos Avançados da Universidade de São Paulo Av. Prof. Luciano Gualberto, 374 Trav. J, Térreo, Cidade Universitária- ina@unimes.com.br

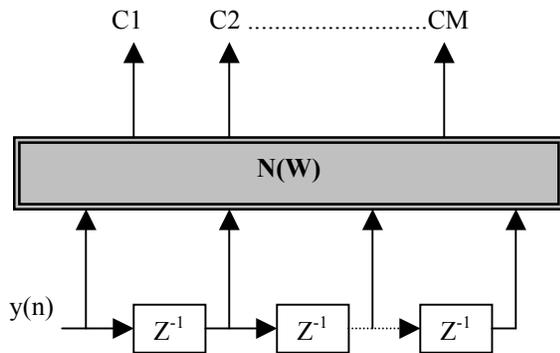


Fig.1 Rede Neural Artificial Multicamada na Classificação de Sinais.

A literatura apresenta vários métodos de classificação de sinais como em [9] onde se faz a parametrização e efetua as medições de amplitudes de pontos fiduciais (ondas P, Q, R, S, T, U) no traçado, áreas associadas, durações e separações. Um método muito estudado é o proposto por Halliday [7], onde a coleção de todos os sinais é modelada por um processo aleatório $Y(t)$ do qual cada traçado individual é uma particular função amostra. Neste processo é feita uma classificação de modo aleatório e em seguida é representado por uma expansão ortonormal como o de Kahunen-Loeve. Por esta expansão, determinam-se a seguir funções ortonormais $\{\Phi_i(t)\}$ chamadas de autofunções da função de covariância associada ao processo, tais que:

$$\int_0^T \Phi_i(t) \Phi_k(t) dt = \delta(i,k)$$

$$y_k(t) = \lim_{n \rightarrow \infty} \sum_{i=1}^n a_{ki} \Phi_i(t)$$

e os coeficientes associados a cada um dos sinais são

$$\text{calculados por: } a_{ki} = \int_0^T y_k(t) \Phi_i(t) dt$$

Ordenando as médias destes coeficientes calculadas sobre toda a população de sinais em ordem decrescente dos seus valores absolutos, pode-se aproximar cada sinal individual apenas pelas L autofunções mais significativas:

$$\hat{y}_k(t) = \sum_{i=1}^L a_{ki} \Phi_i(t)$$

Deste modo, espera-se que o vetor de coeficientes a_k de dimensão $L \ll N$ represente adequadamente cada sinal com um erro de reconstrução suficientemente pequeno.

Com este modelo matemático que foi utilizado em [7] para a classificação de traçados VCG's (vector cardiogramas), pode-se implementar uma rede neural multicamada de dimensão mais viável, que, utilizando apenas estes coeficientes a_k , classifique adequadamente os sinais nas J categorias.

Neste trabalho apresentamos uma nova metodologia onde um Sistema Paraconsistente Classificador de Sinais construído com base na Lógica Paraconsistente Anotada [5] utiliza Células Neurais Artificiais Paraconsistente devidamente configuradas para aprender e classificar sinais

usando Algoritmos muitos simples e capazes de ser controlado para efetuar classificação de categorias de sinais.

2 – A LÓGICA PARACONSISTENTE ANOTADA COM ANOTAÇÕES DE DOIS VALORES - LPA2v

As contradições ou inconsistências são comuns quando descrevemos partes do mundo real. Os sistemas de análises e tratamento de sinais utilizados em Inteligência Artificial funcionam em geral com base na lógica convencional, onde a descrição do mundo é considerada por dois estados: Falso ou Verdadeiro. Estes sistemas binários não conseguem tratar adequadamente as situações contraditórias. As Lógicas Paraconsistentes nasceram da necessidade de se encontrar meios de dar tratamento não trivial às situações contraditórias. Os estudos e propostas das Lógicas Paraconsistentes apresentaram resultados que possibilitam considerar as inconsistências em sua estrutura de um modo não trivial [4] e [1], e por isso, se mostram mais propícias no enquadramento de problemas ocasionados por situações de contradições que aparecem quando lidamos com o mundo real.

A Lógica Paraconsistente Anotada LPA é uma classe de lógica Paraconsistente Evidencial que faz tratamento de sinais representados por anotações permitindo uma descrição e equacionamento por meio de Algoritmos.

Na Lógica Paraconsistente Anotada LPA as fórmulas proposicionais vêm acompanhadas de anotações. Cada anotação, pertencente a um reticulado finito \mathcal{T} , atribui valores à sua correspondente fórmula proposicional. Uma Lógica Paraconsistente Anotada LPA pode ter como reticulado finito, o de “quatro estados”, conforme a figura 2.

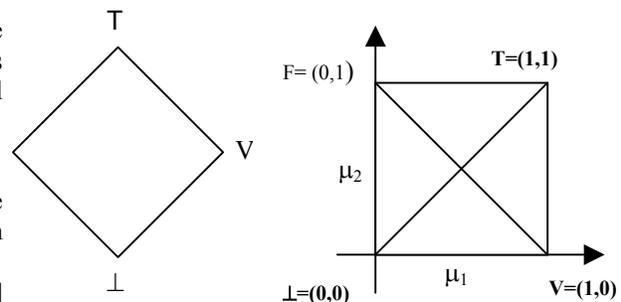


Fig.2 Reticulado finito e Quadrado no Plano Cartesiano.

A Lógica Paraconsistente Anotada com anotação de dois valores - LPA2v é uma extensão da LPA e pode ser representada através de um reticulado de quatro vértices [2] onde podemos estabelecer algumas terminologias e convenções, do seguinte modo:

Seja $\tau = \langle |\tau|, \leq \rangle$ um reticulado finito fixo, onde:

- $|\tau| = [0, 1] \times [0, 1]$
- $\leq = \{((\mu_1, \rho_1), (\mu_2, \rho_2)) \in ([0, 1] \times [0, 1])^2 \mid \mu_1 \leq \mu_2 \text{ e } \rho_1 \leq \rho_2\}$ (onde \leq indica a ordem usual dos números reais). Tal reticulado denomina-se *reticulado de valores-verdade*.

3 – AS CÉLULAS NEURAIS ARTIFICIAIS PARACONSISTENTES

Na análise paraconsistente o objetivo principal é saber com que medida ou grau de certeza podemos afirmar que uma proposição é Falsa ou Verdadeira. Portanto, é considerado como resultado da análise apenas o valor do grau de certeza G_c . O valor do grau de contradição G_{ct} é um indicativo que informa a medida da inconsistência. Se houver um baixo valor de certeza ou muita inconsistência o resultado é uma indefinição. Estes valores podem retornar em dois eixos sobrepostos representando a reticulado finito, agora com valores, conforme a figura 4 abaixo:

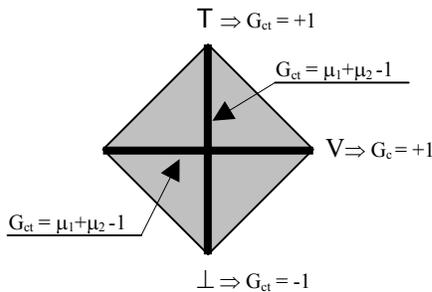


Fig.3 Reticulado finito da LPA2v com valores.

Os valores de controle ajustados externamente são limites que vão servir como referência na análise. Podemos descrever a análise paraconsistente utilizando apenas as equações originadas no Quadrado Unitário do Plano Cartesiano.

Uma descrição do reticulado utilizando os valores obtidos pelas equações resulta no Algoritmo denominado “Para-Analisador” [5]. Este algoritmo é elaborado com base na descrição do reticulado e pode ser escrito na sua forma reduzida expressando assim uma Célula Neural Artificial Paraconsistente básica CNAPb conforme é descrito a seguir:

**/Definições dos valores ajustáveis*/*
 $V_{scc} = C_1$ **/Valor superior de controle de certeza*/*
 $V_{icc} = C_2$ **/Valor inferior de controle de certeza*/*
 $V_{sct} = C_3$ **/Valor superior de controle contradição*/*
 $V_{ict} = C_4$ **/Valor inferior de controle contradição*/*
**/Variáveis de entrada*/*

μ_1, μ_2
**/Variáveis de saída*/*

Saída Digital = S_1
 Saída analógica = S_{2a}
 Saída analógica = S_{2b}

**/Expressões matemáticas*/*

sendo : $G_{ct} = \mu_1 + \mu_2 - 1$

e $G_c = \mu_1 - \mu_2$
**/determinação dos estados lógicos extremos*/*

Se $G_c \geq C_1$ então $S_1 = V$

Se $G_c \leq C_2$ então $S_1 = F$

Se $G_{ct} \geq C_3$ então $S_1 = T$

Se $G_{ct} \leq C_4$ então $S_1 = \perp$

Senão $S_1 = I$ -Indefinição

$G_{ct} = S_{2a}$

$G_c = S_{2b}$

**/ FIM*/*

Denomina-se *Célula Artificial Paraconsistente básica* (CAPb) o elemento capaz de, composto de quando um par de graus de crença e descrença (μ_{1a}, μ_{2a}) for apresentado na sua entrada fornecer um resultado na sua saída na forma de uma tripla dada por: $G_{ct} =$ grau de contradição resultante, $G_c =$ grau de crença resultante e $X =$ constante de anotação resultante Indefinido. A figura 4 mostra a representação de uma CNAPb.

CÉLULA NEURAL ARTIFICIAL PARACONSISTENTE BÁSICA

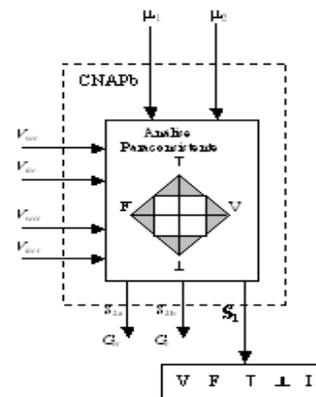


Fig.4 A Célula Neural Artificial Paraconsistente Básica CNAPb.

Na Célula Neural Artificial Paraconsistente básica CNAPb representada pelo algoritmo “Para-Analisador”, foram considerados todos os valores envolvidos nas equações, sendo assim, se os valores dos graus de certeza e de contradição estiverem fora dos valores impostos pelos limites ajustáveis, a saída é um estado denominado de *não-extremo*, sendo a este atribuído um valor indefinido I . A partir da *Célula Artificial Paraconsistente básica* (CAPb) foram criadas outras Células as quais denominamos de *Células Neurais Artificiais Paraconsistentes* (CNAP’s).

Os estudos das CNAP deram origem a uma família de Células Neurais Artificiais Paraconsistentes que constituem os elementos básicos das *Redes Neurais Artificiais Paraconsistentes* (RNAP’s). Neste trabalho para elaboração do Classificador de Sinais foram necessários apenas três tipos [5] de Células denominadas: Célula Neural Artificial Paraconsistente de Aprendizagem CNAPap que consegue

March 16 - 19, 2003, São Paulo, BRAZIL

International Conference on Engineering and Computer Education

aprender e memorizar um padrão aplicado em sua entrada, uma Célula Neural Artificial Paraconsistente de Conexão Lógica Simples de Maximização - CNAPLs que determina a sua saída pelo maior valor aplicado na entrada e uma Célula Neural Artificial Paraconsistente de Decisão - CNAPd que determina a saída final resultante da análise Paraconsistente.

5. O CLASSIFICADOR PARACONSISTENTE DE SINAIS

Neste trabalho o Classificador Paraconsistente de Sinais é composto por módulos básicos denominados de Unidade Neural Artificial Paraconsistente de Comparação de Padrões – UNAPCP que armazenarão sinais para serem comparados com aqueles que serão aplicados na entrada. Cada Unidade – UNAPCP é composta, por sua vez, de três Células Neurais Artificiais Paraconsistentes; uma Célula Neural Artificial Paraconsistente de aprendizagem CNAPCa, uma Célula Neural Artificial Paraconsistente de Conexão Lógica Simples de Maximização - CNAPLs e uma Célula Neural Artificial Paraconsistente de Decisão - CNAPd. O diagrama em blocos da figura 5 mostra como se compõem o Classificador Paraconsistente de Sinais.

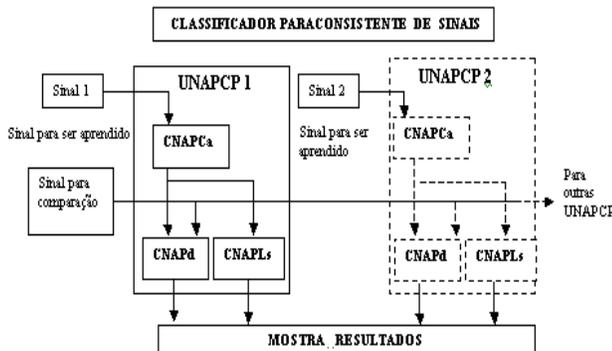


Fig.5. Diagrama mostrando interconexões dos módulos do Classificador paraconsistente de Sinais.

Neste trabalho optamos por construir um Classificador que permite comparação com quatro tipos de sinais, portanto com quatro módulos de Unidade Neural Artificial Paraconsistente de Comparação de Padrões – UNAPCP, no entanto mais módulos podem ser agregados para expandir a capacidade do Classificador. A quantidade de módulos vai depender da natureza da aplicação do projeto.

3.CÉLULA NEURAL ARTIFICIAL PARACONSISTENTE DE APRENDIZAGEM- CNAP-Ap

A Célula Neural Artificial Paraconsistente de aprendizagem CNAPAp é uma Célula Neural Artificial Paraconsistente Básica com uma saída μ_{1r} interligada à entrada do grau de descrença complementado μ_{2c} . Conforme

pode ser visto no Algoritmo sucessivos valores aplicados à entrada do grau de crença μ_1 resulta no aumento gradativo no grau de crença resultante da saída μ_{1r} . Esta Célula pode funcionar de dois modos, para aprendizado do *padrão de verdade*, onde são aplicados valores $\mu_1=1$ sucessivamente até o grau de crença resultante na saída chegar à $\mu_{1r}=1$, e para o aprendizado do *padrão de falsidade* onde são aplicados valores $\mu_1=0$ até o grau de crença resultante chegar à $\mu_{1r}=1$, neste caso a entrada do grau de crença é complementada μ_{1c} conforme mostra a figura 6

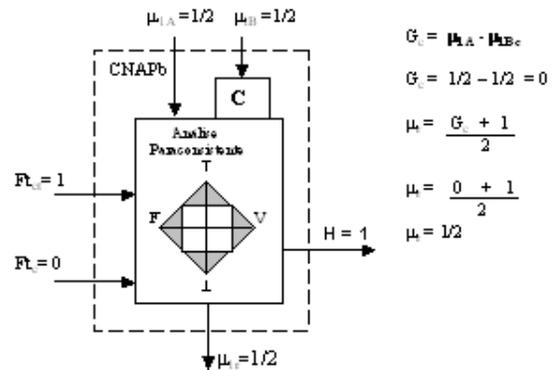


Fig 6- Célula Neural Artificial Paraconsistente de aprendizagem pronta para receber padrões.

Algoritmo completo de Aprendizagem da Célula Neural Artificial Paraconsistente - CNAPa

- 1- Início: $\mu_r = 1/2$ */Célula virgem */
- 2- Defina: $F_A =$ Valor onde: $F_A \geq 1$ */ Entra com o valor do Fator de aprendizagem */
- 3- Defina: $F_{DA} =$ Valor : $F_{DA} \geq 1$ */ Entra com o valor do Fator de desaprendizagem */
- 4- Defina: P */ Padrão de entrada , $0 \leq P \leq 1$ */
- 5- Faça: $G_{ci} = P - \mu_{2c}$ */ Calcula o Grau de crença inicial*/
- 6- Se $G_{ci} < 0$ Faça: $\mu_1 = 1 - P$ */ O grau de crença é o complemento do padrão */
- 7- Se $G_{ci} > 0$ Faça: $\mu_1 = P$ */ O grau de crença é o Padrão */
- 8- Faça : $\mu_2 = \mu_r$ */ Conecta a saída da célula na Entrada do grau de descrença */
- 9- Faça: $\mu_{2c} = 1 - \mu_2$ */ Aplica o Operador Complemento no valor da entrada do grau de descrença */
- 10- Faça: $G_c = \mu_1 - \mu_{2c}$ */ Calcula o Grau de crença */
- 11- Se $G_c \geq 0$ faça $C_1 = F_A$
- 12- Se $G_c < 0$ faça $C_1 = F_{DA}$
- 13- Faça: $\mu_r = \{(G_c \times C_1) + 1\} \div 2$ */ Encontra o grau de crença resultante da saída pela EEB */
- 14- Enquanto $\mu_r \neq 0$ retorne ao passo 8
- 15- Se $\mu_r = 0$ Faça: $\mu_{1r} = 1$ e $\mu_1 = 1 - P$ */ Aplica o Operador NOT e complementa o grau de crença */

6. RESULTADOS PRÁTICOS

Com base na teoria exposta foi elaborado um software Classificador Paraconsistente de Sinais capaz de aprender sinais e compara-los com padrões inseridos na entrada. A figura 7 mostra a tela principal do Classificador Paraconsistente de Sinais proposto neste trabalho com as legendas explicativas.

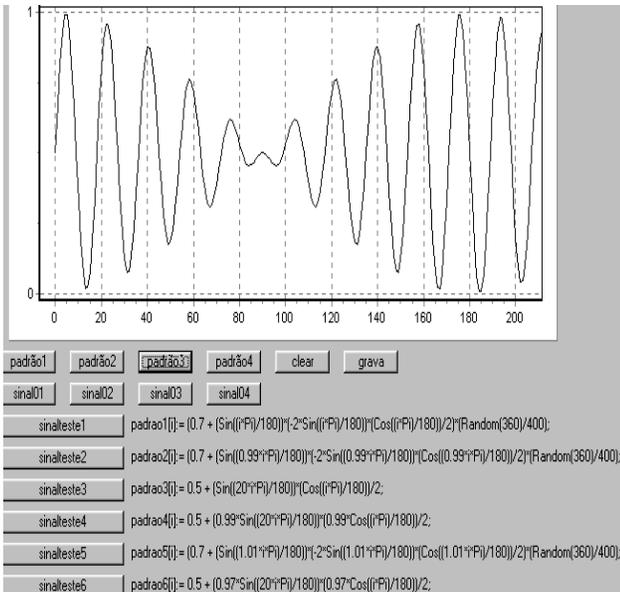


Fig 7- Tela principal do Classificador de Sinais

Os botões na janela do Software permite que o classificador aprenda 4 tipos de sinais com as equações correspondentes. Na análise o software reconhece apenas o sinal apreendido anteriormente, conforme mostra a figura 8.

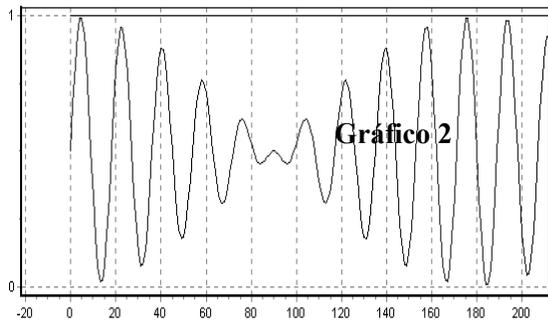


Fig 8- Forma de Onda reconhecida pelo Classificador de Sinais

O exemplo exposto na figura mostra apenas o sinal padrão 3 sendo desnecessário a apresentação dos outros sinais aprendidos.

7-REFERENCIAS

- [1] ABE, J. M “Fundamentos da Lógica Anotada” (Foundations of Annotated Logics), in Portuguese, Ph D thesis, University of São Paulo, FFLCH/USP - São Paulo, 1992.
- [2] ABE, J.M. & DA SILVA FILHO, J.I “Inconsistency and Electronic Circuits”, *Proceedings of The International ICSC Symposium on Engineering of Intelligent Systems (EIS'98)*, Vol3, Artificial Intelligence, Editor: E. Alpaydin, ICSC Academic Press International Computer Science Conventions Canada/Switzerland, ISBN 3-906454-12-6, 191-197, 1998.
- [3] BISHOP, C.M. *Neural Networks for Pattern Recognition*. 1.ed. Oxford University Press, 1995.
- [4] DA COSTA, N.C.A. & ABE, J.M. & SUBRAHMANNIAN, V.S., “Remarks on Annotated Logic” *Zeitschrift für Mathematische Logik und Grundlagen der Mathematik*, Vol.37, 561-570, 1991.
- [5] DA SILVA FILHO, J.I.& ABE, J.M. *Fundamentos das Redes Neurais Artificiais - destacando aplicações em Neurocomputação*. 1.ed. São Paulo, Editora Villipress, Brazil 2001.
- [6] FAUSETT, L. *Fundamentals of Neural Networks, Architectures, Algorithms and Applications* Editora Prentice-Hall, Englewood Cliffs – 1994
- [7] HALLIDAY, J.S., *The Characterization of Vectorcardiograms for Pattern Recognition* – Master Thesis, MIT, Cambridge, 1973.
- [8] HEBB, D. “*The Organization of Behavior*” Wiley, New York, 1949.
- [9] KOVÁCS, ZSOLT LÁSZLO, *Redes Neurais Artificiais – Fundamentos e Aplicações*, collegium cognitio, 1996.
- [10] MCCULLOCH, W & PITTS, W. “A Logical Calculus of the Ideas Immanent in Nervous Activity” *Bulletin of Mathematical Biophysics*, 1943.
- [11] RICH, E. & KNIGHT, K. “*Artificial Intelligence*” *Mc Graw Hill, NY*, 1983.
- [12] SIEBERT, W. “Stimulus Transformation in Peripheral Auditory System in Recognizing Patterns” Ed. Murray Eden, MIT Press, Cambridge, 1968.
- [13] SUZUKI, Y., *Self-Organizing QRS-Wave Recognition in ECG Using Neural Networks* – IEEE Trnsd. On Neural Networks, 6, 1995.
- [14] SUBRAHMANNIAN, V.S “On the semantics of quantitative Logic programs” *Proc. 4 th. IEEE Symposium on Logic Programming, Computer Society press, Washington D.C.*, 1987.
- [15] Rosenblatt, “*Principles of Neurodynamics*” *Spartan Books, NY*, 1962.
- [16] ZERBINI, R.C., *Metodologia para a Classificação de VCG através de redes neurais* –Dissertação de Doutorado, Escola Politécnica da USP, Brazil 1993.