ISSN: 2594-6153

Sistema Paraconsistente de Aprendizado por Repetição Aplicado em *Chip* ATMEGA 328p

Paulino M. Gomes[1,2], João Inácio da Silva Filho[1]

[1]Unisanta - Universidade Santa Cecília - Programa de Pós-Graduação em Engenharia Mecânica - PPGEMec
Rua Oswaldo Cruz, 266- Santos-SP, Brasil

[2]FATEF - Faculdade de Tecnologia de São Vicente
Av. Presidente Wilson, 1013, Itararé, São Vicente, Brasil
e-mail: paulino.machado.g@gmail.com

Resumo: O presente artigo tem por objetivo, apresentar uma modelagem da Lógica Paraconsistente através de uma Célula Neural Artificial Paraconsistente de Aprendizagem Física (CNAPap). Construída a partir de uma Célula Paraconsistente básica, representada pelo algoritmo que descreve o reticulado associado à Lógica Paraconsistente Anotada denominado de "Para-Analisador" (Da Silva Filho, 2006). Neste trabalho uma Célula Paraconsistente de Aprendizagem física é desenvolvida, inserindo o modelo matemático originado do algoritmo em um Chip, implementado por uma linguagem de programação estruturada. Através de ensaios e uma modelagem utilizando o algoritmo de aprendizagem, é demonstrado que uma CNAPap pode aprender e guardar padrões, com características similares aos Neurônios Biológicos do cérebro humano. Os resultados dos ensaios e da modelagem utilizando o algoritmo de aprendizagem, são apresentados no final deste trabalho, demonstrando as características funcionais da CNAPap.

Palavras chave: Célula Neural artificial, Lógica Paraconsistente, Algoritmo, Inteligência Artificial.

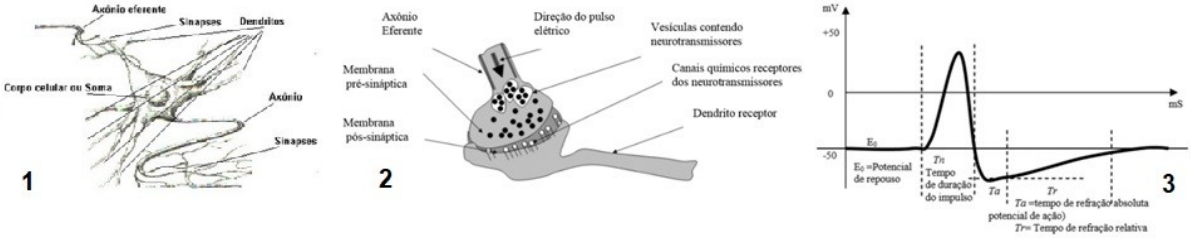
Paraconsistent Repetition Learning System Applied to ATMEGA 328p Chip

Abstract: This article aims to present a modeling of Paraconsistent Logic through a Paraconsistent Artificial Neural Cell of Physical Learning (CNAPap). Constructed from a basic Paraconsistent Cell, represented by the Algorithm that describes the lattice associated with the Annotated Paraconsistent Logic called "Para-Analyzer" (Da Silva Filho, 2006). In this work a Paraconsistent Physical Learning Cell is developed by inserting the mathematical model originated from the algorithm into a Chip, implemented by a structured programming language. Through testing and modeling using the learning algorithm, it is demonstrated that a CNAPap can learn and save patterns with characteristics similar to Biological Neurons in the human brain. The results of the tests and modeling using the learning algorithm are presented at the end of this work, demonstrating the functional characteristics of CNAPap.

Keywords: Artificial Neural Cell, Paraconsistent Logic, Algorithm, Artificial Intelligence.

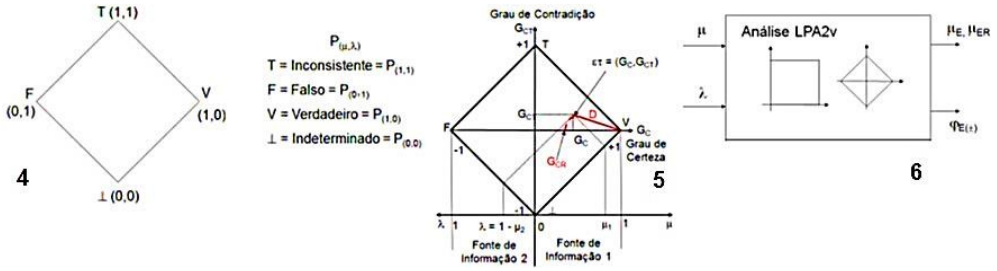
Introdução

Através da biociência, compreendemos que o cérebro humano é um emaranhado de conexões entre prolongamentos de células nervosas chamadas neurônios. Para estas funções mentais existem vários tipos de neurônios, no entanto um neurônio típico pode ser estudado, dividindo em três partes principais: A árvore dendrital, o corpo celular ou soma, e o axônio.



1 - Neurônio Biológico Típico
2 - Representação esquemática da Sinapse de recepção
3 - Forma de onda do *potencial de ação*
Fonte: Da Silva Filho (2008)

No cérebro humano há uma complexa rede composta de bilhões de neurônios interconectados. O contato para a passagem de informação entre os axônios e os dendritos é devido às estruturas chamadas de Sinapses (Figura1). A Sinapse pode ser dividida em duas partes por uma região chamada fenda sináptica. A parte anterior é composta pela membrana do axônio pelo qual chega o sinal de informação (pré-sináptica), na forma de um pulso elétrico e uma parte posterior que é composta pela membrana do dendrito (pós-sináptica) que receberá a informação (Figura2). A informação chegando na forma de um impulso elétrico na membrana pré-sináptica faz com que apareçam vesículas com mediadores químicos chamados de neurotransmissores. O neurônio biológico recebe e envia informações através de conexões sinápticas que propagam as informações nos neurotransmissores possibilitando a formação de um *potencial de ação* no axônio do neurônio receptor. O *potencial de ação* é o responsável pelo envio de informação entre os neurônios na forma de um impulso elétrico. Quando a membrana sofre uma depolarização acentuada a um certo limiar de disparo, gera impulso elétrico. O gráfico (Figura3) mostra a forma de onda de um *potencial de ação*, que é um pulso gerado no axônio de um neurônio típico, onde a duração varia de um micro a um milisegundo (Da Silva Filho, 2008).



4 - Reticulado finito de Hasse associado à LPA2v.; **5** - Reticulado LPAv2 com Grau de certeza (GC); Grau de Contradição (GCT) e Grau de Certeza real (GCR).;**6** - Representação do algoritmo de Análise Paraconsistente LPA2v.; **7** – Símbolo da CNAP padrão. **Fonte:** De Carvalho Jr, A., 2017

Lógica Paraconsistente Anotada com anotação de dois valores LPA2v

Na Lógica Paraconsistente Anotada com Anotação de dois valores (LPA2v) a representação de evidências em um reticulado no plano real, é formada por pares que

compõem umas anotações (μ, λ). A proposição *P* é suportada pelo grau de evidências favoráveis, ou (μ). O grau das evidências desfavoráveis que negam ou rejeitam a proposição *P* é representada pelo elemento (λ). Através de transformações lineares entre um Quadrado Unitário no Plano Cartesiano (QUPC) (Figura4) e o reticulado associado à LPA é possível se chegar às equações matemáticas relacionadas à LPA2v. Onde μ1 é o grau de evidência favorável, proveniente da primeira evidência; λ é o grau de evidência desfavorável, como o complemento da segunda evidência; G_C é o grau de certeza; G_{CT} é o grau de contradição. A análise paraconsistente gera valores de G_C e G_{CT} entre 1 e -1 (Figura5). Para ser consistente com as entradas (μ, λ), em aplicações práticas eles precisam ser normalizados e seus valores limitados novamente entre 0 e 1 para respeitar os limites da lógica paraconsistente. Portanto, o μ_E é o grau de evidência resultante, sendo a saída normalizada de G_C, enquanto que μ_{CRT} é a saída normalizada de G_{CT}. Conforme visto em [1] [8] pode-se obter um resultado mais preciso extraindo-se os efeitos da contradição em sucessivas análises de modo a obter o grau de certeza resultante real (G_{CR}), (Figura 6), e apresentado nas equações matemáticas (Figura7) (DA Silva Filho, 2007):

$$\lambda = 1 - \mu 2 \quad (1)$$
$$G_C = \mu 1 - \lambda \quad (2)$$
$$G_{CT} = \mu 1 + \lambda - 1 \quad (3)$$
$$\mu_E \frac{G_C+1}{2} \Rightarrow \mu_E \frac{\mu-\lambda+1}{2} \quad (4)$$
$$\mu_{CTR} \frac{\mu+\lambda}{2} \quad (5)$$

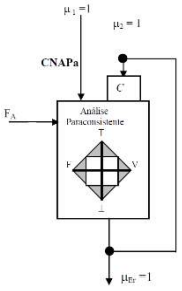
7

$$D = \sqrt{(1-|G_C|)^2 + G_{CT}{}^2} \quad (6)$$
$$\text{Se } G_C > 0$$
$$G_C > 0 \rightarrow G_{CR} = 1 - D; \quad G_C < 0 \rightarrow G_{CR} = D - 1; \quad (7)$$
$$\emptyset = 1 - |G_{CT}| \quad (8)$$
$$\mu_{ER} = \frac{G_{CR}+1}{2} \quad (9)$$
$$\varphi_E = 1 - |2 * \mu_{CTR} - 1| \quad (10)$$

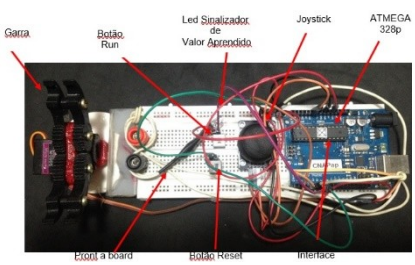
7 – Equações Matemáticas da LPA2v.;
Fonte: De Carvalho Jr, A., 2017

Material e Métodos

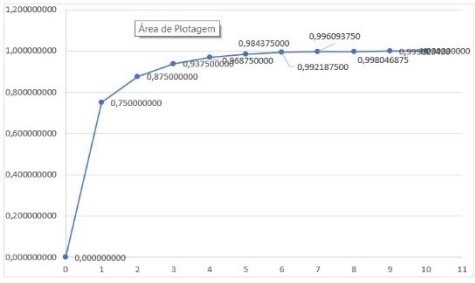
Foram utilizadas funções da programação estruturada em linguagem C++, para implementar o modelo matemático originado do algoritmo da Célula Neural Artificial Paraconsistente de Aprendizagem (CNAPap). Utilizando limites correspondentes aos consolidados e comparando os resultados com os dados, concluiu-se que a modelagem era compatível.



8 - Símbolo da CNAPap.;
Fonte: De Carvalho Jr, A., 2017



9 - Protótipo da CNAPap.;



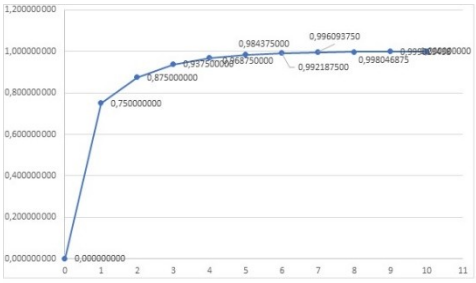
10 - Dados Obtidos com o algoritmo LPAV2 aplicado no CHIP ATMEGA 328p LPA2v.

Resultados

As Figuras 11e 12, apresentam os resultados obtidos durante a inserção de variáveis numéricas, durante a execução do programa. Como próximo passo, foi necessário a criação de uma modelagem física que interpretasse a programação estruturada e os sinais analógicos de entrada e saída.

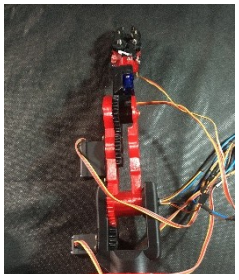
Passos	FA	µ1	Gc	Gct	µRE
0	1,000000000	1,000000000	0,000000000	0,500000000	0,000000000
1	1,000000000	1,000000000	0,500000000	0,500000000	0,750000000
2	1,000000000	1,000000000	0,750000000	0,250000000	0,875000000
3	1,000000000	1,000000000	0,875000000	0,125000000	0,937500000
4	1,000000000	1,000000000	0,937500000	0,062500000	0,968750000
5	1,000000000	1,000000000	0,968750000	0,031250000	0,984375000
6	1,000000000	1,000000000	0,984375000	0,015625000	0,992187500
7	1,000000000	1,000000000	0,992187500	0,007812500	0,996093750
8	1,000000000	1,000000000	0,996093750	0,003906250	0,998046875
9	1,000000000	1,000000000	0,998046875	0,001953125	0,999023438
10	1,000000000	1,000000000	0,999755859	0,000000000	1,000000000

11 - Dados obtidos com o algoritmo LPAV2 aplicado no CHIP ATMEGA 328p

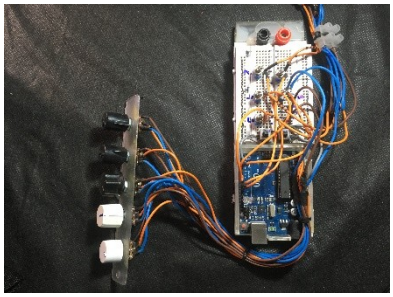


12 - Dados obtidos com o algoritmo LPAV2 aplicado no CHIP ATMEGA 328p Protótipo da CNAPap.;

4. DISCUSSÃO



13 – Braço Articulado com cinco com cinco servo motores.;



14 – CNAPap interface CHIP ATMEGA 328p cinco potenciômetros geradores de sinais analógicos.

O Chip ATMEGA 328p, atendia aos requisitos para utilização da programação estruturada C++, já que é a mesma utilizada em sua configuração. A modelagem física para utilização do chip é composta por uma interface, necessária para a programação do chip, através de microcomputador e conexão dos terminais com o meio físico externo. Cinco potenciômetros, para inserção dos sinais de entrada, e um braço articulado, acionado através dos sinais de saída do ATMEGA 328p, em cinco servos motores (Figura 13, 14).

Conclusão

Através dos resultados consolidados apresentados na modelagem desse artigo, conclui-se que CNAPap física é funcional e pode ser aplicada no desenvolvimento de projetos de Lógica

Paraconsistente com dispositivos articulados. A modelagem apresentada no artigo, durante o treinamento da CNAPap Física, não considerou desvios, sendo um caso a ser estudado em uma futura aplicação.

Referências

1. Da Silva Filho, J. I.. Métodos de Aplicações da Lógica Paraconsistente Anotada de anotação com dois valores-LPA2v. Revista Seleção Documental, ISSN 1809-0648, Santos, São Paulo – Brasil, Número 1, Ano 1, p. 18 – 25, Março 2006.

2. Da Silva Filho, J.I.. Análises de Sinais de Informações em Lógica Paraconsistente Anotada. Revista Seleção Documental, ISSN 1809-0648, Santos, São Paulo – Brasil Número 14, Ano 4, p. 22 – 26, Junho 2009.

3. Da Costa, N. C. A.; Abe, J. M.; Subrahmanian, V.S. "Remarks on annotated logic, Zeitschrift f. math. Logik und Grundlagen d. Math". Vol.37, pp 561-570, 1991.

5. Subrahmanian, V. S. "On the semantics of quantitative logic programs. Proc. 4th. IEEE Symposium on Logic Programming", Computer Society Press, Washington D.C., 1987.

6. Da Silva Filho, J. I.. "Introdução às Células Neurais Artificiais Paraconsistentes" Revista Seleção Documental, No 8, Dezembro, 2007.

7. Da Silva Filho, J. I.; Da Cruz, C. M.; Rocco, A.; Garcia, D. V.; Ferrara, L.F. P. Onuki, A. S.; Mario, . C.; Abe, J. M.. M. A. "Paraconsistent Artificial Neural Network for structuring Statistical Process Control in Electrical Engineering". Towards Paraconsistent Engineering Book, pp 77-102, Springer International Publishing, ISBN 978-3-319-40418-9, 2016.

8. Mario, M. C.; Ferrara, L. F. P.; Da Silva Filho, J. I. "Treinamento de uma Célula Neural Artificial Paraconsistente de Aprendizagem". Seleção Documental Magazine, No 6, pp 11 – 16, 2007, SP. Brazil. ISBN 18090648.

9. Da Silva Filho, J. I.; Torres, C. R.; Abe, J. M. "Robô Móvel Autônomo Emmy: Uma Aplicação eficiente da Lógica Paraconsist Paraconsistente Anotada" Revista Seleção Documental, No 3, Setembro, 2006, SP. Brasil. ISBN 18090648.
10. Da Silva Filho, J. I.; Abe, M. J.; Torres, G. L. Livro: Inteligência Artificial com Redes de Análise Paraconsistentes Teoria e aplicações Rio de Janeiro – LTC, 2008. ISBN 978-85-216-1631-3.