

Configuração de uma Rede Neural Artificial Paraconsistente para o Método de Aprendizado por Demonstração aplicado à um Braço Robótico

Configuration of a Paraconsistent Artificial Neural Network for the Learning from Demonstration Method applied to a Robotic Arm

Configuración de una Red Neuronal Artificial Paraconsistente para el Método de Aprendizaje Demostrativo aplicado a un Brazo Robótico

Recebido: 04/28/2022 | Revisado: 05/08/2022 | Aceitar: 05/16/2022 | Publicado: 05/21/2022

Paulino Machado Gomes

ORCID: <https://orcid.org/0000-0002-2834-8741>
Universidade Santa Cecília, Brasil
E-mail: pmfilter@hotmail.com

Cláudio Luís Magalhães Fernandes

ORCID: <https://orcid.org/0000-0003-3140-5580>
Universidade Santa Cecília, Brasil
E-mail: claudio.lmf@unisanta.br

João Inácio da Silva Filho

ORCID: <https://orcid.org/0000-0001-9715-8928>
Universidade Santa Cecília, Brasil
E-mail: inacio@unisanta.br

Rodrigo Silvério da Silveira

ORCID: <https://orcid.org/0000-0003-0661-6714>
Universidade Santa Cecília, Brasil
E-mail: rodrisilverio@gmail.com

Leonardo do Espírito Santo

ORCID: <https://orcid.org/0000-0003-4477-0544>
Universidade Santa Cecília, Brasil
E-mail: leonardo.santo@sp.senai.br

Maurício Conceição Mario

ORCID: <https://orcid.org/0000-0002-1973-2186>
Universidade Santa Cecília, Brasil
E-mail: cmario@unisanta.br

Vitor da Silva Rosa

ORCID: <https://orcid.org/0000-0002-5549-7611>
Universidade Santa Cecília, Brasil
E-mail: victor@unisanta.br

Germano Lambert Torres

ORCID: <https://orcid.org/0000-0003-3789-4696>
Universidade Santa Cecília, Brasil
E-mail: germanoltorres@gmail.com

Resumo

A Lógica Paraconsistente Anotada – LPA é uma lógica não clássica, baseada em conceitos que permitem, sob certas condições, aceitar a contradição em seus fundamentos, sem invalidar as conclusões. Interpretações matemáticas em seu reticulado associado, possibilitam a obtenção de equações e construções de algoritmos, que formam redes de análise paraconsistentes eficientes, em tratar sinais simulando aprendizagem. O algoritmo utilizado nesta pesquisa, é denominado de Célula Neural Artificial Paraconsistente de aprendizagem (CNAPap), e foi criado a partir das equações baseadas em LPA. Com sinais padronizados repetidamente aplicados à sua entrada, a CNAPap é capaz de armazenar gradativamente estas informações, aumentando ou diminuindo seu nível de resposta na saída com variação assintótica, controlado por um Fator de Aprendizagem (F_A). Para executar os testes, um conjunto de cinco CNAPaps formando uma Rede Neural Artificial Paraconsistente de aprendizagem (RNAPap), foi implementado em um microcontrolador ATMEGA 328p e vários ensaios foram realizados para validar o seu funcionamento, atuando no aprendizado por demonstração (LfD) em um Robô Manipulador. Considerando a frágil estrutura mecânica do Robô Manipulador, e dos dispositivos sensores adaptados para responder aos padrões, os resultados laboratoriais obtidos nos diversos testes apresentados foram satisfatórios, e o Sistema microprocessado construído respondeu de modo eficiente, onde os níveis de acertos, corresponderam entre 75% a 90%, em todas as etapas do método de LfD. Os resultados de estudos

comparativos, mostraram que a RNAPap possui propriedades dinâmicas com capacidade de atuar, tanto no método de aprendizagem por demonstração, como no método de imitação.

Palavras-chave: Lógica paraconsistente anotada; Aprendizagem por demonstração; Ensino; Inteligência artificial; Rede neural artificial paraconsistente.

Abstract

The Annotated Paraconsistent Logic - LPA is a non-classical logic, based on concepts that allow, under certain conditions, to accept the contradiction in its foundations, without invalidating the conclusions. Mathematical interpretations in its associated lattice make it possible to obtain equations and algorithm constructions, which form efficient paraconsistent analysis networks, in treating signals simulating learning. The algorithm used in this research is called Paraconsistent Artificial Neural Cell of Learning (CNAPap), and was created from equations based on LPA. With standardized signals repeatedly applied to its input, CNAPap is capable of gradually storing this information, increasing or decreasing its level of response at the output with asymptotic variation, controlled by a Learning Factor (FA). To run the tests, a set of five CNAPaps forming a learning Paraconsistent Artificial Neural Network (RNAPap), was implemented in an ATMEGA 328p microcontroller and several tests were carried out to validate its operation, acting on learning by demonstration (LfD) in a Robot Manipulator. Considering the fragile mechanical structure of the Robot Manipulator, and the sensor devices adapted to respond to the standards, the laboratory results obtained in the various tests presented were satisfactory, and the microprocessed system built responded efficiently, where the levels of correct answers corresponded to between 75 % to 90%, at all stages of the LfD method. The results of comparative studies showed that RNAPap has dynamic properties capable of acting both in the demonstration learning method and in the imitation method.

Keywords: Paraconsistent annotated logic; Learning from demonstration; Artificial intelligence; Teaching; Paraconsistent artificial neural network.

Resumen

La Lógica Paraconsistente Anotada - LPA es una lógica no clásica, basada en conceptos que permiten, bajo ciertas condiciones, aceptar la contradicción en sus fundamentos, sin invalidar las conclusiones. Las interpretaciones matemáticas en su entramado asociado permiten obtener ecuaciones y construcciones algorítmicas, que forman redes de análisis paraconsistentes eficientes, en el tratamiento de señales simulando aprendizaje. El algoritmo utilizado en esta investigación se denomina Célula Neural Artificial de Aprendizaje Paraconsistente (CNAPap), y fue creado a partir de ecuaciones basadas en LPA. Con señales estandarizadas repetidamente aplicadas a su entrada, CNAPap es capaz de almacenar gradualmente esta información, aumentando o disminuyendo su nivel de respuesta a la salida con variación asintótica, controlada por un Factor de Aprendizaje (FA). Para ejecutar las pruebas, se implementó un conjunto de cinco CNAPaps formando una Red Neuronal Artificial Paraconsistente de aprendizaje (RNAPap), en un microcontrolador ATMEGA 328p y se realizaron varias pruebas para validar su funcionamiento, actuando sobre aprendizaje por demostración (LfD) en un Robot Manipulador. Considerando la frágil estructura mecánica del Manipulador Robot, y los dispositivos sensores adaptados para responder a los estándares, los resultados de laboratorio obtenidos en las diversas pruebas presentadas fueron satisfactorios, y el sistema microprocesado construido respondió eficientemente, donde los niveles de aciertos correspondieron a entre 75 % a 90 %, en todas las etapas del método LfD. Los resultados de los estudios comparativos mostraron que RNAPap tiene propiedades dinámicas capaces de actuar tanto en el método de demostración de aprendizaje como en el método de imitación.

Palabras clave: Lógica paraconsistente anotada; Aprendizaje por demostración; Enseñando; Inteligencia artificial; Red neuronal artificial paraconsistente.

1. Introdução

Um método de inovação em robótica, sempre vai requerer pesquisas na área de inteligência artificial, que inclua outras subáreas a serem estudadas, de forma individual ou correlata. Atualmente, estão sendo desenvolvidas inovadoras pesquisas, que visam tornar as máquinas mais independentes dos métodos de programação, por meio de sistemas especialistas, robótica autônoma e aprendizado de máquina. Isto porque, para certos modelos de controle utilizados em robótica industrial, a reprogramação para mudar tarefas de um robô, pode ser custosa para a produção. Muitas vezes, as novas tarefas exigem adaptações e modificações profundas, no código fonte da programação, que devem ser confirmadas por meio de diversos testes aplicados ao robô, o que resulta em paradas não programadas, trazendo prejuízos a produção (Rosário, 2009).

Em IA, Sistemas Especialistas são empregados para o aprendizado de máquinas, e geralmente utilizam algoritmos baseados em Lógica Clássica, que se fundamenta em princípios estritamente binários. No entanto, em algumas situações em que a informação é incompleta, contraditória ou redundante, as incertezas contidas nos dados podem inviabilizar a sua aplicação

(Abe et al., 2018). Dessa forma, a utilização de sistema desenvolvido com lógicas não clássicas, pode ser um caminho promissor na área da aprendizagem de máquinas que inclui o LfD.

Entre as lógicas denominadas não clássicas, a lógica fuzzy tem se destacado nas pesquisas envolvendo automação, porém existem outros tipos, tais como a Lógica Paraconsistente - PL, que se baseia na oposição ao princípio da não contradição, um dos princípios básicos da lógica clássica. Recentemente a LP, na sua versão especial denominada, Lógica Paraconsistente Anotada com anotação de dois valores - LPA2v, tem se destacado nas aplicações de seus algoritmos, em diversas áreas do conhecimento, como em classificação de padrões (Da Silva Filho et al., 2021) (Mario et al., 2021). Portanto, neste artigo são apresentadas novas alternativas em controle de automação, que requerem aprendizado de máquina para agregar um maior nível de eficiência em LfD, por meio da implementação de estruturas algorítmicas, baseadas em Lógica Paraconsistente – LPA.

Organização do artigo

Este artigo apresenta a seguinte organização: Ainda nesta introdução serão apresentados os principais conceitos do método de aprendizagem por demonstração, com destaque as técnicas que serão utilizadas nos testes. Também serão apresentados os fundamentos da Lógica Paraconsistente Anotada - PAL, suas equações e algoritmos que formam a Célula Neural Artificial Paraconsistente de aprendizagem – CNAPap, e feita a estrutura algorítmica composta por 5 CNAPaps interligadas em cascata. Na seção 2, que se refere a materiais e métodos, apresenta-se as técnicas e procedimentos utilizados para a implementação da CNAPap no microcontrolador ATmega328P, e os procedimentos utilizados nos testes efetuados para o método de aprendizagem por demonstração LfD, aplicado ao Robô Manipulador. Na seção 3, apresenta-se os gráficos e tabelas que mostram os resultados obtidos neste trabalho, relacionados ao desempenho do conjunto de 5 CNAPaps, atuando no aprendizado por demonstração aplicado ao robô, e as discussões sobre os resultados da simulação, e acerca dos valores esperados e obtidos, no método de aprendizagem e imitação do robô. Na seção 5, apresentamos as conclusões, bem como as possibilidades de trabalhos futuros aplicando, esta técnica em robôs manipuladores.

1.1 Aprendizado por Demonstração e Lógica Paraconsistente

O Aprendizado por Demonstração, do inglês Learning from Demonstration (LfD), consiste em fazer com que uma máquina execute novas tarefas, imitando procedimentos que lhe são apresentados, sem a necessidade de reconfiguração ou reprogramação de seu software. Seguindo este princípio, a pesquisa relacionada à aplicação dessa técnica, visa encontrar formas eficientes de substituir o trabalho de programação manual, das atividades da máquina por um método de programação automática, conduzido exclusivamente para repetir, a tarefa demonstrada por um especialista (Mohseni-Kabir et al., 2015). No entanto, em um método de LfD, as comparações dos dados de informação entre o Tutor (ser humano) e o Aprendiz (Robô), são contaminadas por contradições, gerando incertezas nas conclusões (Ijspeert et al. 2002) (Pastor et al. 2013).

Considerando o exposto acima, verifica-se que, devido a estas incertezas geradas por contradição, as análises das informações e comparações, são de tal forma, primordiais para o método LfD, que se forem feitas com base em Lógica Clássica binária, pode atingir um baixo nível de eficiência. Dessa forma, tudo leva a crer, que as condições em que operam os métodos LfD, a aplicação do PL, que tem a propriedade de suportar informações contraditórias, pode amenizar alguns problemas relacionados a essa técnica. Seguindo estas considerações, apresentamos neste artigo a aplicação do algoritmo da Célula Neural Artificial Paraconsistente de aprendizagem (CNAPap), baseado em LPA2v, integrado com o método de aprendizagem por demonstração (LfD) a um Braço Robótico (Ekvall et al., 2008) (Da Silva Filho et al., 2021) (Mario et al., 2021).

1.2 Fundamentos de Aprendizagem da Demonstração (LfD)

Nos sistemas dedicados ao controle de automação, em trajetórias e movimento de robôs, são utilizados algoritmos altamente complexos, para que, possam realizar tarefas integradas à movimentação e transporte de objetos com precisão. A programação dos módulos computacionais dessas máquinas, utilizam códigos dedicados e estruturados, para gerenciar determinadas tarefas específicas para o tipo de robô. Isto faz com que a cada mudança na trajetória da máquina, ou mudanças nos produtos transportados, ocorra a necessidade da intervenção de especialistas em programação, para adequar o sistema a essas novas tarefas (Ekvall, et al. 2008). As pesquisas nesta área mostram que, para alcançar este objetivo, a flexibilidade na programação da máquina, teria que ser adaptada com algoritmos capazes de transferir habilidades por meio de intervenção direta, observação, emulação de metas, imitação e outras interações sociais (Ijspeert et al. 2002) (Pastor et al. 2013).

1.2.1 Aprendizagem Supervisionada

O aprendizado de máquina, é usado para ensinar às máquinas como lidar com os dados de forma mais eficiente. Como pode ser visto em (Billard et al, 2008), os métodos de aprendizado de máquina, podem ser classificados em quatro tipos; 1. Aprendizagem supervisionada (Gabin et al.); 2. Aprendizagem não supervisionada; 3. Aprendizagem semi-supervisionada e 4. Aprendizagem reforçada (Haykin, 2001).

A aprendizagem supervisionada, considerada como LfD, com o apoio de um sistema composto por redes neurais artificiais, pode ser classificada em 3 fases: Demonstração, Aprendizagem e Imitação, conforme detalhadas a seguir (Schaal, S. 2006), (Niekum et al. 2015):

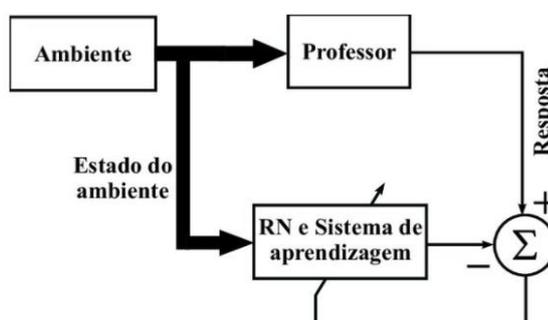
Fase 1. Demonstração – É a fase onde o Tutor realiza as atividades que deseja, que o robô aprenda.

Fase 2. Aprendizagem - É a fase onde ocorre a aprendizagem supervisionada.

Fase 3. Imitação – É a fase na qual, a condição de aprendizagem está completa, e a rede neural artificial finalizou seu processo de treinamento, transferindo o conhecimento obtido do Tutor para o Aprendiz.

A Figura 1, mostra o diagrama de bloco das etapas principais de um aprendizado supervisionado (Haykin, 2001) (Weijjtjens et al., 2017).

Figura 1: Diagrama de blocos de um aprendizado supervisionado.



Fonte: Haykin, (2001).

1.2.1.1 Abordagens das Etapas de Demonstração

Em geral, três tipos de abordagem são usados para o estágio de demonstração (Schaal, S. 2006), (Niekum et al. 2015). São estes:

Tipo 1. Cinestésico - O Tutor demonstra movendo fisicamente o robô, ou a máquina, através dos movimentos ou trajetórias desejadas.

Tipo 2. Teleoperação – O Tutor não entra em contato direto com o Aprendiz, que neste caso é o robô.

Tipo 3. Observações Passivas - O Tutor realiza a tarefa usando seu próprio corpo, que às vezes pode ser instrumentado por sensores adicionais para facilitar o rastreamento.

1.3 Modelagens de Dados para o Aprendizado

A técnica de aprendizado por demonstração (LfD), traz inúmeros desafios para sua aplicação, e um dos principais problemas está no modo de modelagem de dados, para a extração de características possíveis de serem aprendidas pelo Aprendiz. Nesta fase, apresenta-se duas principais questões; sendo a primeira, a de como interpretar e compreender os dados decorrentes dos comportamentos observados, e a segunda, a de como integrar os sistemas de percepção e controle do movimento, para reconstruir o que é observado (Ekvall, et al. 2008).

Outro fato, é que os modelos de sistemas de controle que vão atuar no método LfD, devem conter facilidades de generalização, uma vez que um movimento demonstrado, deve ser possível para modelar diferentes posições de meta (Ekvall, et al. 2008).

Em (Liu & Lemeire, 2017), modelos ocultos de Markov (HMMs) foram usados para codificar e generalizar os exemplos de demonstração. Em outros trabalhos, conforme visto em (Pastor et al., 2013) e (Schaal, 2006), os autores consideraram o fato de que sistemas dinâmicos com estabilidade assintótica global, definem uma função que representa um mapa de navegação global, onde todas as trajetórias espaciais convergem para o alvo. Uma alternativa para o sistema ser robusto a essas perturbações temporais, é que ele seja autônomo, ou sendo, independente do tempo (Nicolescu & Mataric, 2003).

Neste artigo, um sistema de aprendizagem por demonstração, apresenta-se como uma rede Neural Artificial Paraconsistente de aprendizagem (RNAPap), que traz como principal componente um algoritmo baseado na Lógica Paraconsistente, denominado CNAPap - Célula Neural Artificial Paraconsistente de aprendizagem (Da Silva Filho et al., 2010), (Mario et al., 2021).

A CNAPap contém as características da modelação, capaz de ser aplicada no processo de aprendizagem de movimentos de alcance de máquinas robóticas, a partir de um conjunto de demonstrações. Os dados de informação correspondentes às trajetórias, são obtidos através da execução da máquina robótica, controlada por um Joystick operado por um humano, onde as posições são coletadas e transformadas em padrões de aprendizagem. Portanto, o processo de aprendizado, utiliza a teleoperação onde, após o aprendizado, a máquina robótica será capaz de reproduzir trajetórias semelhantes a demonstrações sem interferência humana.

1.4 Lógica Paraconsistente Anotada (LPA)

A Lógica Paraconsistente (LP), pertence a uma categoria de lógicas não clássicas e sua principal característica, é a revogação do princípio da não contradição (Da Costa & Abe, 2000) (Abe et al., 2018).

Em (Da Silva Filho et al., 2010), a LP foi descrita com suas equações e todos os seus predicados sendo formalizados, estudados em sua integralidade, demonstrando ser uma lógica não clássica, proposicional e evidencial. Quando um reticulado é associado à LP, no qual pode-se representar estados lógicos em seus vértices, esta será considerada uma lógica paraconsistente anotada – LPA (Garcia et al., 2019).

Em (Da Silva Filho et al., 2010) foi apresentada uma LPA, que contém um reticulado associado de quatro vértices, onde são consideradas representações de estados lógicos paraconsistentes, formando o conjunto: $\tau = \{v, T, f, \perp\}$, onde: t = verdadeiro, f = falso, T = Inconsistente e \perp = Paracompleto. Neste modo de interpretação, uma anotação composta por dois graus de evidência

(μ, λ) , atribui uma conotação lógica à uma proposição P, de tal forma que: μ representa o grau de evidência a favor da proposição P, e λ representa o grau de evidência que se opõe a proposição P.

Diante disso, é fixado um operador: $|\tau| \rightarrow |\tau|$, tal que: $\tau = \{(\mu, \lambda) \mid \mu, \lambda \in [0, 1] \subset \mathfrak{R}\}$. Assim, a anotação (μ, λ) , na condição de valores máximos, representa os estados lógicos extremos dos vértices da Reticulado, de acordo com os valores de μ e λ a seguir:

(1, 0) - Grau de evidência favorável à proposição P é máximo, e o Grau evidência desfavorável para proposição P é mínimo, transformando-se em um estado lógico v= verdadeiro.

(0, 1) Grau de evidência favorável à proposição P é mínimo, e Grau de evidência desfavorável à proposição P é máximo, transformando-se em um estado lógico f = falso.

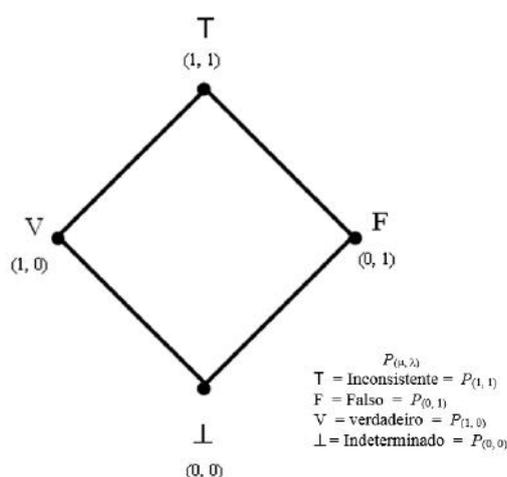
(1, 1) Grau de evidência favorável à proposição P é máximo, e Grau evidência desfavorável para proposição P é máximo, transformando-se em um estado lógico T = Inconsistente.

(0, 0) Grau de evidência favorável à proposição P é mínimo, e o Grau evidência desfavorável para proposição P é mínimo, transformando-se em um estado lógico \perp = Paracompleto.

Esta condição de estados lógicos extremos, definidos pelos valores dos graus de evidência que formam uma anotação, é mostrada na Figura 2.

A LPA funcionando dessa forma, é denominada de Lógica Paraconsistente Anotada com anotação de dois valores (LPA2v), e as interpretações algébricas podem associar os graus de evidência na forma da anotação (μ, λ) , aos estados lógicos do seu Reticulado associado. Como os valores de μ e λ podem variar entre 0 e 1, e pertencem ao conjunto R, então, com suas variações, podem ser considerados internamente no reticulado associado à LPA2v, infinitos estados lógicos paraconsistentes (Da Silva Filho et al., 2010), (Corrêa et al., 2022).

Figura 2: Reticulado associado a Lógica Paraconsistente Anotada LPA2v.



Fonte: Da Silva Filho et al., (2010).

Para o equacionamento, são feitas inicialmente as representações de valores em um Quadrado Unitário no Plano Cartesiano (QUPC), onde o grau de evidência favorável μ tem sua representação no eixo x, e grau de evidência desfavorável λ é representado no eixo y. A análise matemática no QUPC, possibilita que se obtenha uma transformação (T), que representa esses valores em um reticulado associado em PAL2v de coordenadas XY. A transformação T obtida na análise, (Da Silva Filho et al., 2010) é mostrada a seguir:

$$T(X, Y) = (x - y, x + y - 1) \quad (1)$$

Sendo os componentes da transformação $T(X, Y)$, relacionados de acordo com a nomenclatura usual da LPA2v, tal que: $x = \mu$ e $y = \lambda$, então do primeiro termo obtido no par ordenado da equação da transformação (1), tem-se o grau de certeza (GC) (Da Silva Filho et al., 2010), que é obtido por:

$$G_C = \mu - \lambda \quad (2)$$

No segundo termo obtido, no par ordenado da equação da transformação (1), tem-se o grau de contradição (G_{ct}), que é obtido por:

$$G_{ct} = \mu + \lambda - 1 \quad (3)$$

Os valores de GC e de G_{ct} , que estão no conjunto dos números reais \mathfrak{R} , variam no intervalo fechado $+1$ e -1 , e estão expostos no eixo horizontal e vertical, no respectivo reticulado representativo da LPA2v (3). O grau de Certeza da análise pode ser remodelado para grau de evidência, e ser aplicado na análise de outra proposição P (Da Silva Filho et al., 2010), da seguinte forma:

$$\mu_E = \frac{1 + (\mu - \lambda)}{2} \quad (4)$$

onde: μ_E = Grau de evidência resultante.

μ = Grau evidência favorável.

λ = Grau evidência desfavorável.

A partir da estrutura, que será representada algoritmicamente pela equação (4), pode-se introduzir uma família de Células Neurais Artificiais Paraconsistentes - CNAPs (Da Silva Filho et al., 2010), onde, cada uma das células é formada de aperfeiçoamentos, e modificações do algoritmo descritivo das equações da LPA2v.

1.5 Célula Neural Artificial Paraconsistente de Aprendizagem - CNAPap

Em uma Rede Neural Artificial Paraconsistente-RNAP, as células de aprendizagem são projetadas para serem utilizadas como partes de unidades de memórias, ou como sensores de padrões em camadas primárias (Da Silva Filho et al., 2010) (Mario et al., 2021). No método de aprendizagem, se for aplicado na entrada um sinal classificado como o grau de evidência de valor 0, a Célula vai aprender que o padrão é representado por um valor de Falsidade. Surgindo na entrada o valor 0 repetidas vezes, o grau de evidência resultante da análise, vai aumentando gradualmente na saída, até chegar ao valor 1. Nestas condições, dizemos que a Célula aprendeu o padrão de falsidade. O mesmo método é adotado se for aplicada repetida vezes na entrada da Célula, o valor 1. Quando o grau de evidência resultante na saída da Célula chegar ao valor 1, dizemos que a Célula aprendeu o padrão de verdade. Ainda neste método de aprendizagem da CNAPap, é inserido o fator de aprendizado FA, que é ajustado externamente (Da Silva Filho et al., 2010) (Mario et al., 2021). Portanto, o fator de aprendizagem FA é um valor real, dentro do intervalo fechado $[0,1]$, atribuído eventualmente por ajustes externos. Dependendo do valor de FA, será proporcionado uma aprendizagem mais rápida, ou mais lenta da CNAPap. Sendo: $0 \leq FA \leq 1$ e considerando um método de aprendizagem do padrão de verdade, a equação de aprendizagem é obtida através da equação (4), do cálculo do Grau de evidência resultante, ficando:

$$\mu_{E(k+1)} = \frac{\left\{ \mu_1 - \left(\mu_{E(k)C} \right) F_A \right\} + 1}{2} \quad (5)$$

onde: μ_1 = Padrão a ser aprendido, FA = fator de aprendizagem e $\mu_{E(k)C} = 1 - \mu_{E(k)}$.

Considera-se a Célula completamente treinada quando $\mu_{E(k+1)} = 1$.

Para um método de aprendizagem do padrão de Falsidade, é feito também a complementação no grau de evidência favorável, que representa o padrão a ser aprendido. Neste caso a equação fica:

$$\mu_{E(k+1)} = \frac{\left\{ \mu_{1C} - (\mu_{E(k)C}) F_A \right\} + 1}{2} \quad (6)$$

onde: $\mu_{1C} = 1 - \mu_1$.

A Tabela 1, apresenta os resultados obtidos em uma CNAPap, quando são aplicados repetidas vezes o padrão de verdade (1) no método de aprendizagem, e depois a aplicação do padrão de Falsidade (0), no método de desaprendizagem (Da Silva Filho et al., 2010) (Mario et al., 2021).

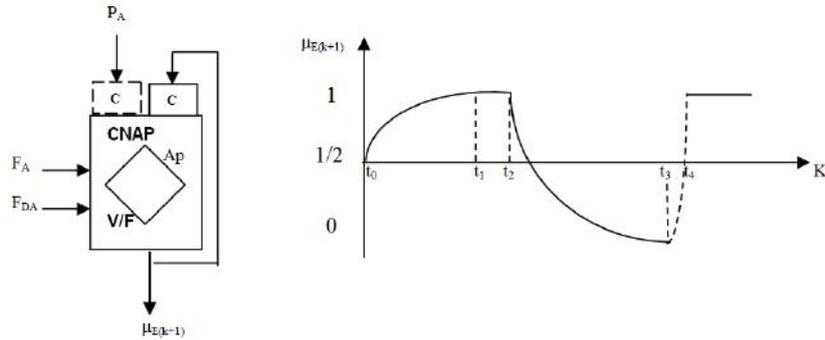
Tabela 1: Tabela de resultados da Célula Neural Artificial Paraconsistente de aprendizagem CNAPap, para um treinamento de aprendizagem.

Passos	Padrão= μ_1	$G_c = \mu_1 - \mu_{2c}$	$G_{ct} = (\mu_1 + \mu_{2c}) - 1$	$\mu_E = \{(G_c \times C_1) + 1\} \div 2$	Observações
0		0,000000000	-1,000000000	0,500000000	Início
1	1	0,500000000	0,500000000	0,750000000	Aprendizagem
2	1	0,750000000	0,250000000	0,875000000	
3	1	0,875000000	0,125000000	0,937500000	
4	1	0,937500000	0,062500000	0,968750000	
5	1	0,968750000	0,031250000	0,984375000	
6	1	0,984375 00	0,015625000	0,992187500	
8	1	0,996093750	0,007812500	0,996093750	
9	1	0,998046875	0,001953125	0,998468750	
10	1	0,999023437	0,000976562	0,999234375	
	1,0	1,000000000	0,000000000	1,000000000	Término
11	0	0,000000000	-1,000000000	0,500000000	Desaprendizagem
12	0	-0,500000000	-0,500000000	0,250000000	
13	0	-0,750000000	-0,250000000	0,125000000	
14	0	-0,875000000	-0,125000000	0,062500000	
15	0	-0,937500000	-0,062500000	0,031250000	
16	0	-0,968750000	-0,031250000	0,015625000	
17	0	-0,984375 00	-0,015625000	0,007812500	
18	0	-0,996093750	-0,007812500	0,003906250	
19	0	-0,998046875	-0,001953125	0,001953125	
20	0	-0,999023437	-0,000976562	0,000488281	
	0	-1,000000000	0,000000000	0,000000000	Término
21	$\mu_1=1,0$	Operador Not			
22	0	+1	0	1	Confirma

Fonte: Autores.

As CNAPs de aprendizagem, são representadas nas interligações das Redes Neurais Paraconsistentes, pela simbologia simplificada, conforme apresentada na Figura 3, onde também é mostrado o gráfico resultante de saída.

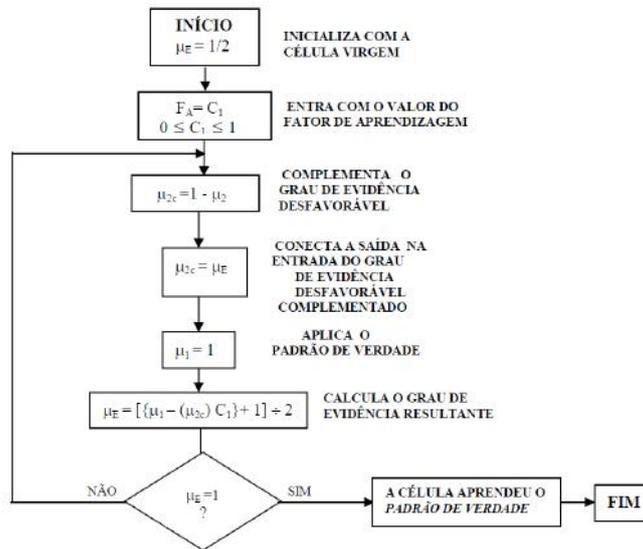
Figura 3: Símbolo simplificado de uma CNAPap e o seu gráfico de saída.



Fonte: Mario et al., (2021).

O gráfico mostrado na Figura 3, apresenta o comportamento do sinal de saída na aplicação de um padrão repetitivo na entrada. Verifica-se que, no instante t_0 a saída apresenta uma Indefinição com grau de evidência resultante valendo $1/2$. Do instante t_0 até o instante t_1 acontece na saída um comportamento monotônico, para então, o grau de evidência resultante ficar constante no instante t_2 . Do instante t_3 até o instante t_4 ocorreu a aplicação do padrão inverso na entrada e, como consequência, o sinal de saída sofreu um decréscimo até zero. No instante t_4 acontece a confirmação do novo padrão aprendido, que é sinalizado através de uma porta de complementação (Da Silva Filho et al., 2010) (Mario et al., 2021). O algoritmo da CNAPap na forma de um fluxograma, é mostrado na Figura 4.

Figura 4: Fluxograma para aprendizagem do *padrão de Verdade* da CNAPap.



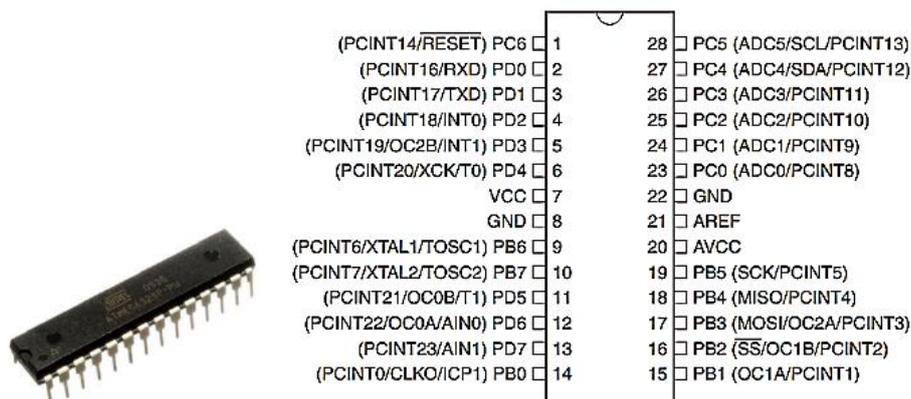
Fonte: Da Silva Filho et al., (2010).

Conforme será visto a seguir, esta pesquisa foi direcionada para a utilização da célula CNAPap no método de aprendizagem por demonstração em um Robô Manipulador. Para isto, o algoritmo da célula de aprendizagem CNAPap foi implementado em um circuito microprocessado, e configurada uma estrutura composta de 5 CNAPaps, constituindo-se em uma Rede Neural Artificial Paraconsistente de aprendizagem (RNAPap), dedicada a atuar no método de aprendizagem de um Robô Manipulador.

2. Material e Métodos

Nesta pesquisa, foi construído um protótipo do Sistema de aprendizagem, composto por um microcontrolador ATMEGA 328p com circuitos eletro/eletrônicos de apoio, de modo que os procedimentos do método LfD, possam ser aplicados em um Robô Manipulador usando a teleoperação, com o uso de um Joystick. A Figura 5, mostra a apresentação física do chip ATmega328P e sua pinagem correspondente.

Figura 5: Apresentação física do *chip* ATmega328P e sua respectiva pinagem.



Fonte: (ATmega328P [Datasheet] 7810D-AVR-01/15).

Os principais dados Técnicos do ATmega328P são apresentados a seguir:

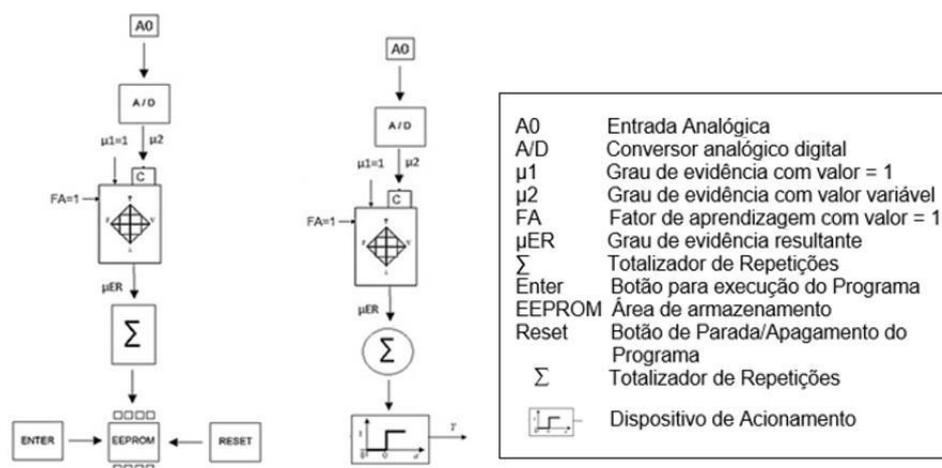
Tensão de operação:	5V
Tensão de alimentação máxima:	5,5 V
Entradas e saídas digitais:	23
Memória Flash:	32 KB
Memória SRAM:	2 KB
Memória EEPROM:	1 KB
Velocidade de clock:	0-20 MHZ

No aprendizado por demonstração utilizando o Hardware, o grau de evidência μ_1 de evidência do algoritmo é o que representa o padrão a ser aprendido. No programa seu valor é igual a 1 e assim, um sinal analógico de voltagem entre 0V e 5V, inserido através do Joystick, é aplicado aos terminais de entrada do chip ATmega328P (A0). Um conversor interno, transforma o sinal analógico para digital, e o conecta em μ_2 , que é outro grau de evidência. Desse modo, μ_2 utiliza variáveis entre os limites 0 e 1.

Após completar o ciclo de aprendizagem, isto é, efetuadas as iterações necessárias para que a saída da CNAPap apresente seu valor máximo $\mu_{ER}=1.0$, uma chave é acionada havendo então a liberação dos sinais de saída do ATmega328P. A programação do sistema, faz com que o microcontrolador ATmega328P ao receber este comando, acione os servo-motores do Braço Robótico para efetuar as ações, conforme foram aprendidas através das demonstrações.

Na Figura 6, são mostrados os fluxogramas dos sinais no método da aprendizagem por demonstração com a CNAPap, implementada no ATmega328P.

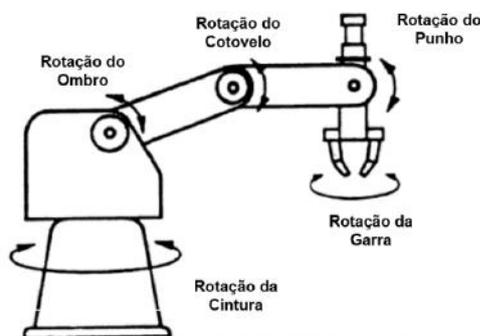
Figura 6: Diagramas do programa. a) Fluxograma representando o funcionamento da CNAPap Física implementada no ATmega328P. b) Fluxograma representando a totalização das repetições e o acionamento.



Fonte: Autores.

Para a demonstração da técnica de LfD, aplicada através de uma estrutura composta de 5 CNAPaps, foi construído um protótipo similar ao Robô Manipulador mostrado na Figura 7.

Figura 7: Partes do Braço Robótico que serão atingidas pelo método de aprendizagem por demonstração através da estrutura de CNAPaps.



Fonte: Haykin (2001).

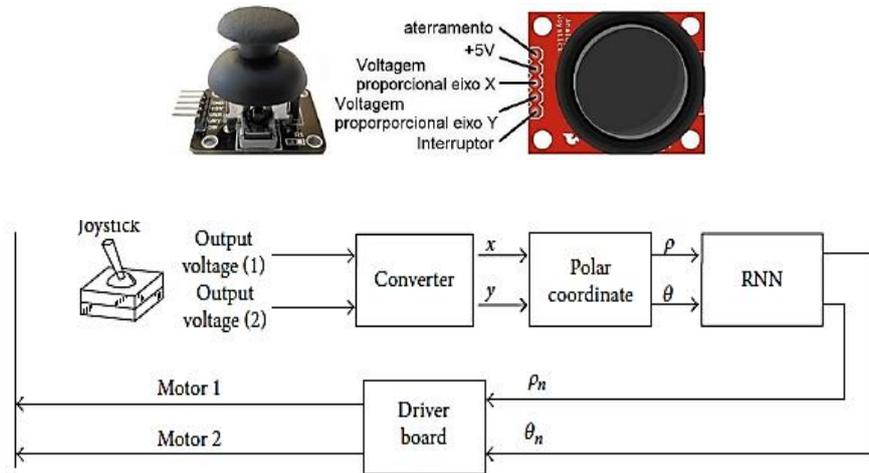
O Braço Robótico utilizado, tem a sua estrutura mecânica classificada, em um tipo de coordenadas esféricas, com algumas adaptações construtivas, onde são destacadas as ações de rotação das partes, que serão atingidas pelo método de aprendizagem por demonstração, através do Sistema microprocessado. Os testes com o protótipo do Sistema de aprendizagem por demonstração, foram direcionados para acionar os cinco principais motores de movimentação atuando nas 5 ações descritas a seguir: 1. Rotação da Cintura. 2. Rotação do Ombro. 3. Rotação do Cotovelo. 4. Rotação do Punho. 5. Rotação da Garra. Como cada parte do Braço Robótico que será atingida pelo método de aprendizagem por demonstração, tem o seu correspondente motor de acionamento, o Sistema de aprendizagem por demonstração terá a sua ação em cinco motores CC.

2.1 Aprendizado por Teleoperação

O sistema construído em hardware, foi projetado para aplicar o método LfD através da Teleoperação, portanto os sinais de aprendizagem das células são inseridos através do Joystick, que, ao acionar a manopla, obtém como resultado a movimentação

do braço robótico, representado pela energização de seus motores. Na Figura 8, é mostrado Joystick utilizado na teleoperação, destacando os sinais de entradas e saídas, e seu diagrama geral do funcionamento.

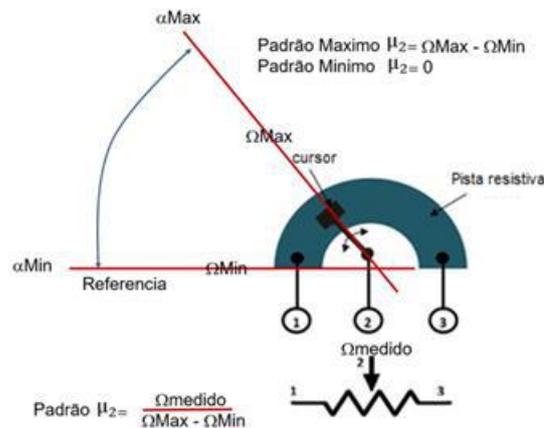
Figura 8: Visão geral física e do diagrama de acionamento do *Joystick*, utilizado para inserir os sinais de aprendizagem na CNAPap.



Fonte: (Datasheet) (Joy – IT /KY-023/1).

O movimento do Joystick gera as informações para o aprendizado por Demonstração do algoritmo da célula CNAPap, que foi implementado no hardware, através da programação do microcontrolador ATmega328P. A monitoração das ações de movimentação do Braço Robótico foi feita através de respostas obtidas por sensores instalados nas juntas e na garra do Braço Robótico. Neste trabalho, utilizou-se potenciômetros instalados em conjunto com os servomotores, onde se estabelece o valor da resistência elétrica, a partir de uma referência de ângulo, conforme mostrado na Figura 9.

Figura 9. Relação dos ângulos limites definidos na movimentação desejada das juntas e da garra, com a geração do grau de evidência padrão μ , que será aplicado na CNAPap para o aprendizado por demonstração.



Fonte: Autores.

A referência estabelecida no potenciômetro permite a correspondência com o ângulo de deslocamento de cada junta do Braço Robótico, portanto, a informação da resposta de aprendizagem das juntas ou da garra, será obtida com base em a sua própria resistência (Ohms). Para a junta e para a garra, são considerados inicialmente o ângulo máximo de deslocamento que será aprendido pelo robô, em relação ao ângulo mínimo considerado como referência. E assim, o ângulo máximo é relacionado

com a resistência máxima do potenciômetro (Ω_{Max}), e considera-se o ângulo mínimo igual a resistência nula do potenciômetro ($\Omega_{Min}=0$), que servirá como referência. A partir desta determinação dos ângulos, se estabelece o Universo do Discurso. Com esta definição que relaciona os ângulos de deslocamento com a resistência, para qualquer medida obtida no potenciômetro (Ω_{Medido}), se obtém a o grau de evidência fazendo a relação:

Se $\Omega_{medido} \geq \Omega_{Max}$, então $\mu_2 = 1$

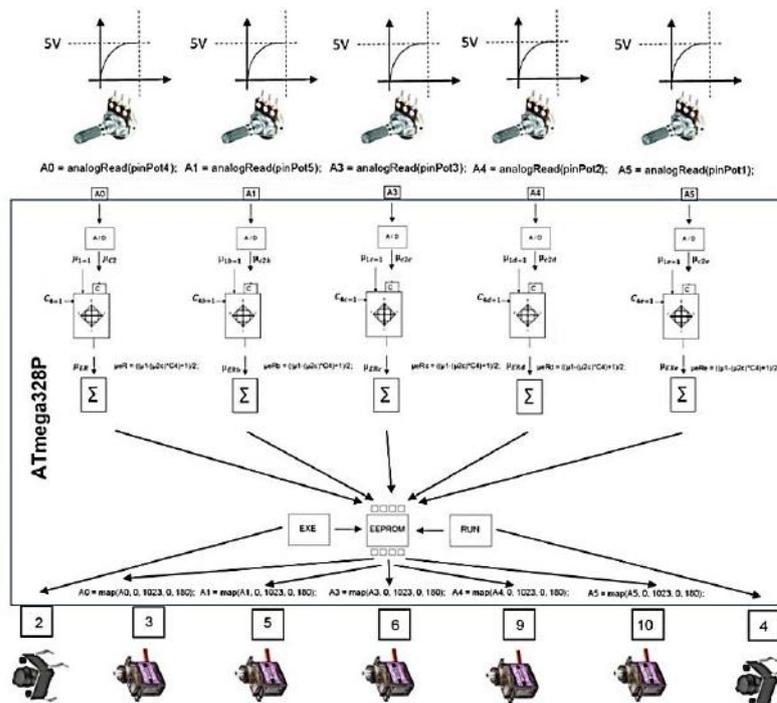
Se $\Omega_{medido} \leq \Omega_{Min}$, então $\mu_2 = 0$

Se $\Omega_{Min} \leq \Omega_{medido} \leq \Omega_{Max}$ então $\mu_2 = \frac{\Omega_{medido} - \Omega_{Min}}{\Omega_{Max} - \Omega_{Min}}$

Definidos os ângulos de deslocamento máximos (α_{Max}), e mínimos (α_{Min}) de cada movimentação das juntas, foram então calculados os limites do Universo de Discurso e inseridos na programação do microcontrolador para o cálculo do padrão μ_2 a ser aprendido. Para haver similaridades, foram considerados no Universo de Discurso, a excursão $\Delta\alpha_{UD} = 90^\circ$ que corresponde a $10\text{ K}\Omega$ para todos os 20 movimentos estudados. Dessa forma, o cálculo para o Fator de aprendizagem aprendido foi feito por: $\mu_2 = \frac{\hat{\text{Ângulo}}\ \alpha_{medido} - \Omega_{Min}}{\Delta\alpha} = \frac{\hat{\text{Ângulo}}\ \alpha_{medido}\ \text{em}\ \text{graus}(^\circ)}{90^\circ} = \frac{\text{Resistência}\ \text{Medida}\ \text{em}\ \Omega}{10\text{K}\Omega}$

Conforme mostrado na Figura 10, foram utilizados como sensores de monitoração da aprendizagem 5 potenciômetros lineares de $10\text{K}\Omega$ que respondem com valores de resistência elétrica, aos correspondentes ângulos limites de deslocamento que serão definidos para cada junta, e para a garra do Braço Robótico. Dessa forma, uma tensão elétrica de 5v é aplicada ao potenciômetro, de modo que ao ser acionado, a junta (ou a garra) correspondente se movimenta, e retorna um valor de tensão relacionado ao deslocamento efetuado na demonstração.

Figura 10: Representação dos potenciômetros que funcionam como sensores geradores de padrão para o processo LfD com a CNAPap no treinando da garra e das juntas do Braço Robótico.

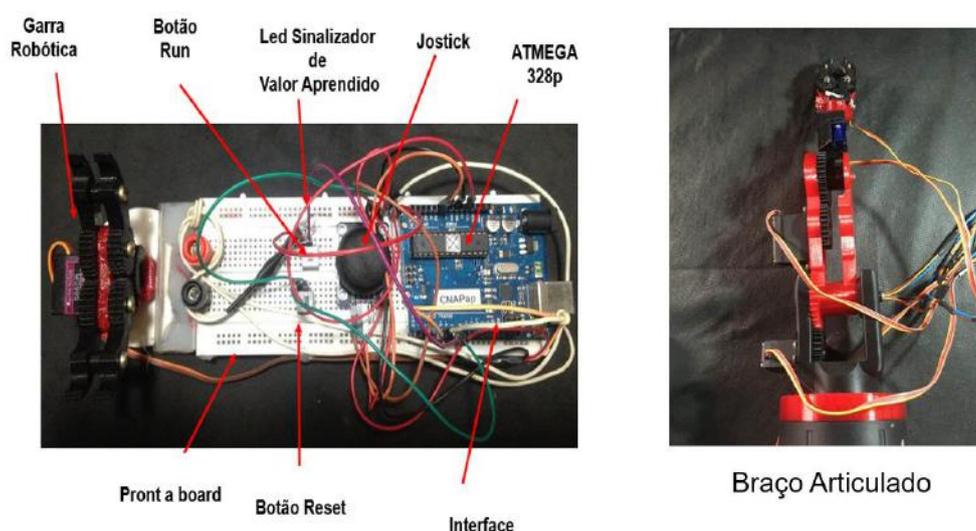


Fonte: Autores.

Através do programa instalado no microcontrolador, o valor de tensão é então, transformado em grau de evidência do deslocamento ($\mu 2$ Padrão) que será aplicado n vezes na entrada da CNAPap, até se consignar um valor aprendido por demonstração, com este estágio sinalizado pelo LED instalado na placa do circuito do microcontrolador ATmega328P.

Os detalhes da montagem do circuito do Sistema de aprendizagem microprocessado, para aplicação do método LfD, para os testes utilizando o Robô Manipulador e seus principais componentes, estão mostrados na Figura 11.

Figura 11: Montagem do protótipo do Sistema de aprendizagem microprocessado para aplicação do método LfD em Braço Robótico.



Fonte: Autores.

No método de aprendizagem por demonstração os dados inseridos na CNAPap, são lidos e armazenados na recorrência estimada de 10 repetições sequenciadas, sem espaços para desvios. Ao final dessas repetições, a CNAPap sinaliza, através da ativação de uma saída analógica (pino 4) conectada a um LED, que aprendeu. Os sinais analógicos convertidos em dados, são então armazenadas em uma área de memória da CNAPap.

Para executar a leitura dos sinais memorizados e iniciar a fase de imitação é necessária a ativação do botão denominado RUN, conectado a uma entrada do microcontrolador (pino 2). Quando executado esse passo, significa que o procedimento de aprendizagem por demonstração foi concluído e um servo motor (μE) (pino3) correspondente é acionado. Este, por sua vez, movimentará a junta ou a garra correspondente, agora sem a necessidade de manipular o joystick ($\mu 2$) (A0), entrando assim na fase da imitação. Para finalizar e limpar os dados inseridos na área de memória da CNAPap, o botão denominado RESET é ativado (pino 6), completando assim o ciclo de aprendizagem e repetição.

3. Resultados e Discussão

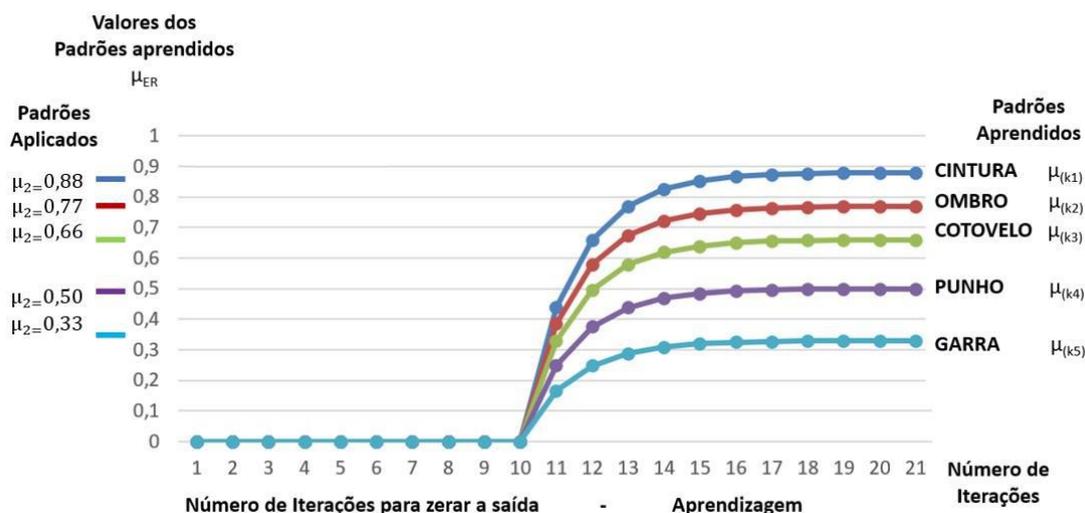
Os padrões de treinamento escolhidos para as análises, foram definidos a partir de cada grau de deslocamento escolhido para a aprendizagem, de cada uma das juntas do Robô Manipulador. Dessa forma, os valores correspondentes ficam sendo:

1. Ângulo da aprendizagem na Rotação da Cintura $\Delta\alpha = 80^\circ$, onde a resistência correspondente do potenciômetro será de $8,8\text{ K}\Omega$, portanto, o Grau de evidência aplicado como Padrão foi de $\mu 2 = 0,88$.

2. Ângulo da aprendizagem na Rotação do Ombro $\Delta\alpha = 70^\circ$, onde a resistência correspondente do potenciômetro será de 7,7 K Ω , portanto, o Grau de evidência aplicado como Padrão foi de $\mu_2 = 0,77$.
3. Ângulo da aprendizagem na Rotação do Cotovelo $\Delta\alpha = 60^\circ$, onde a resistência correspondente do potenciômetro será de 6,6 K Ω , portanto, o Grau de evidência aplicado como Padrão foi de $\mu_2 = 0,66$.
4. Ângulo da aprendizagem na Rotação do Punho $\Delta\alpha = 45^\circ$, onde a resistência correspondente do potenciômetro será de 5,0 K Ω , portanto, o Grau de evidência aplicado como Padrão foi de $\mu_2 = 0,50$.
5. Ângulo da aprendizagem na Rotação da Garra $\Delta\alpha = 30^\circ$, onde a resistência correspondente do potenciômetro será de 3,3 K Ω , portanto, o Grau de evidência aplicado como Padrão foi de $\mu_2 = 0,33$.

Foram feitos testes simulados na CNAPap implementada no microcontrolador ATmega328P aplicando os padrões resultantes dos intervalos dos ângulos de rotação escolhidos para a aprendizagem. Os resultados destas primeiras simulações são apresentados graficamente na Figura 12.

Figura 12: Resultados gráficos dos valores de aprendizagem com as aplicações dos padrões escolhidos para os testes no Robô Manipulador.



Fonte: Autores.

Conforme mostrado nos gráficos da Figura 12, os resultados das simulações efetuadas na CNAPap implementada no microcontrolador ATmega328P, se mostraram muito próximos aos valores esperados para todos os 5 diferentes padrões aplicados, nas partes constituintes do Robô Manipulador.

Através da técnica de Teleoperação com o acionamento do Joystick, estabeleceu-se o estado de “Aprendizado por demonstração”, onde foi aplicado n vezes o padrão μ_2 na entrada da CNAPap até a ocorrência da sinalização pelo acendimento do LED da placa com o alerta, indicando que o aprendizado foi completado. Após este processo, o Braço Robótico foi colocado no estado de “Imitação” onde os dispositivos acionam os motores das juntas movimentando-as nos ângulos limites que foram estipulados no Universo do Discurso seguindo as etapas adquiridas e armazenadas no treinamento.

Para verificação da eficiência do aprendizado por demonstração, efetuado pelo sistema construído, foram comparados os valores dos ângulos no método de Imitação verificando se estes correspondiam aos valores esperados. Os resultados são

mostrados na Tabela 2, onde estão os valores médios das respostas, após 20 verificações dos resultados das imitações. Também na Tabela 2 estão, a correspondente porcentagem de acertos encontradas para as 4 juntas e a garra do Braço Robótico.

Tabela 2: Valores obtidos nos testes onde o total de demonstrações é $N_d=10$, e o total de verificações da imitação é $M_i=20$.

Parte do Robô: CINTURA					
Rotação de aprendizagem	Sensor Resistivo	Voltagem Realimentação	Padrão de aplicação	Acertos nos limites de erro $\Delta\alpha \pm 10\%$	Eficiência
$\Delta\alpha 80^\circ$	8,8 K Ω	4,4 V	$\mu_2 = 0,88$	15	75 %
Parte do Robô: OMBRO					
Rotação de aprendizagem	Sensor Resistivo	Voltagem Realimentação	Padrão de aplicação	Número de acertos nos limites $\Delta\alpha \pm 10\%$	Eficiência
$\Delta\alpha 70^\circ$	7,7 K Ω	3,85 V	$\mu_2 = 0,77$	18	90 %
Parte do Robô: COTOVELO					
Rotação de aprendizagem	Sensor Resistivo	Voltagem Realimentação	Padrão de aplicação	Número de acertos nos limites $\Delta\alpha \pm 10\%$	Eficiência
$\Delta\alpha 60^\circ$	6,6 K Ω	3,3 V	$\mu_2 = 0,66$	14	70 %
Parte do Robô: PUNHO					
Rotação de aprendizagem	Sensor Resistivo	Voltagem Realimentação	Padrão de aplicação	Número de acertos nos limites $\Delta\alpha \pm 10\%$	Eficiência
$\Delta\alpha 45^\circ$	5,0 K Ω	2,5 V	$\mu_2 = 0,50$	17	85 %
Parte do Robô: GARRA					
Rotação de aprendizagem	Sensor Resistivo	Voltagem Realimentação	Padrão de aplicação	Número de acertos nos limites $\Delta\alpha \pm 10\%$	Eficiência
$\Delta\alpha 30^\circ$	3,3 K Ω	1,65 V	$\mu_2 = 0,33$	15	75 %

Fonte: Autores.

Verifica-se pelos valores resultantes da Tabela 2, que o a Cintura e a Garra do Robô mostraram o menor número de acertos com uma eficiência de aprendizagem de 75 %. Entre as partes constituintes do Robô Manipulador, a que resultou em maior eficiência foi o Ombro, que mostrou 90 % de acerto.

Nos testes, foi verificado que as condições estruturais do Robô Manipulador, influenciaram nos resultados com a Teleoperação.

No acionamento dos motores, a estrutura mecânica do robô apresentou-se oscilante, o que dificultou as adaptações dos sensores nas juntas ocasionando erros nas respostas dos padrões de aplicação por meio da Teleoperação. Sendo assim, considera-se que este fator ligado a estrutura mecânica do Robô e dos componentes sensores, foi o que mais influenciou nos resultados obtidos no método de aprendizagem, dos graus de abertura das juntas. Este problema apresentado na etapa do aprendizado, refletiu na etapa da Imitação, no entanto, tanto na aprendizagem como na Imitação, o Robô mostrou que o Sistema de aprendizagem por demonstração construído com CNAPaps, possui condições de recuperação da trajetória ideal, o que mostra certa robustez a distúrbios apresentados.

4. Conclusões

Neste artigo, foi apresentada uma nova forma de aplicar a técnica de LfD, através de uma estrutura computacional construída com algoritmos fundamentados em uma lógica não clássica, denominada de Lógica Paraconsistente Anotada-LPA. Para formar o sistema de aprendizagem, utilizou-se do algoritmo denominado de Célula Neural Artificial Paraconsistente de aprendizagem – CNAPap. Conforme foi visto, a CNAPap é formada por um algoritmo paraconsistente de recorrência, o que a capacita a aprender padrões aplicados em sua entrada. Através de técnicas de recorrência e sua fundamentação em LPA, a CNAPap foi capaz de armazenar gradativamente a informação da entrada, apresentando como resposta sinais com variação assintótica, controlado por um ajuste de um fator de aprendizagem (FA). A estrutura computacional baseada em LPA, foi implementada no microcontrolador ATMEGA 328p, onde um conjunto de 5 CNAPaps foi implementado, para efetuar técnicas de aprendizagem por demonstração. O sistema microprocessado de aprendizagem, foi aplicado no método de aprendizado por demonstração em um Robô Manipulador composto de quatro juntas e uma garra, onde os testes efetuados mostraram que os algoritmos CNAPaps, respondem bem ao aprendizado de padrões normalizados, com valores entre 0 e 1, aplicados repetidamente à sua entrada. Os resultados mostraram, que o Sistema Paraconsistente de aprendizagem por demonstração, possui propriedades dinâmicas, com certa robustez a perturbações, tanto no método de aprendizagem, quanto no método de imitação. Com base nos resultados obtidos neste trabalho, novas pesquisas com algoritmos fundamentados em Lógica Paraconsistente Anotada serão desenvolvidas, com o objetivo de novas adaptações que melhorem a eficiência, do Sistema Paraconsistente de Aprendizagem construído.

Referências

- Abe, J. M., Akama, S., Nakamatsu, K., & Da Silva Filho, J. I. (2018). Some Aspects on Complementarity and Heterodoxy in Non-Classical Logics. *Procedia Computer Science*, 126, 1253–1260. <https://doi.org/10.1016/j.procs.2018.08.068>
- Mario, M. C., Garcia, D. V., Da Silva Filho, J. I., Silveira Junior, L., & Barbuy, H. S. (2021). Characterization and classification of numerical data patterns using Annotated Paraconsistent Logic and the effect of contradiction. *Research, Society and Development*, 10(13), e283101320830. <https://rsdjournal.org/index.php/rsd/article/view/20830>
- Andreas, J., Klein, D., & Levine, S. (2017) Modular multitask reinforcement learning with policy sketches. In Proceedings of the 34th International Conference on Machine Learning. 70, 166–175. JMLR. org, 2017. 1, 2
- Argall, B. D., Chernova, S., Veloso, M., & Browning, B. (2009). A survey of robot learning from demonstration. *Robotics and Autonomous Systems*, 57(5), 469-483, <https://doi.org/10.1016/j.robot.2008.10.024>.
- Billard, A., Calinon, S., Dillmann, R., & Schaal, S. (2008). Robot programming by demonstration. In Siciliano, B., and Khatib, O., eds., Springer Handbook of Robotics. Springer Berlin Heidelberg. 2008, 1371–1394.
- Chi, M., Yao, Y., Liu, Y., & Zhong, M. Learning, Generalization, and Obstacle Avoidance with Dynamic Movement Primitives and Dynamic Potential Fields. *Appl. Sci.* 2019, 9, 1535. <https://doi.org/10.3390/app9081535>
- Corrêa, M. P., Machado, A. C., Da Silva Filho, J. I., Garcia, D. V., Mario, M. C., & Sedano, C. T. S. (2022). Paraconsistent annotated logic applied to industry assets condition monitoring and failure prevention based on vibration signatures. *Research, Society and Development*, [S. 1.], 11(1), e14211125104, 2022. 10.33448/rsd-v11i1.25104.
- Da Costa N. C. A., & Abe J. M. (2000). Paraconsistência em Informática e Inteligência Artificial, *Ciência • Estud. Av.* 14 (39) • <https://doi.org/10.1590/S0103-40142000000200012>
- Da Silva Filho, J. I., Abe, J. M., Marreiro, A. D. L., Martinez, A. A. G., Torres, C. R., Rocco, A., Côrtes, H. M., Mario, M. C., Pacheco, M. T. T., Garcia, D. V., & Blos, M. F. (2021) Paraconsistent annotated logic algorithms applied in management and control of communication network routes *Sensors*, 21(12), 4219 <https://doi.org/10.3390/s21124219>
- Da Silva Filho, J. I., Lambert-Torres, G., & Abe, J. M. (2010). Uncertainty treatment using paraconsistent logic—introducing paraconsistent artificial neural networks. 2010; 320.
- Ekvall, S., & Kragic, D. (2008). Robot learning from demonstration: A task-level planning approach. *International Journal of Advanced Robotic Systems*, 5(3):223234.
- Garcia, D. V., Da Silva Filho, J. I., Silveira Jr, L., Pacheco, M. T. T., Abe, J. M., et al. Analysis of Raman spectroscopy data with algorithms based on paraconsistent logic for characterization of skin cancer lesions. *Vibrational Spectroscopy* 2019;103;102929.

Gienger, M., Mühlig, M., & Steil, J. J. (2010). Imitating object movement skills with robots — A task-level approach exploiting generalization and invariance. *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 1262-1269, 10.1109/IROS.2010.5649990.

Haykin, S. (2001) *Redes Neurais: Princípios e Práticas*. (2a. ed.): Bookman.

Ijspeert, A. J., Nakanishi, J., & Schaal, S. (2002). Learning rhythmic movements by demonstration using nonlinear oscillators. In *Proceedings of the IEEE/RSJ Int. Conference on Intelligent Robots and Systems (IROS2002)*, 2002, pages 958-963.

Liu, T., & Lemeire, J. (2017) Efficient and Effective Learning of HMMs Based on Identification of Hidden States. *Mathematical Problems in Engineering*, vol. 2017, Article ID 7318940, 26 pages, 2017. <https://doi.org/10.1155/2017/7318940>

Mario, M. C., Garcia, D. V., Da Silva Filho, J. I., Silveira Júnior, L., & Barbuy, H. S. (2021). Characterization and classification of numerical data patterns using Annotated Paraconsistent Logic and the effect of contradiction. *Research, Society and Development*, [S. 1.], 10(13), e283101320830, 10.33448/rsd-v10i13.20830.

Mohseni-Kabir, A., Rich, C., Chernova, S., Sidner, C. L., & Miller, D. (2015). Interactive hierarchical task learning from a single demonstration. In *Proceedings of the Tenth Annual - ACM/IEEE International Conference on Human-Robot Interaction, HRI '15*, 2015, 205–212. New York, NY, USA: ACM

Nicolescu, M. N., & Mataric, M. J. (2003). Natural methods for robot task learning: Instructive demonstrations, generalization and practice. In *Proceedings of the second international joint conference on Autonomous agents and multiagent systems*, 2003, 241–248. ACM.

Niekum, S., Osentoski, S., Konidaris, G., Chitta, S., Marthi, B., & Barto, A. G. (2015) Learning grounded finite-state representations from unstructured demonstrations. *The International Journal of Robotics Research*. 2015; 34(2):131-157. 10.1177/0278364914554471

Pastor, P., Kalakrishnan, M., Meier, F., Stulp, F., Buchli, J., Theodorou, E., & Schaal, S. (2013). From dynamic movement primitives to associative skill memories. *Robotics and Autonomous Systems*, 2013, 61(4), 351–361.

Rosário, J. M. (2009) *Automação Industrial*: Editora: Baraúna. 2009. 517 págs. ISBN-13: 978-8579230004

Schaal, S. (2006) Dynamic movement primitives-a framework for motor control in humans and humanoid robotics. in *Adaptive Motion of Animals and Machines*. Springer, 2006, pp. 261–280.

Gebin, L. G. G., Salgado, R. M., & Nogueira, D. A. (2020). Wind Power Forecast: Ensemble Model Based in Statistical and Machine Learning Models. *Research, Society and Development*, 9(12), e38291211251, 10.33448/rsd-v9i12.11251.