

**Revista e-Tec
04-2021**



Faculdade da Indústria SENAI Londrina

Edição V4 – ano 2021 – Londrina Paraná

Corpo Editorial

*Editor Chefe – Edição V4-2021
Prof. Wesley Candido da Silva*

Comitê Executivo

*Prof.^a Adriana G. Carvalho
Prof. Antônio Carlos Rodrigues
Prof. Aparecido S. Dos Santos
Prof.^a Dra. Camila F. Oliveira
Prof. Fábio Rodrigo Milanez
Prof. Renato Kazuo Miyamoto
Prof. Dr. Rodolfo A. Hildebrandt
Prof. Dr. Vicente de Lima Gongora*

*Editora: Faculdade de Tecnologia Senai Londrina
ISSN: 2358-5528*

Direitos reservados

*Faculdade de Tecnologia SENAI Londrina
Rua Belém, 844 – Londrina PR.
43 -3294-5100 – faculdade.londrina@sistemafiep.org.br*

*Partes desta publicação poderão ser reproduzidas, sem a
autorização prévia e escrita da Editora Faculdade de Tecnologia SENAI
Londrina, desde que citada a fonte.*

*Este periódico publica nomes individuais, comerciais, marcas
registradas e produtos pertencentes a diversas companhias. O Editor
utiliza-se destes nomes somente para fins editoriais e em benefício dos
proprietários dos nomes e marcas, sem intenção de atingir seus
direitos. Observa-se ainda que os dados contidos nos artigos são de
responsabilidade dos próprios autores.*

e-TEC

**Revista de
Tecnologia e
Ciência**

Prefácio

As necessidades demandadas pela indústria 4.0?

É notório, no dia a dia, a aproximação da nossa instituição de ensino com o mercado industrial; este fato se verifica de forma continuada, tanto nas ofertas quanto nas soluções apresentadas para o atendimento das demandas da Indústria.

A prática é realizada nas áreas e sub áreas compreendidas dos cursos ofertados pela Faculdade da Indústria SENAI Londrina que reúnem as habilidades do futuro exigidas pelo mercado, tal como a “Aplicabilidade” que representa saber onde e como aplicar determinado conhecimento, aquela determinada ideia e experiência na solução das demandas do mercado.

Nesta publicação, pode ser observada a produção científica de acadêmicos e professores; que divulgam de forma a inspirar através de seus próprios trabalhos, demais interessados pelos diversos temas aqui tratados.

Esta publicação, é uma mostra do resultado de como uma solução de problemas da indústria, pode ocorrer em nosso meio acadêmico; que se percebe tanto através dos projetos de iniciação científica, passando pelo desenvolvimento de projetos aplicados em disciplinas e outros multidisciplinares, desenvolvidos nas Jornadas de aprendizagem; como nos trabalhos de conclusão do curso de graduação e em projetos de pós-graduação.

No tempo de Falar, que ocorre ao final de cada semestre letivo, a nossa comunidade acadêmica mostra as evidências do seu próprio desenvolvimento das soluções demandas, dos problemas estudados transcendendo o período letivo.

Por fim, reunindo as competências técnicas e emocionais, na apresentação pública deste desenvolvimento fica comprovada a prática exitosa da metodologia ativa – Jornadas de Aprendizagem.

Forte abraço e boa Leitura, coordenador do curso de Engenharia Elétrica,
prof. Dr. Vicente Gongora

Sumário

Capítulo I – Open Tec - Tempo de Falar – III° Mostra de Resultados “Jornadas de Aprendizagem”: Equilíbrio das habilidades técnicas e emocionais.....	4
Capítulo II – Estratégias de Controle Inteligente Aplicadas a uma Planta Didática de Nível de Baixo Custo	11
Capítulo III - Desenvolvimento de um Neurônio Artificial: Um estudo de acadêmicos para acadêmicos.....	32
Capítulo IV – Programa de Voluntariado do Sistema FIEP – Abrace uma Criança	45

Capítulo I – Open Tec - Tempo de Falar – IIIº Mostra de Resultados “Jornadas de Aprendizagem”: Equilíbrio das habilidades técnicas e emocionais.

Prof.(a) Esp. Adriana G. carvalho¹

Prof. Esp. Antonio Carlos Rodrigues²

Prof.(a) Dra. Camila F. Oliveira ³

Prof. Esp. Fábio Milanez ⁴

Prof. Dr. Vicente Gongora⁵

Prof. Esp. Wesley Candido da Silva⁶

1. Introdução

Quando se deseja tratar da consolidação de conhecimentos provenientes dos conteúdos formativos provenientes de disciplinas, de práticas exitosas estabelecidas conforme o projeto pedagógico do curso, considerando-se ainda a integração de conteúdos ministrados; é necessário descrever a metodologia ativa das jornadas de aprendizagem. Esta é responsável por consolidar os conhecimentos trabalhados nas demais disciplinas do semestre letivo, dando sentido aos conteúdos, contribuindo de forma inovadora e acrescentando-se novos conteúdos formativos, como um ponto de convergência para a resolução da situação-problema ou do desafio estabelecido no início do semestre; pelo Núcleo docente estruturante dos cursos ofertados pela Faculdade da Indústria SENAI Londrina.

As disciplinas Jornadas de Aprendizagem objetivam interrelacionar conteúdos disciplinares entre os temas abordados nas demais disciplinas do período, ao longo de todo o curso; com novas tecnologias existentes e ou inovadoras. Trata-se de um instrumento de integração da aprendizagem unindo a teoria com a prática profissional, na medida em que proporciona, de forma equilibrada, contato com as demandas

¹ Docente da Faculdade de Tecnologia Senai Londrina. e-mail: adriana.carvalho@sistemafiep.org.br

² Docente da Faculdade de Tecnologia Senai Londrina. e-mail: antonio.rodrigues1@sistemafiep.org.br

³ Docente da Faculdade de Tecnologia Senai Londrina. e-mail: camila.oliveira@sistemafiep.org.br

⁴ Docente da Faculdade de Tecnologia Senai Londrina. e-mail: fabio.milanez@sistemafiep.org.br

⁵ Docente da Faculdade de Tecnologia Senai Londrina. e-mail: vicente.gongora@sistemafiep.org.br

⁶ Docente da Faculdade de Tecnologia Senai Londrina. e-mail: wesley.candido@sistemafiep.org.br

atuais do mercado; podendo incluir diferentes temáticas na área de atuação do curso e necessárias para o desenvolvimento do desafio escolhido.

Como exemplo de operacionalização, mostra-se na Figura 1, algumas fases dos desafios propostos e desenvolvidos, pelas animadas turmas de Engenharia Elétrica, Engenharia Mecânica e Automação Industrial, o kit didático solar noturno, o Robô Agroshock, a Estufa hidropônica e o Robô CNC, nos vários ambientes da Faculdade da Indústria SENAI Londrina.

Figura 1. Projetos Agroshock, kit didático solar noturno e robô CNC.



Fonte: do Autor (2021).

Os conceitos e conteúdo da disciplina de Análise de Circuitos eletrônicos, fundamentando a eletrônica analógica, foram necessários para o desenvolvimento do rastreador solar, utilizado pela turma do quarto período da Engenharia Elétrica, onde o apoio essencial na construção do modelo físico da Turma Engenharia Mecânica, também no mesmo período contribuiu muito com o projeto. Mostra-se na Figura 2, estas turmas em perfeito entrosamento.

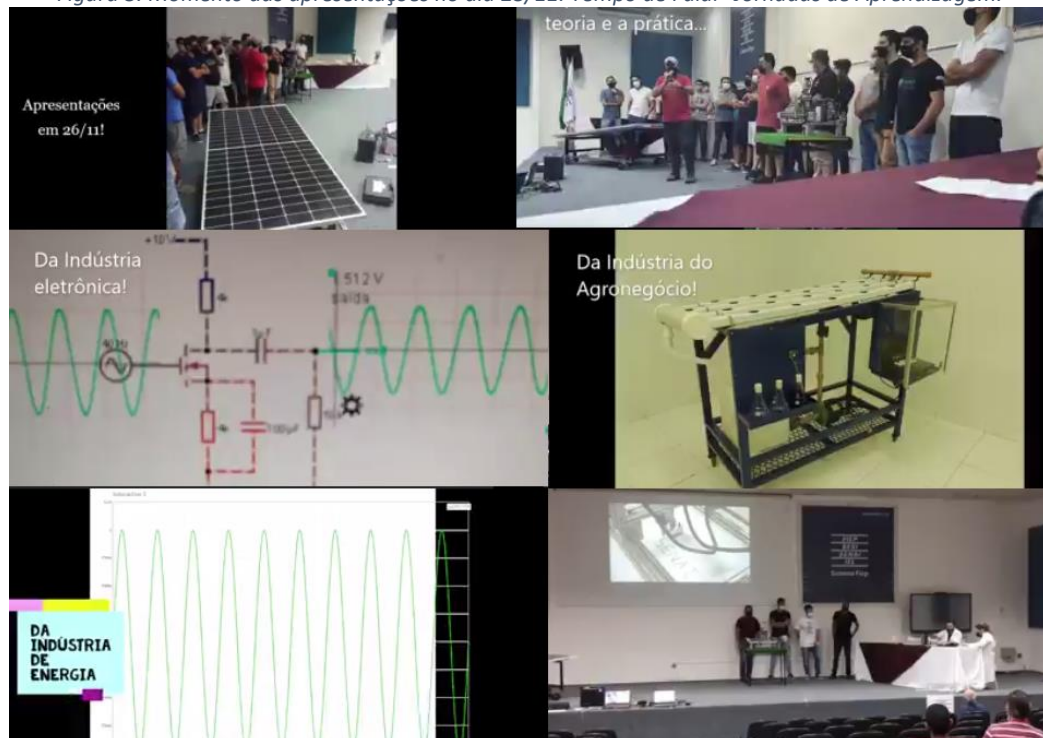
Figura 2. Kit didático – circuito antes e depois da inclusão do rastreador solar.



Fonte: Do Autor, (2021).

As habilidades e competências comportamentais, além de serem exigidas durante as fases de implementação, também são necessárias no momento da apresentação pública dos trabalhos executados. Na Figura 3, mostra-se um dos momentos de apresentações presenciais, no auditório da Faculdade da Indústria SENAI Londrina em 2021.

Figura 3. Momento das apresentações no dia 23/11. Tempo de Falar- Jornadas de Aprendizagem.



Fonte: Do Autor, (2021).

A mostra de resultados deste ano teve o seu ponto alto, nas discussões efetuadas entre as turmas acadêmicas; e, entre acadêmicos e docentes que puderam realizar os devidos esclarecimentos, das execuções, envolvendo as dificuldades encontradas e por fim, dos conteúdos necessários para o desenvolvimento das atividades estabelecidas, salientando-se inclusive as contribuições dos projetos ao perfil profissional do egresso.

Figura 4. Foto comemorativa do encerramento da Terceira mostra de Resultados – Jornadas de Aprendizagem.



Fonte: Do Autor, (2021).

E para quem acreditava que só as atividades das Jornadas bastariam, nossos alunos, salientam que, desejam experimentar todas as possibilidades de sucesso; e, apreciam cada uma delas. Exemplo deste fato, é mostrado na Figura 5, com os alunos do curso de Engenharia Elétrica de diferentes turmas, que se reuniram sob a coordenação do prof. Wesley Candido, no desenvolvimento do projeto “reconhecimento de padrões de imagem, utilizando-se de inteligência artificial”, classificado no Inova Paraná, no momento de sua apresentação que ocorreu de forma remota conectada.

O Prêmio Inova, visa o reconhecimento da conexão e da aplicação integrada da Metodologia Senai de Educação Profissional, da Metodologia do Colégio Sesi (Oficinas de Aprendizagem), da EJA e das Jornadas de Aprendizagem, metodologia ativa das Faculdades da Indústria, além de permitir a visibilidade dos projetos com a exposição às indústrias e à comunidade visitantes.

Os alunos recebem durante o período de realização do Inova, tanto

regional como nacional, uma capacitação em empreendedorismo inovador, aprendem a fazer o Canvas do projeto e como apresentar em formato de Pitch. Nesta plataforma, existe a seleção de projetos e a premiação dos primeiros classificados. Além desta edição do Inova estadual que a Faculdade da Indústria SENAI Londrina participou, os alunos podem participar também do Inova Senai Nacional, caso sejam classificados em primeiro lugar.

Figura 5. Momento da Apresentação do projeto no Inova Estadual.



Fonte: Do Autor, (2021).

Poderíamos encerrar aqui este capítulo, mas a Faculdade da indústria, possui inúmeras realizações, dentre elas cabe ressaltar a turma de formandos neste segundo semestre de 2021, do curso de Tecnologia em Manutenção Industrial.

A Razão além da finalização do curso de graduação, neste momento, comemoramos o fato de estarem 100% bem empregados nas Indústrias da Região.

Este ano, os acadêmicos formandos, escolheram os professores Wesley Candido e Antônio Carlos Rodrigues, para receberem as justas homenagens. Parabéns, formandos e que sua perseverança continue, com objetivos éticos e clareza

de consciência, para em estabelecendo metas estas sejam conquistadas, e coroada com o sucesso que merecem! Contém com a equipe docente da Faculdade da Indústria SENAI Londrina.

A foto foi produzida no momento da apresentação do projeto da Jornada de aprendizagem. Assim sendo, a Figura 6 mostra além dos formandos, o professor Wesley Candido em primeiro plano, alguns alunos de outros cursos e outros que não constam da lista de formandos, mas contribuíram para o desenvolvimento das atividades por eles realizadas.

Figura 6. Turma de Formandos do Curso Superior de Tecnologia em Manutenção Industrial.



Fonte: Do Autor, (2021).

Por fim, a última e grande notícia, uma grande alegria para toda a comunidade acadêmica da Faculdade da Indústria SENAI Londrina; O SENAI

Londrina terá nova Unidade! Cujas previsões de entrega serão entre Julho/ Agosto de 2023; A cidade de Londrina vai poder contar com a mais moderna infraestrutura laboratorial do País, nas áreas de Automação industrial, eletrônica e Tecnologia para Informação da Indústria 4.0!

A nova sede do Senai apresentada ao público geral, em evento oficial na Associação comercial e industrial de Londrina (ACIL), pode ser visualizada na Figura 7, terá próximo de 8 mil m² de área construída na Rua Deputado Fernando Ferrari, jardim bancários em Londrina; o que permitirá atender simultaneamente perto de 700 alunos simultaneamente. Além dos modernos laboratórios de informática, física, robótica, internet das coisas, desenvolvimento de softwares, smart grid, domótica, automação industrial, eletrônica, espaço Maker e de realidade aumentada e virtual.

Figura 7. Nova sede SENAI Londrina – Jardim Bancários



Fonte: Adaptado de SENAI Londrina, 2021.

O espaço abrigará, além da Faculdade da Indústria SENAI Londrina, também o Instituto Senai de Tecnologia da Informação e Comunicação e um HUB de Inteligência Artificial, reforçando ainda mais os laços com a indústria.

Capítulo II – Estratégias de Controle Inteligente Aplicadas a uma Planta Didática de Nível de Baixo Custo

Prof. MS.c. Bruno Leandro Galvão Costa⁷

RESUMO

Este artigo ilustra a aplicação de abordagens de controle inteligente em uma planta didática de controle de nível construída com dispositivos de baixo custo, neste caso, utilizando componentes relacionados à plataforma eletrônica Arduino Uno. Detalhes construtivos deste recurso didático de ensino são comentados ao longo do trabalho. As estratégias de controle inteligente consideradas para a análise no sistema proposto baseiam-se nas redes neurais artificiais e lógica fuzzy, sendo seus desempenhos comparados com o de um controlador proporcional-integrativo clássico. Resultados de ensaios experimentais foram obtidos e são ilustrados no decorrer do trabalho visando evidenciar as vantagens, em termos de controle, de utilização das estratégias inteligentes em relação à convencional.

Palavras-chave: Arduino Uno. Controle PI. Lógica Fuzzy. Redes Neurais Artificiais. Sistema de Controle de Nível.

INTELLIGENT CONTROL STRATEGIES APPLIED TO A LOW COST LEVEL CONTROL DIDACTIC PLANT

ABSTRACT

This paper illustrates the application of intelligent control approaches in a didactic level control plant built with low-cost devices, in this case, using components related to the Arduino Uno electronic platform. Constructive details of this teaching didactic resource are discussed throughout the article. The intelligent control strategies considered for the analysis of the proposed system are based on artificial neural networks and fuzzy logic, and their performances are compared with that of a classical proportional-integrative controller. Results of experimental test were obtained and are illustrated in the course of the work in order to highlight the advantages, in terms of control, of using intelligent strategies in relation to the conventional one.

Key-words: Arduino Uno. PI Control. Fuzzy Logic. Artificial Neural Networks. Level Control System.

⁷ Docente do programa de pós-graduação na Faculdade da Indústria SENAI Londrina. E-mail: brunolgcosta@gmail.com

1. INTRODUÇÃO

Atualmente, nos ambientes industriais, muitas estratégias de controle têm sido utilizadas, visando melhorar o desempenho dos processos envolvidos, bem como aliar questões relacionadas à maximização de produção e minimização de custos. Dentre estas estratégias de controle, os controladores PI/PID são certamente uma das melhores metodologias aplicadas ao controle de processos, principalmente por conta de sua simplicidade de implementação (KOSÁK, 2014).

O projeto de um controlador PI/PID para um dado sistema consiste na definição de alguns parâmetros que caracterizem a dinâmica do processo em questão. Inicialmente, um método de identificação deve ser empregado para a modelagem matemática do sistema de controle. Sendo assim, na literatura podem ser encontrados vários métodos de identificação tais como os de Broida (HAMITI, 1996), Ziegler/Nichols, Sundaresan / Krishnaswamy e de Smith (COELHO; COELHO, 2004).

Após este procedimento é necessário efetuar a sintonia do controlador, ou seja, ajustar seus respectivos ganhos: proporcional, integrativo e derivativo. Dessa forma os métodos de sintonia podem ser empregados, sendo baseados i) em curvas de resposta ao degrau (ASTROM; HÄGGLUND, 1995), ii) no lugar das raízes (OGATA, 2010) e iii) na resposta em frequência (ANGÉLICO; CAMPANHOL; SILVA, 2014).

Apesar da enorme aplicabilidade dos controladores PI/PID, em algumas situações a obtenção de modelos matemáticos de processos de controle pode ser bastante difícil, devido a inúmeros motivos dentre eles às suas características não-lineares. Devido a isso, muitas estratégias alternativas de controle, e que não exigem o conhecimento do modelo do sistema, têm surgido nas últimas décadas, podendo resultar em desempenhos superiores àqueles dos controladores PI/PID. Dentre elas podem ser destacadas àquelas baseadas em sistemas inteligentes (ou inteligência artificial), neste caso, sistemas fuzzy (GOMIDE; GUDWIN, 1994) e redes neurais artificiais (RNAs) (SILVA, 2016).

A lógica fuzzy, também conhecida como lógica nebulosa ou difusa, é uma metodologia computacional que busca desenvolver raciocínios muito próximos aos dos seres humanos, visando a solução de problemas cujas informações sejam imprecisas ou incertas. As RNAs, por sua vez, são algoritmos que se baseiam no comportamento do cérebro humano, o qual possui neurônios, elementos sensoriais

dispostos em uma vasta rede cuja função é a de propagação de impulsos nervosos. No ambiente computacional as RNAs apresentam capacidade de aprendizado e generalização, realizados por meio de informações que lhes são fornecidas.

Vários trabalhos ilustram a aplicação de sistemas inteligentes em processos de controle. Almeida (2014) descreve a aplicação de topologias de controladores fuzzy para um processo multivariável. Por sua vez, o artigo de Andrade, Agulhari, Goedel (2015) ilustra a implementação de um PI-neural para um sistema de controle de vazão. Já em Bertachi et al. (2013), controladores neurais foram desenvolvidos para o controle de uma malha multivariável de nível e temperatura. No trabalho de Freire Júnior, Vargas, Goedel (2013) um PI-neural foi projetado para o controle de um servomecanismo.

Este artigo busca aplicar estratégias de inteligência artificial em sistemas de controle, neste caso, como controladores baseados nos conceitos da lógica fuzzy e das RNAs, sendo analisadas em uma planta didática de controle de nível desenvolvida com equipamentos de baixo custo e controlada via plataforma Arduino Uno. Tal sistema de controle foi desenvolvido com o objetivo de poder auxiliar ambientes de ensino e aprendizagem acadêmicos, de modo a facilitar o entendimento sobre aspectos relacionados à área de controle. Neste contexto, controladores fuzzy e neural foram implementados e seus comportamentos foram analisados com o de um controlador PI convencional, sendo apresentados no decorrer do trabalho. Testes experimentais foram coletados e são mostrados de modo a evidenciar as principais contribuições em termos de controle de tais estratégias inteligentes.

2. FUNDAMENTAÇÃO TEÓRICA

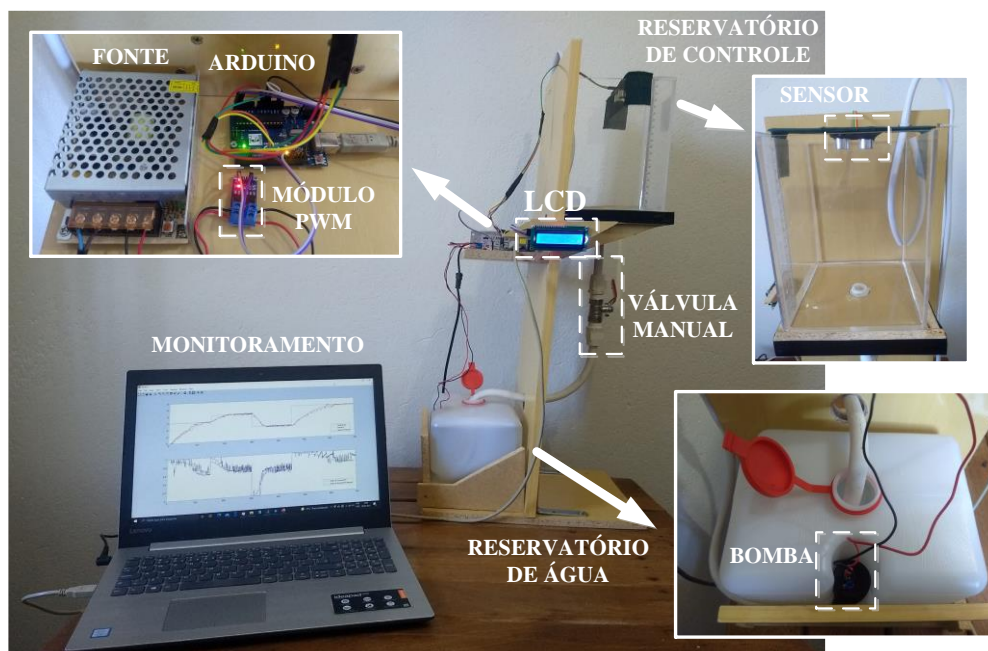
2.1 Planta didática de controle de nível

A Figura 1 ilustra a planta didática de controle de nível desenvolvida neste artigo, constituída principalmente pelos seguintes componentes de fácil acesso:

1. Plataforma de desenvolvimento Arduino;
2. Sensor de nível ultrassônico HC-SR04;
3. Módulo PWM D4184 30A/36V 400W;
4. Fonte de tensão 5A/12V;

5. Display LCD 16x2 (com adaptador I2C);
6. Bomba de água 4A/12V;
7. Válvula manual; e
8. Reservatórios de água e de controle, acoplados com engates de PVC e mangueira para envio/recebimento de água.

Figura 1 – Planta didática de controle de nível



Fonte: Elaborado pelo autor (2021)

Esta planta utiliza a plataforma de desenvolvimento Arduino Uno para a implementação dos controladores que serão analisados ao longo do trabalho, a qual foi adotada no projeto devido ao fato de ser uma plataforma mundialmente difundida e popularizada, tanto nos ambientes acadêmicos quanto nos comerciais e industriais, com grande facilidade de utilização. Dentre as principais características do Arduino Uno podem ser destacadas (MCROBERTS, 2011):

1. Estrutura composta por um microprocessador Atmel AVR, microcontrolador ATmega328P e regulador de tensão linear de 5V (padrão de operação);
2. 6 pinos de entrada analógica (A0 a A5) (10 bits), 14 pinos de entradas/saídas digitais (0 a 13), sendo 6 destes pinos do tipo PWM (8 bits);
3. Corrente CC por pino de entrada/saída igual a 40mA;
4. *Clock* interno a cristal com 16 MHz;

5. Possibilidade de desenvolvimento de algoritmos baseados em linguagem C utilizando o IDE Arduino, ambiente gratuito e fornecido pelo fabricante;
6. Porta USB para comunicação de dados com computadores.

Para realizar as medições de nível no reservatório de controle, optou-se pelo módulo ultrassônico HC-SR04, um dispositivo de sensoriamento de baixo custo e de fácil utilização. Como dispositivo atuador do sistema de nível optou pela utilização de uma bomba de água 12V, acoplada em um reservatório de água (ambos utilizados em limpadores para-brisas de automóveis), sendo operada nos experimentos com uma tensão máxima de 6V para conservar sua vida útil. Devido ao seu consumo de corrente ser de aproximadamente 4A, optou-se pela utilização do módulo PWM D4184, também de baixo custo, que permite operações até 30A. Acoplado ao módulo PWM está uma fonte de tensão 5A/12V, necessária para a alimentação da bomba, de acordo com o nível PWM enviado pelo Arduino. Um pequeno display LCD é considerado para o monitoramento do sistema. Por fim, uma válvula manual é acoplada ao sistema de nível possibilitando possíveis distúrbios.

Vale a pena destacar que para o monitoramento em tempo real do sistema de controle de nível, optou-se por utilizar o software MATLAB, o qual possibilita a criação de algoritmos de controle bastante sofisticados. Neste caso, utiliza-se a comunicação serial como interface para a troca de dados entre a planta de controle e o software.

2.2 Abordagem para o projeto de controladores PI/PID

2.2.1 Identificação de processos

A identificação de um sistema consiste, basicamente, na determinação de um modelo matemático com base em informações do sistema a ser controlado (COELHO; COELHO, 2004). O objetivo principal desta etapa é obter uma função de transferência (FT) que caracterize da melhor forma possível a dinâmica de um determinado sistema de controle, de modo a auxiliar a etapa de sintonia do controlador a ser empregado, ou seja, ajuste de seus ganhos ou parâmetros.

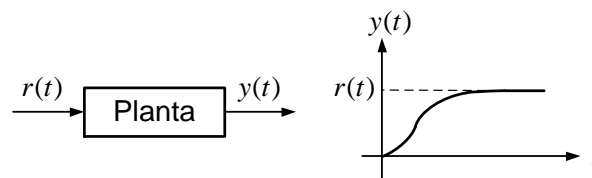
Essa função, em geral, é obtida no domínio do tempo, sendo representada com o auxílio da transformada de Laplace. Alguns processos industriais, tais como sistemas de controle de nível, temperatura, vazão ou pressão, por exemplo, muitas

vezes podem ser representados por FTs de primeira ou de segunda ordem (OGATA, 2010). Neste trabalho, a planta de controle de nível será modelada como um sistema de primeira ordem, cuja FT é dada pela equação (1):

$$(1)$$

onde K é o ganho em regime permanente, L o atraso de transporte e T a constante de tempo do sistema. Para a obtenção desta FT alguns métodos de identificação de sistemas baseados na resposta do processo ao degrau podem ser considerados. Tal resposta, como ilustrado na Figura 2 por $y(t)$, assume um formato em “S” o que facilita a identificação do processo por meio de tais métodos, que já fornecem equações características para a determinação dos parâmetros K , L e T da equação (1).

Figura 2 – Resposta de um processo ao degrau unitário

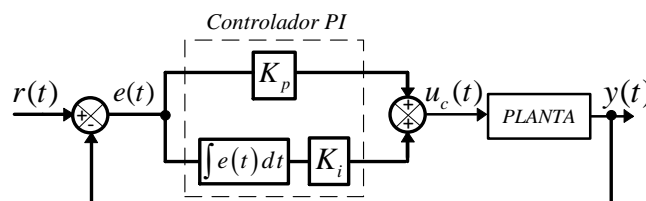


Fonte: Elaborado pelo autor (2021)

2.2.2 Sintonia de controladores PI/PID

Após a etapa de identificação do sistema, métodos de sintonia podem ser aplicados para projetar controladores PI/PID. Neste trabalho, considera-se o projeto de um controlador PI para a planta de nível, cuja estrutura é ilustrada na Figura 3, ação de controle $u_c(t)$ e FT $C(s)$ dadas, respectivamente, pelas equações (2) e (3):

Figura 3 – Estrutura do controlador PI



Fonte: Elaborado pelo autor (2021)

$$(2)$$

$$(3)$$

onde K_p e K_i são os ganhos proporcional e integrativo, respectivamente. Por sua vez, o erro atuante do sistema $e(t)$ é definido como a diferença entre os sinais de referência $r(t)$ e da saída do sistema $y(t)$ (que neste trabalho é o nível de água). A escolha dos parâmetros K_p e K_i geralmente atende a requisitos de desempenho, tais como máximo sobressinal, tempos de subida e de acomodação, por exemplo. Uma das técnicas empregadas no ambiente industrial é a tentativa e erro (do inglês *trial-and-error*), baseada em chutes empíricos para os ganhos. No entanto, esta técnica apresenta problemas: necessidade de inúmeros testes, às vezes exaustivos, e gasto de tempo.

Para solucionar este tipo de problema, métodos de sintonia foram desenvolvidos para uma melhor determinação de parâmetros, tais como os métodos i) baseados na resposta ao degrau (COELHO; COELHO, 2004), ii) baseados na análise do lugar das raízes (OGATA, 2010) e iii) baseados na resposta em frequência, sendo este último considerado neste artigo para o ajuste dos ganhos do controlador PI, cuja metodologia de projeto é detalhada em Angélico, Campanhol, Silva (2014). Além do conhecimento da FT do sistema no qual deseja-se controlar, dois parâmetros de projeto devem ser definidos: i) a frequência de cruzamento de ganho (ω_G) e ii) a margem de fase desejada (γ), estipulados pelo projetista (ou usuário).

2.2.3 Discretização do controlador

Para o desenvolvimento de sistemas de controle em simulações computacionais são utilizados modelos matemáticos baseados em equações diferenciais. Contudo, para a implementação física destes sistemas em microcontroladores, principalmente o controlador PI, devem ser consideradas equações de diferenças, dependentes do tempo de amostragem (T_s), uma variável importante a ser considerada em implementações de hardware digital.

Tais equações são obtidas a partir de métodos de discretização, equações que relacionam o tempo contínuo (s) com o tempo discreto (z), sendo geralmente considerados os métodos de aproximação *Backward*, *Forward* e Trapezoidal (ou Tustin), sendo este último o considerado para o presente trabalho, onde a variável de Laplace é definida como $s=(2/T_s)[z-1]/[z+1]$. Substituindo esta relação na FT da equação (3), e aplicando os conceitos relacionados à transformada z , obtém-se a equação de diferenças para a ação de controle do PI, dada pela equação (4):

(4)

em que as variáveis k e $k-1$ representam uma amostra atual e uma anterior no sistema em tempo discreto, respectivamente.

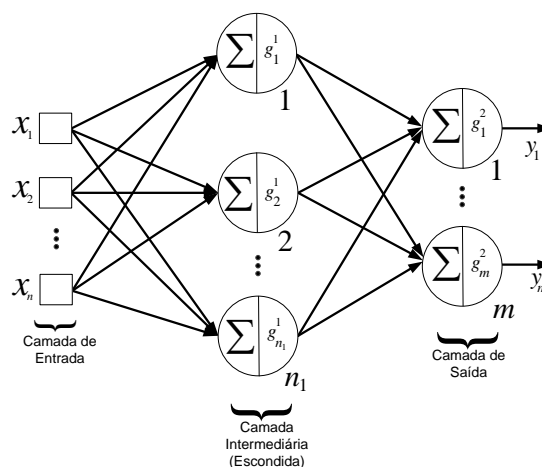
2.3 Abordagens baseadas em inteligência artificial

2.3.1 Redes neurais artificiais

As RNAs, nas últimas décadas, vêm se tornando uma abordagem amplamente estudada em diversas áreas do conhecimento, devido as suas potencialidades, que as tornam muito atrativas para inúmeras aplicações. Estas redes são modelos computacionais inspirados no sistema nervoso de seres vivos, que possuem capacidade de aquisição e manutenção de conhecimento/aprendizado (SILVA, 2016).

Uma das topologias mais empregadas, e que é considerada neste trabalho, é a Perceptron Multicamadas (PMC), pertencente à arquitetura *feedforward* de camadas múltiplas, muito aplicada em problemas que envolvem aproximação de funções, controle de processos, classificação de padrões, por exemplo (SILVA, 2016). Sua estrutura é caracterizada, por pelo menos uma camada intermediária (ou escondida) de n_1 neurônios, entre as camadas de entrada (com x_n sinais de entrada) e de saída (com m neurônios), como ilustra a Figura 4.

Figura 4 – Estrutura de uma rede PMC



Fonte: Elaborado pelo autor (2021)

Basicamente, cada neurônio artificial da rede encontra-se interconectado com suas respectivas entradas por meio dos pesos sinápticos (representados pelas flechas na Figura 4), que são vetores ou matrizes numéricas ajustados por meio dos algoritmos de aprendizagem. Cada neurônio, por sua vez, possui a capacidade de avaliar informações vindas de cada entrada, que são ponderadas a partir dos pesos, e geram uma saída proporcional a estes “estímulos”, com base em funções de ativação (representadas pela variável g na Figura 4).

O tipo de aprendizagem desta rede é o supervisionado, caracterizado pela disponibilidade de amostras de entrada com suas respectivas saídas. Logo, é desejável que o algoritmo busque garantir que as saídas da rede sejam praticamente idênticas às saídas desejadas. Para a rede PMC o algoritmo geralmente empregado é o *Backpropagation* (BP), cujos detalhes podem ser encontrados em Silva (2016).

2.3.2 Lógica fuzzy

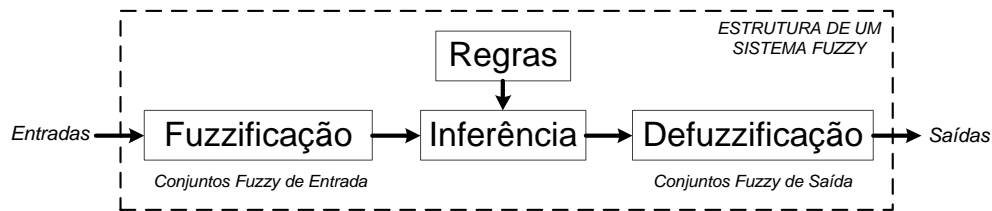
Os conceitos relacionados à lógica fuzzy surgiram da ideia de reprodução do raciocínio de seres humanos, capazes de lidar com problema complexos, a partir de informações imprecisas. Zadeh foi o idealizador dessa lógica, que buscava uma ferramenta matemática para o tratamento de informações vagas e que não fossem extremas (0 ou 1) como ocorre na lógica clássica (GOMIDE; GUDWIN, 1994).

Com base nessa estrutura lógica torna-se possível o desenvolvimento de diversos tipos de sistemas, denominados “sistemas fuzzy”. A Figura 5 ilustra a estrutura de um sistema fuzzy, o qual pode ser dividido em três etapas principais:

1. *Fuzzificação*: consiste no mapeamento das entradas (provenientes de uma dada aplicação) em conjuntos fuzzy (ou termos linguísticos), definidos para as variáveis linguísticas;
2. *Inferência*: processo que irá relacionar as entradas já condicionadas aos conjuntos fuzzy com as saídas do sistema. Nesta etapa consideram-se as regras linguísticas, conjunto de proposições fuzzy combinadas por meio de operadores lógicos, que irão associar as variáveis linguísticas de entrada com as de saída (definidas com base no conhecimento do usuário);
3. *Defuzzificação*: etapa final que irá transformar as ações fuzzy, definidas na inferência, em ações compatíveis com a aplicação, ou seja, é escolhida a

melhor ação para a decisão tomada no processo de inferência, baseada na utilização de métodos matemáticos.

Figura 5 – Estrutura básica de um sistema fuzzy



Fonte: Elaborado pelo autor (2021)

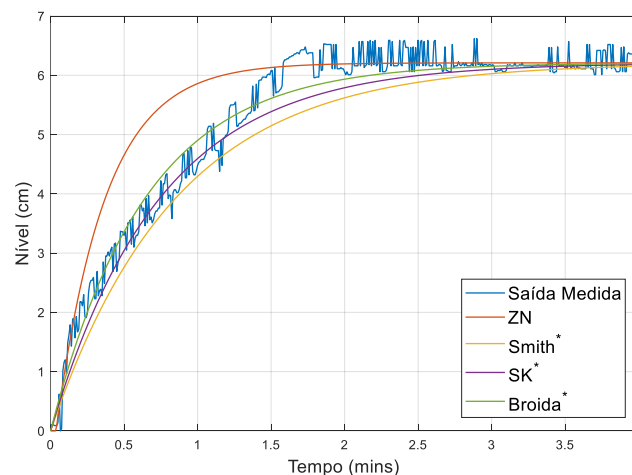
3. METODOLOGIA

3.1 Projeto do controlador PI convencional

A primeira etapa a ser considerada para o projeto do controlador PI é a caracterização do modelo matemático do sistema de controle de nível, ou seja, sua $G(s)$. Sendo assim, neste artigo, para a obtenção dessa função, considerou-se uma análise baseada na comparação das curvas de resposta ao degrau aproximadas fornecidas pelos métodos de identificação de i) Ziegler/Nichols (ZN), ii) Smith, iii) Sundaresan/Krishnaswamy (SK) e iv) Broida, sendo ilustradas na Figura 6.

Durante esta análise foi percebido que, para os métodos de Smith, SK e Broida, o comportamento de resposta ao degrau do sistema ficou mais próximo do sistema real quando o atraso de transporte (L) não foi considerado nas FTs. Na Figura 6 o símbolo (*) na legenda indica justamente esta constatação, ou seja, respostas ao degrau para as funções $G(s)$ resultantes sem considerar o atraso de transporte.

Figura 6 – Comparação das respostas ao degrau obtidas para cada método



Fonte: Elaborado pelo autor (2021)

Logo, para a escolha do melhor modelo da planta, considerou-se o índice “Erro Relativo Médio” (ERM), cuja equação é dada abaixo:

(5)

em que n é o número total de amostras, sendo $e(i) = y(i) - \hat{y}(i)$, onde $y(i)$ representa a saída real medida enquanto $\hat{y}(i)$ representa a saída estimada pelo modelo para uma dada i -ésima amostra. A Tabela 1 mostra os valores do ERM obtidos para cada um dos modelos resultantes.

Tabela 1 – Erro Relativo Médio (ERM) dos modelos aproximados

Modelo	ZN	ZN*	Smith	Smith*	SK	SK*	Broida	Broida*
<i>ERM</i>	0,4536	0,4995	0,4509	0,3854	0,3705	0,2729	0,2550	0,2331

Fonte: Elaborado pelo autor (2021)

Com base nos valores obtidos, é possível concluir que o menor ERM corresponde ao modelo dado pelo método de Broida desconsiderando o atraso de transporte, cuja FT aproximada para a planta de controle de nível é dada por:

(6)

Uma vez obtida a $G(s)$ do sistema de controle é possível desenvolver o projeto do controlador. Neste artigo, optou-se pela utilização do método de resposta em frequência para o projeto dos ganhos do PI, cujos parâmetros são a frequência de cruzamento de ganho (ω_G), dada em rad/s, e a margem de fase (γ), dada em graus.

Sendo assim, cinco controladores PI foram projetados para o sistema em questão, sendo cada um deles testado em protótipo, a fim de analisar a melhor configuração. Vale destacar que para a discretização dos PIs considera-se o método Trapezoidal, sendo adotada uma taxa de amostragem $T_s = 1$ ms. A Tabela 2 descreve os ganhos K_p e K_i obtidos em cada um dos projetos, assim como seus respectivos parâmetros ω_G e γ . Para a escolha do melhor PI, considerou-se os índices de desempenho IAE e IAU , descritos em (7) e (8), nos quais são realizados os cálculos das integrais dos módulos do erro $e(t)$ e da ação de controle $u_c(t)$, respectivamente. Os valores de IAE e IAU para cada um dos controladores também são mostrados na Tabela 2. Por sua vez, a Figura 7 ilustra o comportamento da variável de saída e da $u_c(t)$ do sistema para cada controlador (vale destacar que tais curvas foram obtidas considerando um cenário de degraus comentado em detalhes na seção 4).

(7)

(8)

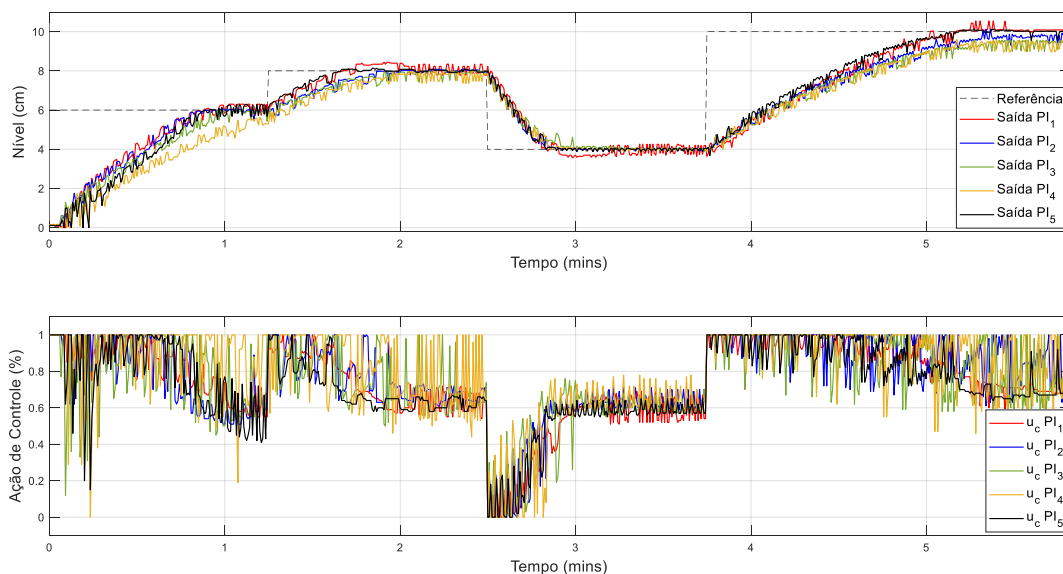
Tabela 2 – Resultados obtidos para os controladores PI

Controlador	ω_G (rad/s)	γ (°)	K_p	K_i	IAE	IAU
PI ₁	5	50	0,29237	2,27771	7,7842	4,3348
PI ₂	10	50	0,68826	7,87726	7,9292	4,4876
PI ₃	10	60	0,81459	6,56244	8,8910	4,3860
PI ₄	15	50	1,08414	16,79865	9,9303	4,6446
PI ₅	8	50	0,52990	5,23882	7,4583	4,1903

Fonte: Elaborado pelo autor (2021)

É possível observar nesta tabela que o controlador “PI₅” é o que resultou em menores valores de IAE e IAU, consequência de um melhor desempenho dentre os controladores analisados. Portanto, este será o controlador PI considerado para o desenvolvimento do projeto do PI-neural, assim como para a análise com os controladores inteligentes.

Figura 7 – Comparação entre os controladores PI projetados



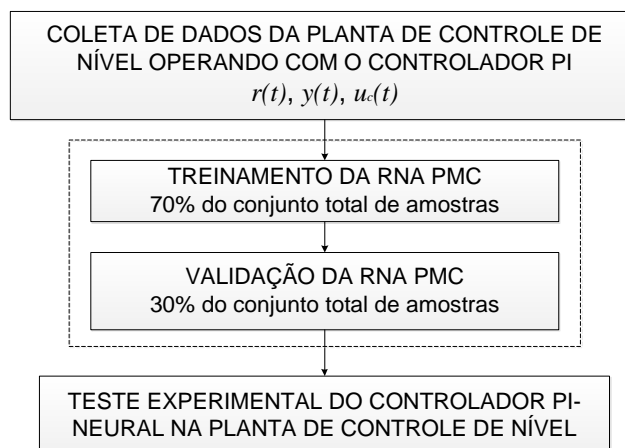
Fonte: Elaborado pelo autor (2021)

3.2 Projeto do controlador PI-neural

A Figura 8 ilustra a metodologia considerada para a implementação do controlador PI-neural neste artigo, composta por 4 etapas principais:

1. *Coleta dos Dados da Planta:* obtenção do conjunto total de amostras que serão consideradas para o treinamento e validação da RNA, obtidas a partir do sistema de controle operando com o controlador PI (escolhido na seção 3.1). Neste trabalho, considerou-se amostras da referência ($r(t)$), da saída ($y(t)$) e da ação de controle ($u_c(t)$) do sistema;
2. *Treinamento da RNA PMC:* etapa de aprendizagem da rede no qual 70% das amostras são consideradas;
3. *Validação da RNA PMC:* finalizado o treinamento, 30% do conjunto total de amostras, que não foram consideradas na etapa de aprendizagem, são utilizadas para a etapa de validação, visando verificar se a RNA retorna valores de saída próximos aos valores desejados;
4. *Teste do Controlador PI-neural na Planta:* uma vez encerrada a etapa de validação, os pesos sinápticos ajustados podem ser utilizados para a implementação em hardware do PI-neural visando avaliá-lo;

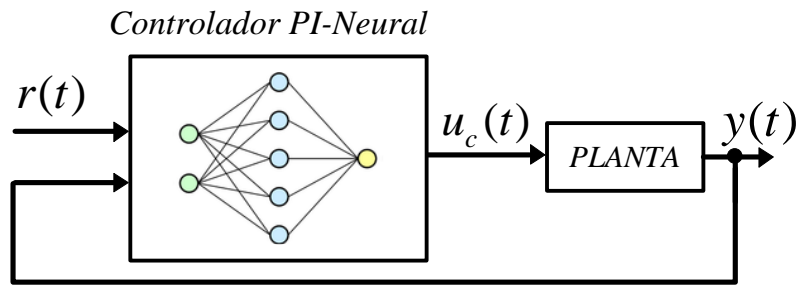
Figura 8 – Metodologia para a implementação do controlador PI-neural



Fonte: Elaborado pelo autor (2021)

Vale a pena destacar que a coleta dos dados, assim como as etapas de treinamento e validação foram feitas utilizando algoritmos desenvolvidos no MATLAB via linhas de código e sem utilização de *toolboxes*. O controlador PI-neural da planta de nível implementado neste trabalho é constituído por três camadas, conforme mostrado na Figura 9: i) *camada de entrada*, com dois neurônios correspondentes aos valores de referência ($r(t)$) e de saída do sistema ($y(t)$), ii) *camada escondida*, composta por cinco neurônios e iii) *camada de saída*, com um único neurônio, referente à ação de controle ($u_c(t)$). Os demais parâmetros considerados para o desenvolvimento do controlador são apresentados na Tabela 3.

Figura 9 – Estrutura do controlador PI-neural



Fonte: Elaborado pelo autor (2021)

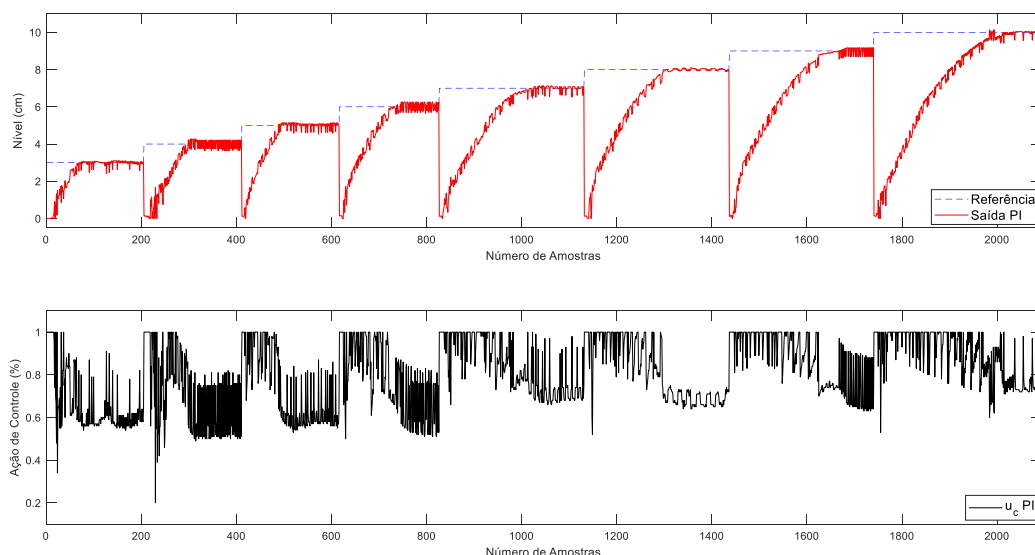
A Figura 10 ilustra o conjunto coletado de amostras composto por valores de referência, saída e ação de controle do protótipo do sistema de nível operando com o controlador PI projetado na seção 3.1. Tais amostras foram obtidas variando-se o valor de referência de 3 cm a 10 cm, com intervalos de 1 cm.

Tabela 3 – Parâmetros da RNA implementada

Neurônios na Camada de Entrada		$x_n = 2$ (referência e saída)	
Neurônios na Camada Escondida		$n_l = 5$	
Neurônios na Camada de Saída		$m = 1$ (ação de controle)	
Função de Ativação dos Neurônios		Tangente Hiperbólica	
Taxa de Aprendizagem do Algoritmo BP		0,001	
Erro requerido		1.10^{-8}	
Número Total de Amostras	Treinamento (70%)	2094	1466
	Validação (30%)		628
Época de Convergência		8568	

Fonte: Elaborado pelo autor (2021)

Figura 10 – Conjunto de amostras coletadas da planta de controle de nível



Fonte: Elaborado pelo autor (2021)

As Tabelas 4 e 5 descrevem os valores dos pesos sinápticos dos neurônios das camadas escondida e de saída obtidos ao final da etapa de treinamento da RNA.

A Figura 11 ilustra o resultado do processo de validação da RNA, no qual é possível observar que a rede é capaz de generalizar de forma correta e aproximada ao comportamento do sistema.

Tabela 4 – Pesos sinápticos dos neurônios da camada escondida

	Limiar (θ)	Referência ($r(t)$)	Saída ($y(t)$)
N1	0,6864	0,1139	1,0349
N2	0,6994	0,9739	-0,0011
N3	0,8202	0,5187	0,4461
N4	0,9360	0,4521	0,6384
N5	-0,9479	3,2381	-3,8754

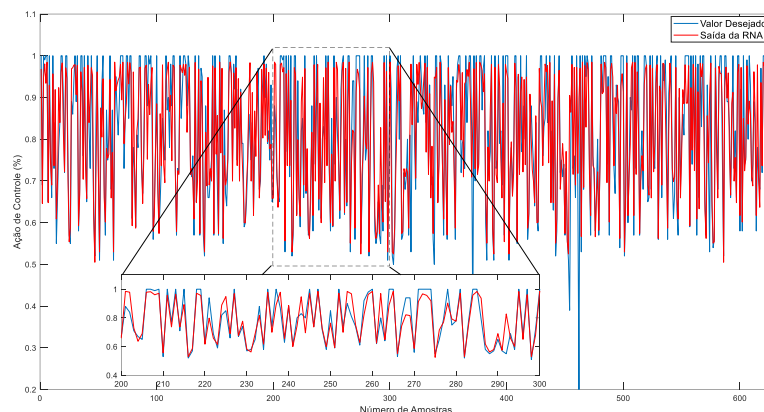
Fonte: Elaborado pelo autor (2021)

Tabela 5 – Pesos sinápticos do neurônio da camada de saída

	Limiar (θ)	N1	N2	N3	N4	N5
N1	0,2612	0,1581	0,7334	0,6548	0,2102	2,6793

Fonte: Elaborado pelo autor (2021)

Figura 11 – Resultado do processo de validação da RNA PMC



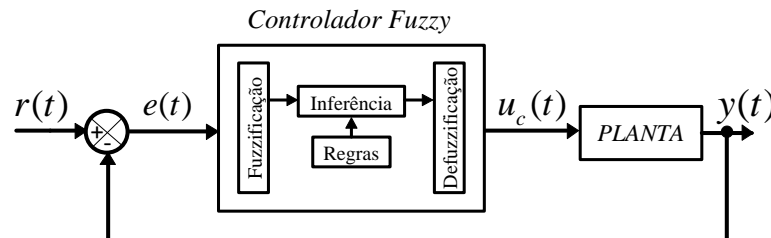
Fonte: Elaborado pelo autor (2021)

3.3 Projeto do controlador fuzzy

A Figura 12 ilustra a estrutura do controlador fuzzy para o controle da planta didática de nível. Como é possível notar, as variáveis linguísticas de entrada e de saída do controlador são o erro ($e(t)$) e a ação de controle ($u_c(t)$) do processo, respectivamente. Esse controlador é basicamente um sistema fuzzy convencional com os estágios de i) fuzzificação, etapa que mapeia o sinal do erro nos conjuntos fuzzy, ii) inferência, etapa caracterizada pelo processamento lógico do sistema, realizando a invocação das regras linguísticas e geração dos resultados, e iii) defuzzificação, conversão do resultado do processo de inferência em uma ação de

controle a ser enviada para a planta. Vale a pena destacar que para este trabalho considerou-se o controlador fuzzy com estrutura do tipo Mamdani, sendo implementado utilizando o *Fuzzy Logic Toolbox* do software MATLAB.

Figura 12 – Estrutura do controlador Fuzzy

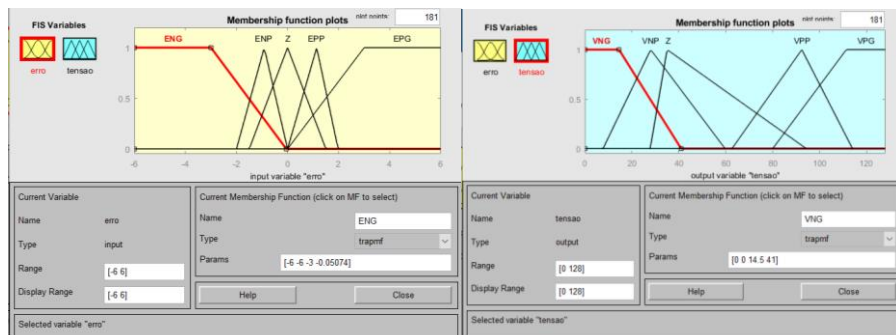


Fonte: Elaborado pelo autor (2021)

A Figura 13 ilustra as configurações definidas para as variáveis linguísticas de entrada, denominada “erro”, e de saída, denominada “tensão”. Para cada uma delas foram definidos cinco termos linguísticos de mapeamento, $T(erro)$ e $T(tensão)$, e os seguintes universos de discurso, $X(erro)$ e $X(tensão)$, sendo descritos a seguir:

1. *Variável “erro”*: ENG (erro negativo grande), ENP (erro negativo pequeno), Z (zero), EPP (erro positivo pequeno) e EPG (erro positivo grande): $T(erro) = \{ENG, ENP, Z, EPP, EPG\}$, $X(erro) = -6$ a 6 ;
2. *Variável “tensão”*: VNG (tensão negativa grande), VNP (tensão negativa pequena), Z (zero), VPP (tensão positiva pequena) e VPG (tensão positiva grande): $T(tensão) = \{VNG, VNP, Z, VPP, VPG\}$, $X(tensão) = 0$ a 128 ;

Figura 13 – Variáveis linguísticas de entrada (à esquerda) e de saída (à direita)



Fonte: Elaborado pelo autor (2021)

A Tabela 6 descreve os métodos considerados para os conectivos lógicos e operadores do controlador fuzzy. Na Figura 14 são descritas as regras linguísticas para a inferência, definidas com base em aspectos fundamentais de controle:

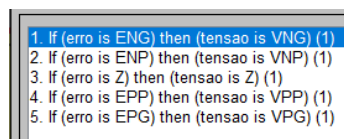
1. Se o erro de controle (positivo/negativo) é grande (ou seja, valor de saída muito distante do sinal de referência), muita ação de controle será necessária (ou seja, a tensão enviada para a bomba deverá ser alta);
2. Se o erro de controle (positivo/negativo) é pequeno (ou seja, valor de saída muito próximo do sinal de referência), pouca ação de controle será necessária (ou seja, a tensão enviada para a bomba deverá ser baixa);
3. Por fim, caso o erro de controle seja zero (indicando que o valor de saída é idêntico ao sinal de referência), nenhuma ação deverá ser realizada;

Tabela 6 – Parâmetros para o controlador fuzzy implementado

Parâmetros	Método
Norma-t e operador de implicação	Min
Conorma-t e operador de agregação	Max
Método de Defuzzificação	Centróide

Fonte: Elaborado pelo autor (2021)

Figura 14 – Regras linguísticas adotadas para o controlador fuzzy



```

1. If (erro is ENG) then (tensao is VNG) (1)
2. If (erro is ENP) then (tensao is VNP) (1)
3. If (erro is Z) then (tensao is Z) (1)
4. If (erro is EPP) then (tensao is VPP) (1)
5. If (erro is EPG) then (tensao is VPG) (1)

```

Fonte: Elaborado pelo autor (2021)

Como mencionado anteriormente, toda a estrutura do controlador fuzzy foi implementada considerando o *toolbox* do MATLAB. Portanto, para a integração deste com a planta de controle de nível, considerou-se um algoritmo de controle baseado em comunicação serial entre Arduino e MATLAB. Neste algoritmo, os valores do erro de controle, processados no Arduino, são enviados via serial para o MATLAB, sendo interpretados pelo controlador fuzzy desenvolvido, que gera sinais de atuação enviados novamente via serial para o Arduino, continuando assim a execução do algoritmo de controle no sistema microprocessado.

4. APRESENTAÇÃO E DISCUSSÃO DOS RESULTADOS

A seguir são ilustrados os resultados experimentais obtidos para a planta de controle de nível operando com cada um dos controladores projetados, segundo procedimentos apontados nas seções anteriores, neste caso, com o: i) controlador PI, ii) controlador PI-neural e iii) controlador fuzzy. Basicamente, para realizar o estudo comparativo entre os controladores, vários degraus de referência foram considerados, sendo adotado o seguinte cenário experimental: i) $0 < t < 1,25$ mins: referência de

nível fixa em 6 cm; ii) $1,25 < t < 2,5$ mins: referência de nível fixa em 8 cm; iii) $2,5 < t < 3,75$ mins: referência de nível fixa em 4 cm; iv) $3,75 < t < 5,8$ mins: referência de nível fixa em 10 cm;

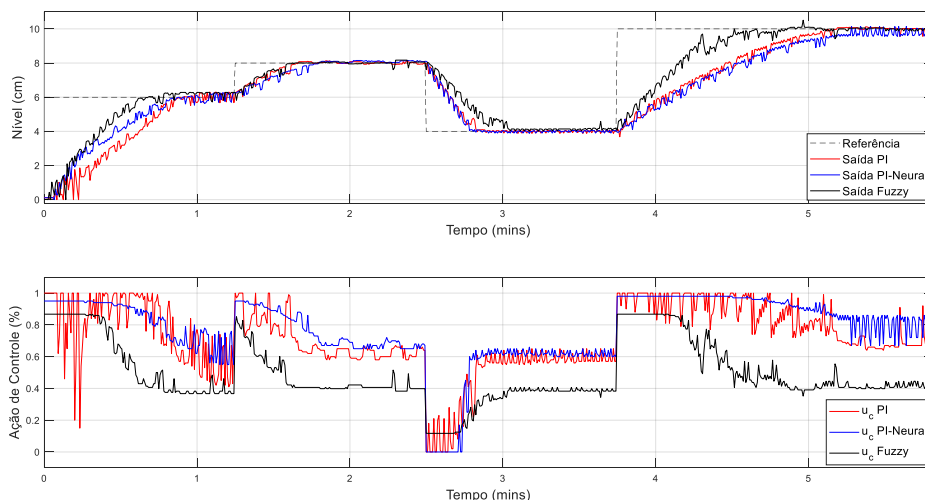
A Figura 15 ilustra os comportamentos do sistema de controle operando considerando os três controladores em estudo. Além disso, para avaliar o desempenho dos controladores, novamente optou-se pela utilização dos índices *IAE* e *IAU*, definidos em (7) e (8), respectivamente. Os valores obtidos considerando estes índices para os controladores projetados são descritos na Tabela 7.

É possível observar nos resultados experimentais, em um primeiro momento, que os três controladores respondem de maneira adequada e efetiva aos degraus de referência impostos, com uma rápida estabilização e sem grandes sobressinais. Analisando minuciosamente o comportamento dos sinais de saída é possível notar que as respostas do sistema operando com os controladores PI e PI-neural são muito similares, com o PI operando mais rápido em quase todos os períodos, exceto entre $0 < t < 1$ min no qual o PI-neural foi mais ágil. Contudo, o sistema operando com o controlador fuzzy obteve, principalmente nos momentos onde houveram degraus de subida, uma resposta mais rápida que os controladores PI e PI-neural.

Além disso, a partir dos resultados obtidos para a ação de controle, é possível notar uma grande característica relacionada aos controladores inteligentes que, neste caso, é o fato de reduzir consideravelmente oscilações de tensão em praticamente todo o cenário de teste, com destaque novamente para o sistema operando com o controlador fuzzy, cujo comportamento da ação de controle é bem menos oscilatório, o que significa menos esforços na atuação da bomba de água.

Por sua vez, os valores retornados pelos índices de desempenho, descritos na Tabela 7, comprovam uma melhor eficiência de controle do sistema operando com o controlador fuzzy, visto que ambos *IAE* (relacionado ao erro de controle) e *IAU* (relacionado à ação de controle) resultaram em menores valores quando comparados aos controladores PI e PI-neural.

Figura 15 – Comportamentos da planta de controle de nível com os controladores projetados



Fonte: Elaborado pelo autor (2021)

Tabela 7 – Comparação de desempenho entre os controladores projetados

Controlador	IAE	IAU
PI	7,4583	4,1903
PI-Neural	7,4373	4,4860
Fuzzy	5,4566	2,8586

Fonte: Elaborado pelo autor (2021)

Um outro aspecto a ser ressaltado com o presente trabalho é com relação aos custos de implementação experimental da planta de controle de nível, os quais estão descritos na Tabela 8. É possível observar que o gasto total do protótipo foi de 435,31 reais, valor consideravelmente baixo quando comparado aos kits disponíveis no mercado, justamente pelo fato de ser composto por dispositivos de baixo custo, além de serem bastante acessíveis.

Tabela 8 – Custos da planta didática de controle

Componentes	Valor
Arduino Uno	R\$ 70,20
Sensor Ultrassônico HC-SR04	R\$ 16,90
Módulo PWM D4184 30A/36V	R\$ 10,90
Fonte Chaveada 60W 12V 5A	R\$ 49,21
Bomba de Água 12 V	R\$ 35,00
Display LCD 16x2 + Módulo I2C	R\$ 22,90
Tanque Acrílico	R\$ 70,00
Reservatório de Água	R\$ 64,90
Cabos 22AWG + Jumpers	R\$ 5,30
Engates Flexíveis de PVC + Mangueira	R\$ 15,00
Válvula Esfera 1/2" Manual	R\$ 15,00
Réguas Plásticas	R\$ 10,00

Placas de Madeira em MDF	R\$ 40,00
Suportes Metálicos + Parafusos	R\$ 10,00
TOTAL	R\$ 435,31

Fonte: Elaborado pelo autor (2021)

5. CONSIDERAÇÕES FINAIS

Este trabalho descreveu uma aplicação de controladores inteligentes baseados em lógica fuzzy e redes neurais artificiais em uma planta didática de controle de nível de baixo custo operando com a plataforma Arduino Uno. Todos os procedimentos de projeto de cada controlador foram comentados. Para o projeto do controlador PI foi realizada a modelagem matemática da planta, a partir do método de identificação de Broida, seguido da sintonia pelo método de resposta em frequência. Já o controlador PI-neural, foi projetado considerando amostras compostas por vários pontos de operação do sistema operando com o controlador PI projetado. Por sua vez, o controlador fuzzy também foi implementado segundo aspectos fundamentais relacionados à teoria de controle. Em seguida, os desempenhos dos três controladores foram comparados entre si, sendo notadas vantagens de operação dos controladores inteligentes principalmente com relação ao nível de oscilação de tensão no atuador do sistema. Após uma análise comparativa baseada em observação dos comportamentos dos sinais de saída e da ação de controle do sistema, assim como nos valores numéricos obtidos por índices de desempenho, pode-se notar que os controladores efetuaram um adequado controle para o sistema, com destaque para o controlador fuzzy, que obteve um desempenho superior.

REFERÊNCIAS

ALMEIDA, J. P. L. S. **Técnicas fuzzy aplicadas ao controle descentralizado**. Dissertação (Mestrado) – Universidade Tecnológica Federal do Paraná. Programa de Pós-Graduação em Engenharia Elétrica. Cornélio Procópio, 2014.

ANDRADE, L. H. S.; AGULHARI, C. M.; GOEDEL, A. Controle PI-neural aplicado a um sistema de vazão. In. XII Simpósio Brasileiro De Automação Inteligente, 2015. UFRN. **Anais...** Natal, 2015. p-468-473.

ANGÉLICO, B. A.; CAMPANHOL, L. B. G.; SILVA, S. A. O. Proportional-integral/proportional-integral-derivative tuning procedure of a single-phase shunt active power filter using Bode diagram. **IET Power Electronics**. v.7, n.10, p.2647–2659, ago. 2014.

ASTROM, K. J.; HÄGGLUND, T. **PID controllers: theory, design, and tuning**. 2. ed. Research Triangle Park: Instrument Society of America, 1995.

BERTACHI, A. H.; SILVA, L. R. B.; SUMAR, R. R.; ANGÉLICO, B. A.; GOEDEL, A. Controle de um processo multivariável em uma planta didática industrial utilizando redes neurais. In. XI Simpósio Brasileiro De Automação Inteligente, UFC. **Anais...** Fortaleza, 2013. p-1-6.

COELHO, A. A. R.; COELHO, L. S. **Identificação de sistemas dinâmicos lineares**. Florianópolis: UFSC, 2004.

FREIRE JÚNIOR, V. A.; VARGAS, A. N.; GOEDEL, A. Uma abordagem PI-neural aplicado ao controle de um servo-mecanismo. In. XI SIMPÓSIO BRASILEIRO DE AUTOMAÇÃO INTELIGENTE (SBAI), 2013. UFC. **Anais...** Fortaleza, 2013. p-1-6.

GOMIDE, F. A. C.; GUDWIN, R. R. Modelagem, Controle, Sistemas e Lógica Fuzzy. **SBA Controle & Automação**. v.4, n.3, p. 97-115, 1994.

HAMITI, K. Position control of a pneumatic actuator under the influence of stiction. **Control Engineering Practice**. v.4, n.8, p.1079–1088, 1996.

MCCROBERTS, M. **Arduino básico**. São Paulo: Novatec Editora, 2011.

OGATA, K. **Engenharia de controle moderno**. 5. ed. São Paulo: Pearson Prentice Hall, 2010.

SILVA, I. N. **Redes neurais artificiais: para engenharia e ciências aplicadas**. São Paulo: Artliber, 2016.

KOZÁK, S. State-of-the-art in control engineering. **Journal of Electrical Systems and Information Technology**. v.1, n.1, p.1-9, 2014.

Capítulo III - Desenvolvimento de um Neurônio Artificial: Um estudo de acadêmicos para acadêmicos.

Danilo Faveri Massi⁸

Diego André Gil⁹

Luis Gustavo Ferrareto Espontão¹⁰

Prof. Esp. Wesley Candido Silva¹¹

RESUMO

Este artigo tem como objetivo fornecer uma introdução para que, a comunidade acadêmica dos cursos de Engenharia, possa praticar de forma perspicaz e criativa a análise de novas tecnologias e as oportunidades trazidas por essas; habilitando, a mesma, em pesquisa e desenvolvimento de conceitos, tais como: a construção de uma Rede Neural Artificial, abordar uma base de dados para aprendizagem do neurônio, identificar o modelo de neurônio, codificar o treinamento do neurônio artificial, utilizar métodos de ajuste de peso da função e obter os parâmetros de forma dinâmica; demonstrando, desta forma, a capacidade de desenvolvimento de uma Rede Neural Artificial de uma camada, programada em Python. Por fim, que possa aplicar estes conhecimentos matemáticos, científicos, tecnológicos e instrumentais em engenharia, considerando os aspectos humanísticos, sociais, éticos, legais, ambientais e econômicos fazendo uso do método científico, preconizados nos projetos pedagógicos da Faculdade da Indústria SENAI Londrina.

Palavras-chave: RNA. Neuronio. Implementação *Python*. Aprendizagem.

⁸ Discente do programa de graduação em Engenharia Elétrica na Fac. Ind. SENAI Londrina.

⁹ Discente do programa de graduação em Engenharia Elétrica na Fac. Ind. SENAI Londrina.

¹⁰ Discente do programa de graduação em Engenharia Elétrica na Fac. Ind. SENAI Londrina.

¹¹ Docente do programa de pós-graduação e graduação na Fac. Ind. SENAI Londrina. E-mail: Wesley.candido@sistemafiep.org.br

Development of an Artificial Neuron: a study by academics for academics

ABSTRACT

This article aims to provide an introduction so that the academic community of Engineering courses can practice insightfully and creatively the analysis of new technologies and the opportunities brought by these; enabling the same in research and development of concepts, such as: the construction of an Artificial Neural Network, approaching a database for neuron learning, identifying the neuron model, coding the training of the artificial neuron, using adjustment methods function weight and get the parameters dynamically; demonstrating, in this way, the ability to develop a one-layer Artificial Neural Network, programmed in Python. Finally, that you can apply this mathematical, scientific, technological and instrumental knowledge in engineering, considering the humanistic, social, ethical, legal, environmental and economic aspects, making use of the scientific method, recommended in the pedagogical projects of the Faculty of Industry SENAI Londrina.

Keywords: RNA. Neuron. Python implementation. Learning.

1. INTRODUÇÃO

As redes neurais artificiais são antigas, e tem se algumas informações históricas que datam de 1943, segundo Furtado (2019), as primeiras informações sobre a neuro computação datam de 1943, em artigos de McCulloch e Pitts, em que sugeriam a construção de uma máquina baseada ou inspirada no cérebro humano. – A partir dessa data foram surgindo livros, construídos alguns computadores neuros, além de estudos e pouco desenvolvimento, até que em 1983, foi fundado um centro de pesquisa em neuro computação, e em 1987 muitas universidades anunciaram formações de institutos, para pesquisa e desenvolvimento.

As redes neurais artificiais, (RNAs) utilizam uma espécie de neurônio artificiais programáveis, que fazem quase que a mesma função de um neurônio humano, porém, foi baseado no humano. Segundo Moreira (2017), os neurônios são unidades do sistema nervoso, que recebem informações (sinais elétricos) de outros neurônios e de neuro receptores especializados, integrando estas informações em suas áreas

operacionais e encaminhando-as, ao final do processo, na forma de uma mensagem, em direção a outros neurônios ou para estruturas efetoras, músculos ou glândulas. – São desta forma que funcionam os neurônios humanos, já o neurônio artificial segundo Furtado (2019), é composta por um elevado número de elementos processadores, os neurônios, amplamente interligados através de conexões com um determinado valor que estabelece o grau de conectividade.

Segundo Carneiro (2001), existem diversas aplicações para redes neurais artificiais, tanto que o volume de investimentos nesta área tem alcançado somas expressivas tanto no Brasil quanto no exterior. Pode-se destacar as aplicações nas áreas médicas, financeiras, robótica, aeroespacial etc. – Sendo assim, temos uma infinidade de aplicações que podem ser utilizadas, e melhorias feitas para melhor desenvolvimento da sociedade das tecnologias.

2. METODOLOGIA

Para a implementação de um neurônio artificial de apenas uma camada, foi realizado utilizando Python através do *jupyter* notebook, trata-se de um ambiente de desenvolvimento *opensource*, amplamente utilizado por estudantes de programação e ciência de dados.

O primeiro passo para realizar a implementação de um neurônio artificial, se trata de ter uma base de dados para realizar o treinamento, que possibilita o neurônio realizar previsões através do aprendizado obtido pela base dos dados. Para esta implementação será utilizado uma base de dados para identificação de cores para pintura de peças de determinados tamanhos, como segue a Tabela 1.

Tabela 1 – Base de dados

Largura	Comprimento	Cor
3	1.5	Vermelho
2	1	Azul
4	1.5	Vermelho
3	1	Azul
3.5	0.5	Vermelho
4	0.5	Vermelho

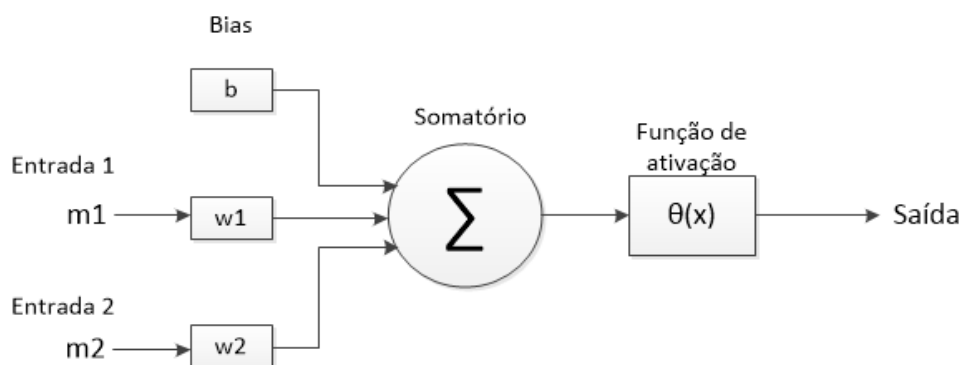
5.5	1	Vermelho
1	1	Azul

Fonte: Dos Autores.

Para codificarmos a Tabela 1 de forma que seja possível para o computador processar, devemos converter as saídas (Cor) para uma forma Booleana, visto que temos apenas duas opções, no caso foi definido que 1 indica vermelho e 0 indica azul.

Com a base de dados obtida e normalizada para uma linguagem que seja possível uma rede neural receber e processar os dados, o próximo passo é identificar o modelo de neurônio ideal para a aplicação, que pode ser verificado que o neurônio deverá ter no mínimo duas entradas e uma saída para retornar o resultado seguindo a base de dados, neste caso também foi inserido uma entrada extra para uma variável “Bias” com a estratégia de auxiliar ao modelo a se adaptar melhor aos dados, e apresenta uma saída que não seja nula (HAYKIN, 2001).

Figura 1 – Modelo Neurônio Artificial



Fonte: Dos Autores.

Como ilustrado pela Figura 1, o modelo proposto é dado por 3 variáveis de entrada, sendo elas, respectivamente: Bias, m1 e m2. Após as variáveis serem balanceadas pelos pesos w1 e w2, os valores passam por uma somatória onde são enviados a uma função de ativação do neurônio, que retorna a saída dele.

Para o neurônio artificial tomar uma decisão ele necessita de uma função que ative ele ou seja uma função de ativação por exemplo função de limiar, função rampa, função sigmoide, entre outras.

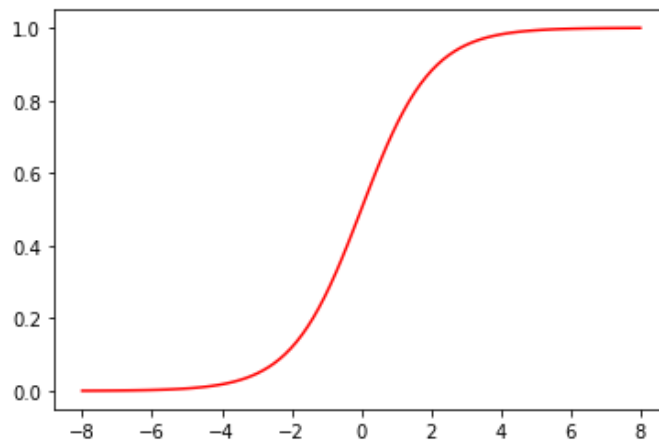
Segundo HAYKIN (2001), neurônios biológicos funcionam de forma binária (ativado ou não ativado), a função sigmoide (Figura 2) é uma boa forma de modelar

esse comportamento, já que assume valores apenas entre 0 (não ativação) e 1 (ativação). Para esta aplicação será utilizado a função sigmoide devido a simplicidade de implementação e fornecer uma resposta de propagação positiva, ou seja, a saída apenas para números positivos, expressa matematicamente pela equação 1.

$$Y(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

Sendo Y a resposta da função sigmoide a uma entrada x , sendo o valor resultante da somatória do neurônio.

Figura 2 – Função Sigmoide



Fonte: Dos Autores.

Será codificado o modelo desse neurônio utilizando a linguagem de programação Python, usando bibliotecas com funções matemáticas e para plotar os gráficos para a análise dos dados (Figura 3).

Figura 3 – Bibliotecas de função matemáticas e plotagem de gráficos

```
#Biblioteca para funções matemáticas
%matplotlib inline
import numpy as np
from matplotlib import pyplot as plt
```

Fonte: Dos Autores.

Definição das funções de somatória e ativação do neurônio (Figura 4).

Figura 4 – Funções do Neurônio Artificial

```
def NN(m1,m2,w1,w2,b):
    z = m1 * w1 + m2 * w2 + b
    return z
#Função Sigmoid
def sigmoid(x):
    return 1/(1 + np.exp(-x))
```

Fonte: Dos Autores.

Sendo NN a função somatória, que tem como parâmetros ilustrados pela Figura 1, e retornando o valor da somatória, e a função sigmoide sendo expressa da mesma maneira que a equação 1 por meio do uso das bibliotecas ilustradas na Figura 3.

Desta forma nosso neurônio está codificado e já é possível realizar previsões, inserindo os valores para cada parâmetro das funções e identificando o retorno da função sigmoide em relação ao valor da somatória do neurônio, entretanto, este valor de previsão não será preciso, pois não realizamos o treinamento deste neurônio, ou seja, obter os valores ideais para cada peso “w1”, “w2” e “b” para obter a resposta desejada para nossa entrada de dados.

Para este treinamento existem métodos empíricos, onde é definido um “passo” para cada valor de peso, e vai alterando até se ajustar para uma saída satisfatória, porém para casos em que temos uma grande quantidade de dados, com muitos pesos envolvidos e deseja-se uma resposta precisa, é necessária uma rotina de treinamento dinâmico, onde busca-se os valores de peso para cada variável, por meio de métodos de cálculo diferencial, buscando a menor taxa de erro.

Nesta aplicação iremos utilizar funções de Custo (*Cost Function*), que resulta o quão assertivo nosso modelo está com as atuais circunstâncias, em outras palavras, é a distância que nossa previsão está do valor alvo, expresso pela equação 2.

$$Erro = (Previsão - Alvo)^2 \quad (2)$$

Por exemplo, digamos que neste caso nosso Alvo seja o valor 1, então nossa previsão tem que se aproximar muito de 1, para que a diferença ao quadrado seja zero ou muito próxima a zero.

Agora vemos um desafio de como saber quando devemos aumentar ou diminuir nossos parâmetros para melhorar nossas saídas, dessa forma a maneira utilizada para este tipo de função, é basicamente calcular a inclinação da curva do erro, assim saberemos quando estamos aproximando do erro zero, ou próximo, aumentando ou diminuindo o valor de nossos pesos.

Para isso vamos utilizar métodos de cálculo, no caso sabemos que a derivada é a inclinação da reta tangente que passa por uma curva, onde no caso é exatamente o que desejamos saber, ao realizarmos a derivada na equação 2, obtemos a equação 3.

$$\text{Inclinação} = 2 (\text{Predição} - \text{Alvo}) \quad (3)$$

Ao analisar o comportamento das funções 2 e 3, é possível notar que quando nosso alvo for 1, e maior for nossa predição, mais tangencial ao plano nossa curva irá estar, pois se aproximara de zero, logo saberemos que estamos no ponto ideal da nossa curva de erro e dos parâmetros de pesos.

Por fim, para atualizar dinamicamente nossos parâmetros, podemos aplicar um passo a função de atualização, ou seja, multiplicar o valor de nossa inclinação por uma constante (chamado de *learning rate*) e subtrair de nosso erro, assim a cada “passo” de nossa função irá tender a 0, ou seja, tangente ao plano x. Na aplicação foi realizado vários testes, ajustando o learning rate e a quantidade de iteração do loop até encontrar um resultado satisfatório.

A implementação deste treinamento em Python segue ilustrado pela Figura 5.

Figura 5 – Loop de Treinamento

```

#Loop de treinamento
learnig_rate = 0.2 # "passo" ou constante que multiplica o valor da inclinação
costs = []

#Assume valores randomicos
w1 = np.random.randn()
w2 = np.random.randn()
b = np.random.randn()

for i in range(50000): #Loop de 50000 iteração, ou seja a rotina de treino irá rodar em um laço de 50000 vezes.

    Sample = np.random.randint(len(data)) #Retorna um valor aleatório da matriz de dados
    Dados = data[Sample]
    z = NN(Dados[0],Dados[1],w1,w2,b)
    pred = sigmoid(z)

    target = Dados[2]

    error = cost(pred, target) #Calculo do erro

    inclin = slope(pred, target) #Inclinação da reta do erro

    dpred_dz = d_sigmoid(z) #Calcula a derivada da predição

    dz_dw1 = Dados[0] #Derivada de w1
    dz_dw2 = Dados[1] #Derivada de w2
    dz_b = 1 #Derivada de b

    #Calcula os erros para cada parametro por meio da multiplicação da inclinação da curva do erro, a derivada da predição
    # e a derivada do parametro a ser calculado, assim obtem-se a derivada do erro em relação ao parametro desejado
    dcost_dw1 = inclin * dpred_dz * dz_dw1
    dcost_dw2 = inclin * dpred_dz * dz_dw2
    dcost_b = inclin * dpred_dz * dz_b

    #Calcula o novo peso para cada parametro, subtraindo o Learning rate, e multiplicando pela parcela de erro do peso
    #calculado anteriormente
    w1 = w1 - learnig_rate * dcost_dw1
    w2 = w2 - learnig_rate * dcost_dw2
    b = b - learnig_rate * dcost_b

    #A cada 100 iterações do loop, coleta uma amostra, realiza uma predição, calcula o erro (cost) e armazena
    #no vetor e realiza a média de erros e plota no gráfico.
    if i % 100 == 0:
        const_sum = 0
        for j in range(len(data)):
            Dados = data[Sample]

            z = NN(Dados[0],Dados[1],w1,w2,b)
            pred = sigmoid(z)

            target = Dados[2]
            const_sum += cost(pred, target)

        costs.append(const_sum/len(data))
plt.plot(costs)

```

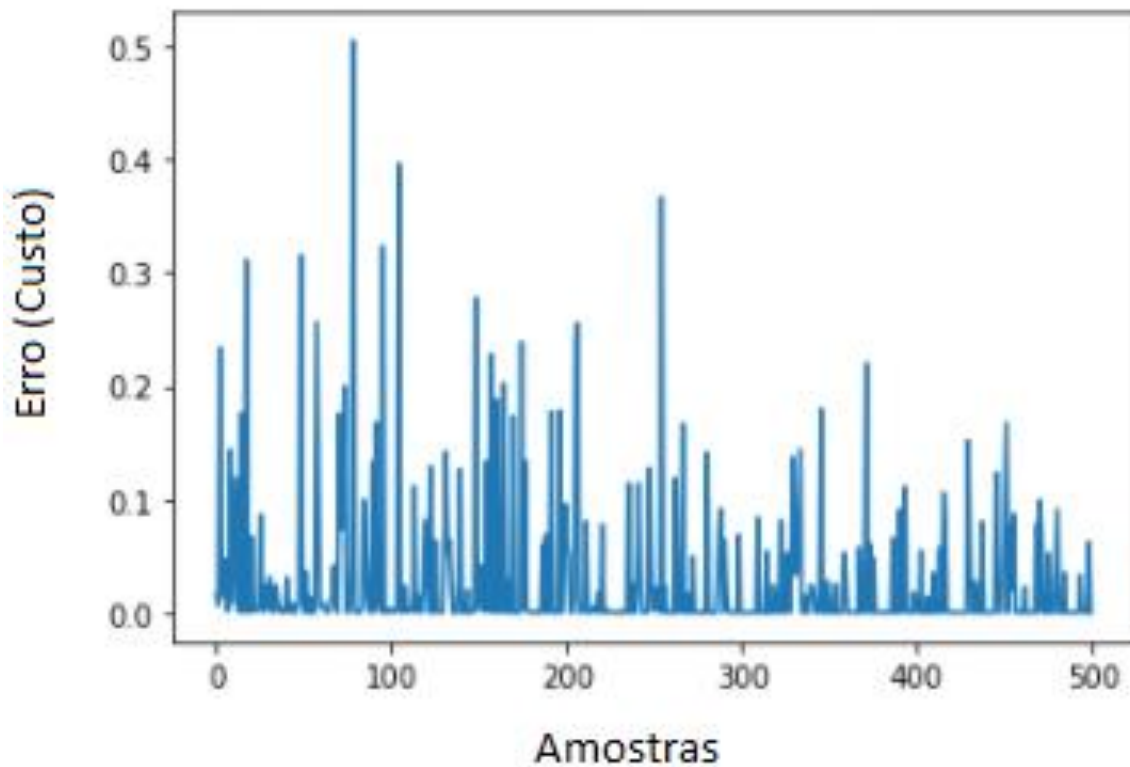
Fonte: Dos Autores.

Seguindo esta metodologia, resumidamente, foi realizada a implementação do neurônio artificial e foi realizado uma coleta dos dados de saída por meio da plotagem da curva de aprendizagem do neurônio e suas respectivas saídas para a base de dados para validação do treinamento.

3 CONCLUSAO

Os resultados obtidos com o treinamento do neurônio artificial foram satisfatórios, como segue a ilustração da curva de aprendizado (Figura 6).

Figura 6 – Curva de aprendizagem



Fonte: Dos Autores.

Analisando o gráfico vemos que a partir de 400 amostras, a curva de erro começou a diminuir para um nível de aproximadamente 0,1 de erro, que para a aplicação foi totalmente satisfatória, por ser com o foco de auxiliar no aprendizado e implementação de algoritmos inteligentes. Para validar esse treinamento, também foi realizado teste com os dados do treinamento, e plotados sequencialmente com o dado analisado e a predição do neurônio (Figura 6).

Por se tratar de uma implementação de um sistema que tenta copiar a complexidade de um neurônio do ser humano, a base de conhecimento e estudo a cerca do tema pode ser muito desafiador, por conter muitas aplicações de conceitos de estáticas e cálculo, como pode ser visto na metodologia deste trabalho.

Figura 6 – Resultados das predição do Neurônio Artificial

[3, 1.5, 1]
Pred: 0.7014570549857383
[2, 1, 0]
Pred: 0.000568134260863994
[4, 1.5, 1]
Pred: 0.9994705814285146
[3, 1, 0]
Pred: 0.31353833397575315
[3.5, 0.5, 1]
Pred: 0.715647787964623
[4, 0.5, 1]
Pred: 0.9861763234026596
[5.5, 1, 1]
Pred: 0.9999998803578608
[1, 1, 0]
Pred: 7.074909868003077e-07

Fonte: Dos autores.

O principal foco é a difusão do conceitos para os alunos e colegas de classe da graduação, ou qualquer estudante que esteja realizando pesquisas sobre o tema. Desta forma é nítido que este trabalho tem a contribuição apenas de realizar um pequeno mergulho no ramo de ciencia de dados e redes neurais, onde foi passado por vários conceitos de maneira resumida que é possível encontrar estudos apenas voltados para o conceito aplicado.

Por fim, para auxílio do estudante interessado em implementar o neurônio artificial aplicado na metodologia deste trabalho, o código é de licença livre, encontrando-se no anexo A deste documento.

REFERÊNCIAS

HAYKIN, S. **Redes Neurais, princípios e práticas**. Porto Alegre: Bookman, 2001. Disponível em: <https://www.deeplearningbook.org>.

Moreira, É. S. **Os neurônios, as sinapses, o impulso nervoso e os mecanismos morfo-funcionais de transmissão dos sinais neurais no sistema nervoso**. Volta Redonda: UniFOA, 2017. v.2. p.81 II. Disponível em <http://editora.unifoa.edu.br/wp-content/uploads/2017/04/Volume-02.pdf>.

Furtado, M. I. V. **Redes neurais artificiais: uma abordagem para sala de aula**. Ponta Grossa: Atena, 2019. Disponível em: <https://www.atenaeditora.com.br/wp-content/uploads/2019/05/e-book-Redes-Neurais-Artificiais-uma-Abordagem-para-Sala-de-Aula.pdf>.

Carneiro, D. L. **Um estudo sobre a aplicabilidade de redes neurais em criptografia.** UFSC, Florianópolis 2001

ANEXO A

```

#Biblioteca para funções matemáticas
%matplotlib inline
import numpy as np
from matplotlib import pyplot as plt

#Dados para treinamento - 1 = vermelho, 0 = azul
#Os dados se tratam de medidas de produtos de uma máquina que são de
tamanhos diferentes
#que precisam ser selecionados por cor para identificação.

data = [[3, 1.5, 1],
        [2, 1, 0],
        [4, 1.5, 1],
        [3, 1, 0],
        [3.5, 0.5, 1],
        [4, 0.5, 1],
        [5.5, 1, 1],
        [1, 1, 0]]

#Função que define nosso neurônio com 2 entradas e 2 pesos para entrada,
com a função de ativação Sigmoid, que retorna valores de 0 a 1.
def NN(m1,m2,w1,w2,b):
    z = m1 * w1 + m2 * w2 + b
    return z

#Função Sigmoid
def sigmoid(x):
    return 1/(1 + np.exp(-x))

#Derivada da função Sigmoid
def d_sigmoid(x):
    return sigmoid(x) * (1 - sigmoid(x))

#Plota os gráficos das funções Sigmoid e Derivada da Sigmoid
T = np.linspace(-8,8,100)
plt.plot(T,sigmoid(T), c='r')
#plt.plot(T,d_sigmoid(T), c='b')

#Função para obter o erro das predições
def cost(pred, target):
    return np.square(pred - target)

#Função para obter a inclinação da curva (derivada do erro (cost))
def slope(pred, target):
    return 2 * (pred - target)

```

```

#Loop de treinamento

learnig_rate = 0.2
costs = []

w1 = np.random.randn()
w2 = np.random.randn()
b = np.random.randn()

for i in range(50000):

    Sample = np.random.randint(len(data)) #Retorna um valor aleatório da
matriz de dados
    Dados = data[Sample]
    z = NN(Dados[0],Dados[1],w1,w2,b)
    pred = sigmoid(z)

    target = Dados[2]

    error = cost(pred, target)

    inclin = slope(pred, target)
    dpred_dz = d_sigmoid(z)

    dz_dw1 = Dados[0]
    dz_dw2 = Dados[1]
    dz_b = 1

    dcost_dw1 = inclin * dpred_dz * dz_dw1
    dcost_dw2 = inclin * dpred_dz * dz_dw2
    dcost_b = inclin * dpred_dz * dz_b

    w1 = w1 - learnig_rate * dcost_dw1
    w2 = w2 - learnig_rate * dcost_dw2
    b = b - learnig_rate * dcost_b

    #A cada 100 iterações do loop, coleta uma amostra, realiza uma
predição, calcula o erro (cost) e armazena
    #no vetor e realiza a média de erros e plota no gráfico.
    if i % 100 == 0:
        const_sum = 0
        for j in range(len(data)):
            Dados = data[Sample]

            z = NN(Dados[0],Dados[1],w1,w2,b)
            pred = sigmoid(z)

            target = Dados[2]
            const_sum += cost(pred, target)

        costs.append(const_sum/len(data))

plt.plot(costs)

#Valida os dados obtidos no treinamento do modelo

```

```
for i in range(len(data)):  
    Dados = data[i]  
    print(Dados)  
    z = NN(Dados[0],Dados[1],w1,w2,b)  
    pred = sigmoid(z)  
    print("Pred: {}".format(pred))
```

Capítulo IV – Programa de Voluntariado do Sistema FIEP – Abrace uma Criança

Prof.(a) Dra. Camila Fogaça de Oliveira¹²

Prof. Esp. Antonio Carlos Rodrigues¹³

Prof. Esp. Wesley Candido da Silva¹⁴

Prof. MS.c. Marco Aurelio Arbex¹⁵

RESUMO

Alunos da Faculdade da Indústria Senai Londrina participaram do Programa de Voluntariado do Sistema Fiep para (i) manutenção, restauração do parquinho e da fachada do Centro de Educação Infantil Alaide Fausto Souza (Instituição filantrópica, sem fins lucrativos, que atende no regime de Educação Infantil, em período integral, crianças de 1 (um) a 4 (quatro) anos de idade em Londrina -PR); (ii) aquisição de brinquedos para parquinho e liquidificador industrial para garantir melhor preparo dos alimentos para os alunos do Centro de Educação Infantil Estrelinha (Instituição filantrópica, sem fins lucrativos, que atende no regime de Educação Infantil, em período integral e parcial, crianças de 2 (dois) a 5 (cinco) anos de idade em Londrina -PR). A ação visou o enriquecimento curricular, a aplicação dos conhecimentos adquiridos ao desenvolvimento social, prestação de serviços à comunidade e a difusão de conhecimentos e técnicas pertinentes às áreas dos cursos oferecidos pela Faculdade da Indústria Senai Londrina.

Palavras-chave: Voluntariado. Projeto de Extensão. Engenharias.

FIEP SYSTEM VOLUNTEERING PROGRAM - HUG A CHILD

ABSTRACT

Students from the Faculdade da Indústria Senai Londrina participated in the Volunteering Program of the Fiep System for (i) maintenance, restoration of the

¹² Doutorado em Ensino de Ciências e Educação Matemática. Docente da Faculdade de Tecnologia Senai Londrina. E-mail: camila.oliveira@sistemafiep.org.br

¹³ Especialização em Engenharia de Manutenção Industrial. Docente da Faculdade de Tecnologia Senai Londrina. E-mail: antonio.rodrigues1@sistemafiep.org.br

¹⁴ Especialização em Engenharia de Automação. Docente da Faculdade de Tecnologia Senai Londrina. E-mail: wesley.candido@sistemafiep.org.br

¹⁵ Mestrado em Administração. Docente da Faculdade de Tecnologia Senai Londrina. E-mail: marco.arbex@sistemafiep.org.br

playground and the facade of the Alaide Fausto Souza Early Childhood Education Center (a non-profit philanthropic institution that serves the Early Childhood Education regime , full-time, children from 1 (one) to 4 (four) years of age in Londrina -PR); (ii) purchase of toys for the playground and industrial blender to ensure better food preparation for the students of Centro de Educação Infantil Estrelinha (a non-profit philanthropic institution that provides full-time and part-time Early Childhood Education services to children from 2 (two) to 5 (five) years old in Londrina -PR). The action aimed at curricular enrichment, the application of acquired knowledge to social development, provision of services to the community and the dissemination of knowledge and techniques relevant to the areas of the courses offered by the Faculdade da Indústria Senai Londrina.

Key-words: Volunteering. Extension Project. engineering.

6. INTRODUÇÃO

O Programa de Voluntariado do Sistema Fiep está vinculado à missão, visão e valores do Sistema Fiep e alia-se ao esforço global de melhorias sociais previstas nos Objetivos de Desenvolvimento Sustentável (ODS) da ONU. O Programa é aberto à participação espontânea de todos os colaboradores com contrato ativo, independentemente do tipo de contrato.

Com 16.321 pessoas beneficiadas em 2021, o Programa de Voluntariado do Sistema Fiep foi contemplado com o Prêmio VoL, que reconhece as melhores práticas de gestão de voluntariado do país. Foram 20 projetos que atenderam 14 cidades paranaenses e transformaram a vida de comunidades carentes (AGÊNCIA SISTEMA FIEP, 2021). Aqui em Londrina, submetemos um projeto que foi aprovado, denominado de Abrace uma Criança.

7. DO PROJETO

O bom funcionamento das unidades escolares está diretamente associado à estrutura oferecida às equipes pedagógicas e aos alunos. Equipamentos adequados trazem mais estrutura aos profissionais e, também, garantem maior qualidade no atendimento oferecido às crianças.

A Educação Infantil, primeira etapa da Educação Básica, tem como finalidade garantir condições adequadas para promover o desenvolvimento integral da criança

de zero a cinco anos em seus aspectos físico, psicológico, intelectual e social, bem como favorecer a construção da identidade e autonomia, complementando a ação da família e da comunidade.

A criança é percebida como “[...] sujeito histórico e de direitos que, nas interações, relações e práticas cotidianas que vivencia, constrói sua identidade pessoal e coletiva, brinca, imagina, fantasia, deseja, aprende, observa, experimenta, narra, questiona e constrói sentidos sobre a natureza e a sociedade [...]” (BRASIL, 2009).

Atualmente, em Londrina, há 56 Centros de Educação Infantil (C.E.I.) Filantrópicos (PREFEITURA DO MUNICÍPIO DE LONDRINA, 2021). Essas instituições possuem convênio com a prefeitura que auxilia em um aporte mensal para custeio de algumas despesas, mas esse valor não é o suficiente para a manutenção das instituições. Dessa forma, recebem contribuições voluntárias, realizam rifas e promoções para sua subsistência e, mesmo assim, apresentam carência de infraestrutura para seu funcionamento.

As instituições escolhidas para o desenvolvimento da ação estão listadas no Quadro 1.

Quadro 1 – Descrição das instituições e necessidades identificadas

	Descrição	Necessidades identificadas
C.E.I. Alaide Fausto Souza	Instituição filantrópica, sem fins lucrativos, que atende no regime de Educação Infantil, em período integral, crianças de 1 (um) a 4 (três) anos de idade, distribuídas conforme a faixa etária, denominadas nas seguintes turmas: Creche 1 – C1 (1-2 anos), Creche 2 – C2 (2-3 anos) e Creche – C3 (3-4 anos).	Restauração de parquinho, manutenção elétrica, pintura de muro e fachada, placa para identificação de escola e número, grama sintética para bebês, porta e divisória para banheiro, microondas para lactário, computadores para secretaria, ventilador de parede ou teto para sala de aula, armários para uso em salas de aula, telhas.
C.E.I. Estrelinha	Instituição filantrópica, sem fins lucrativos, que atende no regime de Educação Infantil, em período integral e parcial, crianças de 2 (dois) a 5 (cinco) anos de idade, distribuídas conforme a faixa etária, denominadas nas seguintes turmas: Creche 2 – C2 (2-3 anos), Creche – C3 (3-4 anos) e Pré-Escola – P4 (4-5 anos).	Brinquedos para parquinho e equipamentos industriais para garantir melhor preparo dos alimentos para os alunos

Fonte: Autor (2022).

Nestas instituições são servidas diariamente: café da manhã, almoço, lanche da tarde e fruta. Nessa rotina, são processados em liquidificador grande quantidade de leite achocolatado, suco de frutas, polpa de frutas, preparo de bolos e massas, etc. As principais vantagens em adquirir um equipamento industrial são a resistência, agilidade e a rapidez em produzir uma quantidade maior de produtos.

A valorização dos espaços de recreação e vivência garante um padrão mínimo de qualidade de ensino, incrementa a interação das crianças, a partir do desenvolvimento de jogos, brincadeiras e atividades coletivas. A inclusão/revitalização de brinquedos para diferentes faixas etárias visa estimular brincadeiras e atividades coletivas, desde que atendam às normas de segurança com manutenção periódicas.

Além da diversão, o momento de brincar no parquinho, proporciona desempenho físico, promove um melhor desempenho motor, estimula a criatividade, favorece o raciocínio, facilita também o convívio social.

Essa campanha faz parte dos projetos de extensão realizados pela Faculdade da Indústria Senai Londrina. As atividades de extensão da Faculdade da Indústria Senai Londrina visam a interação entre a Faculdade e a sociedade, integrando a ciência, o ensino, à pesquisa e a aplicação dos conhecimentos adquiridos ao desenvolvimento social local, assim como o enriquecimento curricular, a prestação de serviços à comunidade e a difusão de conhecimentos e técnicas pertinentes às áreas dos cursos oferecidos.

8. DISCUSSÃO DOS RESULTADOS

a) C.E.I. Alaide Fausto Souza

Para efetivação da restauração do parquinho, a Infocable Infraestrutura e Tecnologia LTDA, empresa especializada em infraestrutura física para ambiente de tecnologia, e Casa das Cores em Iporã, doaram os materiais necessários de pintura. Essa ação iniciou no dia 25/09/2021 e, também, contou com: (i) doação de um ventilador para sala de aula e materiais elétricos por um aluno da Faculdade da Indústria; (ii) doação de dois computadores e um microondas para uso do Centro de

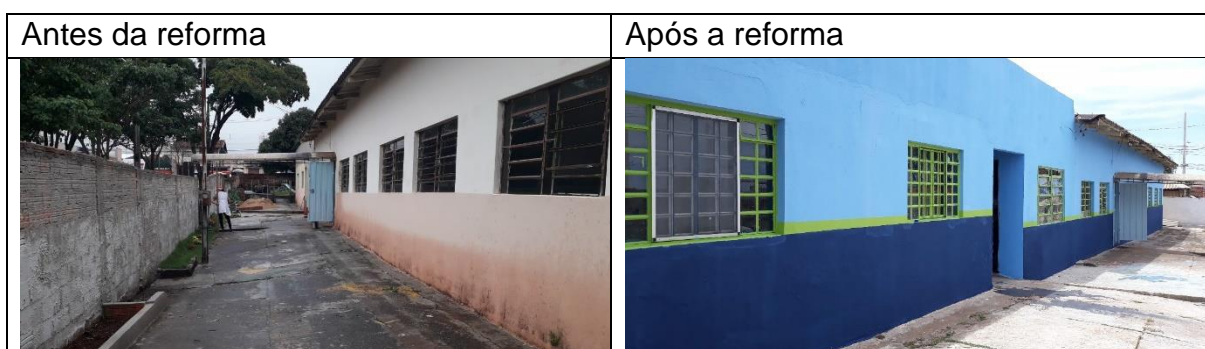
Educação Infantil, pela Green Eletron e Rotary Club Londrina Sul. Esses equipamentos foram descartados em edições anteriores do Rotary Coleta, passaram por manutenção e encontram-se agora em bom estado de uso. Nesse dia também foi realizado um check-list de possíveis melhorias da instituição.

No dia 15/10/2021 foi realizada a doação de 70 livros infantis e 85 brinquedos provenientes da Campanha de Arrecadação de Brinquedos, Livros Infantis e Gibis da Faculdade da Indústria Senai Londrina.

No dia 27/10/2021, a ação de restauração da fachada do CEI Alaíde Fausto Souza, restauração do parquinho, colocação de grama sintética para uso de bebês e manutenção da infraestrutura, contou com o apoio da comunidade escolar, além dos alunos da Faculdade da Indústria Senai Londrina. Funcionários do C.E.I. Alaide Fausto Souza também auxiliaram em outros dias para viabilização da reforma, por exemplo, na colocação de tijolos para aumento da altura do muro, pintura de fachada, pintura de frisos de janelas, pintura do pátio interno, etc.

Para o término da reforma também foram adquiridos por meio de doação: (i) um refletor LED de um aluno da Faculdade da Indústria; (ii) equipamentos para melhoria de um computador da instituição e materiais para pintura da comunidade externa. Nossa última visita no C.E.I. foi no dia 13/11/2021, finalizando pintura do parquinho, realizando desenhos no pátio da escola (centopeia do alfabeto, amarelinha, caracol e zig zague das formas) e doação de placa de identificação da escola. Os registros fotográficos da ação estão listados no Quadro 2.

Quadro 2 – Registros fotográficos – C.E.I. Alaide Fausto Souza







Fonte: Autor (2022)
b) C.E.I. Estrelinha

No dia 26/10/2021 foram realizados a doação de duas gangorras cavalinhos, um liquidificador industrial, 70 livros infantis e 96 brinquedos provenientes da Campanha de Arrecadação de Brinquedos, Livros Infantis e Gibis da Faculdade da Indústria Senai Londrina. No dia 05/11/2021 foi realizada a doação de uma tabela de basquete. Os registros fotográficos da ação estão listados no Quadro 3.

Quadro 3 – Registros fotográficos – C.E.I. Estrelinha





Fonte: Autor (2022).

9. CONSIDERAÇÕES FINAIS

A ação suprirá as principais necessidades apontadas anteriormente do Centro de Educação Infantil porque criou uma rede de voluntariado de alunos da Faculdade da Indústria Senai Londrina, colaboradores do Senai em Londrina e parceiros, trazendo mais estrutura aos profissionais e maior qualidade no atendimento oferecido às crianças.

REFERÊNCIAS

AGÊNCIA SISTEMA FIEP. **Programa de Voluntariado do Sistema Fiep recebe o Prêmio VoL.** Disponível em: < <https://agenciafiep.com.br/2021/12/08/programa-de-voluntariado-do-sistema-fiep-recebe-o-premio-vol/>>. Acesso em: 13 dez. 2021.

BRASIL. **Resolução nº 5, de 17 de dezembro de 2009.** Fixa as Diretrizes Curriculares Nacionais para a Educação Infantil.

PREFEITURA DO MUNICÍPIO DE LONDRINA. **Parcerias com Entidades Filantrópicas** – **2021.** Disponível em:

<<http://repositorio.londrina.pr.gov.br/index.php/menu-educacao/unidades-escolares/1094-cei-filantropicas/file>>. Acesso em: 13 set. 2021.

Seguimos inovando...



Faculdade da Indústria SENAI Londrina



14 de dezembro de 2021 · 🌐

A Faculdade da Indústria SENAI Londrina é 10!
Referência para os cursos de Engenharias, agora terá uma moderna e
super conectada unidade em Londrina,
<https://agenciafiep.com.br/.../senai-parana-tera-nova.../>



AGENCIAFIEP.COM.BR

Senai Paraná terá nova unidade em Londrina - Agência Fiep

Revista e-Tec