

CONTROLE PID NEURO-ADAPTATIVO EMBARCADO EM UM SISTEMA MAGLEV

Data de submissão: 16/04/2024

Data de aceite: 02/05/2024

Eduardo Marek

Instituto Federal de Santa Catarina –
Campus Chapecó
Chapecó – Santa Catarina

Alexandre Dalla’Rosa

Instituto Federal de Santa Catarina –
Campus Chapecó
Chapecó – Santa Catarina
<https://lattes.cnpq.br/0596224519125364>

RESUMO: Este artigo propõe um método de sintonização dos coeficientes de um controlador PID através de uma rede neural em um sistema microcontrolado embarcado. Esta arquitetura de controlador neuro-adaptativo se destaca em aplicações não lineares e com constante mudança de dinâmica da planta, neste caso, a bancada MAGLEV. A Bancada de suspensão magnética possui diversas aplicações na indústria, tais como: trens de alta velocidade, rolamentos magnéticos, dissipadores de vibrações, suspensões magnéticas etc. O desenvolvimento da pesquisa se dará da seguinte maneira: primeiramente será realizado um estudo acerca da planta e da arquitetura da rede neural a ser utilizada e informações sobre

o hardware de processamento, tendo estes dados adquiridos dá-se início do design e construção dos algoritmos para o projeto, somando todas as ferramentas desenvolvidas, começará o processo de implementação dos códigos na bancada de suspensão magnética e a comparação com um controlador PID clássico. Validando o desempenho deste método de sintonização, todos recursos gerados durante as pesquisas serão disponibilizados no GitHub para a comunidade.

PALAVRAS-CHAVE: Microcontroladores, Controle adaptativo, Redes Neurais Artificiais, MAGLEV.

NEURO-ADAPTATIVE PID CONTROL EMBEDDED IN A MAGLEV SYSTEM

ABSTRACT: The present paper proposes a PID controller tuning method based on an artificial neural network, implemented in an embedded microcontroller system. The neuro-adaptative controller architecture is known for its efficiency in non-linear models, with constant parameter changes, in this case, a MAGLEV device. Use largely spread in industrial fields such as: high-speed magnetic levitation trains, magnetic bearings, vibration dampers etc. The

research will be presented in the following sequence: study of the non-linear model, definition of the neural network architecture, evaluation of the hardware specifications, and with that data, the algorithm of the neuro-adaptative controller will be developed, alongside from the firmware. Completed those steps, the test and implementation stage of the proposed control method will begin, reaching its validation through comparison with a classical PID controller. All resources built from this project will be available to all the community in GitHub.

KEYWORDS: Microcontrollers, Adaptative Control, Artificial Neural Network, MAGLEV.

INTRODUÇÃO

A necessidade da contínua melhoria de qualidade no setor industrial fomenta a constante busca e pesquisa de técnicas de otimização dos seus processos e setores. Métodos de controle surgiram na década de 20 (PARK, 2017) e se tornaram importantes ferramentas na indústria no controle de processos. Com a sua implementação busca-se a redução das não conformidades dos procedimentos e ou produtos ali fabricados, fatores, influentes nos quesitos econômicos e comerciais da empresa (BRISK, 2015).

Os controladores são amplamente utilizados em processos de automação da indústria no controle de sistemas (OGATA, 2010). Sendo o PID (proporcional integral derivativo), o mais utilizado na indústria, facilmente aplicado a plantas lentas e com poucos requisitos de performance (RIVAS-ECHEVERRÍA; RÍOS-BOLÍVAR; CASALES-ECHEVERRÍA, 2001). O controlador PID apresenta suas limitações quando a planta possui características como: não linearidades, alta sensibilidade, alta complexidade, perturbações externas significativas e alteração da sua dinâmica e parâmetros conforme o tempo (BI; YING; YONGQUAN, 2003).

Na medida que técnicas de controle e algoritmos novos foram surgindo, o nível de complexidade dos processos e plantas controlados pôde avançar. Entretanto as características: não lineares, modelos de difícil estimação, constante mudança de parâmetros e alta sensibilidade, alinhadas aos requisitos de: precisão, velocidade e robustez, exigem *hardwares* com capacidades de entregar mais operações de cálculo, processadas ainda mais rapidamente (ATHANS, 1966). A possibilidade de implementação destes controladores em dispositivos embarcados é altamente relevante, visto que o determinismo das informações em alguns tipos de processos é indispensável. Somando os fatores mencionados anteriormente a capacidade execução do controlador em um equipamento de baixo custo através de otimizações e criação de novos métodos de abordagem são fundamentais, uma vez que, do ponto de vista comercial é um campo atrativo (ATHANS, 1966).

Neste artigo será abordado um método de controle neuro-adaptativo baseado em um controlador PID, onde, seus ganhos serão atualizados por uma rede neural adaptativa embarcada. Em (HERNÁNDEZ-ALVARADO *et al*, 2016) tem-se uma abordagem similar, entretanto os procedimentos de cálculo dos parâmetros atualizados serão feitos de maneira totalmente embarcada em um microcontrolador de baixo custo.

CONTROLE ADAPTATIVO

Por definição, os controladores adaptativos, possuem parâmetros que podem ser ajustados na tentativa de melhor adaptarem-se as não linearidades e constante mudança das plantas, assim como um mecanismo de fazê-lo (FIDAN; IOANNOU, 2006). A seleção do tipo de controlador e mecanismo é um processo determinante para a qualidade final do controle empregado.

Dentre as primeiras técnicas de controle adaptativo desenvolvidas temos os ganhos escalonados, derivado dos pilotos automáticos e controladores de voo do final da segunda guerra mundial (LEMOS; MARTÍN-SÁNCHEZ; RODELLAR, 2012). Este método consiste em identificar a não linearidade da planta, linearizá-la, gerar uma curva e escolher pontos de interesse nesta curva para sintonizar diferentes controladores, respectivamente (RUGH; SHAMMA, 2000). O número de pontos e a proximidade entre eles é um fator importante a ser observado, pois, quando o controlador estiver atuando em regiões distantes do ponto em que seus coeficientes foram determinados, a ação da não linearidade será alta (LEITH; LEITHEAD, 2010). Exemplificado na Figura 1.

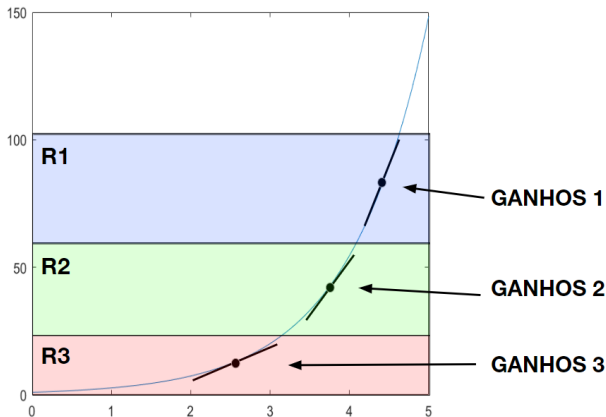


Figura 1: Curva de linearização.

Fonte: autoria própria.

Na abordagem por ganhos escalonados assume-se que a não linearidade é conhecida e pode ser modelada, o que não é verdade para todos os tipos de plantas (ELKHATEM; ENGIN, 2023). Para sistemas cujo as incertezas são desconhecidas ou de difícil modelagem, novas técnicas de controle adaptativo foram criadas.

Os avanços tecnológicos na área da computação possibilitaram a exploração novos algoritmos, que cada vez mais complexos, se tornaram capazes de contornar as incertezas desconhecidas. O uso de redes neurais adaptativas popularizou-se ao final da década de 80, sua capacidade de identificar padrões não lineares e adaptabilidade atraíram interesse da comunidade em aplicações de controle adaptativo, surgindo assim o termo: controle neuro-adaptativo (FIDAN; IOANNOU, 2006).

Controle neuro-adaptativo

As redes neurais artificiais, Figura 2 podem ser conceituadas como uma função matemática de tamanho e complexidade ajustáveis, que possuem a capacidade de se adaptar às situações problemas. Sua construção e princípio de funcionamento foram inspiradas no cérebro humano, dado essa similaridade, também recebem o nome de perceptron multicamadas (WERBOS, 1988), Figura 3.

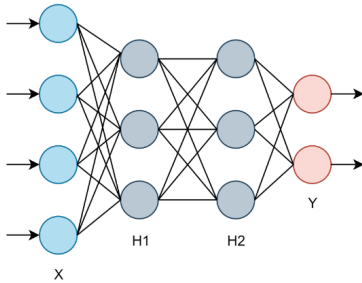


Figura 2: Estrutura da rede neural artificial.

Fonte: autoria própria.

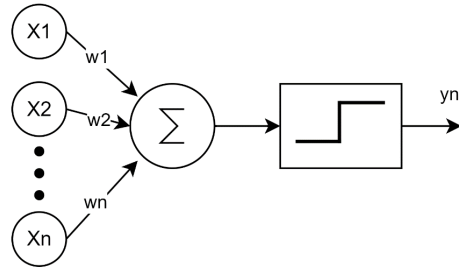


Figura 3: Perceptron de Rosenblatt

Fonte: autoria própria.

Os controladores neuro-adaptativos possuem duas principais arquiteturas: a rede neural substitui o controlador e suas saídas são as ações de controle, Figura 4, observado em (KANG; et al, 2014) (BI; YING; YONGQUAN, 2003), ou a rede neural realiza a sintonização de um controlador, Figura 6, seja qual for sua estrutura (RIVAS-ECHEVERRÍA; RÍOS-BOLÍVAR; CASALES-ECHEVERRÍA, 2001) (HERNÁNDES-ALVARADO *et al*, 2016).

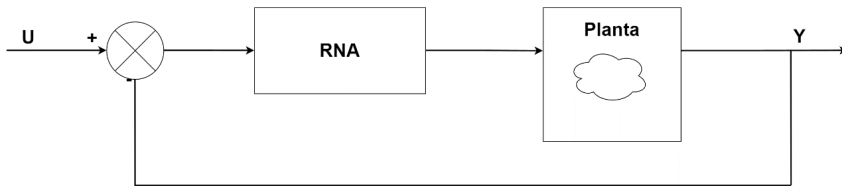


Figura 4: Controlador PID substituído por RNA.

Fonte: autoria própria.

A principal distinção entre os modelos é a certificação de um critério de estabilidade do controlador. No primeiro caso, Figura 4, o autor (KANG; et al, 2014) avalia a estabilidade utilizando a função de Lyapunov em conjunto com as equações de cálculo da variação dos pesos da rede neural. Contudo, no segundo caso, Figura 6, a estabilidade do sistema é garantida na sintonização dos controladores PID utilizados no aprendizado da rede neural (HERNÁNDES-ALVARADO *et al*, 2016). O modelo no qual o controlador é sintonizado por uma rede neural, e não substituído, há uma flexibilidade na escolha da arquitetura do controlador a ser sintonizado e em qual momento os seus ganhos serão atualizados.

CONTROLADOR NEURO-ADAPTATIVO PROPOSTO

O controlador neuro-adaptativo proposto neste artigo, consiste em um sistema totalmente embarcado para identificação em tempo real dos parâmetros de um controlador PID com filtro derivativo de primeira ordem, através de uma rede neural artificial. Com uma lei de controle definida como:

$$U_{(s)} = \left(K_p + K_i \frac{1}{s} + K_d \frac{s}{T_f s + 1} \right) E_{(s)} \quad (1)$$

A rede neural, Figura 5, baseada no modelo de (HERNÁNDEZ-ALVARADO *et al*, 2016), é construída a partir do arquétipo do controlador acima, sendo: k_p , k_d , k_i e T_s correspondente as suas saídas. Suas entradas são: os erros (referência - saída da planta) $E_{[k-2, k-1, k]}$ as ações de controle $U_{[k-2, k-1, k]}$ e saídas da planta (valor lido pelo sensor) $Y_{[k-1, k]}$ nos tempos indicados entre colchetes.

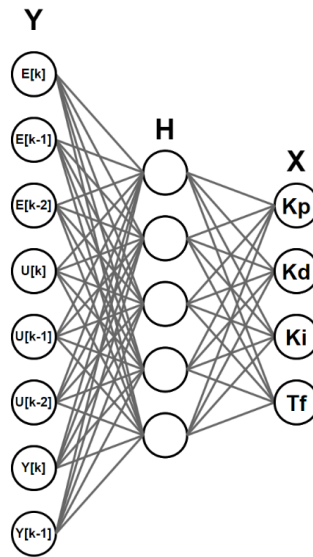


Figura 5: Rede neural implementada.

Fonte: autoria própria.

Além de entradas e saídas, a rede neural requer a definição de mais alguns parâmetros globais, são eles: número de camadas ocultas e respectivos tamanhos, coeficiente de aprendizado e funções de ativação de cada camada.

Assim como elaborado (RASHID, 2016), a rotina *feedforward* é desenvolvida através de operações algébricas entre as matrizes de pesos. Seguindo o modelo de perceptron de Rosenblatt (ROSENBLATT, 1958), Figura 3, agrupa-se os pesos a fim de facilitar a propagação dos sinais pela rede. Observando a Figura 3 constate-se que a saída de cada nó é dada por:

$$y = \sigma \left(b + \sum_{k=1}^l x_k w_k \right) \quad (2)$$

No qual: x_k são as l entradas, multiplicadas por suas respectivas intensidades de sinais w_k , b representa um bias e por fim é aplicado uma função de ativação, neste caso a sigmoide σ . Expandindo para toda a rede mediante álgebra de matrizes, temos que o vetor de saídas Y_n de cada camada i podem ser descritas como:

$$Y_n = \sigma_n(W_n X_n + B_n) \quad (3)$$

A matriz w_n possui o seguinte formato:

$$W_{ixj}^n = \begin{pmatrix} w_{00} & \cdots & w_{i0} \\ \vdots & \ddots & \vdots \\ w_{0j} & \cdots & w_{ij} \end{pmatrix} \quad (4)$$

e a matriz B_n :

$$B_{jx1}^n = \begin{pmatrix} b_0 \\ \vdots \\ b_j \end{pmatrix} \quad (5)$$

Sendo i o índice dos nós da camada n e j o das entradas X_n .

Procedimento de treinamento da rede neural

O formato de treinamento utilizado nesta proposta de controlador neuro-adaptativo é do tipo supervisionada, ou seja, os ganhos da rede são ajustados conforme um conjunto de dados informados pelo usuário (entradas e saídas esperadas).

Antes de definir-se o algoritmo backpropagation, responsável pela correção do valor de cada peso da rede neural, dado um valor de erro, é necessário realizar o cálculo do erro e sua projeção em cada camada. Descrito abaixo, o erro da camada é resultado da: multiplicação escalar da matriz de pesos transposta da camada, pela operação de multiplicação de Hadamard, simbolizada por \odot , com a derivada da função de ativação:

$$E_n = W_n^T \cdot E_{n+1} \odot \partial\sigma(Y_n) \quad (6)$$

e para a última camada:

$$E_{n_{max}} = (Y_{esperado} - Y_{n_{max}}) \odot \partial\sigma(Y_{n_{max}}) \quad (7)$$

Considerando os pesos w_n , bias B_n e um coeficiente de aprendizado α , a variação dos ganhos é dado como:

$$W_n' = W_n + \alpha(E_n Y_n^T) \quad (8)$$

$$B_n' = B_n + \alpha E_n \quad (9)$$

Na finalidade de executar tais operações em sistemas embarcados, antes foi preciso a criação de uma biblioteca para lidar com as matrizes e os cálculos algébricos. Nela estão presentes as operações de: soma, subtração, multiplicação de Hadamard, multiplicação por escalar, transposta e um método para aplicação de uma função em cada valor da matriz, compondo as funções de ativação. O formato de alocação de ponteiros e estrutura foi selecionado como forma de construção da biblioteca, a fim de se garantir um computo rápido dos valores.

Dispondo da biblioteca de matrizes, foram produzidas as rotinas de: feedforward, cálculo de erros, backpropagation, importação e exportação dos ganhos.

Partindo da premissa que, a rede neural será implementada como uma caixa preta para a identificação da planta e sintonização de um controlador, é desejável treiná-la para suportar a maior quantidade possível de modelos. Contudo, critérios de estabilidade devem ser mantidos, diante disso, limitar o espectro de funções fornecidas a rede deve ser considerado, assim como a qualidade dos controladores sintonizados para cada função, respectivamente.

Sintetizando as asserções acima, foi desenvolvido um script na ferramenta computacional MATLAB para produzir o conjunto de dados de treinamento da rede neural. Partindo da equação de transferência da planta:

1. Gera-se uma função de transferência variando todos os seus coeficientes aleatoriamente em $\pm V_f\%$ dos valores da planta fornecida. V_f é definido pelo usuário e dita o quão ampla será a região de atuação do controlador;
2. Sintoniza-se automaticamente um controlador PIDf;
3. Executa simulação do controlador gerado e equação no SIMULINK;
4. Verifica-se estabilidade do controlador;
5. Armazena os valores de ação de controle, erro e saída, seguindo formato de leitura da rede neural;
6. Repete o ciclo acima até que um número satisfatório de dados de plantas distintas possa ser coletado;
7. No intuito de manter os cálculos executados pela rede com grandeza restrita, os dados do conjunto de treinamento são normalizados para valores entre zero e um;
8. Por fim, os dados de: entradas, saídas esperadas e delimitadores das normalizações são armazenados em arquivos de texto.

Considerando que o objetivo, para cada saída da rede é prever um valor, a métrica utilizada em seu treinamento deve ser relacionada ao erro. Logo, optou-se pela utilização do erro quadrático no backpropagation. Os critérios de estabilidade são garantidos através da verificação dos valores máximos e mínimos da simulação, além de um desvio padrão percentual dos valores de erro (referência - saída) da simulação.

Configuração do controlador

O controlador PID implementado é discretizado utilizando o método de *Tustin*, onde:

$$s \simeq \frac{2z-1}{T_a z+1} \quad (10)$$

aplicando (10) em (1), tendo que $N = \frac{1}{T_f}$, temos:

$$U_{[k]} = \frac{E_{[k]}a + E_{[k-1]}b + E_{[k-2]}c + 8U_{[k-1]} - U_{[k-2]}d}{e} \quad (11)$$

onde:

$$a = 4K_p + 4K_d N + 2K_i T_a + 2K_p N T_a + K_i N T_a^2$$

$$b = 2K_i N T_a^2 - 8K_d N - 8K_p$$

$$c = 4K_p + 4K_d N - 2K_i T_a - 2K_p N T_a + K_i N T_a^2$$

$$d = 4 - 2N T_a$$

$$e = 4 + 2N T_a$$

A equação (11) é traduzida e implementada na linguagem C / C ++. Visando futuras pesquisas e aplicações, uma biblioteca foi construída, na qual, a biblioteca de redes neurais fora incluída. Nesta biblioteca o controlador neuro-adaptativo é tratado como uma estrutura. Como principal atributo tem-se a rede neural.

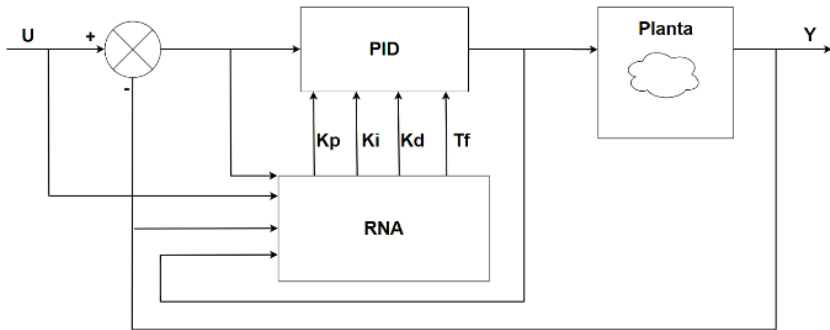


Figura 6: Controlador neuro-adaptativo.

Fonte: autoria própria.

Na Figura 6, configura-se a estrutura realimentada do: controlador, rede neural e planta.

IMPLEMENTAÇÃO DO CONTROLADOR DESENVOLVIDO

O controlador neuro-adaptativo desenvolvido foi aplicado em uma bancada de suspensão magnética, cujo possibilita a realização de testes voltados a exploração das capacidades de adaptação na presença de não linearidades e constante mudança dos parâmetros da planta.

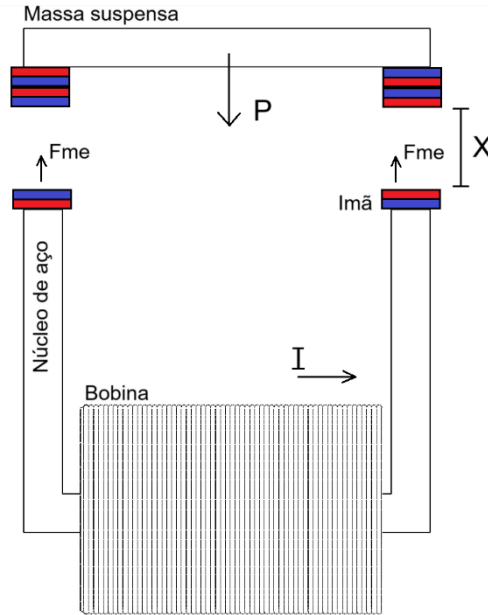


Figura 7: Suspensão magnética.

Fonte: autoria própria.

Modelo utilizado

Caracterizado pela Figura 7, o diagrama de funcionamento pode ser dividido em dois principais componentes, um sistema massa-mola cujo recebe como entrada uma força magnética, e o sistema utilizado para induzir a força magnética, composto por uma bobina de fios de cobre. De acordo com a corrente fornecida a bobina, o módulo do campo eletromagnético gerado aumenta ou diminui, interagindo com os ímãs posicionados nas extremidades dos entreferros do circuito magnético, provocando uma alteração na posição do corpo suspenso.

As equações características são dadas por (12) e (13):

$$F_{me} = C \frac{i^2}{x^2} \quad (12)$$

$$F = m\ddot{x} + B\dot{x} + kx \quad (13)$$

A equação (12) descreve a força magnética gerada em um sistema MAGLEV (WONG, 1986), onde C é uma constante característica da bobina, i a corrente e x a posição. Em (13) uma força resultante é expressa em função da posição, de forma que, os elementos k e B são as constantes elásticas e de atrito. Valores utilizados exibidos na Tabela 1.

Parâmetro	Valor	Unidade
C	$2,77 \text{ e}^{-5}$	$\frac{Nm^2}{i^2}$
\bar{i}	5	A
X	0,005	m
m	0,1427	kg
B	0,6943	$\frac{N \cdot s}{m}$
k	150,1204	Nm

Tabela 1: Coeficientes da planta.

Fonte: autoria própria.

O parâmetro “C”, normalmente obtido de forma experimental, foi encontrado da seguinte forma, equações retiradas de (SADIKU, 2012), R_m , B , ϕ , S , N , p , x , μ_r , μ_0 são respectivamente: relutância magnética, densidade de fluxo magnético, fluxo magnético, área da seção transversal, número de espiras, posição do corpo a ser suspenso, espaço do entreferro, permeabilidade relativa do material e permeabilidade constante do ar, conforme:

$$F_{me} = \frac{SB^2}{\mu_0} [N] \quad (14)$$

o campo magnético B é definido por:

$$B = \frac{\phi}{S} [T] \quad (15)$$

e o fluxo magnético ϕ :

$$\phi = \frac{Ni}{R_m} [Wb] \quad (16)$$

Assumindo que o sistema possui dois entreferros, aplicando (15) e (16) em (14), chega-se em (17):

$$F_{me} = \frac{1}{\mu_0 S} \left(\frac{Ni}{R_m} \right)^2 \quad (17)$$

Considerando o caminho médio p e a região de atuação x a relutância magnética do circuito é (18):

$$R_m = \frac{p}{\mu_0 \mu_r S} + \frac{2x}{\mu_0 S} [Wb] \quad (18)$$

substituindo (18) em (17), obtém-se (19):

$$F_{me} = C \frac{i^2}{x^2} = \frac{N^2 \mu_0 S}{4} \left(\frac{i}{\frac{\rho}{2\mu_r} + x} \right)^2 \quad (19)$$

A parcela expressa abaixo por (20) é desprezível para um núcleo de aço 1020.

$$\frac{\rho}{2\mu_r} = 2e^{-7} \approx 0 \quad (20)$$

Manipulando as equações (19) e (20), definindo C como (21):

$$C = \frac{N^2 \mu_0 S}{4} \left[\frac{Nm^2}{A^2} \right] \quad (21)$$

Utilizando os valores da Tabela 2 é possível calcular C , apresentado na Tabela 1.

Parâmetro	Valor	Unidade
N	980	<i>voltas</i>
μ_r	3000	$\frac{H}{m}$
μ_0	$4\pi e^{-7}$	$\frac{H}{m}$
S	0,0000950625	m^2
ρ	0,0012	m

Tabela 2: Coeficientes do circuito magnético.

Fonte: autoria própria.

A parcela do comportamento de massa-mola é proveniente da interação entre os ímãs de polos opostos dispostos na região superior (levitada) e inferior (bobina e núcleo). Os coeficientes elásticos e foram obtidos de forma experimental. Encontrados a partir da análise do ensaio ao degrau, Figura 8.

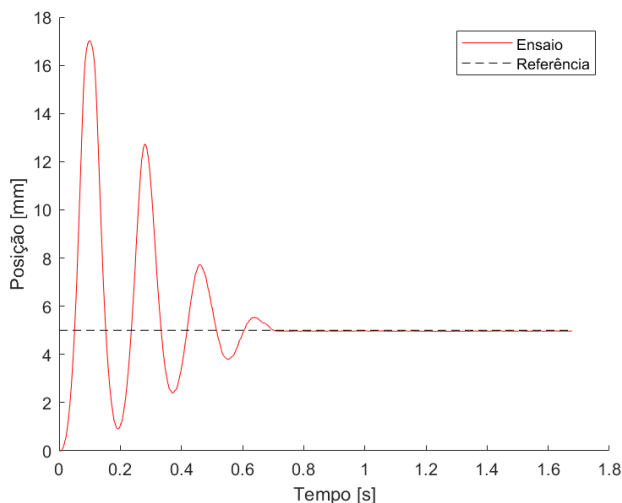


Figura 8: Ensaio ao degrau.

Fonte: autoria própria.

Para uma função de transferência de segunda ordem padrão com acréscimo de um grau de liberdade :

$$G_{(s)} = \frac{\omega_{\eta}^2}{s^2 + \alpha\omega_{\eta}\xi s + \omega_{\eta}^2} \quad (22)$$

utilizando as equações de (OGATA, 2010),

$$\xi = \frac{|\ln M_p|}{\sqrt{\pi^2 \ln(M_p^2)}} \quad (23)$$

$$\omega_{\eta} = \frac{\pi}{T_p \sqrt{1-\xi^2}} \quad (24)$$

Os valores de sobressinal M_p e tempo de pico T_p são 3,4s e 0,1s, respectivamente, de acordo com a Figura 8. A variável α é encontrada de maneira iterativa, orientando-se com objetivo de igualar a frequência das senoides do modelo com o ensaio.

Com os valores das equações (23), (24) derivados da análise do ensaio tem-se que:

$$G1_{(s)} = \frac{1052}{s^2 + 4,8665s + 1052} \quad (25)$$

Linearizando (12) nos pontos \bar{x} e \bar{i} através de Série de Taylor Bidimensional:

$$F_{(x,i)} \approx F(\bar{x}, \bar{i}) + \left[\frac{\partial F}{\partial x} \Big|_{x=\bar{x}, i=\bar{i}} (x - \bar{x}) \right] + \left[\frac{\partial F}{\partial i} \Big|_{x=\bar{x}, i=\bar{i}} (i - \bar{i}) \right] \quad (26)$$

$$F(x, i) \approx \frac{2C\bar{i}^{-2}}{\bar{x}^2} i - \frac{2C\bar{i}^{-2}}{\bar{x}^3} x + C \frac{\bar{i}^{-2}}{\bar{x}^2} \quad (27)$$

Separando a equação acima e transformando no domínio da frequência:

$$G2 = \bar{x} \quad (28)$$

$$W = C \frac{\bar{i}^{-2}}{\bar{x}^2} \quad (29)$$

prossequindo com os valores da Tabela 1:

$$G2 = 0,005 \quad (30)$$

$$W = 0,000184 \quad (31)$$

De modo que, agrupando funções de transferência $G1$, $G2$ e o ganho estático W , em malha aberta, o diagrama de blocos é reproduzido na Figura 9.

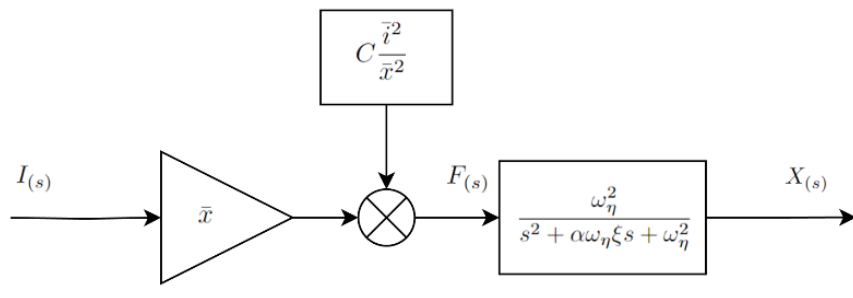


Figura 9: Planta em malha aberta.

Fonte: autoria própria.

Igualando a resposta ao degrau do modelo com a curva do ensaio, plotado na Figura 10:

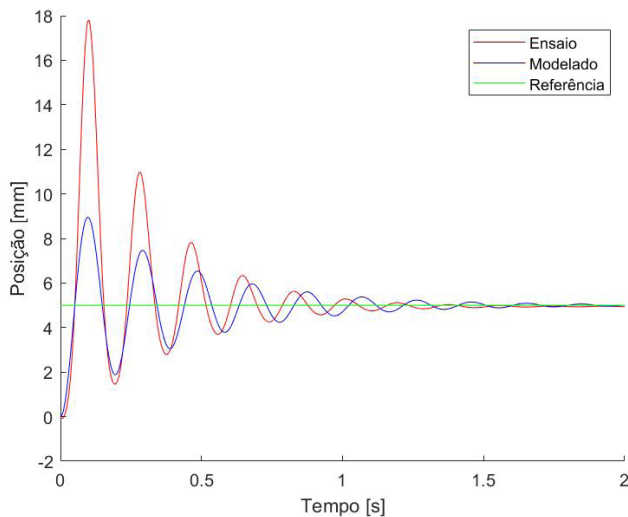


Figura 10: Ensaio vs modelo.

Fonte: autoria própria.

Apesar do modelo calculado ser próximo ao real, a ferramenta IDENT do MATLAB foi usada para obter-se uma função de transferência ainda mais fidedigna. Com ajuste de precisão, exposta na Figura 11. O ajuste da curva obtido resultou em uma função de ordem elevada, devido a tentativa de compensar a dificuldade de incorporar o comportamento não-linear. Logo, assume-se que o modelo obtido não é capaz de compreender todas as não-linearidades da planta.

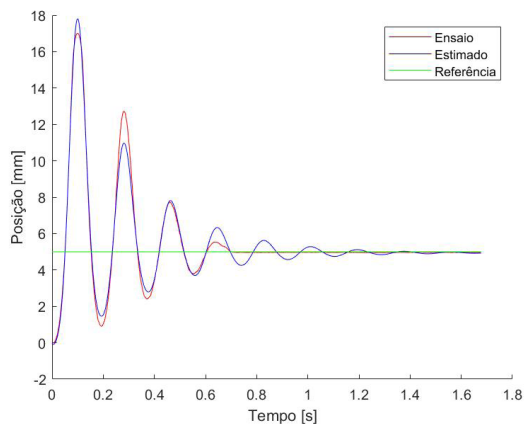


Figura 11: Ensaio vs estimativa IDENT.

Fonte: autoria própria.

Expostos na Tabela 3, os coeficientes gerados pelo MATLAB são posteriormente utilizados na equação (32).

b_n	Valor	a_n	Valor
b_0	-14.85	a_0	27.31
b_1	7718	a_1	7465
b_2	$4.176 e^5$	a_2	$1.412 e^5$
b_3	$8.968 e^7$	a_3	$1.423 e^7$
b_4	$5.865 e^8$	a_4	$1.591 e^8$
b_5	$1.289 e^{11}$	a_5	$7.669 e^9$
b_6	$2.836 e^{11}$	a_6	$2.902 e^{10}$

Tabela 3: Parâmetros função da transferência.

Fonte: autoria própria.

Função de transferência base utilizada para o treinamento:

$$G_{(s)} = \frac{b_0 s^6 + b_1 s^5 + \dots + b_5 s + b_6}{s^7 + a_1 s^6 + \dots + a_5 s + a_6} \quad (32)$$

Treinamento da rede

Dispondo de um modelo matemático próximo ao real, a rotina de geração de dados foi executada no MATLAB, resultando em 1,4 Gb de dados preparados para o treinamento da rede neural. A função de erro quadrático foi utilizada como métrica na avaliação do treinamento, pois o objetivo da rede é prever valores do tipo pontos flutuantes, assegurando que os coeficientes do controlador neuro-adaptativo estejam em sintonia com o estado atual da planta. Um total de 10 épocas com coeficiente de aprendizado de módulo 0,1 foram usados no treinamento, exibido na Figura 12. Após testes, foi constatado que usando apenas uma camada oculta com cinco nós, a rede neural é capaz de realizar o aprendizado para esta aplicação. Topologia final da rede exibida na Figura 5.

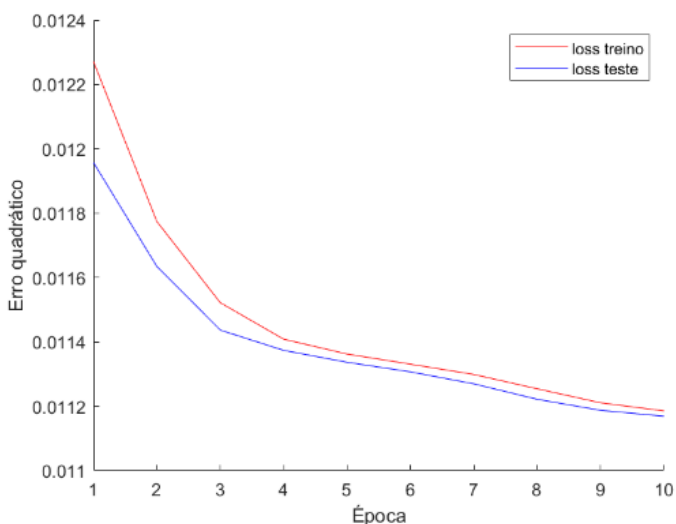


Figura 12: Curva de treino da RNA implementada.

Fonte: autoria própria.

Implementação no sistema embarcado

A bancada de suspensão magnética funciona a partir do microcontrolador de tecnologia AVR *Esp32 Dev Module*, fabricado pela ESPRESSIF. Selecionado por ter qualidades como: ser um dispositivo de baixo custo, alta disponibilidade no mercado, pode ser programado através da plataforma Arduino. A atuação da bobina do circuito magnético é feita através do chaveamento de um mosfet com um sinal PWM provindo do microcontrolador. A posição da peça levitada é obtida realizando a leitura do valor ADC de um sensor linear de efeito Hall. Como forma de maximizar o potencial de processamento do microcontrolador, não foram utilizadas interrupções por alarme, pois desta forma, o programa executa um ciclo após o outro, sem pausas. *Firmware* representado pelo fluxograma da Figura 13.

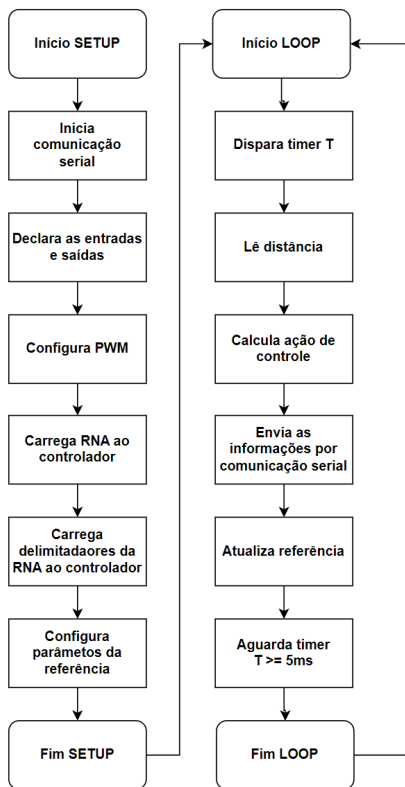


Figura 13: Fluxograma do *firmware*.

Fonte: autoria própria.

Implementado a rede treinada com *clock* de 240MHz, foi constatado que o tempo para rede realizar a rotina do *feedforward* leva em média 550 μ s. O cálculo da ação de controle em conjunto com a atualização de parâmetro é realizado em 570 μ s. Apesar do sistema permitir uma frequência de 1,754kHz, a utilizada foi 0,2kHz, dado que a correlação entre tempo de pico do modelo T_p , 100ms, e o tempo de amostragem do controlador, 5 ms, deve ser de 5 a 20 vezes mais rápida, segundo o autor (OGATA, 2010).

RESULTADOS OBTIDOS

A eficiência do controlador neuro-adaptativo proposto é comparada em relação a um controlador PID clássico, sintonizado pelo método Ziegler–Nichols. A Figura 14 expõe a comparação da resposta obtida dos controladores a um degrau de diferentes valores de módulo. É possível notar uma clara distinção entre o comportamento dos controladores, nos tempos 2s, 10s e 20 s o PID clássico apresenta um valor de sobressinal elevado. Em adição de ter um tempo de estabilização mais lento que o do controlador neuro-adaptativo, observado nos tempos 3s, 16s e 22s.

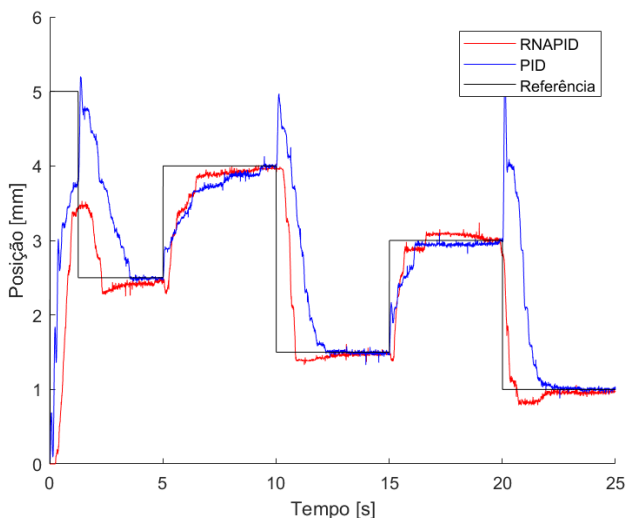


Figura 14: Comparação entre controladores.
Fonte: autoria própria.

Como forma de avaliar a adaptabilidade e robustez do controlador, foram realizados dois ensaios utilizando os mesmos controladores da comparação da Figura 14. No primeiro é acrescentando uma inclinação de 20° a bancada, já no segundo, uma massa de $22g$ é adicionada como forma de perturbação fixa.

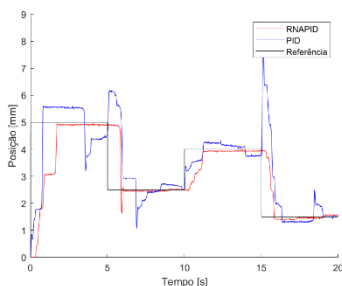


Figura 15: Ensaio com elevação.
Fonte: autoria própria.

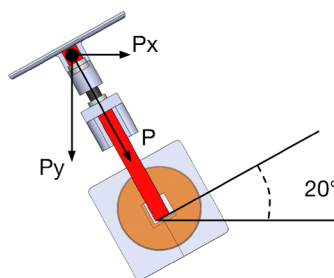


Figura 16: Representação 3D da elevação.
Fonte: autoria própria.

Na Figura 15 a dinâmica do circuito magnético e sistema massa-mola são alterados, uma vez que, a força peso não é mais normal a superfície de sua base. A temperatura do circuito magnético também deve ser levada em conta, uma vez que devido à alta corrente, o efeito Joule se intensifica, reduzindo a força máxima produzida pelo sistema. Na Figura 15 nota-se que o controlador PID padrão apresenta oscilações e sobressinal do início ao fim. Já o controlador neuro-adaptativo proposto mantém uma atuação similar à do primeiro experimento.

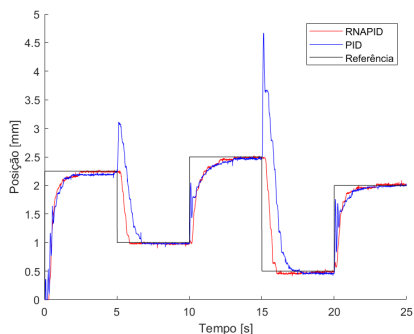


Figura 17: Ensaio com perturbação fixa.
Fonte: autoria própria.

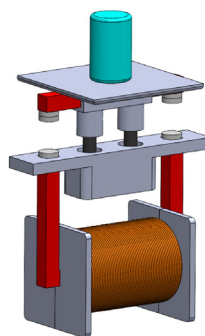


Figura 18: Representação 3D da perturbação fixa.
Fonte: autoria própria.

No ensaio da Figura 17 o controlador PID clássico apresentou considerável sobressinal e pequenas oscilações nos tempos de 0,1s, 10,1s e 20,1s. Características, que o controlador neuro-adaptativo implementado não exibiu, reforçando a sua capacidade adaptativa proposta.

CONCLUSÕES

Neste projeto é proposto a construção e implementação de um controlador neuro-adaptativo totalmente embarcado em um sistema microcontrolado em uma bancada com parâmetros de difícil estimação. Ambos os modelos apresentados nas Figuras 10 e 11 não conseguem representar todas as não linearidades da planta.

O controlador PID com filtro derivativo de primeira ordem, sintonizado pela rede neural, foi capaz de manter uma robustez no sistema de malha fechada, diminuindo o sobressinal e reduzindo o valor de erro em para uma referência de degrau, em diferentes regiões de atuação. Somado aos resultados obtidos nos ensaios com inclinação de e acréscimo de um corpo maciço, tais características constataam a alta capacidade de adaptação do controlador neuro-adaptativo a mudanças nos parâmetros da planta, perturbações externas e não linearidades.

Ressalta-se a ótima performance obtida no computo das operações de sintonização dos parâmetros através da RNA e cálculo da ação de controle, com tempo de , realizadas de forma totalmente embarcada em um microcontrolador de baixo custo, e alta disponibilidade, tendo portabilidade a outros hardwares similares. O desenvolvimento e aplicação de uma biblioteca de redes neurais adaptativas voltada para microcontroladores foi o ponto chave para a obtenção dos resultados deste trabalho.

Pesquisas futuras se mostram necessárias na comparação do controlador proposto com outros métodos de controle adaptativo, assim como implementações em outras aplicações práticas.

REFERÊNCIAS

ATHANS, M. **The status of optimal control theory and applications for deterministic systems.** IEEE Transactions on Automatic Control, 1966. Disponível em: <https://ieeexplore.ieee.org/abstract/document/1098353>. Acesso em: 15 de novembro de 2023.

BAZANELA, A. S.; SEHNEM R. M. **Controle Ótimo Não Linear em Malha Fechada por Redes Neurais Profundas.** SBA, 2022. Disponível em: https://www.sba.org.br/open_journal_systems/index.php/cba/article/view/3434. Acesso em: 16 de novembro de 2023.

BI, Z.; YING, H.; YONGQUAN, Y. **A PID neural network controller.** Proceedings of the International Joint Conference on Neural Networks, 2003. Disponível em: <https://ieeexplore.ieee.org/document/1223703>. Acesso em: 15 de novembro de 2023.

BRISK, M. L. **Process control: potential benefits and wasted opportunities.** Australian Journal of Electrical and Electronics Engineering, 2015. Disponível em: <https://www.tandfonline.com/doi/abs/10.1080/1448837X.2005.11464113>. Acesso em: 15 de novembro de 2023.

ELKHATEM, A. S.; ENGIN, S. N. **Enhancing performance and stability of gain-scheduling control system using evolutionary algorithms: A case study on transport aircraft.** Expert Systems with Applications, 2023. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0957417422018772>. Acesso em: 15 de novembro de 2023.

FIDAN, B.; IOANNOU, P. **Adaptive Control Tutorial.** 11ª edição. Filadélfia: Society for Industrial and Applied Mathematics Philadelphia, 2006.

GE, S. S.; HANG, C. C.; ZHANG, T. **Nonlinear adaptive control using neural networks and its application to CSTR systems.** Journal of Process Control, 1998. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S0959152498000547>. Acesso em: 15 de novembro de 2023.

HERNÁNDEZ-ALVARADO, L. *et al.* **Neural Network-Based Self-Tuning PID Control for Underwater Vehicles.** Sensors, 2016. Disponível em: <https://www.mdpi.com/1424-8220/16/9/1429>. Acesso em: 15 de novembro de 2023.

KANG, J. *et al.* **An adaptive PID neural network for complex nonlinear system control.** Neurocomputing, 2014. Disponível em: <https://www.sciencedirect.com/science/article/abs/pii/S092523121301134X>. Acesso em: 15 de novembro de 2023.

LEITH, D. J.; LEITHEAD, W. E. **Survey of gain-scheduling analysis and design.** International Journal of Control, 2010. Disponível em: <https://www.tandfonline.com/doi/abs/10.1080/002071700411304>. Acesso em: 15 de novembro de 2023.

LEMONS, J. M.; MARTÍN-SÁNCHEZ, J. M.; RODELLAR, J. **Survey of industrial optimized adaptive control.** International Journal of Adaptive Control, 2012. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/acs.2313>. Acesso em: 15 de novembro de 2023.

LU J.; ZHAO, N. **Neural network algorithm and its application in temperature control of distillation tower.** ARXIV, 2021. Disponível em: <https://arxiv.org/abs/2101.00582>. Acesso em: 16 de novembro de 2023.

OGATA, K. **Engenharia de controle moderno.** 5ª edição. Rio de Janeiro: Pearson Universidades, 2010.

PARK, M. *et al.* **Process control and economic cost design for total quality management.** Total Quality Management & Business Excellence, 2016. Disponível em: <https://www.tandfonline.com/doi/full/10.1080/14783363.2015.1133243>. Acesso em: 15 de novembro de 2023.

RASHID, T. **Make Your Own Neural Network.** Edição ilustrada. Carolina do Sul: Createspace Independent Publishing Platform, 2016.

RIVAS-ECHEVERRÍA, F.; RÍOS-BOLÍVAR, A.; CASALES-ECHEVERRÍA, J. **Neural Network-based Auto-Tuning for PID Controllers.** Neural Network World, 2001. Disponível em: <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=286101d6bd52c9cc7eeb02a6c15495f43e5bc187> Acesso em: 15 de novembro de 2023.

ROSENBLATT, F. **The perceptron: a probabilistic model for information storage and organization in the brain.** Psychological Review, 1958. Disponível em: <https://psycnet.apa.org/record/1959-09865-001>. Acesso em: 15 de novembro de 2023.

SADIKU, M. **Elementos de Eletromagnetismo.** 5ª edição. Bookman 2012.

RUGH, W. J.; SHAMMA, J. S. **Research on gain scheduling.** Automatica, 2000. Disponível em: <https://www.sciencedirect.com/science/article/pii/S000510980000583>. Acesso em: 15 de novembro de 2023.

WERBOS, P. J. **Backpropagation: past and future.** IEEE 1988 International Conference on Neural Networks, 1988. Disponível em: <https://ieeexplore.ieee.org/abstract/document/23866>. Acesso em: 15 de novembro de 2023.

WONG, T. H. **Design of a magnetic levitation control system??? An undergraduate project.** IEEE Transactions on Education, 1986. Disponível em: <https://ieeexplore.ieee.org/abstract/document/5570565>. Acesso em: 15 de novembro de 2023.