

# Controlador PID Adaptativo Via Redes Neurais-LMS Para Controle de Altitude de Quadricoptero

Fernando Antonio Moreira Serra Abreu  
Departamento de Engenharia Elétrica  
Universidade Federal do Maranhão -UFMA  
São Luís, Brasil  
e-mail: fernando.abreu.ms@hotmail.com

João Viana da Fonseca Neto  
Departamento de Engenharia Elétrica  
Universidade Federal do Maranhão -UFMA  
São Luís, Brasil  
e-mail: jviana@dee.ufma.br

**Resumo** — O desenvolvimento de uma metodologia para sintonia online de controladores PID adaptativo via Redes Neurais Artificiais baseado no algoritmo de treinamento LMS é abordada neste artigo. Sistemas físicos sujeitos a variações paramétricas e distúrbios externos requerem um controlador com auto ajuste em seus ganhos. O algoritmo proposto ajusta continuamente os parâmetros do controlador PID de forma adaptativa. A metodologia é aplicada a um controle de altitude de um veículo aéreo não tripulado (VANT) do tipo quadricoptero. Este veículo é uma aeronave do tipo multirotor e apresenta quatro motores que são responsáveis por sua movimentação e possui a capacidade de decolagem e pousos verticais. Resultados experimentais confirmam o bom desempenho do método, seguindo uma trajetória de referência mesmo na presença de variações paramétricas na planta.

**Palavras-chave**— PID; Redes Neurais Artificiais(RNA); Controle Adaptativo; Quadricoptero.

## I. INTRODUÇÃO

Os Veículos Aéreos Não Tripulados (VANT) do tipo quadricoptero, são aeronaves de pequeno porte que são controladas remotamente ou autônomas e receberam nos últimos anos bastante atenção das indústrias e pesquisadores no ramo aeroespacial, controle e robótica, podendo ser aplicados em monitoramento de tráfego, topografia e mapeamento, inspeção agrícola e transporte [1].

Controladores PID são amplamente utilizados na indústria e tem sido utilizado em todo o mundo para sistemas de controle industrial, por apresentar estrutura simples e desempenho robusto, onde inúmeros métodos de sintonia têm sido desenvolvidos para alcançar os melhores valores práticos dos ganhos.

Controladores PID requerem a sintonia de três parâmetros: ganho proporcional, constante de tempo integral e constante de tempo derivativo. Devido a variações das condições de operações em processos industriais, é necessário o ajuste dos parâmetros do PID para atender os requisitos de produção. O método de sintonia PID tradicional [2] apresenta algumas limitações, no entanto manteve-se muito popular pela sua simplicidade. O grande desafio para projetos de controle PID, são sistemas não lineares e sistemas que apresentam variações paramétricas.

Controle adaptativo é uma técnica de controle que apresenta a característica de mudar o seu comportamento de acordo com a dinâmica do sistema, ajustando os seus parâmetros continuamente a partir de uma lei de adaptação paramétrica [3]. O algoritmo adaptativo dos mínimos quadrados (LMS) um dos algoritmos mais importantes no processamento de sinal adaptável, sendo muito utilizado em filtragem adaptativa por sua facilidade de implementação, simplicidade, menor complexidade computacional e boa propriedade de convergência [4].

Aplicações de Redes Neurais Artificiais (RNA), Algoritmos Genéticos e Lógica Fuzzy, tem sido utilizadas para métodos de sintonia dos controladores PID de forma adaptativa [5], [6], [7]. Em [8], é abordado um controlador PID adaptativo via redes Neurais para controle atitude de quadricopteros.

Grande esforço foi dedicado por parte de pesquisadores para reduzir o tempo gasto para a sintonia dos parâmetros do controlador PID. Neste artigo é apresentado um controlador PID adaptativo via redes neuronais juntamente com a teoria de convergência do LMS, que atualiza continuamente os ganhos do controlador PID, apresentando suas características e resultados de simulação aplicados ao controle de altitude de um quadricoptero.

O artigo está organizado em seções, que estão dispostas da seguinte forma: na Seção II, apresenta-se a descrição do sistema, seu funcionamento, e a modelagem. Já na Seção III, é abordado os controladores propostos e suas características. Na Seção IV é realizado os experimentos computacionais. Finalmente na Seção V é apresentada a conclusão.

## II. DESCRIÇÃO DO SISTEMA

O quadricoptero é uma aeronave acionada por quatro motores com ângulos fixos, que representam quatro forças que são basicamente o empuxo gerado pelo conjunto motor e hélice, conforme apresentado na Fig. 1.

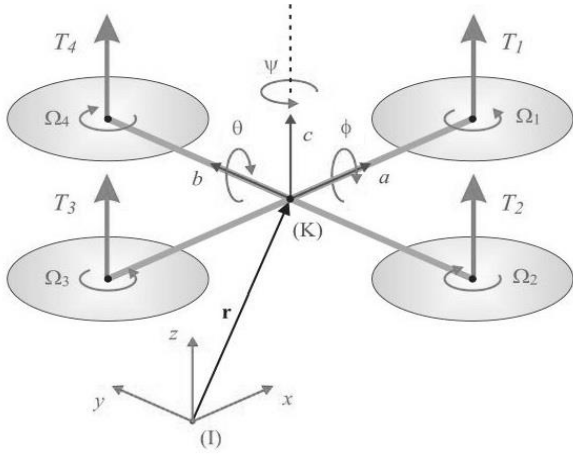


Fig. 1. Configuração do quadricóptero.

Ainda na Fig. 1,  $x$ ,  $y$  e  $z$  denota a posição do centro de massa do quadricóptero,  $\phi$  é o ângulo de rolagem,  $\theta$  é o ângulo de arfagem e  $\psi$  o ângulo de guinada e  $\Omega_1, \Omega_2, \Omega_3$  e  $\Omega_4$  representam a velocidade angular dos motores. As propriedades do quadricóptero são não lineares, multivariável (MIMO), altamente acopladas devido ao sistema de acionamento composto por quatro entradas e seis graus de liberdade [9].

Utilizando a notação de [10] e [11], o modelo dinâmico do quadricóptero é representado por

$$\begin{aligned}\ddot{x} &= (\cos \phi \sin \theta \cos \psi + \sin \phi \sin \psi) \frac{1}{m} U_1, \\ \ddot{y} &= (\cos \phi \sin \theta \sin \psi + \sin \phi \cos \psi) \frac{1}{m} U_1, \\ \ddot{z} &= -g + (\cos \phi \cos \theta) \frac{1}{m} U_1,\end{aligned}\quad (1)$$

em que  $x$ ,  $y$  e  $z$  representa os movimentos de translação, e seus ângulos de orientação,  $\phi$  (rolagem),  $\theta$  (arfagem) e  $\psi$  (guinada) são representados por

$$\begin{aligned}\ddot{\phi} &= \dot{\theta} \dot{\psi} \left( \frac{I_y - I_z}{I_x} \right) - \frac{J_r}{I_x} \dot{\theta} \Omega + \frac{l}{I_x} U_2, \\ \ddot{\theta} &= \dot{\phi} \dot{\psi} \left( \frac{I_z - I_x}{I_y} \right) - \frac{J_r}{I_y} \dot{\phi} \Omega + \frac{l}{I_y} U_3, \\ \ddot{\psi} &= \dot{\phi} \dot{\theta} \left( \frac{I_x - I_y}{I_z} \right) + \frac{1}{I_z} U_4.\end{aligned}\quad (2)$$

em que  $I_x, I_y$  e  $I_z$ , representam os momentos de inercia do quadricóptero em seus respectivos eixos,  $J_r$  o momento de inercia da hélice,  $l$  o comprimento do braço.

As entradas do sistema são representadas por

$$\begin{aligned}U_1 &= b(\Omega_4^2 + \Omega_2^2 + \Omega_3^2 + \Omega_1^2), \\ U_2 &= bl(\Omega_4^2 - \Omega_2^2), \\ U_3 &= bl(\Omega_3^2 - \Omega_1^2), \\ U_4 &= d(-\Omega_4^2 + \Omega_2^2 - \Omega_3^2 + \Omega_1^2),\end{aligned}\quad (3)$$

sendo  $U_1, U_2, U_3$  e  $U_4$ , as entradas referentes ao empuxo e  $b$  e  $d$  as constantes do motor.

O quadricóptero apresenta um sistema com diversas variáveis de entrada e saída. A fim de simplificar a dinâmica do sistema. O artigo propõe realizar somente controle de altitude do quadricóptero, transformando o sistema de equações de várias entradas e várias saídas MIMO em um sistema SISO e a sua dinâmica pode ser representada por

$$U_1 = b(\Omega_4^2 + \Omega_2^2 + \Omega_3^2 + \Omega_1^2), \quad (4)$$

e a equação que representa a aceleração no eixo Z representada por

$$\ddot{z} = -g + (\cos \phi \cos \theta) \frac{1}{m} U_1, \quad (5)$$

sendo  $U_1$  a tração produzida pelos quatro conjuntos motor-hélice no eixo z.

### III. PROJETO DE CONTROLE PID ADAPTATIVO

Controladores de processos industriais são responsáveis por manter o equipamento em um ponto de operação ideal, e estes possuem algoritmos de controle específicos. A seguir são apresentadas as técnicas de controle PID convencional e o método de atualização dos seus ganhos via Rede Neural.

#### A. Controlador PID Digital

O controlador PID no tempo contínuo depende dos ajustes desses três parâmetros. A sua função transferência é dada por

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s, \quad (6)$$

sendo  $K_p, K_i$  e  $K_d$  são dos ganhos proporcional, integral e derivativo, respectivamente. Para a realização da sintonia de controladores PID é utilizado o método de Ziegler-Nichols. [12].

Para a presente proposta, é utilizado um controlador PID digital com a seguinte estrutura com seus ganhos ajustados pela rede neural

$$U_{PID}(k) = K_p e(k) + K_I T_s \sum_{i=0}^k e(k) + K_D \frac{\Delta e(k)}{T_s}, \quad (7)$$

sendo o erro  $e(k) = y(k) - y_d(k)$ , em que  $y(k)$  é o sinal de referência,  $y_d(k)$  é a saída medida, e  $T_s$  o intervalo de amostragem.

### B. Controlador PID-NEURAL

A abordagem utilizada para o desenvolvimento do controlador PID adaptativo tem como base uma rede neuronal do tipo Perceptron Simples (PS) de uma camada representado pela Fig. 2, com o mecanismo de ajuste dos ganhos fundamentados nos métodos do LMS. O sinal de erro é processado pelas entradas da rede, o combinador linear representa a estrutura do ganho PID e a função de ativação é regida pelo comportamento da planta. O ajuste de ganho é visto como uma unidade neural no sentido de Rosenblat e é realizado pelo método LMS que representa a estrutura do controlador de alto desempenho.

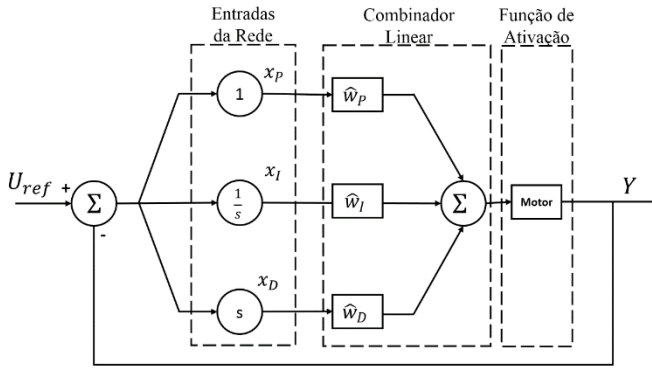


Fig. 2. Diagrama geral representando o controlador PID-RNA.

O mecanismo de ajuste do LMS atualiza os ganhos do controlador PID em tempo real apresentando ao sistema de controle propriedades de adaptabilidade. Estes modelos estão inseridos no contexto do diagrama de blocos do sistema apresentado na Fig. 3 que representa a estrutura de alto desempenho no contexto adaptativo.

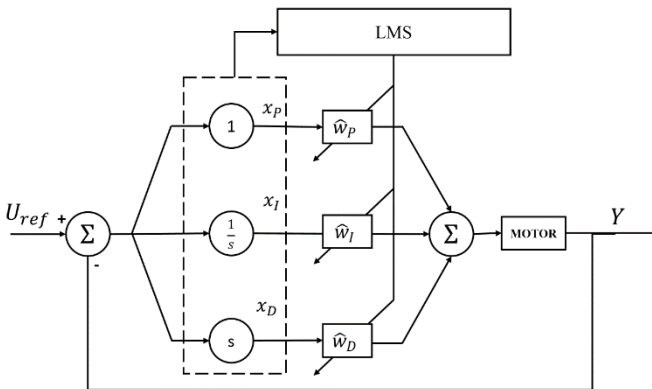


Fig. 3. Diagrama representando o ajuste dos ganhos.

De acordo com a Fig. 3,  $x_p$ ,  $x_i$  e  $x_d$ , representam as entradas do neurônio que são ponderadas pelos ganhos do controlador PID. A saída da rede é uma combinação linear da saída dos pesos, gerando o sinal de controle que é dado por

$$U_{RNA}(k) = \hat{w}_p(k)x_p(k) + \hat{w}_i(k)x_i(k) + \hat{w}_d(k)x_d(k), \quad (8)$$

em que  $\hat{w}_p, \hat{w}_i, \hat{w}_d$  e correspondem aos pesos da rede que representam os ganhos do controlador PID.

O mecanismo de ajuste adaptativo fundamenta-se no treinamento LMS (Least Mean Square), gradiente descendente ou mínimos quadrados que tem como objetivo minimizar o erro de realimentação. No algoritmo LMS, o vetor de pesos  $\hat{w}_{PID}(t)$  executa uma trajetória aleatória e por essa razão o algoritmo LMS é denominado um algoritmo de gradiente estocástico. O algoritmo LMS é fundamentado na utilização de valores instantâneos para a função de custo que é representado pela energia do erro. Utilizando a notação de [13], energia do erro é dada por

$$E(w) = \frac{1}{2} e^2(k), \quad (9)$$

sendo  $e(n)$  o sinal de erro que aciona o mecanismo de controle.

Diferenciando (9) em relação ao vetor de pesos ( $w$ ), obtém-se

$$\frac{\partial E(w)}{\partial w} = e(k) \frac{\partial e(k)}{\partial w}. \quad (10)$$

O algoritmo LMS opera com um neurônio linear de forma que podemos expressar o sinal de erro como

$$e(k) = d(k) - x^T(k)w(k), \quad (11)$$

com isso

$$\frac{\partial e(w)}{\partial w} = -x(k), \quad (12)$$

e

$$\frac{\partial E(w)}{\partial w(k)} = -x(k)e(k). \quad (13)$$

Utilizando (13) como um cálculo para o vetor gradiente, temos

$$\nabla E(w) = g(k) = -x(k)e(k), \quad (14)$$

$$w(k+1) = w(k) - \eta_{PID}g(k). \quad (15)$$

O ajuste dos ganhos para as condições operacionais de projeto se dá atualizando o vetor de pesos da rede neural a cada interação que é dada por

$$\hat{W}_{PID}(k+1) = \hat{W}_{PID}(k) + \eta_{PID} e(k) \hat{x}_{PID} \quad (16)$$

sendo  $\hat{W}_{PID}$  os ganhos estimados para a condição de operação,  $\eta_{PID}$  o parâmetro da taxa de aprendizagem e  $\hat{x}_{PID}(t)$  as entradas em atraso, atual e em avanço.

#### IV. EXPERIMENTO COMPUTACIONAL

Nesta Seção são apresentadas simulações do sistema de controle PID-Neural proposto e comparações com o método de controle PID tradicional. Nos dois sistemas é utilizado o mesmo modelo de planta e mesmo controlador PID discreto.

O quadricoptero utilizado para os experimentos computacionais é o Rolling Spider, apresentado na Fig. 4. É utilizada esta plataforma pois seus parâmetros podem ser facilmente acessados pelo Toolbox Aerospace Blockset componente do software MATLAB© versão 2018.

Os parâmetros utilizados para simulações deste artigo são evidenciados na tabela 1.

TABELA I. PARAMETROS DE SIMULAÇÃO

Variável	Unidade	Valor
Passo de Simulação (Ts)	s	0,005
Aceleração da Gravidade (g)	m/s <sup>2</sup>	9,8
Massa (m)	Kg	0,08
Coefficiente de força (b)	Ns <sup>2</sup>	4,72 x 10 <sup>-8</sup>
Saturação Máxima (Smax)	N	1,2272
Saturação Mínima (Smin)	N	0,0245
Arfagem (φ)	°	0
Rolagem (θ)	°	0



Fig. 4. Quadricoptero Rolling Spider.

Em sistemas reais como o empregado neste trabalho, apresentam não linearidades como a saturação de drivers. A função de saturação representa os limites impostos ao atuador, e representa a tração máxima e mínima pelo conjunto motor e hélice do quadricoptero e é limitada entre 0,0245 N e 1,2272 N. A Fig. 5 representa o diagrama de blocos do sistema com a função de saturação que representa o próprio sinal de controle, quando o mesmo está ente 0,0245 N e 1,2272 N.

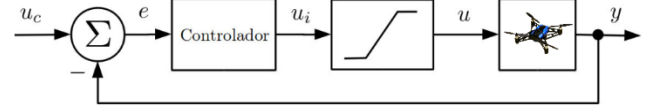


Fig. 5. Sistema de controle do quadricoptero em malha fechada com saturação.

A função de saturação é expressa por

$$U = \begin{cases} 1.2272, & \text{se } U_i > 1.2272 \\ U & \text{se } 0.0245 \leq U_i \leq 1.2272 \\ 0.0245, & \text{se } U_i < 0.0245. \end{cases} \quad (17)$$

Na Fig. 6 é apresentado o desempenho do controlador PD e o controlador PID Adaptativo-Neural. Os ganhos do controlador PD tomando como base o método de Ziegler Nichols foi Kp= 0,8 e Kd= 0,3 , o ganho integral foi considerado zero.

Para análise do comportamento da planta é utilizado uma trajetória de referência que representa a variação de altitude do quadricoptero, que é dada por

$$z = 1.6 \sin(t), \quad (18)$$

O treinamento da rede está fundamentado em (16). É necessário atribuir a taxa de aprendizagem dos ganhos do controlador, que são ajustados de forma empírica. De acordo com [13] a taxa de aprendizado deverá estar compreendida entre  $0 < \eta_{PID} < 1$  . A taxa de aprendizado  $\eta_{PID}$  é desmembrada em  $\eta_P, \eta_I, \eta_D$ , que são referentes às taxas de aprendizado dos ganhos proporcional, integral e derivativo, respectivamente. Para uma melhor convergência foram utilizados os seguintes valores:  $\eta_P = 0,9; \eta_I = 0,3; \eta_D = 0,3$ .

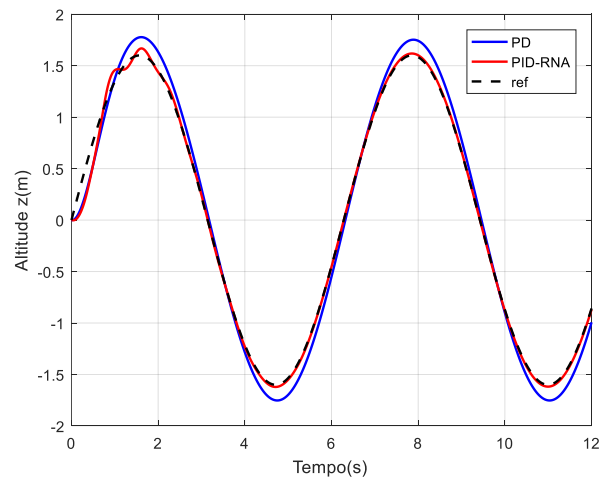


Fig. 6. Resposta do sistema com controle PD e PID-Neural.

O controlador PID-Neural apresenta uma melhor resposta seguindo a trajetória de referência (18), podendo observar que é capaz de acompanhar a trajetória estipulada. Em contrapartida apresenta uma leve oscilação, mas sempre próximo da trajetória estipulada

O erro de trajetória do quadricoptero é observado na Fig. 7, obtendo valores altos no início da simulação e estabiliza-se em torno de zero, que indica que o controlador PID-Neural consegue seguir a trajetória, em contrapartida o controlador PD sempre apresenta elevado erro de trajetória quando há uma mudança brusca na trajetória.

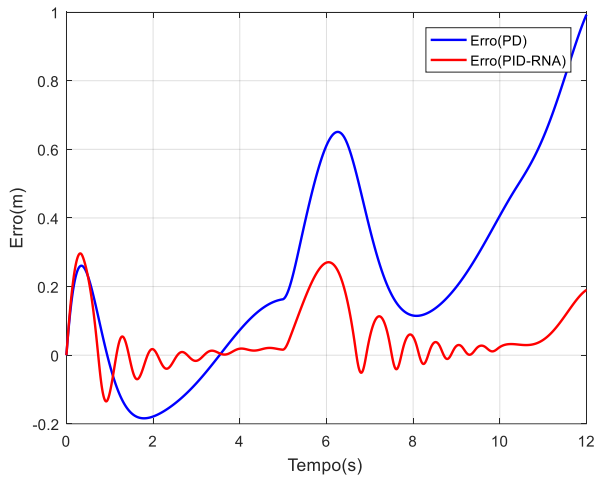


Fig. 7. Análise do erro de trajetória de referência.

A evolução dos treinamentos dos ganhos fundamentados em (16) pode ser observado na Fig. 9. Os ganhos da rede iniciam-se em zero, demonstrando que a rede é capaz de sintonizar os melhores ganhos PID, apresentando suas características de adaptabilidade e atingindo a convergência próximo de 0.8 segundos.

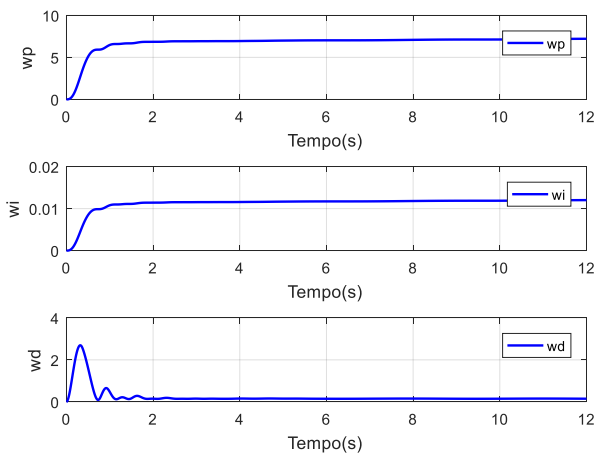


Fig. 8. Atualização dos ganhos do PID-RNA.

Na Fig. 10 pode ser observado o esforço de controle do PD tradicional (7) e do controlador PID-Neural (8), como o sinal

de controle é limitado entre 1,22 N e 0,0245 N, no início da simulação atingiu-se a saturação do sinal de controle. O esforço de controle do controlador PID-Neural, apresenta oscilações no início da simulação para minimizar o erro de trajetória no início da simulação, e na sequencia apresenta o mesmo comportamento do controlador PD.

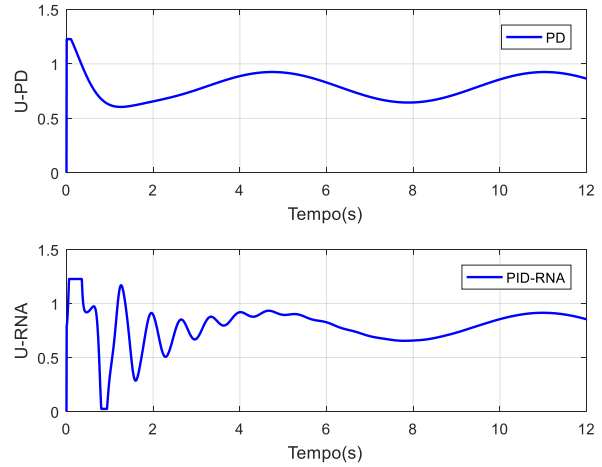


Fig. 9. Esforço do controlador PD e PID-Neural.

Para confirmar a eficiência do método de sintonia proposto é aplicado uma variação paramétrica na planta. Após 5 segundos de simulação é inserido uma variação paramétrica, com acréscimo de 30% na massa do drone, simulando um transporte de um equipamento. A Fig. 11, representa a resposta do quadricoptero, comprovando que mesmo com a variação paramétrica, o quadricoptero consegue seguir a trajetória de referência.

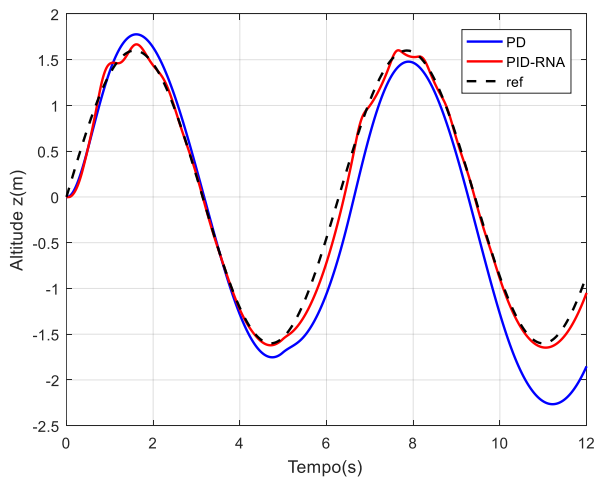


Fig. 10. Resposta do sistema com variação paramétrica.

No entanto o controlador PD não atende muito bem quando a planta sofre variações paramétricas e o mesmo não consegue seguir a trajetória definida.

O sinal de erro de trajetória dos controladores é mostrado na Fig. 12, comprovando que controlador PID-Neural é capaz

de se auto ajustar e manter o erro de trajetória próximo de zero, já o controlador PD apresenta um aumento deste erro.

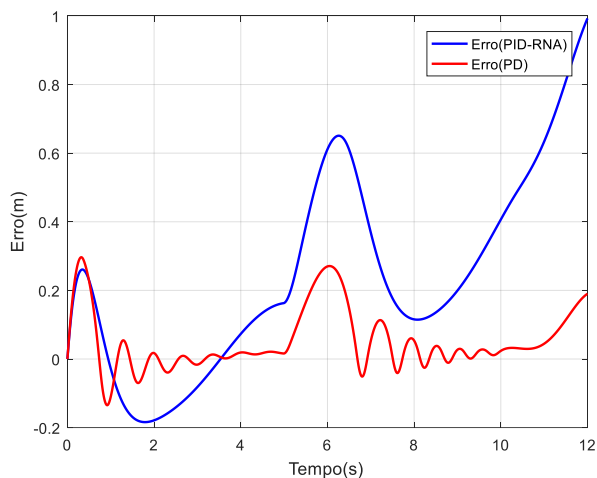


Fig. 11. Análise do erro de trajetória de referência com variação paramétrica.

Na Fig. 14 observa-se o comportamento dos ganhos do controlador PID-Neural quando ocorre a variação paramétrica na planta fazendo com que ainda consiga seguir a trajetória com erro de percurso reduzido.

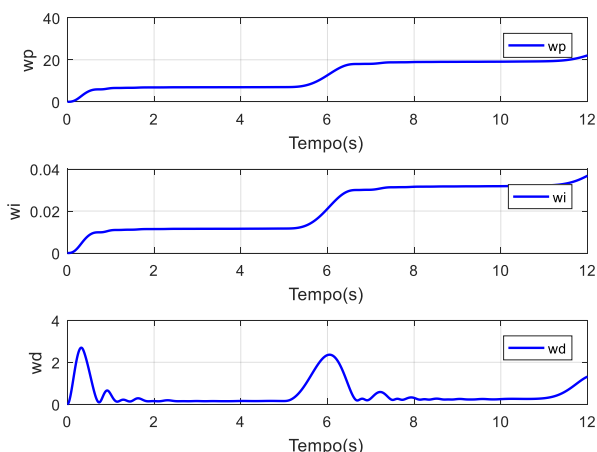


Fig. 12. Atualização dos ganhos com variação paramétrica.

## V. CONCLUSÃO

Neste artigo foi apresentado um método para sintonia de controladores PID via Redes Neurais fundamentada na teoria geral de ajuste do LMS, com os ganhos do controlador ajustados de forma online. Foi levantado o modelo matemático da dinâmica do quadricóptero para o controle de altitude. O sistema dinâmico foi visto como um neurônio do tipo perceptron com sua função de ativação regida pelo

comportamento da planta. E por fim, resultados e simulações foram apresentados.

A superioridade do método proposto foi demonstrada a partir dos resultados dos experimentos simulados, com uma análise comparativa do controlador PID-Neural proposto com o controlador PD tradicional. O PID-Neural apresentou um menor erro de trajetória e foi capaz de seguir uma trajetória de referência mesmo na presença de variações paramétricas.

## AGRADECIMENTOS

Agradecemos a Universidade Federal do Maranhão (UFMA) e Laboratório de Sistemas Embarcados e Controle Inteligente (LABSECI), pelos conhecimentos técnico/científicos e práticos para o desenvolvimento deste trabalho. E finalmente à Fundação de Amparo à Pesquisa e Desenvolvimento do Maranhão (FAPEMA), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPQ) pelo financiamento desta pesquisa.

## REFERÊNCIAS

- [1] F. Javidi-Niroumand, and A. Fakharian. "Trajectory Tracking via Adaptive Nonlinear Control Approach for a Quadrotor MAV," IEEE Artificial Intelligence and Robotics, 2015.
- [2] J. G. Ziegler, N. B. Nichols. "Optimum settings for automatic controllers." trans. ASME, v.64, n. 11. 1942
- [3] K. J. Astrom, B. Wittenmark., Adaptive control, 2nd ed, Courier Corporation, 2013.
- [4] S. Haykin. Adaptive Filter Theory, 3<sup>rd</sup> ed, New York: Prentice-Hall, 2002.
- [5] M. S. Aftab, and M. Shafiq. "Adaptive PID Controller based on Lyapunov Function Neural Network for Time Delay Temperature Control," IEEE 8th Conference & Exhibition, 2015.
- [6] R. Hernandez-Alvarado, L. G. Garcia-Valdovinos, T. Salgado-Jimenez, A. Gomez-Espinosa, and F. Fonseca-Navarro, "Neural network-based self-tuning pid control for underwater vehicles," Sensors, vol. 16, no. 9, p. 1429, 2016.
- [7] R. Wang, Z. Zhou, and G. Qu. "Fuzzy Neural Network PID Control Based on RBF Neural Network for Variable Configuration Spacecraft," IEEE 3rd Advanced Information Technology, Electronic and Automation Control Conference, pp 1203-1207, 2018.
- [8] S. Bari, S. S. Z. Hamdani, H. U. Khan, M. Rehman, and H. Khan. "Artificial Neural Network Based Self-Tuned PID Controller for Flight Control of Quadcopter," IEEE International Conference on, Engineering and Emerging Technologies, 2019.
- [9] M. N. Duc, T. N. Trong, Y. S. Xuan. "The Quadrotor MAV System using PID Control," IEEE International Conference on Mechatronics and Automation, 2015.
- [10] Bouabdallah, S., Noth, A., Siegwart, R. "PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor," IEEE International Conference on Intelligent Robots and Systems, pp 2451-2456, 2004.
- [11] S. Saha, and S. Wadoo. "Linear Optimal Control of a Parrot AR Drone 2.0" IEEE Undergraduate Research Technology Conference, 2017.
- [12] K. Ogata. Engenharia de Controle Moderno, 5rd ed, Pearson, 2010.
- [13] S. S. Haykin. Neural networks and learning machines, vol. 3, Pearson, 2009.