19th Brazilian Congress of Thermal Sciences and Engineering
November 06th-10th, 2022, Bento Gonçalves - RS - Brazil

ENCIT2022-0301
A FEEDBACK LINEARIZATION CONTROLLER USING A HYBRID CONTROL WITH WAVELET AND PERCEPTRON NEURAL NETWORKS APPLIED TO A HYDRAULIC ACTUATOR

Fabio Augusto Pires Borges
Thomaz Pereira da Silva Junior
Vitor Mauro Fiori
Letieri Rodrigues de Ávila
N2E Research Group, Engineering School, Federal University of Rio Grande - FURG, Av Italia, km 8, 96203-900, Rio Grande, RS, Brazil.
e-mails: *fborges@furg.br, hbkthomaz@gmail.com, vmfiori@hotmail.com, letieriavila@gmail.com*

Abstract. *In this work, we use a neural network as a substitute for the traditional analytic functions employed as an inversion set in feedback linearization control algorithms applied to hydraulic actuators. Although very effective and with strong stability guarantees, feedback linearization control depends on parameters that are difficult to determine, requiring large amounts of experimental effort to be identified accurately. On the other hand, neural networks require little effort regarding parameter identification the control hardware. Here, we combine these techniques to control the positioning of a hydraulic actuator, without requiring extensive identification procedures. We compare the use of wavelet and perceptron neural networks applied in the proposed schema. Simulation results are obtained, and the effectiveness of the controller is confirmed with low position errors when compared with a classical PID controller in a piston position tracking trajectory control. Advantages and weakness of the use of both perceptron and wavelet network are outlined in the final conclusions.*

Keywords: *hydraulic actuator control, neural network-based identification, feedforward multilayer perceptron, feedback linearization, wavelet neural network.*

1. INTRODUCTION

The Hydraulic actuators are usually applied in industrial tasks when the process requiring high forces with limited dimensions. However, when we aimed high precision control responses, several problems must be overcome when hydraulic actuators are used. Some strongly nonlinear phenomena are present in these actuators such as valve saturation, behavior of the flow rates through the valve orifices and friction forces in the piston (Watton, 2009). Therefore, the design of controllers that can overcome such characteristics in high precision applications is a challenging task. Several authors have been proposed control strategies that enhancing the precision of hydraulic actuators, e.g., backstepping (Guo *et al.*, 2016; Yao and Deng, 2017; Yao *et al.*, 2017; Sun *et al.*, 2018; Yang *et al.*, 2018), feedback linearization (Seoa *et al.*, 2007; Fales and Kelkar, 2009; Mintsa *et al.*, 2012) and sliding mode control (Ghazali *et al.*, 2011; Xinliang and Wang, 2013; Zhu *et al.*, 2016). Many other algorithms that combining these and other techniques have been proposed, as illustrated in Coelho and Cunha (2011), Ji *et al.* (2015) and Na *et al.* (2020)**,** for instance. In these mentioned papers, strategies to overcome the parametric uncertainties and external disturbances present in hydraulic actuators are applied, such as the online learning methods, where parameters and disturbances are estimated in real time, e.g., adaptive control (Coelho and Cunha, 2011; Xinliang and Wang, 2013; Yao and Deng, 2017), Extended State Observer (ESO) (Guo *et al.*, 2016; Yao and Deng, 2017), Extended Disturbance Observer (EDO) (Sun *et al.*, 2018) and Extended Differentiator (Yang *et al.*, 2018). Despite they provide reduced position errors when compared with fixed-structure controllers, they also have important drawbacks for practical applications, such as the need for tuning large numbers of control parameters. Usually, such tuning process must take into account a compromise between the robustness of the controller, the computing capabilities of the available control hardware, and the convergence speed of the estimation procedure, which is a rather difficult task. Furthermore, other factors like sensor noise tend also to affect the convergence performance. Thus, at least some of the aforementioned online estimation approaches may require highly sophisticated and expensive sensors for attaining accurate results.

An effective approach for controlling hydraulic systems is based on *feedback linearization* methods, which compensate for nonlinear phenomena by estimating their effects and applying a control signal that opposes them. In this work, we propose a hybrid control scheme applied to a hydraulic positioning system. It employs two control loops, based on the main phenomena involved in the system: the outer loop is concerned with its mechanical variables, whereas the inner loop is a feedback linearization algorithm, which aims at compensating for the nonlinearities due to the hydraulic dynamics. Neural Networks have been widely used in many different applications, mainly because of their ability of "learning by themselves" how to adapt to their working environment and have been used in the control of hydraulic actuators to compensate for the unknown disturbances and unmodeled dynamics (Pedro and Ekoru, 2013; Yang *et al.*, 2013; Borges *et al.*, 2016; Borges, 2017; Lin *et al.*, 2018; Zhu *et al.*, 2018; Yao *et al.*, 2020). In the outer loop, we use a neural network to compensate the nonlinear effect of the friction forces of the piston. In the inner loop, we substitute just one part of the traditional analytic model with neural networks, namely, the computing of the actual control signal to be applied to the control valve. Both neural network parameters are acquired by an extensively offline training. Since most of the critical uncertain parameters of the system are necessary in this specific step of the control loop, we still retain the basic advantages of feedback linearization control while avoiding large amounts of experimental effort to find suitable parameters for the inverse set and compensation of the friction force, a significant advantage when compared with the traditional offline methods. Furthermore, the offline training procedure breaks the application of the proposed controller into two stages: training and trajectory tracking, which significantly reduces the computing efforts involved in each one, thus making the proposed method amenable to control hardware's more modest than those required for fully online-trained controllers. In Borges *et al.* (2016), the same approach was introduced using multilayer perceptron neural networks. In the present work, we introduce the use of wavelet neural networks for both the inner and outer loop neural networks. According to Alexandridis and Zapranis (2013), wavelet neural networks have important advantages when compared with the traditional feedforward multilayer perceptron neural networks such as a better generalization capability for networks with reduced inputs dimension, a linear relation between the hidden and output layers and an analytical methodology for the initialization of the parameters in the core of hidden layer. Wavelet neural networks have been successfully applied in control systems as an alternative to the others neural networks as the feedforward multilayer perceptron and RBF neural networks in papers as Leu and Hong (2010) and Wang *et al.* (2013). In the present work, a comparation highlighting the advantages and drawbacks of both neural networks strategies in the proposed schema is performed. The effectiveness of the proposed strategy applying perceptron and wavelet neural networks is demonstrated with comprehensive simulation results.

The remainder of this paper is structured as follows. In Section 2, the hydraulic actuator model is discussed. In Section 3, we present the overall control strategy, whereas Section 4 and 5 are dedicated to theoretical issues of multilayer perceptron and wavelet neural networks. In section 6, the training and validation methodology is detailed, while, in Section 7, the proposed method is evaluated by means of simulation results. Finally, the main conclusions are outlined in Section 8.

2. HYDRAULIC ACTUACTOR MODEL

The Figure 1 illustrates the Hydraulic Actuator applied in the present work. Following other works (Watton, 2009; Coelho and Cunha, 2011; Sobczyk *et al.*, 2012), the mathematical modeling of its dynamic behavior is based on Newton's second law combined with flow-continuity considerations. According to the system described in Fig. 1, the mechanical dynamics of the actuator piston is:

$$F_H = p_1 A_1 - p_2 A_2 = M\ddot{y} + F_A \tag{1}$$

where y, $\dot{y}$ and $\ddot{y}$ are the position, velocity and acceleration of the piston-load assembly, respectively, p_1 and p_2 are the pressures in the chambers, A_1 and A_2 are the piston areas, M represents the mass of the piston rod and the load, F_H is the hydraulic force applied to the piston resulting from the action of p_1 and p_2 and F_A is the friction force in the piston. The dynamic behavior of such pressures is modeled according to the principle of mass conservation that Applied to each cylinder chamber yields:

$$\dot{p}_1 = \frac{\beta}{v_1 + A_1 y}\left(Q_1 - A_1 \dot{y}\right) \tag{2}$$

$$\dot{p}_2 = -\frac{\beta}{v_2 + A_2 y}\left(Q_2 - A_2 \dot{y}\right) \tag{3}$$

where β is the hydraulic fluid bulk modulus, v_1 and v_2 are the initial volumes of fluid in each cylinder´s chamber. Q_1 and Q_2 are the volumetric flow rates through the valve orifices and may be expressed as functions of the pressures in the chambers and the input signal u applied to the valve as follow:

$$Q_1 = K_{v1} u g_1, \quad g_1 = \begin{cases} \sqrt{p_s - (p_1 + l_1)}, & u \geq 0 \\ \sqrt{p_1 - l_3}, & u < 0 \end{cases}$$
$$Q_2 = K_{v2} u g_2, \quad g_2 = \begin{cases} \sqrt{p_2 - l_4}, & u \geq 0 \\ \sqrt{p_s - (p_2 + l_2)}, & u < 0 \end{cases} \tag{4}$$

where K_{v1} and K_{v2} are the flow-rate gains that characterize each orifice of the valve, and $l_1 \ldots l_4$ are the pressure losses caused by the hydraulic line couplers, which are significant and must be taken into account when high-precision tasks are considered.

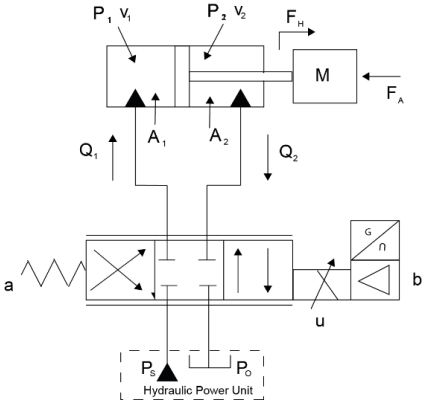

Figure 1. Hydraulic actuator.

Taking the auxiliary terms f_1 and f_2 as follow:

$$f_1 = \frac{\beta}{v_1 + A_1 y}, \quad f_2 = \frac{\beta}{v_2 - A_2 y} \tag{5}$$

and replacing Q_1, Q_2, f_1, and f_2 in (2) and (3) by their corresponding terms given in (4) and (5), resulting:

$$\dot{F}_H = \left(A_1 f_1 K_{v1} g_1 + A_2 f_2 K_{v2} g_2\right) u - \left(A_1^2 f_1 + A_2^2 f_2\right) \dot{y} \tag{6}$$

The parameters of the actuator used in the present work were obtained by the experimental plant found in Borges (2017) and Borges *et al.* (2021). Several of experimentation procedures are applied to obtain a satisfactory simulation model. Such experiments may be found in Borges (2017) for the hydraulic actuator used in the present work. The Table. **1** outlines the parameters applied in the model. The friction force F_A is simulated by means of the Gomes model (Gomes and Rosa, 2003). The complete Gomes model parameters for the present actuator were obtained in Borges (2017) and are described in Appendix.

3. CONTROL STRATEGY

The cascade control strategy applied to a hydraulic actuator consists in interpreting the hydraulic actuator mathematical model as two interconnected subsystems (Cunha *et al.*, 2002) or two control loops: a hydraulic subsystem (inner loop) and mechanical one (outer loop). Such a strategy can be summarized as:

(i) Compute a control law F_d (desired hydraulic force) for the mechanical subsystem such that the output y tracks the desired trajectory y_d as closely as possible.

(ii) Compute a control law u for the hydraulic subsystem such that F_H tracks F_{Hd} as closely as possible.

Table. 1. Model Parameters.

Parameter	Value	Estimation method
V_{10}	1.2446×10^{-4} m^3	Direct measurement
V_{20}	9.9060×10^{-5} m^3	Direct measurement
y_0	0.1 m (center)	Direct measurement
K_{v1} ; $K_{v2,}$	$\sqrt{2}.15.11\times10^{-9}$ m^3/(s×$\sqrt{\text{Pa}}$)	Manufacturer catalog
l_1	5.68×10^{10}Q$_1$ Pa	Interactive simulation
l_2	4.35×10^{10}Q$_2$ Pa	Interactive simulation
l_3	3.59×10^{10}Q$_1$ Pa	Interactive simulation
l_4	3.59×10^{10}Q$_2$ Pa	Interactive simulation
M	14.54 kg	Direct measurement
A_1	4.91×10^{-4} m^2	Direct measurement
A_2	2.37×10^{-4}m^2	Direct measurement
β	1.0×10^{9}N/m^2	Typical value
P_s	50×10^{5}Pa	Input choice

According to the strategy proposed, the outer loop control law is the same of the Slotine and Li controller (Slotine and Li, 1988), described by Eq. (8) where K_d is a positive constant, $\ddot{y}_r$ is a reference acceleration and z is a measure tracking error. Such auxiliary terms and the resulting expression for the desired hydraulic force are:

$$\dot{y}_r = \dot{y}_d - \lambda\tilde{y}, \quad \tilde{y} = y - y_d, \quad z = \dot{y} - \dot{y}_r = \dot{\tilde{y}} + \lambda\tilde{y} \tag{7}$$

$$F_{Hd} = M_0\ddot{y}_r - K_d z + \Omega \tag{8}$$

where the term Ω is a neural network that stands for the friction force.

In the inner loop, the control law is given by Eq. (12) that represents a feedback linearization control strategy where the system nonlinearities are perfectly cancelled when all necessary parameter values are known without uncertainty. The idea of feedback linearization, i.e. of canceling the nonlinearities and imposing desired linear dynamics, can be easily grasped in the case of nonlinear systems described by the so-called *companion form*. A system is said to be in companion form if its dynamics is represented by

$$x^{(n)} = f(\mathbf{x}) + b(\mathbf{x})u \tag{9}$$

where u is the scalar control input, x is the scalar output of interest, and $\boldsymbol{x} = [x, \dot{x}, \ddot{x}, \ldots, x^{(n-1)}]^T$is the state vector, and $f(\boldsymbol{x})$ and $b(\boldsymbol{x}) \neq 0$ are nonlinear functions of the states. Using this representation, it is clear that choosing the control input

$$u = \frac{1}{b}(v - f), \tag{10}$$

we can cancel the nonlinearities and obtain a simple input-output relation of multiple-integrator form:

$$x^{(n)} = v \tag{11}$$

Comparing Eq. (7) and (10), we have:

$f = -(A_1{}^2 f_1 + A_2{}^2 f_2) \cdot \dot{y}$ and Z a neural network that represents the inverse of b. Defining $v = \dot{F}_{Hd} - K_p\tilde{F}$, the proposed control law is:

$$u = \left(\dot{F}_{Hd} - K_p\tilde{F} + \left(A_1{}^2 f_1 + A_2{}^2 f_2\right)\dot{y}\right)Z \tag{12}$$

where the force error is defined as $\tilde{F} = F_H - F_{Hd}$. With this control law, the closed loop system dynamics when all model parameters match their real values becomes:

$$\dot{\tilde{F}} = -K_p \tilde{F} \tag{13}$$

i.e., the force applied to the piston converges asymptotically to its desired value defined in Eq. (9), which, in turn, ensures that the trajectory performed by the actuator converges to its reference. These facts are proven in Borges *et al.* (2021) by means of Lyapunov stability arguments, which also show that the tracking force error converges to zero faster as the gain K_p is raised. If the parameters are uncertain, the same analysis allows to prove that the trajectory tracking errors of the closed loop system converge to a bounded region whose amplitude is determined by such uncertainty amount.

4. FEEDFORWARD MULTILAYER PERCEPTRON

The feedforward multilayer perceptron (MLP) neural network is the most common approach for neural networks. A matrixial representation is:

$$o = \mathbf{\Gamma}\left[\mathbf{W_n}\mathbf{\Gamma}\left[\mathbf{W_{n-1}}...\mathbf{\Gamma}\left[\mathbf{W_1 u} + \mathbf{b_1}\right] + ... + \mathbf{b_{n-1}}\right] + \mathbf{b_n}\right], \tag{14}$$

where $\mathbf{W}_n$ is the weighting matrix of the n-th layer, $\mathbf{b}_n$ is the bias vector associated with each layer node, and $\mathbf{\Gamma(x)}=[\gamma_1(x), \gamma_2(x), \ldots, \gamma_n(x)]$ is a nonlinear operator where each $\gamma_n(.)$ is a monotonic and continuously differentiable activation function. In the present work is used the sigmoidal logistic function.

5. WAVELET NEURAL NETWORKS

The Wavelet Neural Networks WNN are a generalization of radial basis function networks (RBFN) and, according to Alexandridis and Zapranis (2013) preserve de universal approximation capacity of the NN proved in Cybenko (1989). WNN are one hidden layer networks that use a wavelet as an activation function, instead of the classic sigmoidal family. According to Alexandridis and Zapranis (2013), wavelets have a better performance than other transference functions. A wavelet transfer function has high compression abilities and the task of computing the value at a single point or updating the function estimate from a new local measure involves only a small subset of coefficients. The idea of a WNN is to adapt the wavelet basis to the training data. Hence, the wavelet estimator is expected to be more efficient than an MLP (Zhang, 1997).

A wavelet family is a set of functions generates by means of the translation and dilatation of a mother wavelet ψ. The structure of the WNN used in the present work is similar that applied in Alexandridis and Zapranis (2013) and Oussar and Dreyfus (2000). The network output, considering only one output, is given by the following expression:

$$y = \mathbf{W}_1\mathbf{\Psi(A, B, x)} + b_w + \mathbf{W}_2\mathbf{x}\,, \tag{15}$$

where $\mathbf{W_1}$ is the weighting vector that connect the hidden layer to the output layer, b_w is the bias associated with output layer, $\mathbf{W_2}$ is the weighting vector that connect the input vector $\mathbf{x}$ to the output layer and $\mathbf{\Psi}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{x})$ is the wavelons vector, computed according to the input vector $\mathbf{x}$ and the matrices of dilatations and translation $\mathbf{A}$ and $\mathbf{B}$.

If we worked with a WNN where the number of wavelons is m, that is, the number of hidden nodes, and the number of inputs is p, the expression in Eq. (15) can be written as:

$$y = \sum_{i=1}^{m} w_{1i}\Psi_i(A_i, B_i, \mathbf{x}) + b_w + \sum_{g=1}^{p} w_{2_g} x_g \tag{16}$$

where:

$$\Psi_i(A_i, B_i, \mathbf{x}) = \prod_{j=1}^{p} \psi\left(z_{i,j}\right) \tag{17}$$

being that ψ is the mother wavelet chosen for the WNN. The scalar $z_{i,j}$ id given by:

$$z_{i,j} = \frac{x_j - B_{i,j}}{A_{i,j}} \tag{18}$$

The Mexican Hat is the mother wavelet adopted in the present work. It is given by:

$$\psi\left(z_{i,j}\right)=\left(1-z_{i,j}{}^{2}\right)e^{-\frac{1}{2}z_{i,j}{}^{2}} \tag{19}$$

6. NEURAL NETWORKS TRAINING AND VALIDATION

The building of the training ad validation sets of the neural network functions was performed offline by using a simple proportional controller to lead the system to track the desired position trajectory, as illustrated in Fig. 2. This profile corresponds to moving the actuator piston from one fixed position to another, using a set of sinusoidal trajectories with different frequencies to ensure a smooth transition between them as described in Eq. (20). For the process of validation was applied the cross validation method (Haykin, 1999).

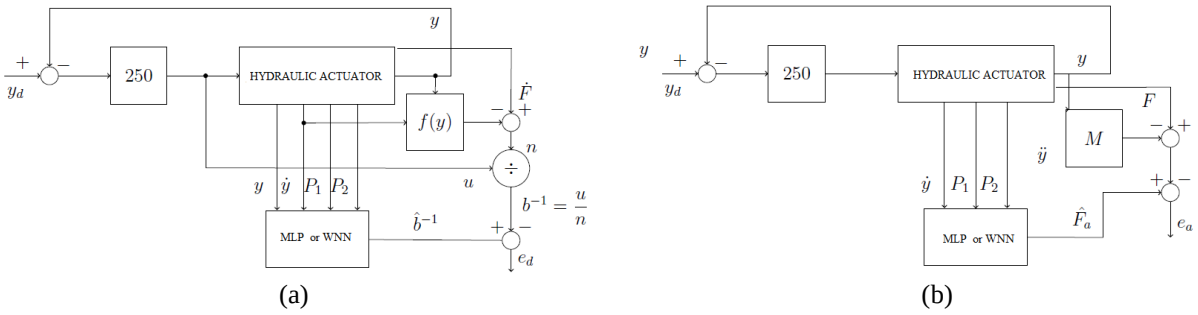

Figure. 2. Neural Network: training and validation set generation. (a) Z. (b) Ω.

The position trajectory applied in the generation of the training and validation sets is:

$$y_d = 0.1 + Asen\left(\omega t\right) \tag{20}$$

where A is the amplitude and ω is the angular frequency. The amplitude is kept in a fixed value of 0.08m. The Tables 2 and 3 outline the frequencies applied in training and validation sets generation.

Table 2. Frequencies in training set.

Frequency (rad/s)	Time limit (s)
1	11
0.875	18.8
0.625	28.23
0.375	44.98
0.25	71.65

Table 3. Frequencies in validation set.

Frequency (rad/s)	Time limit (s)
0.93	11.81
0.81	19.5

The Quickprop algorithm (Fahlman, 1988) was applied in both MLP and WNN training. The core of both WNN and MLP were chosen with thirty neurons in the hidden layer. We used the root-mean-square error in Eq. (21) as the performance criterion to compare the different neural network errors.

$$RMSE = \sqrt{\frac{1}{P}\sum_{n=1}^{P}\left(y_t - y_p\right)^2} \tag{21}$$

where y_t and y_p are the target and predicted values, respectively, and P is the number of samples.

The Figures 3 and 4 show the progress of RMSE along of training epochs for Ω and Z networks, applying MLP and WNN architectures. We can observe in the figures and in the Tab. 4 that the WNN offers a minor value for the final RMSE and converge faster to a suitable RMSE value than the MLP.

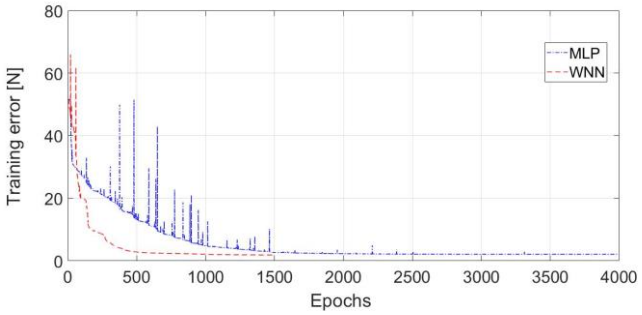

Figure. 3. Training of Ω network.

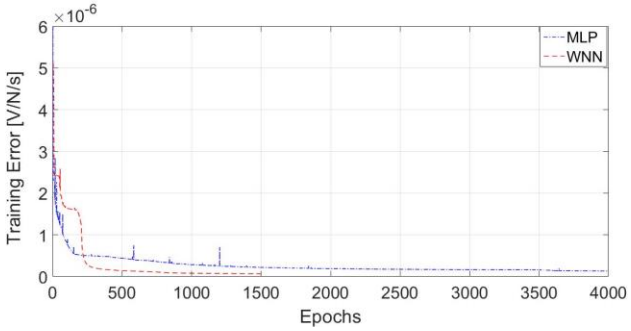

Figure. 4. Training of Z network.

The Figures 5 and 6 show the validation set and the outputs of Ω and Z networks, applying MLP and WNN architectures. We can observe in the figures that WNN and MLP present close results. The Table. **4** outline the RMSE for both the curves, where WNN is a little better for Z network and MLP for Ω.

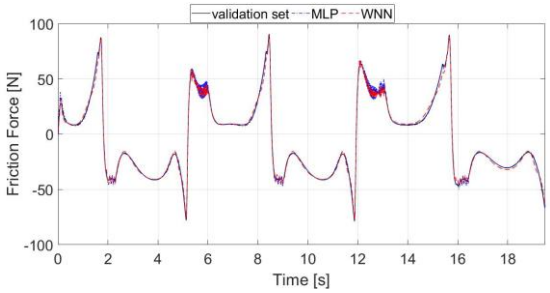

Figure. 5. Validation of Ω network.

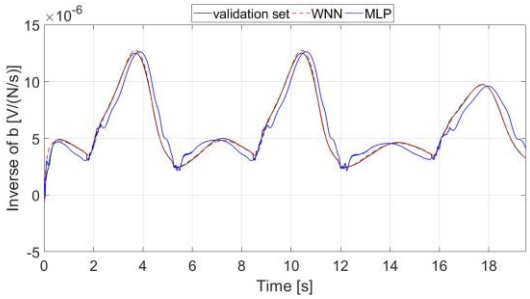

Figure. 6. Validation of Z network.

7. SIMULATION RESULTS

Simulation evaluation was carried out by means of a position tracking control involving a trajectory test: a sinusoid with amplitude 0.08 m and period 8.3 s was tracked.

The feedback gains used in the proposed controller were kept fixed with the values of K_d = 5000 s^{-1}, λ = 150 s^{-1} and K_p = 200 s^{-1}. For comparison purposes, we also performed a simulation using a classical PID controller. The feedback gains values for PID were K_p = 420, K_i = 2018, K_d = 0.9. Simulation results are shown in Figure. **7**. The Table. **5** presents the RMSE results. We can observe that the results of both WNN and MLP are very close. The error of the proposed controller is about seven times lower than the PID controller.

Table. 4. Training and Validation errors.

	Neural Network	Training error	Validation error
Ω (Friction Force)	MLP	2.11 N	1.74 N
	WNN	1.86 N	2.29 N
Z (b^{-1})	MLP	1.30 10^{-7} V/N/s	1.89 10^{-7} V/N/s
	WNN	0.67 10^{-7} V/N/s	1.67 10^{-7} V/N/s

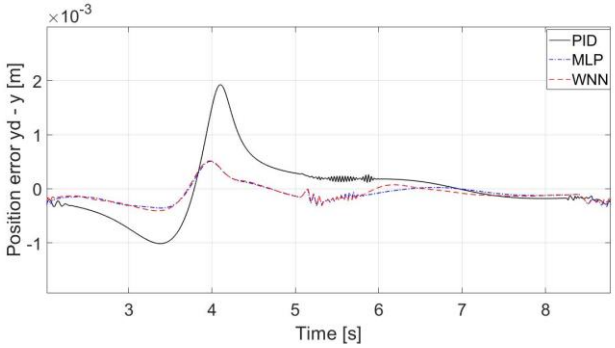

Figure. 7. Trajectory tracking control position errors.

Table. 5. Trajectory tracking control position errors.

Neural Network	position error
PID	13.8 10^{-4} m
MLP	1.91 10^{-4} m
WNN	1.95 10^{-4} m

8. CONCLUSIONS

In the present work, we propose the use of a feedforward multilayer neural network and a wavelet neural network to compensate the nonlinearities present in the plant of a hydraulic actuator. This strategy aims to facilitate and improve the application of feedback linearization-based control schemes to such system. We showed by means of simulation results that the proposed controller is very effective in reduction of the position error compared with a traditional PID controller in a tracking position control applied to a hydraulic actuator, confirming the experimental results obtained in Borges (2017), Borges *et al.* (2016) and Borges *et al.* (2021). Moreover, we performed a comparation between the results of MLP and WNN applied in the core of the proposed controller. The results showed that both the strategies have similar performance in the effectiveness of the controller, highlighting the superior training speed of WNN, that can be an advantage when used in online control proposes. Future work will focus the expansion of the proposed method to encompass unknown external load disturbances in the hydraulic plant.

9. REFERENCES

Alexandridis, A. K. and Zapranis, A. D., 2013. 'Wavelet neural networks: A practical guide', Neural Networks, 42, pp. 1–27. .

Borges, F. A. P., Perondi, E. A., Cunha, M. A.B., Sobczyk, M. R, 2016. 'A hybrid feedback linearization + neural network control algorithm applied to a hydraulic actuator', in 9th FPNI Ph.D. Symposium on Fluid Power, FPNI 2016. Florianópolis, SC, Brazil: ASME, p. V001T01A037.

Borges, F. A. P., 2017. Controle em cascata de um atuador hidráulico utilizando redes neurais. Universidade Federal do Rio Grande do Sul.

Borges, F. A. P., Perondi, E. A., Cunha, M. A.B., Sobczyk, M. R., 2021. 'A neural network-based inversion method of a feedback linearization controller applied to a hydraulic actuator', Journal of the Brazilian Society of Mechanical Sciences and Engineering. Springer Berlin Heidelberg, 43(5), pp. 1–19.

Coelho, L. D. S., Cunha, M. A. B., 2011. 'Adaptive cascade control of a hydraulic actuator with an adaptive dead-zone compensation and optimization based on evolutionary algorithms', Expert Systems with Applications, 38(10), pp. 12262–12269.

Cunha, M. A. B., Guenther, R., De Pieri, E. R., De Negri, V. J., 2002. 'Design of Cascade Controllers for a Hydraulic Actuator', International Journal of Fluid Power, 3(2), pp. 35–46.

Cybenko, G., 1989. 'Approximation by superpositions of sigmoidal functions', Mathematics of Control, Signals, and Systems.

Fahlman, S. E., 1988. 'An Empirical Study of Learning Speed in Back-Propagation Networks', (4976).

Fales, R., Kelkar, A., 2009. 'Robust control design for a wheel loader using H∞ and feedback linearization based methods', ISA Transactions. Elsevier Ltd, 48(3), pp. 312–320.

Ghazali, R., Ngadengon, R., Sam, Y. M., Rahmat, M. F., Hamzah, N., 2011. 'Chaotic trajectory tracking of an electro-hydraulic actuator system using discrete sliding mode control', 2011 IEEE International Conference on Control System, Computing and Engineering, pp. 500–506.

Gomes, S. C. P., Rosa, V. S., 2003. 'A new approach to compensate friction in robotic actuators', Proceedings - IEEE International Conference on Robotics and Automation, 1, pp. 622–627.

Guo, Q, Zhang, Y., Celler, B. G, Su, S. W., 2016. 'Backstepping Control of Electro-Hydraulic System Based on Extended-State-Observer With Plant Dynamics Largely Unknown', IEEE Transactions on Industrial Electronics, 63(11), pp. 6909–6920.

Ji, Y., Huang, G. H., Sun, W., 2015. 'Risk assessment of hydropower stations through an integrated fuzzy entropy-weight multiple criteria decision making method: A case study of the Xiangxi River', Expert Systems with Applications. Elsevier Ltd, 42(12), pp. 5380–5389.

Leu, Y. G., Hong, Y. X., 2010. 'Neural networks output feedback controllers using nonlinear parametric wavelet functions', International Journal of Intelligent Computing and Cybernetics, 3(4), pp. 672–685.

Lin, Y. C., Chen, D., Chen, M., Chen, X., Li, J., 2018. 'A precise BP neural network-based online model predictive control strategy for die forging hydraulic press machine', Neural Computing and Applications, 29(9), pp. 585–596.

Mintsa, H., Venugopal, R., Kenne, J. P., Belleau, C., 2012. 'Feedback linearization-based position control of an electrohydraulic servo system with supply pressure uncertainty', IEEE Transactions on Control Systems Technology, 20(4), pp. 1092–1099.

Na, J. , Li, Y., Huang, Y., Gao, G., Chen, Q., 2020. 'Output Feedback Control of Uncertain Hydraulic Servo Systems', IEEE Transactions on Industrial Electronics. IEEE, 67(1), pp. 490–500.

Oussar, Y., Dreyfus, G., 2000. 'Initialization by selection for wavelet network training', Neurocomputing, 34(1–4), pp. 131–143.

Pedro, J., Ekoru, J., 2013. 'NARMA-L2 Control of a Nonlinear Half-Car Servo-Hydraulic Vehicle Suspension System', Acta Polytechnica Hungarica, 10(4), pp. 5–26.

Seoa, J., Venugopal, R., Kennéa, J. P., 2007. 'Feedback linearization based control of a rotational hydraulic drive', IFAC Proceedings Volumes (IFAC-PapersOnline), 7(PART 1)

Haykin, S., 1999. Neural networks: a comprehensive foundation. 2nd edn. Pearson.

Slotine, J.-J. E., Li, W., 1988. 'Adaptive manipulator control: A case study', IEEE Transactions on Automatic Control, 33(11), pp. 995–1003.

Sobczyk, M. R., Perondi, E. A., Cunha, M. A. B., 2012. 'A continuous extension of the LuGre friction model with application to the control of a pneumatic servo positioner', in 2012 IEEE 51st IEEE Conference on Decision and Control (CDC), pp. 3544–3550.

Sun, C., Fang, J., Wei, J., Hu, B. O., 2018. 'Nonlinear Motion Control of a Hydraulic Press Based on an Extended Disturbance Observer', IEEE Access, 6, pp. 18502–18510.

Wang, G., Guo, L., Duan, H., 2013. 'Wavelet neural network using multiple wavelet functions in target threat assessment', The Scientific World Journal, 2013.

Watton, J., 2009. Fundamentals of Fluid Power Control. Cambridge University Press.

Xinliang, L., Wang, X., 2013. 'Adaptive Sliding Mode Position Control of Electro-Hydraulic Servo System with Single-Rod Actuators', 2013 IEEE International Symposium on Robotic and Sensors Environments (ROSE), pp. 220–225.

Yang, X., Zheng, X., Chen, Y., 2018. 'Position Tracking Control Law for an Electro-Hydraulic Servo System Based on Backstepping and Extended Differentiator', IEEE/ASME Transactions on Mechatronics. IEEE, 23(1), pp. 132–140.

Yang, Y. Balakrishnan, S. N., Tang, L., Landers, R.G., 2013. 'Electro-hydraulic piston control using neural MRAC based on a modified state observer', IEEE/ASME Transactions on Mechatronics, 18(3), pp. 867–877.

Yao, J., Deng, W., 2017. 'Active Disturbance Rejection Adaptive Control of Hydraulic Servo Systems', IEEE Transactions on Industrial Electronics, 64(10), pp. 8023–8032.

Yao, J., Deng, W. , Jiao, Z., 2017. 'RISE-Based Adaptive Control of Hydraulic Systems With Asymptotic Tracking', IEEE Transactions on Automation Science and Engineering. IEEE, 14(3)

Yao, Z. Yao, J.,Yao, F., Xu, Q., Xu, M., Deng, W., 2020. 'Model reference adaptive tracking control for hydraulic servo systems with nonlinear neural-networks', ISA Transactions. Elsevier Ltd, 100, pp. 396–404.

Zhang, Q., 1997. 'Using wavelet network in nonparametric estimation', IEEE Transactions on Neural Networks, 8(2), pp. 227–236.

Zhu, L., Wang, Z., Liu, Y., Song, W., 2016. 'Sliding-mode dynamic surface control for MDF continuous hot pressing hydraulic system', 2016 Chinese Control and Decision Conference (CCDC). IEEE, pp. 2509–2514.

Zhu, L., Wang, Z., Zhou, Y., 2018. 'Adaptive Neural Network Saturated Control for MDF Continuous Hot Pressing Hydraulic System With Uncertainties', IEEE Access. IEEE, 6, pp. 2266–2273.

10. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.

11. APPENDIX

The friction force in the Gomes model is described by the trajectories illustrated in Fig 8.

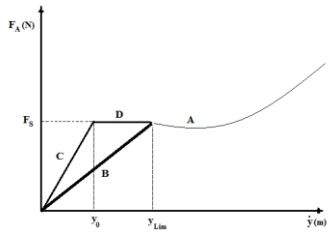

Figure.8. Trajectories of Gomes friction model.

$$\dot{y} \geq \dot{y}_{\lim} \rightarrow F_A = f_{vA}(\dot{y}) - K\dot{y}. \qquad \text{Trajectory A.} \tag{22}$$

$$f_{vA}(\dot{y}) = c_6\dot{y}^6 + c_5\dot{y}^5 + c_4\dot{y}^4 + c_3\dot{y}^3 + c_2\dot{y}^2 + c_1\dot{y} + c_0 \;\; ; \;\; \dot{y} > 0, \tag{23}$$

$$f_{vA}(\dot{y}) = d_5\dot{y}^5 + d_4\dot{y}^4 + d_3\dot{y}^3 + d_2\dot{y}^2 + d_1\dot{y} + d_0 \;\; ; \;\; \dot{y} < 0, \tag{24}$$

where: $\mathbf{c^T}$ = [3616055991,90211 -1259149259,42947 171035516,506081 1509039,8703813424512,463097214 -6778,25791735762 166,069950482143]^T, $\mathbf{d^T}$=[-338186182,095526 -65714639,6349697 -4701264,37738616 -194099,979524457 -2677,58352606057 -113,147119937170]^T and K is a term applied to compensate the quick coupler losses present in the actuator. K were set as 2,680 Ns/m.

$$\dot{y} < \dot{y}_{\lim} \text{ and } F \geq F_S \rightarrow F_A = f_{vB}\dot{y} - K\dot{y}. \qquad \text{Trajectory B (stick region)} \tag{25}$$

where $f_{vB} = \frac{F_S}{\dot{y}_{lim}}$.

$$\dot{y}_0 \leq \dot{y} < \dot{y}_{\lim} \text{ and } F < F_S \rightarrow F_A = F_S - K\dot{y}_0. \qquad \text{Trajectory D (slip region)} \tag{26}$$

$$\dot{y} < \dot{y}_0 \text{ and } F < F_S \rightarrow F_A = f_{vC}\dot{y} - K\dot{y}. \qquad \text{Trajectory C (stick region)} \tag{27}$$

where $f_{vC} = \frac{F_S}{\dot{y}_0}$.

Table. 6. The Gomes Friction model parameters.

Parameter	values	operation
F_S	145 N	$\dot{y} > 0$
$\dot{y}_{lim}$	0.004 m/s	
$\dot{y}_0$	0.004*0,95 m/s	
F_S	-108 N	$\dot{y} < 0$
$\dot{y}_{lim}$	-0.002 m/s	
$\dot{y}_0$	-0.002*0,95 m/s	