

# Red neuronal perceptrón multicapa aplicada a control de posición de un motor de DC para seguimiento de trayectorias

A. Cervantes-Marquez\*

\*Universidad Autónoma de Querétaro  
aldocema04@gmail.com

**Resumen**—El presente trabajo muestra el uso de una red neuronal de tipo perceptrón multicapa utilizada para controlar un sistema consistente en un motor de corriente continua que se regulará su posición angular, se realizará el análisis del sistema dinámico aislado, el cual tendrá una aplicación en una pata robótica, siendo el motor el actuador del sistema y un elemento del total del sistema. La red neuronal será entrenada mediante el uso de métodos de control ya conocidos como es el controlador PID, el cual es utilizado para realizar control sobre sistemas, teniendo resultados variables e inconsistentes al momento de cambiar sus ganancias.

**Index Terms**— Red neuronal perceptrón multicapa, Controlador PID, Control de sistemas, Sistema de control.

## I. INTRODUCCIÓN

Los sistemas de control, son de gran utilidad para realizar regulaciones de variables de interés, con el fin de obtener estabilidad y comportamientos deseados en el desempeño de los sistemas.

Por lo que el uso y desarrollo de técnicas que permitan realizar un control y rendimiento esperado, son requeridas en los diferentes campos del conocimiento.

El uso de controladores PID es ampliamente utilizado en el control de sistemas lineales y sus variaciones en aplicaciones más complicadas (sistemas con grandes retardos, sistemas no lineales, sistemas de múltiples entradas y salidas, etc.) [1].

La red neuronal perceptrón multicapa consiste en un conjunto de redes neuronales conectadas entre si, teniendo una buena aplicación en el control de sistemas mediante predicción e interpolación de valores, observando el aprendizaje de que un controlador PID puede dar a un algoritmo de este tipo [2].

## II. MARCO TEÓRICO

Para el desarrollo del trabajo, es necesario conocer los siguientes conceptos y

### II-A. Motor de DC

Un motor de corriente continua consta de una rotor y un estator (de imán permanente), que mediante la excitación de corriente, se crean campos magnéticos y se induce una fuerza de atracción o repulsión, generando el movimiento mecánico del sistema. Mediante el uso de la caracterización del sistema se pudo obtener un sistema en transformada de Laplace de segundo orden como se muestra a continuación [3], [4]:

$$\frac{\theta(s)}{I(s)} = \frac{K}{s(s+a)} \quad (1)$$

Como se observa la función de transferencia, relaciona la entrada  $I$  y la salida  $\theta$  en un espacio continuo  $s$  y las constantes del motor  $K$  y  $a$ .

Para poder aplicar este sistema de una manera computacional, será necesario aplicar la transformada  $z$  y el teorema del corrimiento para poder obtener una ecuación recursiva del sistema [5]. Obteniendo:

$$\theta(k) = K(-e^{-a}+1)I(k-1) + (e^{-a}+1)\theta(k-1) - e^{-a}\theta(k-2) \quad (2)$$

Donde, para cada instante  $k$  a través del periodo de tiempo  $T$  se puede obtener un sistema discreto a lo largo del tiempo  $t = Tk$ .

Tiene la característica de que en lazo abierto (al excitarlo con voltaje sin realimentar la salida) su valor tiende a infinito, además de que al excitar un voltaje negativo (cambio de polaridad) el valor de la posición tiende a infinito negativo, así como también que la convención del ángulo en incremento y decremento (véase Figura 1).

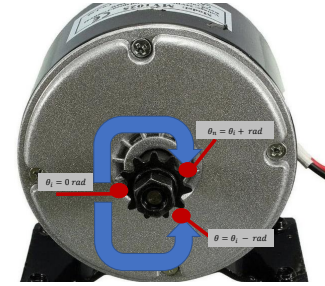


Figura 1. Convención de sentido de giro del motor y tendencia de valores de posición.

### II-B. Controlador PID

El controlador PID consiste en un algoritmo que permite regular un sistema de control mediante su acción proporcional, integral y derivativa [6], teniendo la siguiente forma.

$$I(t) = U(t) = k_p e(t) + k_i \int e(t)dt + k_d \frac{de(t)}{dt} \quad (3)$$

Teniendo de entrada una función de error que es medida a lo largo del tiempo y como salida un esfuerzo de control que será la entrada al motor para excitarlo.

Donde las constantes  $k_p$ ,  $k_i$  y  $k_d$  son las ganancias de cada parte del sistema. Sin embargo, este sistema es aplicado en

un tiempo continuo, por lo que también se le deberá aplicar la transformada  $z$  con el teorema del corrimiento, obteniendo [5], [7].

$$U(k) = U(k-1) + (c_0 e(k)) + (c_1 e(k-1)) + (c_2 e(k-2)) \quad (4)$$

Tomando en cuenta:

$$\begin{aligned} k_{pd} &= k_p, & k_{id} &= k_i T, & k_{dd} &= \frac{k_d}{T} \\ c_0 &= k_{pd} + k_{id} + k_{dd}, & c_1 &= -k_{pd} - 2k_{dd}, & c_2 &= k_{dd} \\ \theta_{final}(k) &= \theta(U(k)) & e(k) &= sp - \theta_{final}(k) \end{aligned} \quad (5)$$

### II-C. Red neuronal Perceptrón

Las redes neuronales son un método de inteligencia artificial que permite la capacidad de predecir comportamientos [8], el cual se basa en el funcionamiento del cerebro humano mediante neuronas.

Las redes neuronales constan básicamente de 3 partes que son (véase Figura 2):

- **Capa de entrada:** consistentes de los valores de entrada del modelo.
- **Capas ocultas:** consisten en los conjuntos de neuronas intermedias.
- **Capa de salida:** consiste de la capa de salidas de la red (valores esperados)

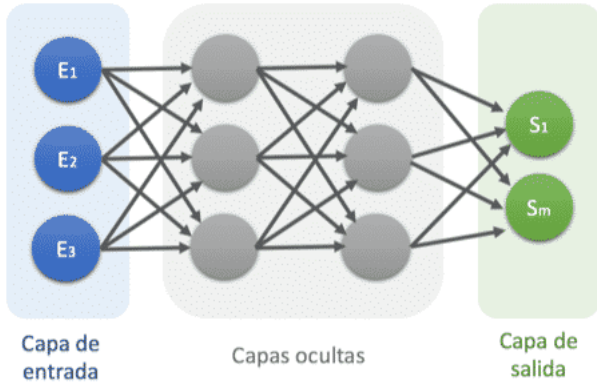


Figura 2. Estructura de red neuronal.

La nomenclatura a utilizar para representar en forma vectorial una red neuronal es la siguiente:

$$r_n = [n_1(f_{a1}), n_2(f_{a2}), n_3(f_{a3}), \dots, n_p(f_{ap})] \quad (6)$$

Por otro lado, la red neuronal de tipo perceptrón consta de la siguiente arquitectura Obteniendo una ecuación de recta como se muestra a continuación (vease Figura 3).

Por lo que se puede generalizar que la neurona tiene un comportamiento:

$$\left( \sum_{i=1}^n (\omega_i x_i) \right) + b \quad (7)$$

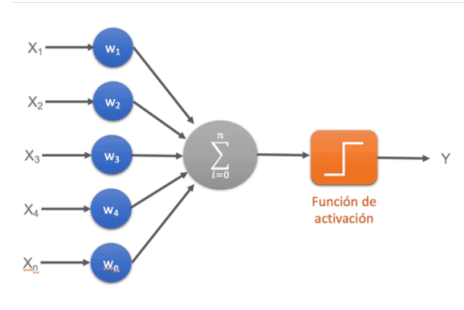


Figura 3. Perceptrón monocapa.

en donde se puede destacar que se tiene una sumatoria de pesos por los valores de la variable  $x$  y finalmente sumándole un bias ( $b$ ), algo análogo a  $\omega_0$  o termino independiente.

**II-C1. Función de pérdida:** En este caso, donde se manejan variables continuas normalizadas (función de pérdida para regresión), es necesario ocupar una función de pérdida adecuada, por lo que el uso de la raíz media cuadrática es la indicada para estos casos [9].

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y})^2}{n}} \quad (8)$$

**II-C2. Optimizador:** Se utilizó un optimizador de Adam, con el fin de poder obtener una velocidad de entrenamiento rápida con una tasa de aprendizaje  $\alpha$  baja y agregando otros hiperparámetros  $\beta_{1,2}$  y  $\epsilon$  [10]. Tomando en cuenta la media  $m$ , la varianza  $v$ ,  $g$  el valor del algoritmo de gradiente descendente y  $\eta$  el tamaño del paso.

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t, & \hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2, & \hat{v}_t &= \frac{v_t}{1 - \beta_2^t} \end{aligned} \quad (9)$$

Es posible obtener el valor de la siguiente iteración de la optimización de Adam:

$$\omega_t = \omega_{t-1} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon} \quad (10)$$

**II-C3. Función de activación:** El uso de funciones de activación para la regresión, deben ser funciones continuas, por su simplicidad y el manejo de aplicaciones donde se manejen valores positivos y negativos [11]. Con la forma  $f(x) = mx + b$ , sin embargo, teniendo la desventaja de utilizar una función lineal, su gradiente nunca tenderá a 0, pero para la aplicación, no es mandatorio que se tenga una pérdida de 0.

### III. METODOLOGÍA

Para el desarrollo del proyecto fue necesario aplicar la siguiente metodología (véase Figura 4).

Donde primeramente se realizaron los cálculos y transformaciones de las funciones del sistema para poder meterlas en el programa, posteriormente se creó un *dataset* con los valores del desempeño del controlador PID sintonizado, donde se pudiera tener valores estables y con poco sobrepaso. Posteriormente se estableció la red y se entrenó con los datos

del desempeño del PID, para poder probar y predecir la salida del controlador de las trayectorias requeridas. Para finalmente ser medidas las métricas de desempeño y observar que tan bien se comportaron los datos.

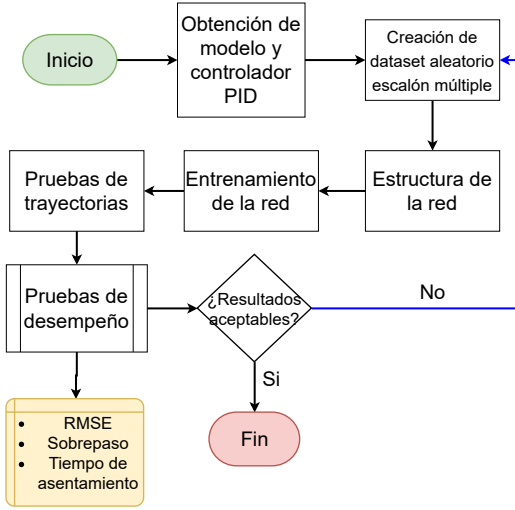


Figura 4. Metodología del trabajo.

La prueba del entrenamiento es realizado mediante el siguiente diagrama (véase Figura 5).

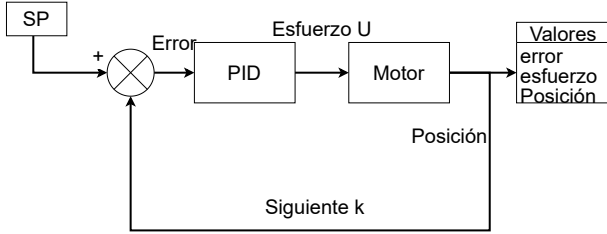


Figura 5. Obtención de valores para el entrenamiento.

Finalmente el diagrama de pruebas es como se muestra a continuación (véase Figura 6):

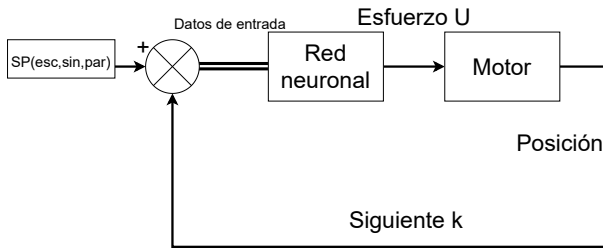


Figura 6. Obtención de valores para el entrenamiento.

#### IV. ESTRUCTURA DE LA RED

La red propuesta fue de [8,14,28,14,1] (véase Figura 7) en sus capas, teniendo las siguientes características generales.

- Entradas: 3

- Salidas: 1
- Función de activación: linear
- Optimizador: Adam
- función de coste: RMSE
- Pesos: 999
- Épocas de entrenamiento: 7

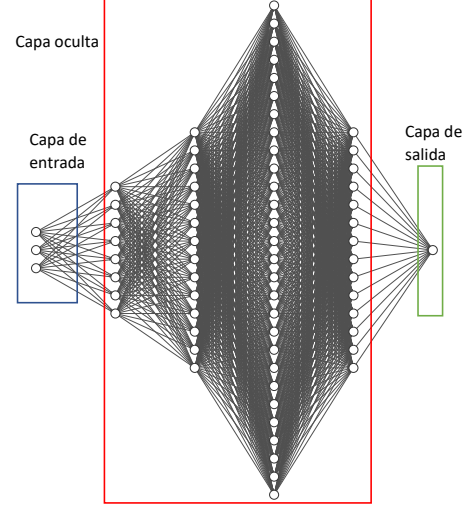


Figura 7. Obtención de valores para el entrenamiento.

teniendo como entradas y salidas:

$$x = \begin{bmatrix} e(0) & e(-1) & e(-2) \\ e(1) & e(0) & e(-1) \\ \vdots & \vdots & \vdots \\ e(k) & e(k-1) & e(k-2) \end{bmatrix} \quad (11)$$

$$y = \begin{bmatrix} out(0) \\ out(1) \\ \vdots \\ out(k) \end{bmatrix}$$

siendo los errores actuales y anteriores de hasta 2 iteraciones atrasadas.

#### V. RESULTADOS

Para el entrenamiento se trabajó con el siguiente patrón de datos (*Dataset*), el cual consta de pruebas de escalón, que al ser alcanzado cambia aleatoriamente a valores en el intervalo  $[-1.5, 1.5]$  teniendo 6000 valores de entrenamiento (véase Figura 8).

Los valores de desempeño fueron: Raíz media cuadrática, sobrepaso y tiempo de asentamiento (véase Figura 4). Teniendo las siguientes fórmulas.

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (y_i - \hat{y})^2}{n}} \quad (12)$$

$$m_p[\%] = (1 - \frac{|y_{max}|}{s_p}) * 100 \%$$

$$t_s = t(e_{min})$$

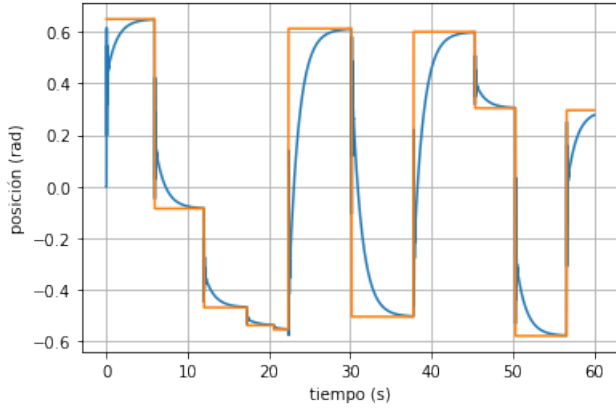


Figura 8. Valores del patrón de entrenamiento

Se realizaron 4 pruebas de trayectoria, siendo: al escalón, a función sinusoidal y prueba parabólica (sigmoidea y campana de Gauss).

#### V-A. Prueba al escalón

Esta prueba consta de seguir una trayectoria con la forma:

$$s_p(\theta) = \begin{cases} 0 & t \leq 0 \\ a & t > 0 \end{cases} \quad (13)$$

Obteniendo la Figura 10 como resultados por el controlador PID y el basado en la red neuronal Perceptrón multicapa con  $a = -1.5$ .

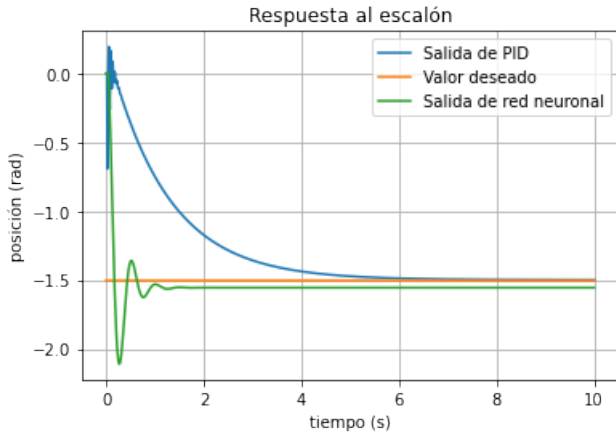


Figura 9. resultados de la red y PID en respuesta al escalon  $a = -1.5$ .

Teniendo como resultados de las métricas en la Tabla I.

Tabla I  
RESULTADOS DE DESEMPEÑO AL ESCALÓN.

Método	Valor inicial	RMSE	Sobrepaso	Tiempo de asentamiento
PID	0 rad	$2.38 \times 10^{-15}$	100%	0.06 s
Red neuronal	0 rad	$2.27 \times 10^{-15}$	140.6%	0.03 s

#### V-B. Prueba sinusoidal

La prueba consiste en la trayectoria de tipo sinusoidal de la forma:

$$s_p(\theta) = a \sin(\omega t + \alpha) \quad (14)$$

con  $\omega = 1.5$ ,  $a = 1$  y  $\alpha = 0$

Obteniendo la Figura 10 como resultados por el controlador PID y el basado en la red neuronal Perceptrón multicapa.

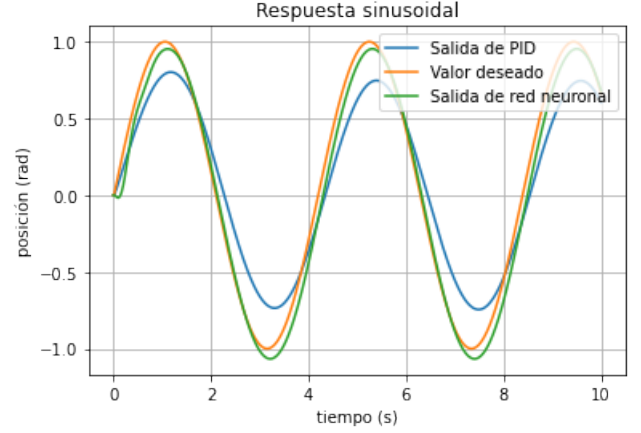


Figura 10. resultados de la red y PID en respuesta sinusoidal.

Teniendo como resultados de las métricas en la Tabla II.

Tabla II  
RESULTADOS DE DESEMPEÑO SINUSOIDAL.

Método	Valor inicial	RMSE	Sobrepaso	Tiempo de asentamiento
PID	0 rad	$1.78 \times 10^{-15}$	0%	2.76 s
Red neuronal	0 rad	$1.78 \times 10^{-15}$	17.6%	6.27 s

#### V-C. Prueba Parabólica

Esta prueba es la que más se asemeja al comportamiento de la trayectoria de una pata robótica (siendo este el principal interés del trabajo), por lo que se analiza de 2 grados de libertad de la pata.

**V-C1. Prueba sigmoidea:** La prueba consta de una función periódica, con forma sigmoidea en el intervalo de  $\xi=[0,\pi]$ , representada por la formula:

$$\theta_1 = d\phi - A\phi(\cos(\xi) - 1) \quad (15)$$

Con  $d\phi = 350$ ,  $A\phi = 350$ , obteniendo los resultados en la Figura 11.

Las pruebas de desempeño se muestran en la Tabla III.

Tabla III  
RESULTADOS DE DESEMPEÑO PARABÓLICO/SIGMOIDEO.

Método	Valor inicial	RMSE	Sobrepaso	Tiempo de asentamiento
PID	6 rad	$3.94 \times 10^{-15}$	600%	3.94 s
Red neuronal	6 rad	$6.96 \times 10^{-15}$	600%	0.29 s

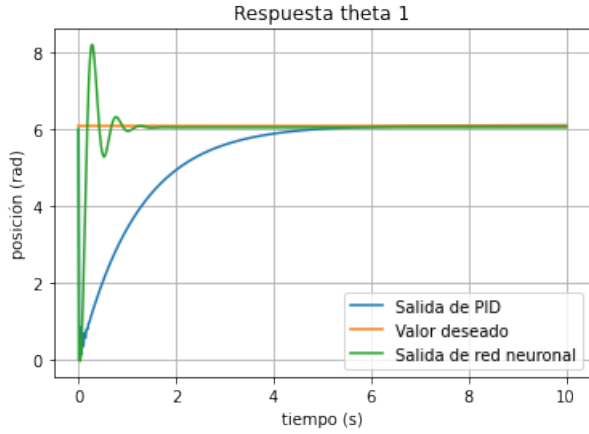


Figura 11. resultados de la red y PID en respuesta parabólica/sigmoidea.

V-C2. *Prueba Gaussiana*: La prueba consta de una función periódica, con forma sigmoidea en el intervalo de  $\xi=[0,\pi]$ , representada por la formula:

$$\theta_{2,3} = d\chi - A\chi(\cos(\xi) - 1)e^{k_1\xi} \quad (16)$$

con  $d\chi = 40$ ,  $A\chi = 1$  y  $k_1$ :

$$k_1 = \frac{1}{c^2\sqrt{2\pi}}e^{-\frac{(\xi-b)^2}{2c^2}} \quad (17)$$

$$c = 0.6$$

$$b = 1.2$$

Se obtuvieron los siguientes resultados (véase Figura 12).

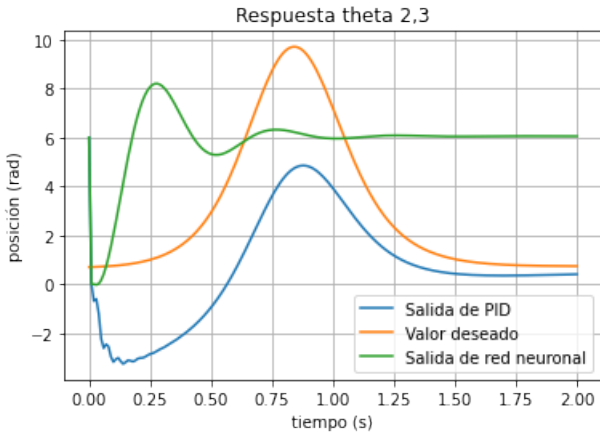


Figura 12. resultados de la red y PID en respuesta parabólica/Gaussiana.

Teniendo como resultados en las métricas de desempeño mostradas en la Tabla IV

Tabla IV  
RESULTADOS DE DESEMPEÑO PARABÓLICO/SIGMOIDEO.

Método	Valor inicial	RMSE	Sobrepaso	Tiempo de asentamiento
PID	6 rad	$2.53 \times 10^{-15}$	0%	0.29 s
Red neuronal	6 rad	$3.75 \times 10^{-15}$	800%	0.01 s

## VI. CONCLUSIONES

En este trabajo se muestra el desempeño de una red neuronal en un sistema de control aplicado a controlar trayectorias de posición, en donde se realizaron normalizaciones y cálculos de saturación para poder hacer más realista el problema, incrementando las no linealidades (desde un punto de vista del control) del sistema total.

La red neuronal tiene un desempeño bastante decente, tomando en cuenta que únicamente aprendió de funciones escalón y de un controlador PID sintonizado manualmente, que tenía una respuesta relativamente lenta y no era capaz de seguir de una manera tan satisfactoria la función sinusoidal.

Se observa en la prueba del escalón que llega mucho más rápido acercándose al objetivo, a cambio de tener sobrepasos y oscilaciones, así como también un error de estado estacionario.

En la prueba sinusoidal, realiza de manera muy aceptable el seguimiento de la trayectoria, con un pequeño desfase y un sobrepaso muy bajo. Caso contrario al controlador PID.

Finalmente en la prueba parabólica/sigmoidea, se observa que a pesar de encontrarse relativamente cerca del resultado, se obtuvo una respuesta brusca, oscilatoria y con error de estado estacionario.

Por otro lado, en la prueba parabólica/gaussiana, se observa que fue imposible para el sistema poder seguir la trayectoria requerida, y por lo tanto su desempeño fue insuficiente.

Para trabajos futuros, se propone implementar el modelo completo de la pata robótica y utilizar otro tipo de configuraciones para la red, más adecuadas para el sistema de control, tanto optimizadores, funciones de pérdida, funciones de activación (subir el grado de la función), etc.

Adicionalmente se observó que al agregar más parámetros de entrada como de salida y los esfuerzos tanto actuales como anteriores, se tenía un desempeño deplorable, por lo que al observar que únicamente con los errores actuales y anteriores es posible realizar control sobre el sistema en algunos casos. Esto implica que será un reto nuevo encontrar los valores que más puedan permitir un mejor aprendizaje de la red y su uso en trayectorias múltiples.

## REFERENCIAS

- 1 Nimbalkar, R., Lavhate, S., Lawande, V., and Toradmal, N., "A review on significance of pid controller for speed control of dc motor," *International Journal IJRITCC*, vol. 3, no. 4.
- 2 "Statistical control of multivariate processes through the artificial neural network multilayer perceptron and the mewma graphic analysis — iee latin america transactions," <https://latam.ieee9.org/index.php/transactions/article/view/1284>, (Accessed on 06/09/2022).
- 3 V.M. Hernández-Guzmán and R. Silva-Ortigoza and R.V. Carrillo Serrano, *Control Automático: Teoría de diseño, Construcción de Prototipos, Modelado, Identificación y Pruebas Experimentales*. Colección CIDETEC del Instituto Politécnico Nacional, 2013.
- 4 V.M. Hernández-Guzmán and R. Silva-Ortigoza, *Automatic Control with Experiments*. Springer, 2019.
- 5 y M. Martinez, A. A., *Identificación y control adaptativo*. Prentice Hall, 2003. [Online]. Available: <http://books.google.com/books?id=W-xMPgAACAAJ>
- 6 Mazzone, V., "Controladores PID," <https://www-eng.newcastle.edu.au/~jhb519/teaching/caut1/Apuntes/PID.pdf>, 2002.

- 7 M. Nasreldin, A. Abdalgfar, A. Alfatih y A. Abdelrahman, "IMPLEMENTATION OF A PID CONTROL SYSTEM ON MICROCONTROLLER (DC MOTOR CASE STUDY)," *International Conference on Communication, Control, Computing and Electronics Engineering*, 2017.
- 8 "Perceptrón multicapa - red neuronal - diego calvo," <https://www.diegocalvo.es/perceptron-multicapa/>, (Accessed on 09/12/2022).
- 9 "How to choose loss functions when training deep learning neural networks," <https://machinelearningmastery.com/how-to-choose-loss-functions-when-training-deep-learning-neural-networks/>, (Accessed on 10/09/2022).
- 10 "Adam — latest trends in deep learning optimization. — by vitaly bushaev — towards data science," <https://towardsdatascience.com/adam-latest-trends-in-deep-learning-optimization-6be9a291375c>, (Accessed on 10/09/2022).
- 11 "Activation functions — fundamentals of deep learning," <https://www.analyticsvidhya.com/blog/2020/01/fundamentals-deep-learning-activation-functions-when-to-use-them/>, (Accessed on 06/09/2022).