

WaveNet-IIR PID Controllers Applied to a 2DoF Helicopter Model

O. F. Garcia-Castro* M. A. Vega-Navarrete**
L. E. Ramos-Velasco** R. Garcia-Rodriguez**
C. R. Domínguez-Mayorga* E. Escamilla-Hernández***
L. N. Oliva-Moreno****

* Postgraduate Program of Aerospace Engineering
(e-mails: {213220003, cdominguez}@upmh.edu.mx)

** Metropolitan Polytechnic University of Hidalgo, Aeronautical Engineering Program. Tolcayuca, Hidalgo, Mexico (e-mails: {mvega, lramos, rogarcia}@upmh.edu.mx)

*** Escuela Superior de Ingeniería Mecánica, Campus Culhuacán, Instituto Politécnico Nacional (e-mail: eescamillah@ipn.mx)

**** Unidad Profesional Interdisciplinaria de Ingeniería Campus Hidalgo, Instituto Politécnico Nacional (e-mail: loliva@ipn.mx)

Abstract—This article presents a WaveNet-IIR PID controller applied to Quanser two degree of freedom helicopter. The Wavenet-IIR is based on a radially based neural network with wavelet functions as function activations. At the same time, a pair of infinite impulse response filters are used to prune some neuronal networks. Furthermore, an auto-tuning method is presented for PID control gains of the MIMO system. Finally, simulation results, using LabVIEW, are shown under different conditions.

Keywords: helicopter, WaveNet, PID controller, Quanser, identification

1. INTRODUCCIÓN

El diseño de controladores libres de modelo son de gran utilidad para aplicaciones donde se desconoce la dinámica exacta de un sistema. Uno de los más conocidos y utilizados es el controlador proporcional-integral-derivativo (PID), ya que se caracteriza por asegurar un buen desempeño estático, especialmente para sistemas lineales e invariantes en el tiempo; sin embargo, padecen de un bajo rendimiento en sistemas no lineales variables e inciertos. Pese a lo anterior, los controladores PID también son de gran relevancia en el control de aeronaves, pues se requiere de robustez para mantener controlado alguno de los tres ejes de la aeronave: longitudinal, vertical y transversal (Rouis et al., 2021).

Para mejorar la respuesta dinámica en diferentes condiciones de operación de un sistema y, así, compensar ciertas desventajas, se han propuesto esquemas de control PID que integren alguna técnica adicional, p. ej.: modos deslizantes o control inteligente por medio de redes neuronales y/o lógica difusa. Además, en los últimos años se han presentado enfoques de auto-sintonización de ganancias para este tipo de controladores (Zou et al., 2022; Ouyang, 2019; Zhao, He, Yang y Li, 2022; Emamia et al., 2018; Lui y Ye, 2019).

Entonces, la aportación de este trabajo es presentar un algoritmo de control neuronal para la guiñada y el cabeceo de un helicóptero de 2 grados de libertad (GDL), sistema MIMO; utilizando controladores PID discretos donde una red neuronal se encarga de ajustar las ganancias. Dicha red está formada por una capa interna donde sus funciones de activación son definidas por *wavelets*¹, a esta configuración se le denomina WaveNet (García-Castro, 2020; Ramos-Velasco et al., 2016). A la salida de la red se tiene un conjunto de filtros filtro de respuesta de impulso infinito IIR, los cuales se encargan de podar a las neuronas con poca actividad. Cabe mencionar que la WaveNet tiene dos funciones: identificar la dinámica del sistema y auto-sintonizar las ganancias de los controladores PID discretos.

Para la selección de la *wavelet* se realizará un estudio comparativo de acuerdo a su respuesta transitoria y estacionaria utilizando el modelo lineal del helicóptero *Quanser* de 2 GDL. Finalmente, diversas simulaciones bajo diferentes condiciones de operación se presentan y una parte de la identificación se emplea para ajustar las ganancias p , i y d de los controladores PID discretos

¹ También llamadas onditas u onduletas, y se caracterizan por trabajar en el tiempo y la frecuencia de manera simultánea.

El resto del artículo está organizado de la siguiente manera: en la Sección 2 se exponen los algoritmos de redes neuronales directamente sobre el helicóptero; en la Sección 3 se muestran los resultados obtenidos de las simulaciones numéricas en lazo cerrado; y en la Sección 4 se dan las conclusiones.

2. CONTROLADOR PID WAVENET-IIR PARA EL HELICÓPTERO QUANSER

El esquema de control PID *WaveNet-IIR* se muestra en la Figura 1, en él se aprecian básicamente cinco bloques: *i)* Helicóptero, donde su dinámica es representada por ecuaciones diferenciales lineales; *ii)* *WaveNet-IIR*, en este bloque se realiza la identificación del modelo lineal; *iii)* Auto sintonización donde se ajustan las ganancias de los controladores PID cada periodo de muestreo; finalmente, se tiene a los *iv)* Controladores del tipo PID discreto.

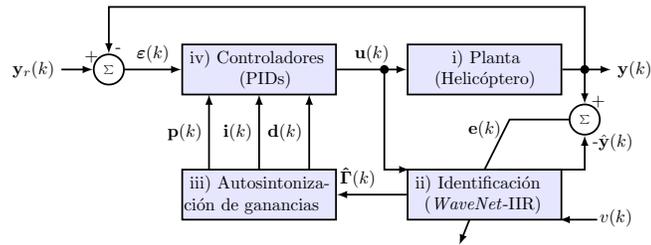


Figura 1. Diagrama a bloques del controlador PID *WaveNet-IIR* para el helicóptero *Quanser* de 2 GDL.

A continuación se describe de manera detallada el contenido de todos los bloques.

i) Planta (Helicóptero). Este bloque hace referencia al helicóptero de la empresa *Quanser* mostrado en la Figura 2 y algunos de sus parámetros son descritos en la Tabla 1 (*Quanser Inc.*, 2012).



Figura 2. Modelo experimental del helicóptero *Quanser* de 2 GDL y su diagrama de cuerpo libre.

Para llevar a cabo la simulación de la dinámica del helicóptero se utilizaron sus funciones de transferencia (*Ouyang*, 2019), las cuales son

$$H_{\theta\theta}(s) = \frac{y_{\theta}(s)}{V_{\theta}(s)} = \frac{2.361}{s^2 + 9.26s} \quad (1)$$

$$H_{\phi\phi}(s) = \frac{y_{\phi}(s)}{V_{\phi}(s)} = \frac{0.2402}{s^2 + 3.487} \quad (2)$$

Tabla 1. Parámetros del modelo del helicóptero *Quanser* de 2 GDL.

Símbolo	Descripción	Valor
K_{pp} ,	Constante de torque debido a la	0.2040 N m/V
K_{py} ,	fuerza de empuje sobre los ejes	0.0068 N m/V
K_{yy} ,	de cabeceo y guiñada desde las	0.0720 N m/V
K_{yp}	hélices de cabeceo y guiñada,	0.0219 N m/V
	respectivamente.	
V_{θ} , V_{ϕ}	Voltaje en el motor de cabeceo y guiñada, respectivamente.	± 24 V y ± 15 V

$$H_{\theta\phi}(s) = \frac{y_{\theta}(s)}{V_{\phi}(s)} = \frac{0.07871}{s^2 + 9.26s} \quad (3)$$

$$H_{\phi\theta}(s) = \frac{y_{\phi}(s)}{V_{\theta}(s)} = \frac{0.7895}{s^2 + 3.487} \quad (4)$$

donde y_{θ} y y_{ϕ} son las salidas del modelo para cada eje de movimiento; se aprecia cómo el modelo es marginalmente estable, es decir, hay polos en el eje imaginario. Además, en la Figura 3 se aprecia que el modelo está acoplado.

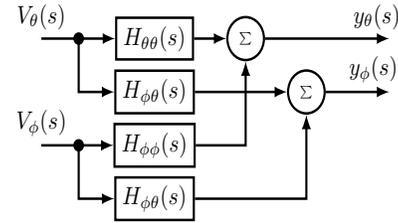


Figura 3. Diagrama de bloques basado en las funciones de transferencia.

Adicionalmente, el helicóptero posee un bloque donde se realiza la interpretación eléctrica de la señal de control $\mathbf{u} = [u_{\theta} \ u_{\phi}]^T$, es decir la relación mecánica-eléctrica:

$$\mathbf{u} = \begin{bmatrix} K_{pp}V_{\theta} + K_{py}V_{\phi} \\ K_{yy}V_{\phi} + K_{yp}V_{\theta} \end{bmatrix}, \quad (5)$$

resolviendo para V_{θ} y V_{ϕ} y sustituyendo los valores de la Tabla 1 se obtiene lo siguiente

$$V_{\theta} = 0.4899u_{\theta} - 0.7387u_{\phi} \quad (6)$$

$$V_{\phi} = 0.0793u_{\phi} - 0.7387u_{\theta}. \quad (7)$$

donde u_{θ} y u_{ϕ} son las señales de control.

ii) Identificación (WaveNet-IIR). En este bloque se identifica la dinámica entrada-salida del sistema y su estructura se muestra en la Figura 4, la cual está compuesta por una red neuronal de base radial (*Haykin*, 1999) cuyas funciones de activación son *wavelets* “hijas”.

Asimismo, en la Figura 5 se observa la configuración interna de los bloques de filtros IIR donde sus salidas son las posiciones del helicóptero $\hat{\mathbf{y}} = [\hat{y}_{\theta}, \hat{y}_{\phi}]$. Estos filtros “podan” a las neuronas con poca contribución en el proceso de identificación y poseen la estructura de un sistema no lineal (*Carillo-Santos et al.*, 2018).

La función $\Psi(\tau)$ se llama *wavelet* “madre” porque a partir de ella se generan otras funciones al establecer valores que escalen a la señal y también la trasladen sobre el eje

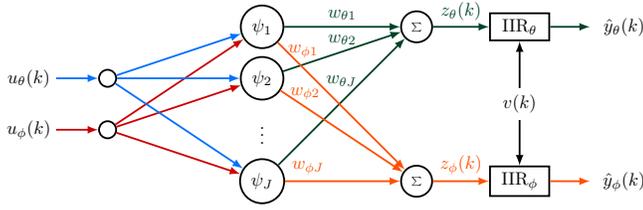


Figura 4. Diagrama de la red neuronal *WaveNet* con filtros IIR.

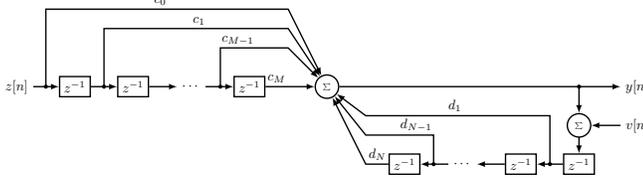


Figura 5. Filtro IIR donde M y N representan el número de coeficientes de adelanto y retraso, respect.

del tiempo. Estas nuevas funciones se nombran *wavelets* “hijas” y son dadas por (García-Castro, 2020)

$$\psi_{qs}(\tau) = \frac{1}{\sqrt{s}} \psi(\tau), \quad \tau(k, q, s) = \left(\frac{k-q}{s} \right), \quad s \leq 0 \quad (8)$$

donde q y s trasladan y escalan a la señal, respect.; k indica el instante de la operación y τ es una variable discreta en el tiempo.

En este artículo se emplean las *wavelets* “madre” de la Tabla 2 (Domínguez-Mayorga et al., 2012) para obtener las *wavelets* “hijas”. Por ejemplo, la i -ésima *wavelet* “hija” de la función *Polywog2* está dada por

$$\psi_i(\tau_i) = (\tau_i^3 - 3\tau_i)e^{-0.5\tau_i^2}, \quad \tau_i = \left(\frac{k-q_i}{s_i} \right), \quad (9)$$

donde $i \in \{1, 2, \dots, J\}$ siendo J el número de *wavelets*. Además, la derivada parcial de esta función con respecto a la variable de dilatación s es

$$\frac{\partial \psi_i(\tau_i)}{\partial s_i} = \frac{1}{s_i} (\tau_i^4 - 6\tau_i^2 + 3) e^{-0.5\tau_i^2}. \quad (10)$$

Tabla 2. *Wavelets* usadas en este artículo.

Nombre	Expresión
<i>Morlet</i>	$\cos(w_0\tau)e^{-0.5\tau^2}$
<i>Rasp1</i>	$\tau(\tau^2 + 1)^{-2}$
<i>Rasp2</i>	$\tau \cos \tau (\tau^2 + 1)^{-1}$
<i>Rasp3</i>	$\sin \pi \tau (\tau^2 - 1)^{-1}$
<i>Polywog1</i>	$\tau e^{-0.5\tau^2}$
<i>Polywog2</i>	$(\tau^3 - 3\tau)e^{-0.5\tau^2}$
<i>Polywog3</i>	$(\tau^4 - 6\tau^2 - 3)\tau e^{-0.5\tau^2}$
<i>Polywog4</i>	$(1 - \tau^2)e^{-0.5\tau^2}$
<i>Polywog5</i>	$(3\tau^2 - \tau^4)e^{-0.5\tau^2}$
<i>Shannon</i>	$(\sin 2\pi\tau - \sin \pi\tau)(\pi\tau)^{-1}$
<i>Slog1</i>	$\sum_a (1 + e^{a-\tau})^{-1} - \sum_b (1 + e^{b-\tau})^{-1}$
<i>Slog2</i>	$\sum_a 3(1 + e^{a-\tau})^{-1} - \sum_b (1 + e^{b-\tau})^{-1}$
	$a \in \{-1, 1\}, b \in \{-3, 3\}$

Los parámetros de la red neuronal y los filtros IIR en su forma matricial son: la señal de control $\mathbf{u} = [u_\theta, u_\phi]^T$,

el parámetro de traslación $\mathbf{q} = [q_1, q_2, \dots, q_J]^T$, el parámetro de dilatación $\mathbf{s} = [s_1, s_2, \dots, s_J]^T$, las *wavelets* “hijas” $\boldsymbol{\psi} = [\psi_1, \psi_2, \dots, \psi_J]^T$, la directriz de la red neuronal $\hat{\mathbf{\Gamma}} = [\hat{\Gamma}_\theta, \hat{\Gamma}_\phi]^T$, la salida de la red neuronal $\mathbf{z} = [z_\theta, z_\phi]^T$, la posición estimada $\hat{\mathbf{y}} = [\hat{y}_\theta, \hat{y}_\phi]^T$, y las matrices de los pesos sinápticos $\mathbf{W} \in \mathbb{R}^{2 \times J}$; y los coeficientes $\mathbf{C} \in \mathbb{R}^{2 \times M}$ y $\mathbf{D} \in \mathbb{R}^{2 \times N}$ para los filtros son:

$$\mathbf{W} = \begin{bmatrix} w_{\theta 1} & w_{\theta 2} & \dots & w_{\theta J} \\ w_{\phi 1} & w_{\phi 2} & \dots & w_{\phi J} \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} c_{\theta 0} & c_{\theta 1} & \dots & c_{\theta M} \\ c_{\phi 0} & c_{\phi 1} & \dots & c_{\phi M} \end{bmatrix},$$

$$\mathbf{D} = \begin{bmatrix} d_{\theta 1} & d_{\theta 2} & \dots & d_{\theta N} \\ d_{\phi 1} & d_{\phi 2} & \dots & d_{\phi N} \end{bmatrix}. \quad (11)$$

La salida $\mathbf{z}(k)$ de la *WaveNet* está dada por

$$\mathbf{z}(k) = \mathbf{u}^T(k) \mathbf{W}(k) \boldsymbol{\psi}^T(k), \quad (12)$$

la cual pasa a través de los filtros IIR para obtener la posición estimada $\hat{\mathbf{y}}(k)$, la cual es

$$\hat{\mathbf{y}}(k) = \hat{\mathbf{\Gamma}}(k) + \mathbf{D} \hat{\mathbf{Y}}(k) \mathbf{v}(k) \quad (13)$$

donde

$$\hat{\mathbf{\Gamma}}(k) = \mathbf{C} \mathbf{z}(k), \quad (14)$$

$$\hat{\mathbf{Y}}(k) = \begin{bmatrix} \hat{y}_\theta(k-1) & \hat{y}_\theta(k-2) & \dots & \hat{y}_\theta(k-N) \\ \hat{y}_\phi(k-1) & \hat{y}_\phi(k-2) & \dots & \hat{y}_\phi(k-N) \end{bmatrix} \quad (15)$$

y $\mathbf{v}(k) = [v(k) \ v(k)]^T$ es la señal persistente de los filtros.

La optimización de los parámetros descritos se lleva a cabo utilizando el algoritmo de retro-propagación donde la función de costo $\mathbf{E}(k)$ se define como

$$\mathbf{E}(k) = \frac{1}{2} \sum_t [\mathbf{e}_t(k)]^2 \quad (16)$$

para la cual se requiere el error de estimación $\mathbf{e}_t(k) = \mathbf{y}_t(k) - \hat{\mathbf{y}}_t(k)$, donde \mathbf{y}_t indica la salida del sistema y $\hat{\mathbf{y}}_t$ es la salida estimada de la *WaveNet*-IIR, $t = \{\theta, \phi\}$. Para minimizar la función de costo $\mathbf{E}(k)$, se aplica el método del gradiente descendente utilizando las siguientes derivadas

$$\frac{\partial \mathbf{E}}{\partial \mathbf{W}(k)} = \mathbf{u}(k) \mathbf{e}^T(k) \mathbf{C}(k) \boldsymbol{\psi}_a^T(\tau) = \mathbf{U} \mathbf{I}_e (\mathbf{C} \boldsymbol{\psi}_a)^T \quad (17)$$

$$\frac{\partial \mathbf{E}}{\partial \mathbf{q}(k)} = -\mathbf{e}(k) \mathbf{u}(k) \mathbf{C}^T(k) \boldsymbol{\psi}_b = \mathbf{U} (\mathbf{I}_f (\mathbf{C} \boldsymbol{\psi}_b)^T) \quad (18)$$

$$\frac{\partial \mathbf{E}}{\partial \mathbf{s}(k)} = \tau \frac{\partial \mathbf{E}}{\partial \mathbf{q}(k)} = \frac{\partial \mathbf{E}}{\partial \mathbf{q}(k)} \odot \boldsymbol{\tau} \quad (19)$$

$$\frac{\partial \mathbf{E}}{\partial \mathbf{C}(k)} = -\mathbf{u}(k) \mathbf{e}^T(k) \mathbf{z}(k-M) = \mathbf{U} \mathbf{I}_e \mathbf{z}^T \quad (20)$$

$$\frac{\partial \mathbf{E}}{\partial \mathbf{D}(k)} = -\mathbf{v}(k) \mathbf{e}^T(k) \hat{\mathbf{y}}(k-N) = \mathbf{v} \mathbf{I}_e (\hat{\mathbf{y}} \boldsymbol{\psi}_a)^T \quad (21)$$

donde

$$\mathbf{U} = u_\theta(k) + u_\phi(k), \quad \mathbf{I}_e = \text{diag}\{e_\theta(k), e_\phi(k)\}, \quad (22)$$

$$\mathbf{I}_f = [e_\theta(k) \ e_\phi(k)]^T, \quad \boldsymbol{\tau} = [\tau_1 \ \tau_2 \ \dots \ \tau_J] \quad (23)$$

$$\boldsymbol{\psi}_a = \begin{bmatrix} \psi_1(\tau_1) & \psi_2(\tau_2) & \dots & \psi_J(\tau_J) \\ \psi_1(\tau_1 - 1) & \psi_2(\tau_2 - 1) & \dots & \psi_J(\tau_J - 1) \\ \vdots & \dots & \ddots & \vdots \\ \psi_1(\tau_1 - M) & \psi_2(\tau_2 - M) & \dots & \psi_J(\tau_J - M) \end{bmatrix} \quad (24)$$

$$\boldsymbol{\psi}_b = \begin{bmatrix} \frac{\partial \psi_1(\tau_1)}{\partial s_1} & \dots & \frac{\partial \psi_J(\tau_J)}{\partial s_J} \\ \frac{\partial \psi_1(\tau_1 - 1)}{\partial s_1} & \dots & \frac{\partial \psi_J(\tau_J - 1)}{\partial s_J} \\ \vdots & \ddots & \vdots \\ \frac{\partial \psi_1(\tau_1 - M)}{\partial s_1} & \dots & \frac{\partial \psi_J(\tau_J - M)}{\partial s_J} \end{bmatrix}. \quad (25)$$

La actualización de los parámetros cumple con una regla:

$$\boldsymbol{\lambda}(k+1) = \boldsymbol{\lambda}(k) + \mu_\lambda \Delta \boldsymbol{\lambda}(k), \quad \Delta \boldsymbol{\lambda}(k) = -\frac{\partial \mathbf{E}}{\partial \boldsymbol{\lambda}(k)} \quad (26)$$

donde $\boldsymbol{\lambda}$ es alguno de los parámetros \mathbf{C} , \mathbf{D} , \mathbf{W} , \mathbf{q} y \mathbf{s} , con su correspondiente tasa de aprendizaje μ_λ .

Para fines de programación en la plataforma LabVIEW 2013 se utilizó el tercer elemento de cada una de las ecuaciones (17)–(21) donde \odot representa el operador Hadamard². Además $k \in \{1, 2, \dots, K\}$ donde K es el número de épocas³. Cuando se hace la identificación del modelo fuera de línea, el diseñador propone un valor para K ; en la identificación en línea, $K = 1$.

iii) Auto sintonización de las ganancias. Este bloque se encarga de ajustar las ganancias del controlador $\mathbf{p}(k) = [p_\theta \ p_\phi]^T$, $\mathbf{i}(k) = [i_\theta \ i_\phi]^T$ y $\mathbf{d}(k) = [d_\theta \ d_\phi]^T$, lo cual se realiza de la siguiente manera

$$\mathbf{p}(k) = \mathbf{p}(k-1) + \mu_p \mathbf{e}(k) \hat{\Gamma}(k) [\boldsymbol{\varepsilon}(k) - \boldsymbol{\varepsilon}(k-1)] \quad (27)$$

$$\mathbf{i}(k) = \mathbf{i}(k-1) + \mu_i \mathbf{e}(k) \hat{\Gamma}(k) \boldsymbol{\varepsilon}(k) \quad (28)$$

$$\mathbf{d}(k) = \mathbf{d}(k-1) + \mu_d \mathbf{e}(k) \hat{\Gamma}(k) [\boldsymbol{\varepsilon}(k) - 2\boldsymbol{\varepsilon}(k-1) + \boldsymbol{\varepsilon}(k-2)] \quad (29)$$

donde $\boldsymbol{\mu}$ es la tasa de aprendizaje y $\hat{\Gamma}(k)$ es la directriz de la red neuronal dada por (14).

iv) Controladores (PIDs). En este bloque se genera la señal de control $\mathbf{u}(k)$ la cual varía la salida del sistema $\mathbf{y}(k)$. Dicha etapa solo necesita los valores de las ganancias $\mathbf{p}(k)$, $\mathbf{i}(k)$ y $\mathbf{d}(k)$ así como el error de seguimiento $\boldsymbol{\varepsilon}(k) = \mathbf{y}_r - \mathbf{y}$, donde $\mathbf{y}_r(k)$ es la posición de referencia y $\mathbf{y}(k)$ el valor real de la planta. Entonces, la señal de control $\mathbf{u}(k+1)$ está dada por

$$\mathbf{u}(k+1) = \mathbf{u}(k) + \mathbf{p}(k) [\boldsymbol{\varepsilon}(k) - \boldsymbol{\varepsilon}(k-1)] + \mathbf{i}(k) [\boldsymbol{\varepsilon}(k) + \mathbf{d}(k) [\boldsymbol{\varepsilon}(k) - 2\boldsymbol{\varepsilon}(k-1) + \boldsymbol{\varepsilon}(k-2)]] \quad (30)$$

recordando que $\boldsymbol{\varepsilon}(k)$ es el error de seguimiento.

v) Convergencia del algoritmo WaveNet-IIR Debido al espacio limitado que se tiene, solo se presenta el teorema que establece una condición suficiente de la convergencia del algoritmo WaveNet-IIR mediante la selección de una tasa de aprendizaje adecuada y una idea general de la demostración, (Díaz-López et al., 2013).

² Sean los vectores $a, b, c \in \mathbb{R}^{n \times 1}$ donde $c_x = a_x \cdot b_x$ con $x \in \{1, 2, \dots, n\}$ (Deisenroth et al., 2020).

³ El algoritmo se realiza K veces en un periodo de muestreo determinado.

Teorema 1. La convergencia del error de identificación $\mathbf{e}_t(k)$ y de los parámetros $\boldsymbol{\lambda}$ se garantiza, si la tasa de aprendizaje de los parámetros de la WaveNet-IIR, μ_λ , satisface

$$\mu_\lambda = \frac{\mu}{1 + \varphi(k)^T \varphi(k)} \quad (0 < \mu \leq 1) \quad (31)$$

con

$$\varphi(k) = \frac{\partial \hat{\mathbf{y}}}{\partial \boldsymbol{\lambda}} = \begin{bmatrix} \frac{\partial \hat{y}}{\partial s_1} & \dots & \frac{\partial \hat{y}}{\partial s_J} & \frac{\partial \hat{y}}{\partial q_1} & \dots & \frac{\partial \hat{y}}{\partial q_J} & \frac{\partial \hat{y}}{\partial w_1} \\ \dots & \frac{\partial \hat{y}}{\partial w_J} & \frac{\partial \hat{y}}{\partial c_0} & \dots & \frac{\partial \hat{y}}{\partial c_M} & \frac{\partial \hat{y}}{\partial d_1} & \dots & \frac{\partial \hat{y}}{\partial d_N} \end{bmatrix}. \quad (32)$$

Prueba. Sea la función candidata de Lyapunov definida como

$$V(k) = \frac{1}{2} (\mathbf{y}_t(k) - \hat{\mathbf{y}}_t(k))^2 = \frac{1}{2} \mathbf{e}_t^2(k) \quad (33)$$

Si $\Delta V(k) = V(k+1) - V(k)$ tenemos que

$$\begin{aligned} \Delta V(k) &= V(k+1) - V(k) = \frac{1}{2} (\mathbf{e}_t^2(k+1) - \mathbf{e}_t^2(k)) \\ &= \frac{1}{2} \Delta \mathbf{e}_t(k) (2\mathbf{e}_t(k) + \Delta \mathbf{e}_t(k)). \end{aligned}$$

Asumiendo que $\Delta \mathbf{e}_t(k) \approx \left[\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right]^T \Delta \boldsymbol{\lambda}(k)$ para un periodo de muestro rápido tenemos

$$\begin{aligned} \Delta V(k) &= \frac{1}{2} \left[\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right]^T \Delta \boldsymbol{\lambda}(k) \\ &\quad \left(\left[\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right]^T \Delta \boldsymbol{\lambda}(k) + 2\mathbf{e}_t(k) \right) \\ &= \left[\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right]^T \mu_\lambda \mathbf{e}_t(k) \left(-\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right) \\ &\quad \left(\left[\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right]^T 2\mu_\lambda \mathbf{e}_t(k) \left(-\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right) + 2\mathbf{e}_t(k) \right) \\ &= -2\mu_\lambda \left(\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right)^2 \mathbf{e}_t^2(k) \left[1 - \mu_\lambda \left(\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(k)} \right)^2 \right]. \end{aligned}$$

Usando (31) tenemos que $\Delta V(k) = -\gamma \mathbf{e}_t^2(kT) < 0$ con $\gamma = 2\mu_\lambda \left(\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(kT)} \right)^2 \left[1 - \mu_\lambda \left(\frac{\partial \mathbf{e}_t(k)}{\partial \boldsymbol{\lambda}(kT)} \right)^2 \right]$.

De tal forma, (26) minimiza la función de coste (16) a lo largo de las variaciones de los parámetros y coeficientes, es decir, la convergencia asintótica global al equilibrio único $\mathbf{e}_t(k) = 0$ i.e. $\mathbf{y}_t(k) \rightarrow \hat{\mathbf{y}}_t(k)$ cuando $k \rightarrow \infty$. Obsérvese que bajo un ajuste adecuado de μ_λ , (30) se convierte en $\Delta V(k) < \alpha_0 V(k)$, para algún $\alpha_0 > 0$ que garantice la estabilidad exponencial.

3. RESULTADOS EN SIMULACIÓN NUMÉRICA

Para comparar el rendimiento entre un PID clásico y el PID WaveNet-IIR propuesto en este artículo, se implementaron ambos esquemas de control en la plataforma

de LabVIEW 2013 aplicándolos al modelo lineal del helicóptero *Quanser* de 2 GDL.

3.1 PID Clásico

Primero se pone en marcha un controlador PID clásico para el seguimiento de las trayectorias de cada eje definidas como

$$y_{r_\theta} = \begin{cases} 14 \sin(\pi k/250), & 0 \leq k < 2.5 \\ 15, & 2.5 \leq k < 5.0 \\ -15, & 5.0 \leq k < 7.5 \\ 8 \sin(\pi k/250) + 2 \sin(\pi k/320) + 3 \sin(\pi k/100), & 7.5 \leq k < 10 \end{cases} \quad (34)$$

$$y_{r_\phi} = \begin{cases} 40 \sin(2\pi k/1000), & 0 \leq k < 5 \\ 40 \sin(2\pi k/1000) + 2.5 \sin(2\pi k/100), & 5 \leq k < 10 \end{cases} \quad (35)$$

El control PID clásico es incapaz de seguir las trayectorias deseadas de forma aceptable, como se muestra en la Figura 6. Aunque en la Figura 7 se aprecia cómo las señales de control permanecen acotadas.

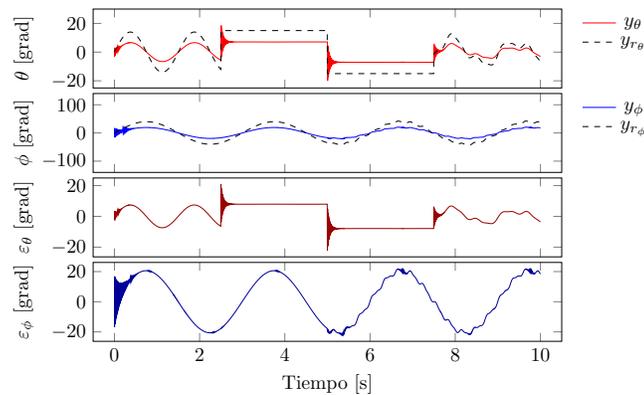


Figura 6. Control en lazo cerrado utilizando un PID clásico y sus errores de seguimiento. Las ganancias fijas son dadas por el fabricante y para el controlador PID del eje de cabeceo las ganancias fijas fueron: $p = 10$, $i = 4$ y $d = 5$; para la guiñada: $p = 10$, $i = 1.75$ y $d = 7$.

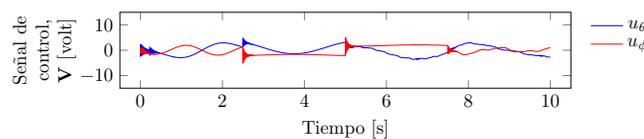


Figura 7. Comportamiento de la señal de control $u(k)$ generada por el controlador PID clásico.

3.2 PID WaveNet-IIR

Ahora la *WaveNet-IIR* se encarga de identificar la dinámica de la planta utilizando las *wavelets* de la Tabla 2. Basándose en el análisis de respuestas transitorias y estacionarias dadas por el error de aproximación $e(k)$ (v.

Figura 8), se seleccionó a la que posee el mejor comportamiento: la *Polywog2*; ya que en estado estacionario no presenta oscilaciones. La Tabla 3 indica los parámetros de la *WaveNet-IIR* para la *wavelet* seleccionada.

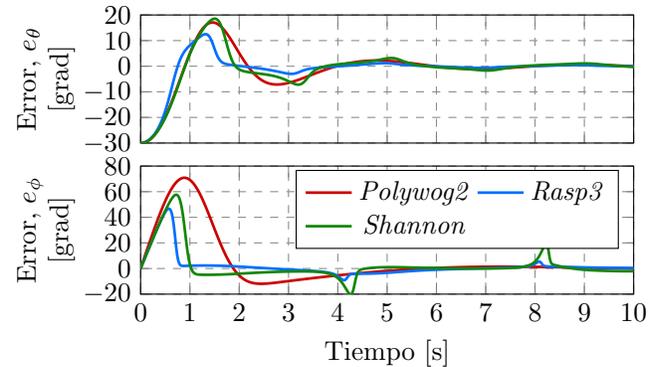


Figura 8. Error de aproximación $e_t(k)$ en cada eje.

Tabla 3. Configuración de la *WaveNet-IIR* para la función *Polywog2*.

Parámetro	Valor	Tasa de aprendizaje	
		μ_W	Valor
Neuronas, J	3	μ_q	1.0E-17
Épocas, K	4	μ_s	1.0E-17
Retrasos, M	4	μ_C	9.0E-13
Adelantos, N	3	μ_D	5.0E-12
Señal persistente, v	0.2		

Tras el entrenamiento realizado por la *WaveNet-IIR* se obtienen sus valores iniciales de operación, dados en la Tabla 4. Cabe mencionar que después del entrenamiento algunos de los parámetros se mantienen constantes, esto es propio de la teoría *WaveNet* (Carillo-Santos et al., 2018).

Tabla 4. Valores iniciales para los parámetros de la *WaveNet-IIR*.

Parámetro	Valor
$W(k)$	$\begin{bmatrix} 0.6959, 0.4507, -0.1162 \\ 0.7896, 0.5042, 0.8387 \end{bmatrix}$
$C(k)$	$\begin{bmatrix} -1.24E-5 & -1.61E-5 & -1.29E-5 & -1.13E-5 \\ -2.93E-4 & -2.87E-4 & -2.93E-4 & -2.92E-4 \end{bmatrix}$
$D(k)$	$\begin{bmatrix} 0.5868, 0.3134, -0.0876 \\ 0.3225, 0.3156, -0.0594 \end{bmatrix}$
$q(k)$	$[-0.7249, -0.9080, -0.6680]$
$s(k)$	$[0.9786, 0.1266, 0.5347]$

Tabla 5. Valores iniciales de las ganancias para la *WaveNet-IIR* PID y sus tasas de actualización.

Ganancias	Tasas de aprendizaje
$p(k) = [2.90E-2, 9.04E-3]$	$u_p = [1.00, 0.90]$
$i(k) = [1.58E-2, 1.10E-2]$	$u_i = [0.80, 0.60]$
$d(k) = [1.45E-2, 5.50E-3]$	$u_d = [0.75, 0.80]$

Enseguida se cierra el lazo de control y se utilizan los datos propuestos en la Tabla 5, los cuales se especifican

los valores de las ganancias $\mathbf{p}(k)$, $\mathbf{i}(k)$ y $\mathbf{d}(k)$, así como sus tasas de aprendizaje; y fueron hallados a prueba y error.

Por último, se realiza la simulación utilizando las ecuaciones (34) y (35) para obtener las señales de referencia; además, se indujo un ruido a la salida de la planta utilizando el bloque *Uniform White Noise* de LabVIEW, cuya expresión es

$$f(x) = \begin{cases} 1/2a, & -a \leq k < a \\ 0, & \text{en otro caso} \end{cases} \quad (36)$$

donde k es un número aleatorio y a toma el valor de 1.

Los resultados obtenidos se indican en la Figura 9 y la señal de control $\mathbf{u}(k)$ generada se presenta en la Figura 10. En este caso, el esquema de control PID *WaveNet-IIR* es capaz de realizar el seguimiento de la trayectoria manteniendo un error acotado, de igual manera la señal de control permanece acotada.

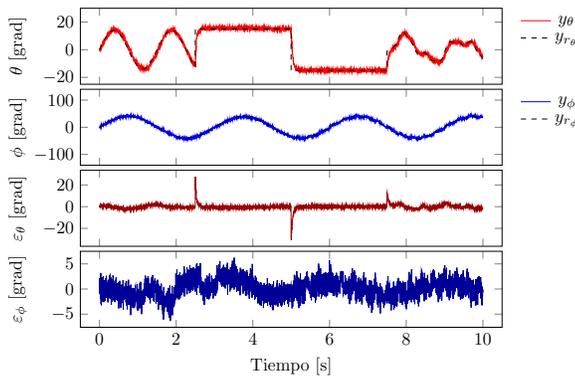


Figura 9. Control en lazo cerrado del modelo lineal aplicando la *WaveNet-IIR* y errores de seguimiento.

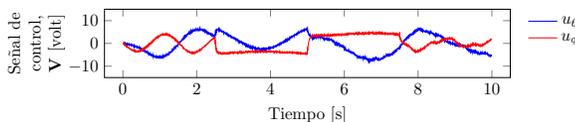


Figura 10. Comportamiento de la señal de control $\mathbf{u}(k)$ aplicando el PID *WaveNet-IIR*.

4. CONCLUSIONES

A partir de los resultados mostrados en este artículo, es posible decir que la aplicación de las redes neuronales *wavelets* junto a los filtros IIR ostentan una enorme capacidad para resolver problemas, tal es el caso de identificar la dinámica de una planta y la auto-sintonización de una ley de control a pesar de desconocer la dinámica del modelo a manipular.

Este trabajo fue resultado de una tesis de ingeniería (García-Castro, 2020). Actualmente, se está desarrollando el trabajo experimental con la finalidad de obtener los resultados de laboratorio y, de esta manera, compararlos con los obtenidos a través de la simulación numérica.

REFERENCIAS

- Carrillo-Santos, C., Seck-Tuoh-Mora J., Hernández-Romero N. y Ramos-Velasco L.E., 2018. Wavenet identification of dynamical systems by a modified PSO algorithm. *Eng Appl Artif Intell*, 79, pp. 1–9.
- Emamia, S.A., Paolo, C. y Banazadeh, A., 2020. Neural network-based flight control systems: present and future. *Annual Reviews in Control*, 59, pp. 97-137.
- Deisenroth, M., Faisal A. y Ong C., 2020. *Mathematics for machine learning*. Cambridge University Press.
- Díaz-López, O.A., Ramos-Velasco, L.E., Domínguez-Ramírez, O.A. y Parra-Vega, V., 2013. Multiresolution WaveNet PID control for global regulation of robots. *9th Asian Control Conference (ASCC 2013)*.
- Domínguez-Mayorga, C., Espejel-Rivera, M., Ramos-Velasco, L.E., Ramos-Fernández, J. y Escamilla Hernández, E., 2012. Wavenet algorithms with applications in approximation signals: a comparative study. *RIAI*, 9(4), pp. 347–358.
- García-Castro, O.F., 2020. *Controladores PID wavenet aplicados a un modelo de helicóptero de 2 GDL*. Ing. en Aeronáutica, Universidad Politécnica Metropolitana de Hidalgo, México.
- Haykin, S., 1999. *Neural networks, a comprehensive foundation*. EUA: Pearson Education Inc.
- Liu, Z y Ye, X., 2021. Research on attitude control strategy of Two-DOF helicopter based on LabVIEW. *2021 IEEE 4th Advanced Information Management, Communication, Electronic and Automation Control Conference (IMCEC)*, pp. 165-172.
- Ouyang, Y., Dong, L., Xue, L. y Sun, C., 2019. Adaptive control based on neural networks for an uncertain 2-DOF helicopter system with input deadzone and output constraints. *IEEE/CAA Journal of Automatica Sinica*, 6(3), pp. 807-815.
- Quanser Inc., 2012. *User manual 2 DOF helicopter experiment, set up and configuration*. Markham, Ontario.
- Ramos-Velasco, L.E., Domínguez-Ramírez, O. y Parra-Vega, V., 2016. Wavenet fuzzy PID controller for nonlinear MIMO systems: experimental validation on a high-end haptic robotic interface. *Applied Soft Computing*, 40, pp. 199–205.
- Rouis, N., N'Doye, I. y Laleg-Kirati, T.M., 2021. Model-free LQR based PID controller for trajectory tracking of 2-DOF helicopter.
- Zhao, Z., He, W., Yang, J. y Li, Z., 2022. Adaptive neural network control of an uncertain 2-DOF helicopter system with input backlash and output constraints. *Neural Computing and Applications*, publicado en línea.
- Zhao, Z., Zhang, J., Liu, Z., Mu, C. y Hong, K.S., 2022. Adaptive neural network control of an uncertain 2-DOF helicopter with unknown backlash-like hysteresis and output constraints. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–10.
- Zou, T., Wu, H., Sun, W. y Zhao, Z., Ouyang, Y., Dong, L., Xue, L. y Sun, C., 2022. Adaptive neural network sliding mode control of a nonlinear two-degrees-of-freedom helicopter system. *Asian J. Control*.