

# Neural Adaptive PID Control of a Quadrotor using EFK

Claudio. D. Rosales, Santiago. R. Tosetti, Carlos. M. Soria and Francisco.G. Rossomando

**Abstract**— In this paper, we present a novel trajectory tracking algorithm for a four-rotor air vehicle (quadrotor). The PID controller is developed following an adaptive neuronal technique, and the discrete theory of Lyapunov verifies its stability. Also, the neuronal identification of the UAV dynamic model is presented. Besides, an extended Kalman filter is used in order to filter the signals from the aerial vehicle that are contaminated by measurement noises, and that can affect the quality of the identification. Then, the output errors are re-propagated to adjust the PID gains to reduce the control errors. Finally, the experimental results are presented using a four-rotor aerial vehicle (quadrotor), by comparing the presented proposal with a classical fixed-gain PID.

**Index Terms**— Adaptive PID; Discrete Stability Analysis; Identification; Neural Networks; Quadrotor.

## I. INTRODUCCIÓN

En los últimos años, los vehículos aéreos autónomos (UAV, Unmanned Aerial Vehicles) de cuatro rotores denominados cuadricópteros o quadrotores, se han vuelto muy populares debido a que tienen la misma maniobrabilidad que un helicóptero convencional, como son la capacidad de despegue y aterrizaje vertical, poder realizar vuelos estacionarios y principalmente la capacidad de movimiento omnidireccional [1]. Los cuadricópteros en la actualidad son más populares que los helicópteros convencionales, debido a que sus mecanismos son más simples y fáciles de controlar. Aunque todavía el control de estos vehículos es un desafío por las no-linealidades inherentes al sistema y al acoplamiento entre las variables de control producidos por el momento giroscópico ya que son sistemas subactuados [2].

En artículos recientes, un gran número de estrategias de control se han utilizado para abordar el problema de la estabilidad de estas aeronaves. La gran mayoría son enfoques que requieren contar con un modelo matemático que permita describir el comportamiento dinámico del vehículo aéreo. Los autores en [3] utilizan métodos numéricos y álgebra lineal para obtener las acciones de control de mínima energía a partir del modelo dinámico del sistema. Además, se encuentran en la literatura trabajos en control predictivo basado en modelo [4], superficie deslizante [5, 6], control de dinámica inversa, linealización por retroalimentación, entre otros.

El desempeño del sistema de control depende de la exactitud del modelo matemático. Esto motivó el desarrollo de controladores diseñados a partir de los PID clásico [7, 8] donde no se requiere conocimiento previo de la planta, pero su desempeño depende de que no existan cambios en los parámetros intrínsecos del sistema ya que la sintonización de

este controlador normalmente es para un punto fijo de operación del sistema. Como consecuencia de esto, algunos trabajos de la literatura presentan estrategias de control PID pero que tienen características dinámicas de inteligencia artificial como en [9], lo cual mejora el desempeño frente a los controladores estáticos. Las redes neuronales artificiales (NN) tienen varias características como son su no linealidad, autoaprendizaje y adaptabilidad. El control neuronal utiliza redes neuronales artificiales es una de las estrategias de control inteligentes que utiliza una red neuronal artificial como controlador o identificador para resolver el problema de control del sistema no lineal e incierto en un entorno no conocido [10, 11]. Algunos sistemas híbridos de control PID y NN han sido estudiadas ampliamente, donde se utiliza la estructura neuronal para poder retropropagar el error de control y poder ajustar las ganancias de los controladores PID. En [12] se presenta un controlador que combina una estructura PID y una red neuronal de base radial (RBF-NN) para retropropagar el error de control y ajustar las ganancias de controlador. El mismo se enfoca en el control del ángulo de cabeceo (pitch) del vehículo aéreo, el cual genera el desplazamiento longitudinal del mismo. En dicho trabajo no se demuestra la estabilidad del sistema de control y se verifica su desempeño a través de simulaciones. En [13] se proponen dos lazos anidados de controladores PID desarrollados en tiempo continuo. En el lazo interno se propone un controlador PID Neuronal para cada grado de libertad rotacional (pitch, roll y yaw o por sus términos en español cabeceo, rolido y guiñada), donde cada PID este acoplado a una red neuronal multi-capas. Además, se utiliza un lazo de control externo para alcanzar las referencias de trayectoria basado en cuatro PID, uno para cada grado de libertad (x, y, z). Se prueba la estabilidad del lazo de control PID Neuronal y se valida utilizando simulaciones numéricas.

En este trabajo se presenta un esquema de control PID neuronal que se ha diseñado en tiempo discreto lo que permite una implementación más eficiente y evita el problema de implementar controladores diseñados para tiempo continuo en sistemas digitales. El controlador ajusta las ganancias del controlador PID sin la necesidad de contar con conocimiento previo de la planta. El presente controlador disminuye las incertidumbres en el UAV y las perturbaciones creadas por el ambiente, además, no se requiere conocimiento del modelo dinámico del cuadricóptero. De esta manera, se garantiza la robustez frente a perturbaciones y dinámicas no modeladas. Por lo tanto, el presente controlador al ser diseñado en tiempo discreto mejora su desempeño frente a otras propuestas. En [14], se muestra un PID Neuronal utilizado para controlar un vehículo terrestre de tracción diferencial. El controlador se valida experimentalmente con un muy buen desempeño, sin embargo, el análisis de estabilidad puede mejorarse cuando se

considera las ganancias del PID dentro de la función candidata de Lyapunov. El objetivo de este trabajo es mostrar el diseño e implementación de un controlador PID Neuronal Adaptable sobre un vehículo aéreo tipo quadrotor o cuadricóptero, donde los parámetros del mismo pueden variar (por diferentes causas) durante la ejecución de la tarea.

El trabajo se organiza de la siguiente manera, en la sección II se muestra el modelo dinámico utilizado para describir el comportamiento del vehículo aéreo utilizado. La sección III presenta el identificador neuronal utilizado para realizar la estimación de algunas variables del vehículo y su correspondiente demostración de estabilidad. En la sección IV se presenta el desarrollo del controlador PID Neuronal Adaptable y en la sección V se muestra el desarrollo de la demostración de estabilidad. Por último, la sección VI presenta los resultados experimentales utilizados para validar la propuesta de control desarrollada, y en la sección VII se muestran las conclusiones obtenidas.

## II. MODELO DINÁMICO DE UN CUADRICÓPTERO

Los autopilotos que se utilizan para el control de estas aeronaves tienen en cuenta los efectos aerodinámicos para conseguir la estabilización del vuelo, aunque no es posible tener acceso a sus parámetros para poder ajustarlos en línea [15]. Sin embargo, en varios trabajos de la literatura, [16, 17, 18], se fundamenta la posibilidad de analizar la respuesta de los cuadricópteros fundamentado en la respuesta de sus sensores internos, en lugar de lidiar con la dinámica compleja que describe su comportamiento. La relación más importante de esta plataforma es la relación entre las acciones de control y la posición del cuadricóptero. Esta relación se estima mediante un modelo lineal de dos estados que representa la dinámica del cuadricóptero [10] de la siguiente manera:

$$\ddot{\mathbf{q}}^w = \mathbf{R}\mathbf{K}_u\mathbf{u} - \mathbf{K}_v\dot{\mathbf{q}}^w \quad (1)$$

el superíndice  $w$  indica que la variable está expresada con respecto al sistema de referencia inercial,  $\mathbf{q}^w = [x^w, y^w, z^w, \psi^w]^T$  y  $\dot{\mathbf{q}}^w = [\dot{x}^w, \dot{y}^w, \dot{z}^w, \dot{\psi}^w]^T$  y  $\mathbf{u} = [u_x, u_y, u_z, u_\psi]^T$  representan la posición, las velocidades del vehículo en el sistema inercial y las acciones de control normalizadas, respectivamente. A su vez,  $\mathbf{K}_u = \text{diag}[k_1 k_3 k_5 k_7]$  y  $\mathbf{K}_v = \text{diag}[k_2 k_4 k_6 k_8]$  son matrices diagonales definidas positivas. La matriz  $\mathbf{R}$  es una matriz de rotación de  $\mathbf{R}^3$  que solo considera la rotación en un ángulo  $\psi^w$  alrededor del eje  $z^w$  del sistema inercial. En la Figura 1 se presenta el vehículo aéreo utilizado para validar la ley de control propuesta y los sistemas de referencia asociados. Las variables  $\ddot{x}^w$ ,  $\ddot{y}^w$  y  $\ddot{z}^w$  son las aceleraciones en los ejes  $x^w$ ,  $y^w$  y  $z^w$ , respectivamente, en el referencial  $\langle w \rangle$  y  $\ddot{\psi}^w$  representa la aceleración angular relacionada al eje  $z$ . Las señales de control  $u_x$ ,  $u_y$  y  $u_z$  son las variables de entradas conectadas con los ejes  $x^w$ ,  $y^w$  y  $z^w$ , respectivamente, mientras que  $u_\psi$  es la variable de entrada asociada con el movimiento angular alrededor del eje  $z^b$ .

Con el objetivo de obtener un modelo discreto del modelo dinámico, se utilizó una discretización directa con  $\dot{x} = \frac{x(k) - x(k-1)}{T_o}$  y  $\ddot{x} = \frac{x(k+1) - 2x(k) + x(k-1)}{T_o^2}$ , y lo mismo para  $y$ ,  $z$  y  $\psi$ .

De este modo, se expresa (1) en tiempo discreto:

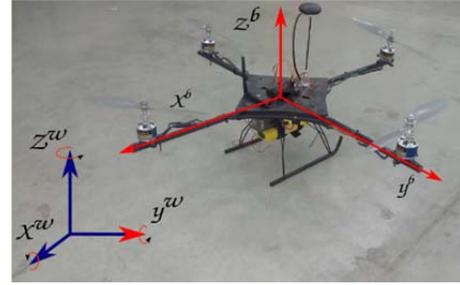


Figura 1. Cuadricóptero y sus sistemas de referencias asociados, el sistema inercial  $\langle w \rangle$  y el sistema asociado al vehículo  $\langle b \rangle$ .

$$\mathbf{q}^w(k+1) = \mathbf{\Lambda}_2\mathbf{u} - \mathbf{\Lambda}_1\mathbf{q}^w(k) + \mathbf{\Lambda}_0\mathbf{q}^w(k-1) + \mathbf{\Delta}(k) \quad (2)$$

con  $\mathbf{\Lambda}_2 = \mathbf{R}\mathbf{K}_u T_o^2$ ,  $\mathbf{\Lambda}_1 = (-2\mathbf{I} + \mathbf{K}_v T_o)$  y  $\mathbf{\Lambda}_0 = (\mathbf{K}_v T_o - \mathbf{I})$ . El último término adicionado en (2) ( $\mathbf{\Delta}(k)$ ) representa todas las incertidumbres dinámicas y perturbaciones que puedan afectar al cuadricóptero.

$$\underbrace{\begin{pmatrix} \mathbf{q}^w(k) \\ \mathbf{q}^w(k+1) \end{pmatrix}}_{\mathbf{x}(k+1)} = \underbrace{\begin{pmatrix} 0 & \mathbf{I} \\ \mathbf{\Lambda}_0 & -\mathbf{\Lambda}_1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} \mathbf{q}^w(k-1) \\ \mathbf{q}^w(k) \end{pmatrix}}_{\mathbf{x}(k)} + \underbrace{\begin{pmatrix} 0 \\ \mathbf{\Lambda}_2 \end{pmatrix}}_{\mathbf{B}} \mathbf{u}(k) + \underbrace{\begin{pmatrix} 0 \\ \mathbf{\Delta}(k) \end{pmatrix}}_{\mathbf{\delta}(k)} \quad (3)$$

Para obtener el modelo completo solo se requiere la identificación de los parámetros  $k_1, k_2, \dots, k_8$  [19].

La ecuación (2) puede expresarse como un modelo lineal en espacio de estados donde  $\mathbf{u}(k)$  y  $\mathbf{x}(k+1)$  son la entrada y la salida medida, respectivamente, y  $\mathbf{x}(k)$  es el vector de estado.

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) \quad (4)$$

Los parámetros del modelo (2) son constantes para uso en ambientes interiores sin perturbaciones (indoor). Si utilizamos el UAV en ambientes externos (outdoor), el aparato estará sometido a perturbaciones, donde las matrices  $\mathbf{K}_u$  y  $\mathbf{K}_v$  serán variantes en el tiempo, y los parámetros de (2) resultan:

$$\begin{aligned} \mathbf{\Lambda}_2(k) &= \mathbf{\Lambda}_0^{-1} \mathbf{R}\mathbf{K}_u(k) T_o^2 \\ \mathbf{\Lambda}_1(k) &= \mathbf{\Lambda}_0^{-1} (-2\mathbf{I} + \mathbf{K}_v(k) T_o) \\ \mathbf{\Lambda}_0(k) &= (-\mathbf{I} + \mathbf{K}_v(k) T_o) \end{aligned} \quad (5)$$

En la sección siguiente se identificará el modelo del UAV.

## III. IDENTIFICACIÓN EN LÍNEA DEL MODELO

La implementación del control PID con ganancias adaptables requiere una buena estimación de los estados, y su correspondiente realimentación de los errores de control para obtener un buen ajuste de las ganancias del PID. El vector de errores de estimación se define del siguiente modo:

$$\tilde{\mathbf{q}}(k) = \mathbf{q}^w(k) - \hat{\mathbf{q}}^w(k) = \begin{pmatrix} \tilde{q}_x(k), \tilde{q}_y(k), \tilde{q}_z(k), \tilde{q}_\psi(k) \end{pmatrix}^T \quad (6)$$

Donde  $\hat{\mathbf{q}}^w(k)$  es el vector de señales estimadas del drone y  $\tilde{q}_x(k) = x^w(k) - \hat{x}^w(k)$ ,  $\tilde{q}_y(k) = y^w(k) - \hat{y}^w(k)$ ,  $\tilde{q}_z(k) = z^w(k) - \hat{z}^w(k)$  y  $\tilde{q}_\psi(k) = \psi^w(k) - \hat{\psi}^w(k)$  son sus componentes. El modelo (7) puede ser representado por una red neuronal

$$\mathbf{q}^w(k+1) = \mathbf{A}\mathbf{q}^w(k) + \mathbf{W}^{*T} \xi(\tilde{\mathbf{q}}(k)) + \varepsilon(k) \quad (7)$$

donde  $\mathbf{A} = \text{diag}[a_x, a_y, a_z, a_\psi]$  es una matriz Hurwitz diagonal,  $\mathbf{W}^{*T}$  son los pesos óptimos de la red neuronal que minimizan el error de aproximación. La variable  $\varepsilon(k)$  es el error de

aproximación de la red neuronal. Finalmente,  $\xi(\bar{\mathbf{q}}(k))$  es una función de base radial que en este trabajo se utilizó una función del tipo Gaussiana y  $\bar{\mathbf{q}}(k)$  es el regresor del modelo neuronal. Debido a que los estados de salida son afectados por perturbaciones y ruido de medición, se propone un estimador de estados basado en (7)

$$\hat{\mathbf{q}}(k+1) = \mathbf{A}\hat{\mathbf{q}}^w(k) + \hat{\mathbf{W}}^T(k) \xi(\bar{\mathbf{q}}(k)) \quad (8)$$

De (6) y realizando la diferencia entre (7) y (8), el error de identificación puede describirse como:

$$\tilde{\mathbf{q}}(k+1) = \mathbf{A}\tilde{\mathbf{q}}(k) + \hat{\mathbf{W}}^T(k) \xi(\bar{\mathbf{q}}(k)) + \varepsilon(k) \quad (9)$$

Donde  $\tilde{\mathbf{W}} = \mathbf{W}^* - \hat{\mathbf{W}}$ . Expresando la ecuación anterior en forma escalar

$$\tilde{q}_i(k+1) = a_i \tilde{q}_i(k) + \tilde{\mathbf{W}}_i^T(k) \xi(\bar{\mathbf{q}}(k)) + \varepsilon_i(k) \quad (10)$$

el subíndice  $i$  denota el número de fila de  $\tilde{\mathbf{q}}(k)$ , además  $\xi(\bar{\mathbf{q}}(k))$  es la función de base radial (RBF) que representa cada neurona en las capas ocultas (hidden layers).

$$\xi_i(\bar{\mathbf{q}}(k)) = \exp\left(-\frac{(\bar{\mathbf{q}}(k) - c_i)^T(\bar{\mathbf{q}}(k) - c_i)}{2\sigma_i}\right) \quad (11)$$

$l$  es el número de la neurona de la capa oculta,  $c_i$  es la posición central de la neurona  $l$ -ésima, y  $\sigma_i$  es el factor de ancho (igual a uno en este caso) de la función Gaussiana. El vector  $\bar{\mathbf{q}}(k)$  es definido de la siguiente manera

$$\bar{\mathbf{q}}(k) = [\hat{\mathbf{q}}(k), \hat{\mathbf{q}}(k-1), \mathbf{u}(k), \mathbf{q}_{ref}(k+1)] \quad (12)$$

El filtrado mediante Kalman proporciona una estima del estado  $\hat{q}_i(k)$  y su error de covariancia  $\mathbf{P}_i(k)$  de forma recursiva. La calidad de la estimación depende de los parámetros del sistema y sus perturbaciones. Para la identificación, la salida está contaminada por ruidos de medición y se debe realizar su filtrado. Pero debido al hecho que la aproximación de la red neuronal es no lineal, se requiere un filtro de Kalman extendido [20].

$$\hat{\mathbf{W}}_i(k+1) = [\hat{\mathbf{W}}_i(k) + \eta \mathbf{K}_i \tilde{q}_i(k)] \quad (13)$$

$$\mathbf{K}_i(k) = \mathbf{P}_i(k) \mathbf{H}_i(k) \mathbf{M}_i(k) \quad (14)$$

$$\mathbf{P}_i(k+1) = \mathbf{P}_i(k) - \mathbf{K}_i(k) \mathbf{H}_i^T(k) \mathbf{P}_i(k) \mathbf{M}_i(k) + \mathbf{Q}_i(k) \quad (15)$$

con  $\mathbf{M}_i(k) = [\mathbf{R}_i(k) + \mathbf{H}_i^T(k) \mathbf{P}_i(k) \mathbf{H}_i(k)]^{-1}$ .

$\tilde{q}_i(k)$  es el error del estado estimado,  $\mathbf{P}_i(k) \in \mathfrak{R}^{m \times m}$  es la matriz de covariancia de error estimada en el paso  $k$ ,  $\hat{\mathbf{W}}_i(k)$  es el vector de pesos de la red neuronal,  $m$  es el número de neuronas de la red,  $q_i^w(k)$  es el estado  $i$ -ésimo del UAV,  $\hat{q}_i(k)$  es la estimación neuronal del  $i$ -ésimo estado,  $n$  es el número de estados que para nuestro caso resulta en este caso  $n = 4$ ,  $\mathbf{K}_i(k) \in \mathfrak{R}^m$  es el vector de ganancias Kalman,  $\mathbf{Q}_i(k) \in \mathfrak{R}^m$  es vector de peso de la matriz de covarianza estimada,  $\mathbf{R}_i(k) \in \mathfrak{R}^m$  es la covarianza del ruido medido;  $\mathbf{H}_i(k) \in \mathfrak{R}^m$  es un vector en donde cada elemento  $H_{ij}$  es la derivada del estado  $\hat{q}_i(k)$  de la red neuronal con respecto al peso de la red  $\hat{W}_{ij}(k)$

$$\mathbf{H}_i(k) = \left[ \frac{\partial \hat{q}_i(k+1)}{\partial \hat{W}_{ij}(k)} \right]^T = \xi^T(\bar{\mathbf{q}}(k)) \quad (16)$$

donde  $i = x, y, z, \psi$  y  $j = 1, 2, \dots, m$ . En este caso  $\mathbf{P}_i(k)$  y  $\mathbf{Q}_i(k)$  se inicializan como matrices diagonales con valores de los elementos aleatorios. Además, el conjunto de las matrices  $\mathbf{H}_i(k)$ ,  $\mathbf{K}_i(k)$  y  $\mathbf{P}_i(k)$  para la ley de ajuste del EKF están acotadas [21, 22].

Por lo tanto, el vector de errores estimados  $\tilde{\mathbf{W}}_i(k+1)$  se define de la siguiente manera

$$\tilde{\mathbf{W}}_i(k+1) = \mathbf{W}_i^* - \hat{\mathbf{W}}_i(k+1) = \mathbf{W}_i^* - [\hat{\mathbf{W}}_i(k) + \eta \mathbf{K}_i \tilde{q}_i(k)] = \tilde{\mathbf{W}}_i(k) - \eta \mathbf{K}_i \tilde{q}_i(k) \quad (17)$$

el índice  $i$  denota la  $i$ -ésima fila de  $\hat{\mathbf{W}}(k)$  y  $\eta$  es un factor de diseño que representa el factor de aprendizaje de la red neuronal.

**Hipótesis:** El error de aproximación óptimo  $\varepsilon$  está acotado  $\|\varepsilon\| \leq \varepsilon_{\max}$ .

**Teorema:** Considerando el sistema definido por (7) y el modelo del estimador (9) utilizando una ley de ajuste EKF definida por (13). El observador neuronal basado en un EKF es semi-globalmente uniformemente acotado (SGUUB, semiglobally uniformly ultimate bounded).

**Demostración:** Se propone la siguiente función candidata de Lyapunov

$$V_i(k) = \tilde{q}_i^2(k) + (\tilde{\mathbf{W}}_i^T(k) \tilde{\mathbf{W}}(k)) \quad (18)$$

Realizando la diferencia en tiempo discreta,  $\Delta V_i$  se escribe como:

$$\Delta V_i(k) = V_i(k+1) - V_i(k) = (\tilde{\mathbf{W}}_i^T(k+1) \tilde{\mathbf{W}}(k+1)) - (\tilde{\mathbf{W}}_i^T(k) \tilde{\mathbf{W}}(k)) + \tilde{q}_i^2(k+1) - \tilde{q}_i^2(k) \quad (19)$$

A continuación, se reemplaza (10) y (18) en (19).

$$\begin{aligned} \Delta V_i(k) = & \tilde{\mathbf{W}}_i^T(k) \tilde{\mathbf{W}}(k) - 2\eta \tilde{q}_i(k) \tilde{\mathbf{W}}_i^T(k) \mathbf{K}_i + \eta^2 \tilde{q}_i^2(k) \mathbf{K}_i^T \mathbf{K}_i \\ & a_i^2 \tilde{q}_i^2(k) + (\tilde{\mathbf{W}}_i^T(k) \xi(\bar{\mathbf{q}}(k)))^2 \\ & + \varepsilon_i^2(k) + 2a_i \tilde{q}_i(k) \tilde{\mathbf{W}}_i^T(k) \xi(\bar{\mathbf{q}}(k)) + 2a_i \tilde{q}_i(k) \varepsilon_i(k) \\ & + 2\varepsilon_i(k) \tilde{\mathbf{W}}_i^T(k) \xi(\bar{\mathbf{q}}(k)) - \tilde{\mathbf{W}}_i^T(k) \tilde{\mathbf{W}}(k) - \tilde{q}_i^2(k) \end{aligned} \quad (20)$$

Considerando la desigualdad vectorial  $\mathbf{X}^T \mathbf{X} + \mathbf{Y}^T \mathbf{Y} \geq |\mathbf{2X}^T \mathbf{Y}|$ , donde  $\mathbf{X} \wedge \mathbf{Y} \in \mathfrak{R}^{m \times 1}$

$$\begin{aligned} \Delta V_i(k) = & \eta^2 \tilde{q}_i^2(k) \mathbf{K}_i^T \mathbf{K}_i - 2\eta \tilde{q}_i(k) \tilde{\mathbf{W}}_i^T(k) \mathbf{K}_i + 3a_i^2 \tilde{q}_i^2(k) + \\ & 3(\tilde{\mathbf{W}}_i^T(k) \xi(\bar{\mathbf{q}}(k)))^2 + 3\varepsilon_i^2(k) - \tilde{q}_i^2(k) \end{aligned} \quad (21)$$

Operando matemáticamente la ecuación anterior y aplicando norma matricial y sus propiedades se obtiene,

$$\begin{aligned} \Delta V_i \leq & |\tilde{q}_i^2(k)| \|\eta \mathbf{K}_i\|^2 - 2\eta |\tilde{q}_i(k)| \|\tilde{\mathbf{W}}_i(k)\| \|\mathbf{K}_i\| + 3a_i^2 |q_i(k)|^2 \cdot \\ & 3\|\tilde{\mathbf{W}}_i(k)\|^2 \|\xi(\bar{\mathbf{q}}(k))\|^2 + 3|\varepsilon_i(k)|^2 - |\tilde{q}_i(k)|^2 \end{aligned} \quad (22)$$

Recordando que  $\|\xi(\bar{\mathbf{q}}(k))\|^2 \leq 1$ , por lo tanto, de la ecuación anterior se obtienen las condiciones para que  $\Delta V_i < 0$  [23]

$$\begin{aligned} |\tilde{q}_i(k)|^2 & > \frac{3|\varepsilon_i(k)|^2}{1 - \|\eta \mathbf{K}_i\|^2 - 3a_i^2} \\ \|\tilde{\mathbf{W}}_i(k)\| & > 2\eta |\tilde{q}_{i\max}(k)| \|\mathbf{K}_i\| \end{aligned} \quad (23)$$

La matriz  $\mathbf{K}_i$  es acotada [21,22] y por hipótesis  $\varepsilon(k)$  es una función acotada (23) prueba que el observador neuronal es semi-globalmente uniformemente acotado y es estable. De esta manera, puede utilizarse como retropropagación del error del controlador PID adaptivo que se explayará en la siguiente sección.

#### IV. CONTROLADOR PID NEURONAL ADAPTIVO

El objetivo es obtener un método de auto-ajuste utilizando un controlador PID neuronal adaptable que permita a un cuadricóptero siga una trayectoria deseada con un error mínimo de control. El error de control se define la siguiente manera:

$$\mathbf{e}(k) = (e_1(k), e_2(k), e_3(k), e_4(k))^T \quad (24)$$

donde  $e_1(k) = (x_{ref}(k+1) - x^w(k+1))$ ,  $e_2(k) = (y_{ref}(k+1) - y^w(k+1))$ ,  $e_3(k) = (z_{ref}(k+1) - z^w(k+1))$  y  $e_4(k) = (\psi_{ref}(k+1) - \psi^w(k+1))$ .

En la Figura 2 se presenta la estructura de control utilizada en este trabajo.

El modelo clásico del controlador PID en tiempo discreto está definido por la siguiente ecuación,

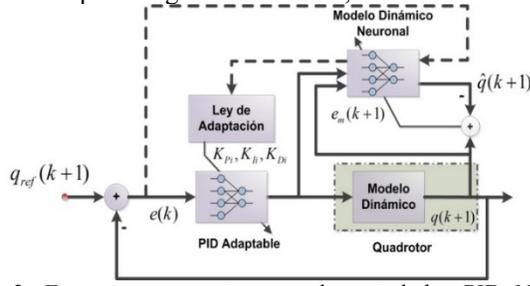


Figura 2. Esquema propuesto para el controlador PID Neuronal Adaptivo.

$$o_i(k) = K_{Pi}ei(k) + K_{Ii} \sum_{j=1}^k e_i(j)T_o + \frac{K_{Di}}{T_o} (ei(k) - ei(k-1)) \quad (25)$$

donde el subíndice  $i$  indica la variable del vehículo aéreo a controlar. Considerando la acción de control en el instante de tiempo  $(k-1)$ , resulta:

$$o_i(k-1) = K_{Pi}ei(k-1) + K_{Ii} \sum_{j=1}^{k-1} e_i(j)T_o + \frac{K_{Di}}{T_o} (ei(k-1) - ei(k-2)). \quad (26)$$

Realizando la diferencia (25) y (26) se puede expresar a la señal de control  $o_i(k)$  como:

$$o_i(k) = o_i(k-1) + K_{Pi}(ei(k) - ei(k-1)) + K_{Ii}ei(k)T_o + \frac{K_{Di}}{T_o} (ei(k) - 2ei(k-1) + ei(k-2)). \quad (27)$$

El objetivo de adaptar las ganancias del controlador PID va detrás de la premisa de obtener un error de control menor y mayor velocidad de convergencia. Sin embargo, el PID convencional es un controlador estático, dado que las ganancias son valores constantes que no se modifican durante la ejecución de la tarea de control. Como resultado, las acciones no tienen la capacidad de adaptarse frente a las variaciones que pueden surgir en la planta a controlar. El controlador propuesto permite resolver este problema dado que las ganancias del PID son sintonizadas para minimizar el error de control con la capacidad de mejorar su robustez frente a perturbaciones externas. Para obtener un controlador basado en la estructura de un PID es esencial definir una función de activación del controlador. Por lo tanto, se utiliza la tangente hiperbólica  $\tanh(\cdot)$  como función de saturación continua. De este modo, la acción de control es menor que el máximo admisible o posible de lograr por parte del actuador. De acuerdo con la hipótesis 1 de la sección anterior:

$$|u_i(k)| \leq u_{iMax}$$

La principal ventaja del uso de la función  $\tanh(\cdot)$  para cada neurona es que se garantiza la estabilidad, que será probada en la sección V.

**Definición 1:** La función de activación  $\varphi(\cdot)$  está definida de la siguiente manera:

$$u_i(k) = \varphi(o_i(k)) = u_{iMax} \tanh\left(\frac{o_i(k)}{u_{iMax}}\right) \quad (28)$$

Aplicando la función de activación en cada acción de (28) resulta,

$$o_i(k) = \quad (29)$$

$$\rho(o_i(k-1)) + K_{Pi}\varphi(e_i(k) - e_i(k-1)) + \tau_{Ii}\varphi(e_i(k))T_o + \frac{K_{Di}}{T_o}\varphi(e_i(k) - 2e_i(k-1) + e_i(k-2))$$

Para asegurar la convergencia y el desempeño de la función de optimización, se define el índice  $E(k)$  que es una función de los errores de posición del cuadricóptero.

$$E(k) = \frac{1}{2} \sum_{i=1}^4 e_i^2(k) \quad (30)$$

Los parámetros  $K_{Di}$ ,  $K_{Ii}$  y  $K_{Pi}$  para el controlador AN-PID deben sintonizarse utilizando el método del gradiente descendente, similar a los pesos de una red neuronal tipo perceptrón cuyo esquema se presenta en la Fig. 3.

Las reglas para adaptar los parámetros del controlador se definen a continuación:

$$\mathbf{K}_{\theta i}(k+1) = \begin{pmatrix} K_{Pi}(k) \\ K_{Ii}(k) \\ K_{Di}(k) \end{pmatrix} + \begin{pmatrix} \eta_{Pi} \frac{\partial E(k)}{\partial K_{Pi}} \\ \eta_{Ii} \frac{\partial E(k)}{\partial K_{Ii}} \\ \eta_{Di} \frac{\partial E(k)}{\partial K_{Di}} \end{pmatrix} \quad (31)$$

Los parámetros  $\eta_{Pi}$ ,  $\eta_{Ii}$  y  $\eta_{Di}$  son los factores de aprendizaje del controlador PID neuronal. De (31) y utilizando la regla de la cadena, se obtienen las siguientes ecuaciones:

$$\begin{aligned} \frac{\partial E(k)}{\partial K_{Pi}} &= \frac{\partial E(k)}{\partial q_i^w(k+1)} \frac{\partial q_i^w(k+1)}{\partial u_i(k)} \frac{\partial u_i(k)}{\partial o_i(k)} \frac{\partial o_i(k)}{\partial K_{Pi}} \\ \frac{\partial E(k)}{\partial K_{Ii}} &= \frac{\partial E(k)}{\partial q_i^w(k+1)} \frac{\partial q_i^w(k+1)}{\partial u_i(k)} \frac{\partial u_i(k)}{\partial o_i(k)} \frac{\partial o_i(k)}{\partial K_{Ii}} \\ \frac{\partial E(k)}{\partial K_{Di}} &= \frac{\partial E(k)}{\partial q_i^w(k+1)} \frac{\partial q_i^w(k+1)}{\partial u_i(k)} \frac{\partial u_i(k)}{\partial o_i(k)} \frac{\partial o_i(k)}{\partial K_{Di}} \end{aligned} \quad (32)$$

Después de realizar las operaciones matemáticas correspondientes se obtiene

$$\begin{aligned} K_{Pi}(k+1) &= K_{Pi}(k) - \eta_{Pi}e_i(k) \left( \frac{\partial \hat{q}_i^w(k+1)}{\partial u_i(k)} \right) \\ u_{iMax}[1 - \varphi^2(o_i(k))]f(e_i(k) - e_i(k-1)) \\ K_{Ii}(k+1) &= K_{Ii}(k) - \eta_{Ii}e_i(k) \left( \frac{\partial \hat{q}_i^w(k+1)}{\partial u_i(k)} \right) \\ u_{iMax}[1 - \varphi^2(o_i(k))]f(e_i(k)T_o) \\ K_{Di}(k+1) &= K_{Di}(k) - \eta_{Di}e_i(k) \left( \frac{\partial \hat{q}_i^w(k+1)}{\partial u_i(k)} \right) \dots \\ u_{iMax}[1 - \varphi^2(o_i(k))]f\left(\frac{e_i(k) - 2e_i(k-1) + e_i(k-2)}{T_o}\right) \end{aligned} \quad (33)$$

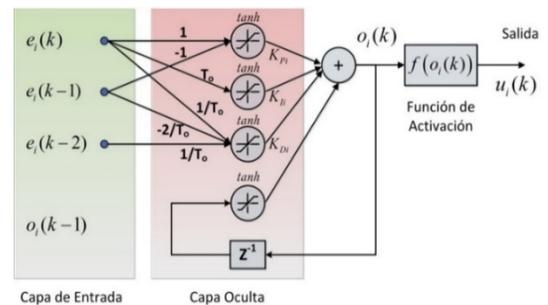


Figura 3. Estructura interna del controlador red PID neuronal adaptivo.

## V. ANALISIS DE ESTABILIDAD DEL AN-PID

**Hipótesis 2:** Definiendo los parámetros ideales del AN-PID  $\mathbf{K}_\theta^*$ , el error de estimación se define,

$$\tilde{\mathbf{K}}_{\theta i}(k) = \mathbf{K}_{\theta i}(k) - \mathbf{K}_\theta^*$$

donde  $\Omega_K$  y  $\Omega_q$  son conjuntos de valores acotados en  $\tilde{\mathbf{K}}_{\theta i}(k)$  sobre  $\mathbf{q}$ , respectivamente, definidas como  $\Omega_K = \{\tilde{\mathbf{K}}_{\theta i}(k) / \|\tilde{\mathbf{K}}_{\theta i}(k)\| \leq \mathbf{K}_{max}\}$  y  $\Omega_q = \{\mathbf{q}/\mathbf{q} \leq Q\}$  donde  $\mathbf{K}_{max}$  y  $Q$  son constantes positivas. Se asume que dentro de este conjunto  $\Omega_q$  el error de aproximación del AN-PID es muy pequeño cuando el vector de parámetros  $\mathbf{K}_{\theta i}(k)$  es igual al vector de parámetros óptimo denominado  $\mathbf{K}_\theta^*$ .

**Lema 1:** La diferencia del error  $\Delta e(k)$  en el proceso de aprendizaje esta expresada por:

$$\Delta e_i(k) = \frac{\partial e_i(k)}{\partial q_i^w(k+1)} \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right)^T \Delta \mathbf{K}_{\theta i} \quad (34)$$

Teniendo en cuenta (32) y considerando  $\bar{\eta} = \max(\eta_{Pi}, \eta_{Ii}, \eta_{Di})$

$$\Delta \mathbf{K}_{\theta i} = -\bar{\eta} e_i(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \quad (35)$$

reemplazando (34) en (35)

$$\Delta e_i(k) = -\frac{\partial e_i(k)}{\partial q_i^w(k+1)} \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right)^T \bar{\eta} e_i(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \quad (36)$$

$$\Delta e_i(k) \leq \rho_i e_i(k)$$

la variable  $\rho_i$  se define

$$\rho_i \leq \max \left( \bar{\eta} \left| \frac{\partial e_i(k)}{\partial q_i^w(k+1)} \right| \left\| \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right\|^2 \right) \quad (37)$$

con

$$\left\{ \begin{array}{l} \left| \frac{\partial e_i(k)}{\partial q_i^w(k+1)} \right| = 1 \\ \left\| \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right\| = \left| \frac{\partial q_i^w(k+1)}{\partial u_i(k)} \right| \left| \frac{\partial u_i(k)}{\partial o_i(k)} \right| \left\| \frac{\partial o_i(k)}{\partial \mathbf{K}_{\theta i}} \right\| \end{array} \right.$$

De la sección III, la dinámica desconocida  $\frac{\partial q_i^w(k+1)}{\partial u_i(k)}$  se aproxima mediante  $\frac{\partial \hat{q}_i^w(k+1)}{\partial u_i(k)}$ . En (23) se demuestra la estabilidad, por lo tanto, la diferencia de norma de  $\|\bar{\mathbf{q}}(k) - \mathbf{c}_i\|$  y  $\|\bar{\mathbf{e}}(k) - \mathbf{c}_i\|$  son acotadas por constantes positivas  $\kappa_1$  y  $\kappa_2$ , respectivamente y las funciones RBF  $\|\xi(\bar{\mathbf{q}}(k))\|$  y  $\|\chi(\bar{\mathbf{q}}(k))\|$  están acotadas ( $\leq 1$ ). Además, teniendo en cuenta (25) la red neuronal es estable, por lo tanto, los pesos de la red  $\bar{\mathbf{W}}$  están acotados por una constante, y la norma de (33) se puede representar mediante:

$$\left| \frac{\partial q_i^w(k+1)}{\partial u_i(k)} \right| \cong \left| \frac{\partial \hat{q}_i^w(k+1)}{\partial u_i(k)} \right| \leq \bar{W}. \quad (38)$$

Considerando

$$\begin{aligned} \left| \frac{\partial u_i(k)}{\partial o_i(k)} \right| &= |1 - \varphi^2(o_i(k))| \leq u_{iMax} \\ \left\| \frac{\partial o_i(k)}{\partial \mathbf{K}_{\theta i}} \right\| &= |\varphi(e_i(k) - e_i(k-1))| + |\varphi(e_i(k)T_o)| + \\ &\left| \varphi \left( \frac{e_i(k) - 2e_i(k-1) + e_i(k-2)}{T_o} \right) \right| \leq 3u_{iMax} \end{aligned} \quad (29)$$

A partir de (38) y (39), (37) puede expresarse de la siguiente manera:

$$\rho_i \leq \max \left( \bar{\eta} \left| \frac{\partial e_i(k)}{\partial q_i^w(k+1)} \right| \left\| \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right\|^2 \right) = 9\bar{\eta}\bar{W}^2 u_{iMax}^4 \quad (40)$$

Se define la siguiente función discreta definida positiva:

$$V(k) = \sum_{i=1}^4 (e_i^2(k) + \tilde{\mathbf{K}}_{\theta i}^T(k) \tilde{\mathbf{K}}_{\theta i}(k)) \quad (41)$$

Luego, se calcula la diferencia discreta la de la función

$$\Delta V(k) = V(k+1) - V(k) = \sum_{i=1}^4 \left( e_i^2(k+1) - e_i^2(k) + (\tilde{\mathbf{K}}_{\theta i}^T(k+1) \tilde{\mathbf{K}}_{\theta i}(k+1) - \tilde{\mathbf{K}}_{\theta i}^T(k) \tilde{\mathbf{K}}_{\theta i}(k)) \right) \quad (42)$$

De (35), el producto entre  $\tilde{\mathbf{K}}_{\theta i}^T$  y  $\tilde{\mathbf{K}}_{\theta i}$  puede determinarse

$$\begin{aligned} \tilde{\mathbf{K}}_{\theta i}^T(k+1) \tilde{\mathbf{K}}_{\theta i}(k+1) &= \dots \\ &= \left( \Delta \mathbf{K}_{\theta i} + \bar{\eta} e_i(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right)^T \left( \Delta \mathbf{K}_{\theta i} \right. \\ &+ \bar{\eta} e_i(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \left. \right) \left( \Delta \mathbf{K}_{\theta i} \right. \\ &+ \bar{\eta} e_i(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \left. \right) \tilde{\mathbf{K}}_{\theta i}^T(k+1) \tilde{\mathbf{K}}_{\theta i}(k+1) \\ &- \tilde{\mathbf{K}}_{\theta i}^T(k) \tilde{\mathbf{K}}_{\theta i}(k) \\ &= -2\bar{\eta} e_i(k) \tilde{\mathbf{K}}_{\theta i}^T(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \\ &+ \bar{\eta}^2 e_i^2(k) \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right)^T \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \end{aligned} \quad (43)$$

El valor de  $e_i(k+1)$  se puede calcular a través de la siguiente ecuación:

$$e_i(k+1) = e_i(k) + \Delta e_i(k) \quad (43)$$

Sustituyendo (44) en  $e_i^2(k+1)$  y calculando la diferencia de cuadrados se obtiene

$$e_i^2(k+1) - e_i^2(k) = 2e_i(k)\Delta e_i(k) + \Delta e_i^2(k) \quad (44)$$

Donde

$$\begin{aligned} \Delta V(k) &= \sum_{i=1}^4 \left( 2e_i(k)\Delta e_i(k) + \Delta e_i^2(k) - \bar{\eta}^2 e_i^2(k) \tilde{\mathbf{K}}_{\theta i}^T(k) \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right) + \right. \\ &\left. \bar{\eta}^2 e_i^2(k) \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right)^T \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right) \end{aligned} \quad (45)$$

De la Hipótesis 1, (46) puede escribirse como:

$$\begin{aligned} (k) &= \sum_{i=1}^4 \left( -2\rho_i e_i^2(k) + \rho_i^2 e_i^2(k) - 2\bar{\eta} e_i(k) \tilde{\mathbf{K}}_{\theta i}^T(k) \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right) \right. \\ &\left. + \bar{\eta}^2 e_i^2(k) \left( \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right)^T \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right) \end{aligned} \quad (47)$$

Aplicando norma a la ecuación anterior

$$\begin{aligned} V(k) &\leq \sum_{i=1}^4 -2\rho_i |e_i(k)|^2 + \rho_i^2 |e_i(k)|^2 \\ &- 2\bar{\eta} |e_i(k)| \left\| \tilde{\mathbf{K}}_{\theta i}^T(k) \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right\| \\ &+ \bar{\eta}^2 |e_i(k)|^2 \left\| \frac{\partial q_i^w(k+1)}{\partial \mathbf{K}_{\theta i}} \right\|^2 \end{aligned} \quad (48)$$

Reemplazando (39) y (40) en la diferencia discreta de Lyapunov (48)

$$\begin{aligned} \Delta V_i(k) &\leq -2\rho_i |e_i(k)|^2 + \rho_i^2 |e_i(k)|^2 \\ &- 6\bar{\eta} |e_i(k)| \left\| \tilde{\mathbf{K}}_{\theta i}^T(k) \right\| \bar{W} u_{iMax}^2 \\ &- 9\bar{\eta}^2 |e_i(k)|^2 \bar{W}^2 u_{iMax}^4 \end{aligned} \quad (49)$$

De (49), se obtiene lo siguiente

$$\Delta V_i(k) \leq 0 \quad |e_i(k)| > \frac{6\bar{\eta}\|\bar{\mathbf{K}}_{\theta_i}^T(k)\|\bar{W}u_{iMax}^2}{2\rho_i - \rho_i^2 + 9\bar{\eta}^2\bar{W}^2u_{iMax}^4} \quad (50)$$

De la ecuación anterior,  $V(k)$  es acotada, lo que implica que  $e_i(k)$  finalmente uniformemente acotado (Uniformly Ultimately Bounded, UUB). Los errores en (50) permiten establecer que estos están acotados por el error de aproximación del controlador PID neuronal adaptable.

## VI. RESULTADOS EXPERIMENTALES

En esta sección se presentan los resultados experimentales llevados a cabo para validar el controlador presentado en este trabajo. El cuadricóptero utilizado consta de una estructura de bajo peso que posee cuatro brazos que soportan en sus extremos a los sistemas propulsores, compuestos por un motor de corriente continua sin escobillas (BLDCM, Brushless DC Motor), su correspondiente hélice acoplada al eje del motor, y finalmente, el controlador electrónico de velocidad (ESC, Electronic Speed Control). El autopiloto utilizado es el arducopter que tiene la característica de ser de código abierto [24]. Además, cuenta con una IMU y un GPS integrado que permiten obtener las variables de postura de la aeronave. El sistema de control corre sobre Linux y fue implementado utilizando la SDK DroneKit-Python [25]. Esta SDK permite crear aplicaciones en python que pueden comunicarse con vehículos sobre el protocolo MAVLink. Para validar la propuesta se escogió una trayectoria de referencia tipo ocho inclinado. La tarea de seguimiento se va a desarrollar utilizando el controlador PID adaptable desarrollado en la sección IV y un controlador PID con ganancias estáticas. En ambos casos, los PID fueron inicializados con ganancias arbitrarias obtenidas después de realizar varias pruebas. En el del controlador PID adaptable, las ganancias son actualizadas durante la ejecución de la tarea a través de una ley de aprendizaje a través de un modelo de identificación neuronal. En la 4 se grafican los puntos GPS de la trayectoria real utilizando el software Google Earth<sup>®</sup>, donde se puede apreciar la forma tipo ocho. Las tareas se desarrollaron en un ambiente alejado de cualquier concentración urbana, por cuestiones de seguridad y las condiciones climáticas eran muy buenas casi sin viento.



Figura 4. Vista superior de la trayectoria realizada utilizando Google Earth<sup>®</sup>.

En 5 se muestra la evolución 3D de las trayectorias desarrolladas por el vehículo aéreo, utilizando el controlador PID con ganancias estáticas y con ganancias adaptables. En 6 se presenta la evolución de la norma del error de control, el cual contempla el error de las cuatro variables controladas ( $x, y, z, \psi$ ). Durante toda la tarea se puede observar como el

error de control del controlador AN-PID es menor que el PID clásico.

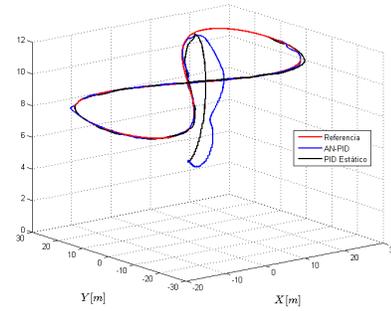


Figura 5. Evolución 3D de las trayectorias para el controlador AN-PID y un controlador PID con ganancias estáticas.

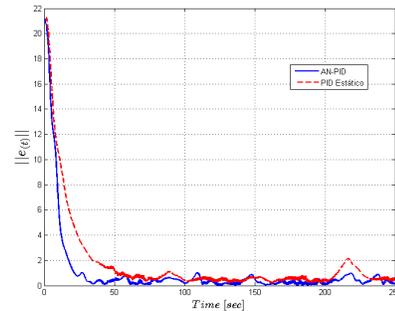


Figura 6. Norma euclidiana del error de control durante la ejecución de la tarea.

Durante el desarrollo de la misión el PID neuronal modifica el valor de sus ganancias en función del error de la variable de control en cuestión. En cambio, las ganancias del PID estático fueron definidas al inicio de la tarea y por más que el vehículo y las condiciones de vuelo cambien, el sistema de control ya está definido. En 7, 8, 9 y 10 se presentan la evolución temporal de las variables  $x, y, z$  y  $\psi$ , respectivamente. El objetivo es ver la evolución para cada una de las variables a controlar de manera individual, dado que la norma del error presentado anteriormente sintetiza en demasía el desempeño de ambos sistemas de control.

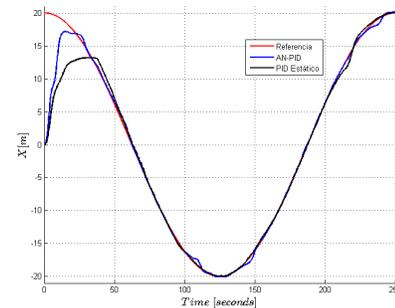


Figura 7. Valor de  $x$  durante el desarrollo de la tarea, utilizando el controlador AN-PID (azul) y un PID estático (negro).

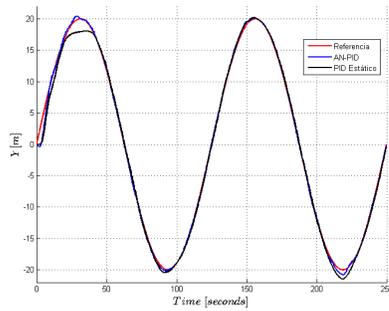


Figura 8. Valor de  $y$  durante el desarrollo de la tarea, utilizando el controlador AN-PID (azul) y un PID estático (negro).

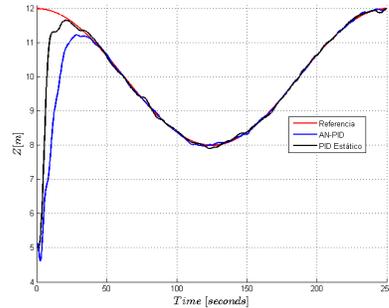


Figura 9. Valor de  $z$  durante el desarrollo de la tarea, utilizando el controlador AN-PID (azul) y un PID estático (negro).

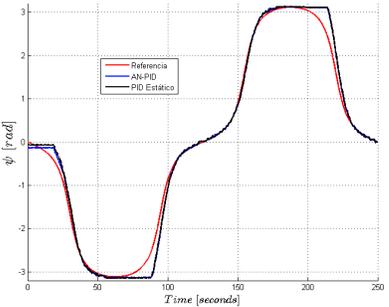


Figura 10. Valor de  $\psi$  durante el desarrollo de la tarea, utilizando el controlador AN-PID (azul) y un PID estático (negro).

En las Figuras anteriores se observa claramente la mejora que representa el controlador propuesto en la sección IV frente a un controlador clásico PID con ganancias fijas.

El estimador neuronal propuesto en la sección III utilizado para el cálculo de las acciones de control se valida en las Figs. 11, 12, 13 y 14 representan la comparativa entre la dinámica identificada por el estimador neuronal basado en EKF y la dinámica real de la aeronave.

Las perturbaciones no deseadas que actúan sobre el modelo van desde dinámicas no modeladas hasta variaciones en los parámetros del vehículo aéreo. Los resultados presentados demuestran la robustez del controlador neuronal PID adaptativo. La estabilidad del sistema de lazo cerrado se demostró analíticamente a través de la teoría de la estabilidad de tiempo discreto de Lyapunov (Sección 5). Esta propuesta de control inteligente puede considerarse como una solución general para el control de sistemas no lineales y en particular

para el caso de sistemas UAV o cuando la dinámica es variable o tiene incertidumbres en el modelo.

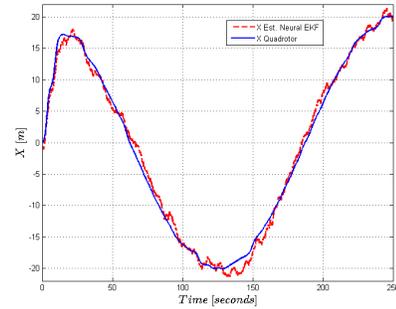


Figura 11. Evolución de la dinámica para  $x$  estimada por el identificador neuronal y la real dinámica del cuadricóptero.

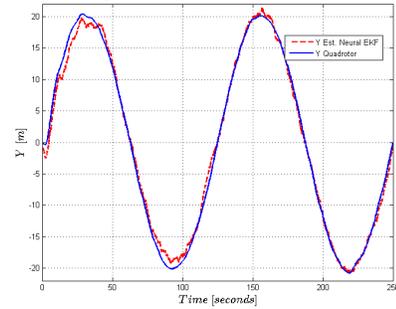


Figura 12. Evolución de la dinámica para  $y$  estimada por el identificador neuronal y la real dinámica del cuadricóptero.

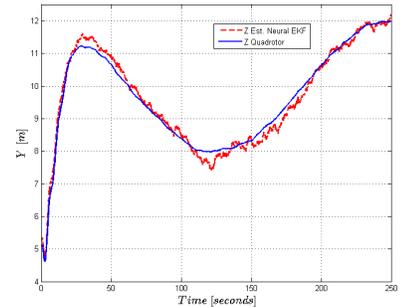


Figura 13. Evolución de la dinámica para  $z$  estimada por el identificador neuronal y la real dinámica del cuadricóptero.

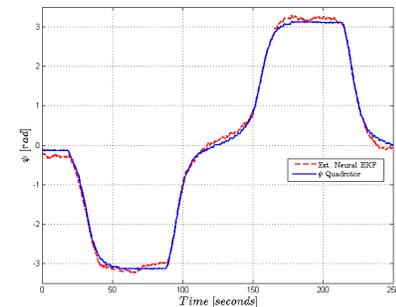


Figura 14. Evolución de la dinámica para  $\psi$  estimada por el identificador neuronal y la real dinámica del cuadricóptero.

Esta área de investigación está abierta a las cuestiones de modelado, estabilidad de la ley de control, convergencia y análisis de robustez de los sistemas de control con el fin de diseñar controladores cada vez más precisos.

## VII. CONCLUSIONES

El controlador PID neural adaptativo propuesto resuelve el problema de modificar las ganancias de un controlador PID con el objetivo de minimizar los errores de control. Se demostró que el sistema de lazo cerrado finalmente uniformemente estable (UUB). Para retro-propagar los errores de control se utilizó una red neuronal RBF y así poder ajustar las ganancias del controlador. Debido a la capacidad de adaptación del controlador, las ganancias de los PID pueden ajustarse a cualquier variación de la dinámica del UAV y de esta forma mejorar el desempeño del sistema de control.

El AN-PID propuesto se aplicó en un vehículo aéreo de cuatro rotores (quadrotor) mejorando el rendimiento original en comparación con un PID clásico de ganancias fijas. Los resultados experimentales han demostrado un buen desempeño del controlador PID neural adaptativo y su adaptación a la dinámica del UAV. Por último, la ley de control propuesta y el modelo de identificación pueden aplicarse a cualquier sistema con dinámicas no lineales y desconocidas.

## REFERENCIAS

- [1] Islam, S., Liu, P. X., & El Saddik, A. (2015). Robust control of four-rotor unmanned aerial vehicle with disturbance uncertainty. *IEEE Transactions on Industrial Electronics*, 62(3), 1563-1571.
- [2] Bouabdallah, S., Noth, A., & Siegwart, R. (2004, September). PID vs LQ control techniques applied to an indoor micro quadrotor. In *Intelligent Robots and Systems, 2004.(IROS 2004). Proceedings. 2004 IEEE/RSSJ International Conference on* (Vol. 3, pp. 2451-2456). IEEE.
- [3] Rosales, C., Gandolfo, D., Scaglia, G., Jordan, M., & Carelli, R. (2015). Trajectory tracking of a mini four-rotor helicopter in dynamic environments—a linear algebra approach. *Robotica*, 33(08), 1628-1652.
- [4] Raffo, G. V., Ortega, M. G., & Rubio, F. R. (2008). MPC with Nonlinear  $\mathcal{H}_\infty$  Control for Path Tracking of a Quad-Rotor Helicopter. *IFAC Proceedings Volumes*, 41(2), 8564-8569.
- [5] Reinoso, M. J., Minchala, L. I., Ortiz, P., Astudillo, D. F., & Verdugo, D. (2016). Trajectory tracking of a quadrotor using sliding mode control. *IEEE Latin America Transactions*, 14(5), 2157-2166.
- [6] Bristeau, P. J., Callou, F., Vissiere, D., & Petit, N. (2011). The navigation and control technology inside the ar. drone micro uav. *IFAC Proceedings Volumes*, 44(1), 1477-1484.
- [7] Xiong, J. J., & Zhang, G. B. (2017). Global fast dynamic terminal sliding mode control for a quadrotor UAV. *ISA transactions*, 66, 233-240.
- [8] Khan, H. S., & Kadri, M. B. (2014, December). Position control of quadrotor by embedded PID control with hardware in loop simulation. In *Multi-Topic Conference (INMIC), 2014 IEEE 17th International* (pp. 395-400). IEEE.
- [9] Sa, R. C., De Araujo, A. L. C., Varela, A. T., & Barreto, G. D. A. (2013, October). Construction and PID control for stability of an unmanned aerial vehicle of the type quadrotor. In *Robotics Symposium and Competition (LARS/LARC), 2013 Latin American* (pp. 95-99). IEEE.
- [10] An, S., Yuan, S., & Li, H. (2016, August). Self-tuning of PID controllers design by adaptive interaction for quadrotor UAV. In *Guidance, Navigation and Control Conference (CGNCC), 2016 IEEE Chinese* (pp. 1547-1552). IEEE.
- [11] Chen, Y. M., He, Y. L., & Zhou, M. F. (2015). Decentralized PID neural network control for a quadrotor helicopter subjected to wind disturbance. *Journal of Central South University*, 22, 168-179.
- [12] ul Amin, R., Aijun, L., Khan, M. U., Shamsirband, S., & Kamsin, A. (2017). An adaptive trajectory tracking control of four rotor hover vehicle using extended normalized radial basis function network. *Mechanical Systems and Signal Processing*, 83, 53-74.
- [13] Wang, S., Li, B., & Geng, Q. (2013, June). Research of RBF neural network PID control algorithm for longitudinal channel control of small UAV. In *Control and Automation (ICCA), 2013 10th IEEE International Conference on* (pp. 1824-1827). IEEE.
- [14] Chen, Y. M., He, Y. L., & Zhou, M. F. (2015). Decentralized PID neural network control for a quadrotor helicopter subjected to wind disturbance. *Journal of Central South University*, 22, 168-179.
- [15] Rossomando, F. G., & Soria, C. M. (2015). Design and Implementation of Adaptive Neural PID for Non Linear Dynamics in Mobile Robots. *IEEE Latin America Transactions*, 13(4), 913-918.
- [16] Santana, L. V., Brandao, A. S., Sarcinelli-Filho, M., & Carelli, R. (2014, May). A trajectory tracking and 3d positioning controller for the ar. drone quadrotor. In *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on* (pp. 756-767). IEEE.
- [17] Krajník, T., Vonásek, V., Fišer, D., & Faigl, J. (2011, June). AR-drone as a platform for robotic research and education. In *International Conference on Research and Education in Robotics* (pp. 172-186). Springer Berlin Heidelberg.
- [18] Hernandez, A., Copot, C., De Keyser, R., Vlas, T., & Nascu, I. (2013, October). Identification and path following control of an AR. Drone quadrotor. In *System Theory, Control and Computing (ICSTCC), 2013 17th International Conference* (pp. 583-588). IEEE.
- [19] Santana, L. V., Brandão, A. S., & Sarcinelli-Filho, M. (2016). Navigation and cooperative control using the ar. drone quadrotor. *Journal of Intelligent & Robotic Systems*, 84(1-4), 327-350.
- [20] Santos, M. C. P., Rosales, C. D., Sarapura, J. A., Sarcinelli-Filho, M., & Carelli, R. (2018). An Adaptive Dynamic Controller for Quadrotor to Perform Trajectory Tracking Tasks. *Journal of Intelligent & Robotic Systems*, 1-12.
- [21] Rossomando, F. G., Soria, C., & Carelli, R. (2013). Algoritmo compensador neuronal discreto de dinámica en robots móviles usando filtro de Kalman extendido. *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería*, 29(1), 12-20.
- [22] Song, Y., & Grizzle, J. W. (1992, June). The extended Kalman filter as a local asymptotic observer for nonlinear discrete-time systems. In *American Control Conference, 1992* (pp. 3365-3369). IEEE.
- [23] Alanis, A. Y., Sanchez, E. N., & Ricalde, L. J. (2010). Discrete-time reduced order neural observers for uncertain nonlinear systems. *International journal of neural systems*, 20(01), 29-38.
- [24] Sanchez, E. N., Alanis, A. Y., & Loukianov, A. G. (2008). Discrete-time high order neural control: trained with Kalman filtering (Vol. 112). Springer Science & Business Media.
- [25] Ardupilot, <http://ardupilot.org/>.
- [26] DroneKit-Python, <http://python.dronekit.io/>.



**Claudio D. Rosales**, nació en San Juan en 1984. Egresó de la Universidad Nacional de San Juan como Ingeniero Electrónico en 2009, en 2014 obtuvo el Doctorado en Ingeniería de Sistemas de Control y en 2018 el grado de Doctor en ingeniería eléctrica en la Universidad Federal de Espirito Santo, ES, Brasil, en 2018. Actualmente es y profesor de la Universidad Nacional de San Juan. Sus intereses son algoritmos para navegación de sistemas multi-robot, control no lineal y robótica aérea.



**Santiago R. Tosetti**, nació en San Luis, Argentina. Se graduó como Ingeniero Electrónico en 2003 y completó su Doctorado en Sistemas de Control en 2009, ambos de la Universidad Nacional de San Juan, Argentina. Actualmente trabaja como investigador en el Consejo Nacional de Investigación Científica y Técnica (CONICET). Sus intereses de investigación incluyen Robótica, Agricultura de Precisión y Sistemas de Control.



**Carlos M. Soria**, Carlos Miguel Soria, nació en Tucumán, Argentina. Se graduó como ingeniero eléctrico en la Facultad de Ciencias de la Universidad Nacional de Tucumán (UNT) in 1996. En 2000 se graduó como magister en Ingeniería de sistemas de Control en la facultad de Ingeniería de la Universidad Nacional de San Juan (UNSJ). Obtuvo el grado de doctor en Ingeniería en Sistemas de Control en la misma Universidad. Actualmente es Profesor en la Universidad Nacional de San Juan e Investigador de CONICET, Argentina.



**Francisco Guido Rossomando**, nació en San Juan, Argentina. Se graduó en ingeniería electrónica y magister en ingeniería de sistemas de control en la Universidad Nacional de San Juan (UNSJ), Argentina, en 1997 y 2002 respectivamente. Obtuvo el grado de Doctor en ingeniería eléctrica orientación sistemas de control en la Universidad Federal de Espirito Santo, ES, Brasil, en 2006. Realizo su post-doctorado en la UNSJ en 2008, a través de una beca posdoctoral del Consejo Nacional de Investigaciones Científicas y Tecnológicas (CONICET). Actualmente es investigador adjunto del CONICET. Sus áreas de investigación son control de robots, control adaptable, identificación y control de procesos industriales.