

Identificación y control de un Vehículo Aéreo no Tripulado tipo Quadcopter

Eduardo Giraldo Suarez
Universidad Tecnológica de Pereira
egiraldos@utp.edu.co

Pablo Andrés Muñoz Gutiérrez
Universidad del Quindío
pamuoz@uniquindio.edu.co

Jhon Abraham Bonilla Becerra
Universidad Tecnológica de Pereira
jabonilla@utp.edu.co

(Tipo de Artículo: Investigación Ciencia y Tecnología. Recibido el 16/10/2015. Aprobado el 15/04/2016)

Resumen. En este artículo se describe el modelo dinámico de un vehículo aéreo no tripulado tipo QuadCopter, se linealiza alrededor de un punto de operación y se aplica un controlador por realimentación de estados que logra llevar el sistema a un estado estable en 100ms. Se aplica un algoritmo de identificación MIMO por mínimos cuadrados al modelo no linealizado obteniendo un error medio cuadrático de 0.2562 grados en una simulación de 50s; se aplica un controlador LQR que logra controlar e identificar el sistema inicial en un tiempo de 3.7s y se compara con un controlador PID convencional obteniendo como resultado un menor tiempo de establecimiento y mejor seguimiento de referencia en el controlador LQR.

Palabras clave. Control, Identificación de sistemas, QuadCopter, Sistemas MIMO.

Identification and control of an Unmanned Aerial Vehicle Quadcopter

Abstract. In this article a dynamic model of an UAV QuadCopter is described, it is linearized around an operating point and a state feedback algorithm is applied which manages to arrive to a steady state in 100ms. A least square MIMO identification algorithm is applied to the nonlinear model obtaining a mean square error of 0.2562 degrees in a 50 seconds simulation; subsequently an LQR algorithm is applied which achieves to identify and control the initial system in 3.7s The LQR algorithm is compared to a conventional PID controller obtaining as result a shorter stablishing time and better reference tracking in the LQR controller.

Keywords. Control, System identification, QuadCopter, MIMO systems.

Nomenclatura

$V = [U, V, W]^T$: Velocidad lineal del UAV

$w = [P, Q, R]^T$: Velocidad angular del UAV

$\Theta = [\phi, \theta, \psi]^T$: Ángulos de Euler

1. Introducción

El vuelo aéreo no tripulado es un área en constante desarrollo debido al gran número de aplicaciones de los vehículos aéreos no tripulados (UAV's); los sistemas de control tradicionales implementados en vehículos de bajo costo son controladores PID que son vulnerables a posibles perturbaciones internas y externas al vehículo tales como el desgaste de la batería, la rotura de una hélice, añadir una carga desconocida, las condiciones atmosféricas entre otras. [1] [2]

El UAV objeto de estudio es un QuadCopter, este es un vehículo en forma de "X" con 4 rotores que controlan su orientación y su posición, es una planta altamente no lineal con inherente efecto de acoplamiento.

Para realizar un buen controlador es necesario tener un buen conocimiento del sistema; al realizar un modelo matemático, principalmente en un sistema no lineal, se obtiene un alto grado de incertidumbre ya que es complejo modelar todas las perturbaciones que afectan al sistema, como el efecto de cercanía a tierra, ráfagas de viento y turbulencia, además este está propenso a variar con el tiempo. Para darle una solución a este problema este artículo se centra principalmente en el método de identificación para el UAV.

Además de los clásicos PID, diferentes técnicas de control e identificación se han implementado a un QuadCopter; en [3] se implementa un controlador por paso atrás con lógica difusa (BFL) y un controlador por

paso atrás con mínimos cuadrados (BLMS), estos se compararon y se determinó que el desempeño del controlador BFL fue mejor que el de mínimos cuadrado ya que este último presentó una respuesta 6 o 7 veces más lenta.

En [2] se utilizan redes neuronales artificiales para la identificación de los parámetros del UAV, calculando los pesos de la red con el método de Levenberg Marquardt y validando los datos con simulación "Hardware in the Loop" en tiempo real; se calcula el error mediante el criterio del error medio cuadrático dando como resultado un RMSE de 0.1729°, 0.0869° y 0.0495° para el roll, pitch y yaw respectivamente.

En [4] se implementa una red neuronal modificada en la que se utilizan algoritmos de mínimos cuadrados recursivos y gradiente descendente para actualizar los parámetros de la red, en este estudio se realizaron mediciones encontrando un error medio cuadrático de 0.018 grados entre el pitch actual y el estimado por medio de la red.

En [5] los autores se enfocan en la estimación de la dinámica de un QuadCopter mediante giroscopios a bordo y un conjunto de 3 cámaras que detectan leds infrarrojos adheridos al vehículo, se utilizan mínimos cuadrados para realizar una predicción del comportamiento del QuadCopter y así contrarrestar la latencia de las cámaras; finalmente utilizan filtros de Kalman para fusionar la información de los giroscopios y el sistema de visión para así obtener el modelo.

En [6] se diseña un sistema de control para la estabilidad y orientación de un QuadCopter, se realiza su modelamiento usando los ángulos de Euler, se identificó un modelo paramétrico aplicando el método de la predicción del error que se expresó en espacio de

estados, se aplicaron controladores PID por ganancia programada y PID autosintonizado por lógica difusa los cuales fueron de tipo SISO.

En [7] se utilizan controladores PID para controlar las velocidades angulares de un vehículo aéreo no tripulado, sobre las cuales aplicando un controlador proporcional se logra controlar su estabilidad.

En [1] se realiza una identificación paramétrica de un QuadCopter recolectando la información de pruebas de vuelo reales con una plataforma PX4 y XBee, se realiza un pre procesamiento con un filtro de Kalman y posteriormente se utilizan algoritmos genéticos para hallar los parámetros; para la validación de los datos miden el error entre los parámetros previamente conocidos y los hallados en la identificación.

En el presente artículo se implementarán métodos de identificación paramétrica como el algoritmo de mínimos cuadrados, además se implementaran controladores por realimentación de estados y regulador óptimo cuadrático (LQR). Para probar y verificar los algoritmos de identificación y control se usará una planta de un QuadCopter simulada en un entorno Matlab-Simulink la cual se puede encontrar en [8] y lleva por nombre "Quadcopter Dynamic Modeling and Simulation".

En este artículo se presentará el modelo dinámico del QuadCopter en la sección 2, en las sección 3 se aplicarán el algoritmo de identificación, la sección 4 estará dedicada al diseño y simulación de los controladores se mostrarán resultados y diferencias entre estos y finalmente se presentan algunas conclusiones en la sección 5.

2. Materiales y métodos

2.1 Estructura general del controlador

La estructura general del sistema de control de un QuadCopter se muestra en la Figura 1. Las variables Roll, Pitch, Yaw, Throttle hacen referencia al alabeo, cabeceo, guiñada y aceleración respectivamente. El mezclador situado a la salida del sistema de control combina las entradas para obtener una salida para cada uno de los 4 motores del QuadCopter; la unidad de medición inercial (IMU) ligada al vehículo mide la dinámica de este y realimenta el sistema de control.

2.2 Modelamiento matemático

Inicialmente el QuadCopter es una planta en la que se controlan 3 variables (Roll, Pitch, Yaw) cambiando la velocidad en sus 4 motores. Para facilitar el problema de control, este se puede convertir en un sistema de 3 entradas, 3 salidas agregando un mezclador que relacione el Roll, Pitch, Yaw deseados con sus salidas [8].

Es muy importante tener en cuenta las posibles configuraciones que puede adoptar un QuadCopter la más común y estudiada en este artículo es la configuración "X", aunque también puede adoptar una configuración "+". Para la configuración estudiada, la matriz que relaciona las entradas con las salidas se define como:

$$\begin{bmatrix} \Sigma T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} Ct & Ct & Ct & Ct \\ -d * Ct & d * Ct & d * Ct & -d * Ct \\ -d * Ct & -d * Ct & d * Ct & d * Ct \\ -Cq & Cq & -Cq & Cq \end{bmatrix} \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix} \quad (1)$$

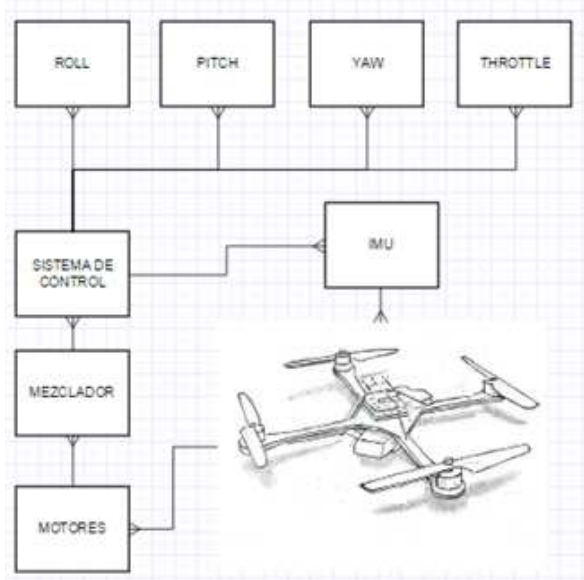


Figura 1. Esquema general de control de un QuadCopter

Es muy importante tener en cuenta las posibles configuraciones que puede adoptar un QuadCopter la más común y estudiada en este artículo es la configuración "X", aunque también puede adoptar una configuración "+". Para la configuración estudiada, la matriz que relaciona las entradas con las salidas se define como:

$$\begin{bmatrix} \Sigma T \\ \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} Ct & Ct & Ct & Ct \\ -d * Ct & d * Ct & d * Ct & -d * Ct \\ -d * Ct & -d * Ct & d * Ct & d * Ct \\ -Cq & Cq & -Cq & Cq \end{bmatrix} \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix} \quad (1)$$

Donde Cq es el coeficiente de torque que se asume igual para todos los motores del QuadCopter, $d = dist * \sin(45^\circ)$, $dist$ la distancia desde cada motor al centro del vehículo, Ct el coeficiente de empuje (Thrust), w_i la velocidad de cada motor y ΣT el empuje total [8].

Para describir el momento de inercia del QuadCopter se utiliza la matriz de momentos de inercia; asumiendo que el vehículo es completamente simétrico, la matriz está dada por:

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (2)$$

Sobre el QuadCopter también actúan fuerzas giroscópicas, estas son generadas por la precesión giroscópica de cada uno de los rotores. Estas se denotan como Gp_ϕ y Gp_θ para el alabeo y cabeceo respectivamente [8].

La ecuación de estado para la velocidad angular se define como:

$$\begin{bmatrix} \dot{P} \\ \dot{Q} \\ \dot{R} \end{bmatrix} = I^{-1} [M - \Omega I \begin{bmatrix} P \\ Q \\ R \end{bmatrix}] \quad (3)$$

Donde M es la matriz que contiene todos los momentos del sistema y está definida por:

$$M = \begin{bmatrix} \tau_\phi + Gp_\phi \\ \tau_\theta + Gp_\theta \\ \tau_\psi \end{bmatrix} \quad (4)$$

Ω es el tensor para la velocidad angular y está definido por:

$$\Omega = \begin{bmatrix} 0 & -R & Q \\ R & 0 & -P \\ -Q & P & 0 \end{bmatrix} \quad (5)$$

La posición angular del QuadCopter se definirá utilizando los ángulos de Euler, estos forman un conjunto de 3 coordenadas que sirven para representar la orientación de un sistema de ejes ortogonales respecto a otro sistema de ejes ortogonales normalmente fijos, donde ϕ representa el alabeo que es la rotación en el eje longitudinal "x", θ representa el cabeceo que es la rotación en el eje transversal "y", y finalmente ψ que es la rotación en el eje vertical "z" [9].

Estas rotaciones se representan mediante la matriz de rotación R con respecto a un eje de coordenadas en tierra, donde C=Coseno y S=Seno [10].

$$R = \begin{bmatrix} C\psi C\theta & C\psi C\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ S\psi C\theta & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\phi & C\theta C\phi \end{bmatrix} \quad (6)$$

Por convención de la secuencia de rotación aeroespacial, la rotación de una aeronave se puede expresar como la rotación del eje z, seguido a la rotación del eje y, y por último la rotación en el eje x; usando las matrices de rotación secuencial se obtiene la ecuación cinemática de los ángulos de Euler:

$$\dot{\theta} = \begin{bmatrix} 1 & T\theta S\phi & T\theta C\phi \\ 0 & C\phi & -S\phi \\ 0 & S\phi/C\theta & C\phi/C\theta \end{bmatrix} \begin{bmatrix} P \\ Q \\ R \end{bmatrix} \quad (7)$$

Estas ecuaciones se utilizan en [8] para simular un QuadCopter en un entorno Matlab-Simulink.

2.3 Algoritmo de mínimos cuadrados

Mínimos cuadrados es un método de análisis numérico que permite obtener un ajuste (de acuerdo al criterio del error mínimo cuadrático) entre un conjunto de pares ordenados y una función.

Para un sistema MIMO, en el caso de 4 entradas, 3 salidas, los vectores $y(k)$ y $u(k)$ son de la siguiente forma:

$$y(k) = \begin{bmatrix} y_1(k) \\ y_2(k) \\ y_3(k) \end{bmatrix} \quad u(k) = \begin{bmatrix} u_1(k) \\ u_2(k) \\ u_3(k) \\ u_4(k) \end{bmatrix} \quad (8)$$

Donde $y(k)$ es el vector de salidas del sistema y $u(k)$ el vector de entradas, además el sistema se puede describir como:

$$y(k) = B_1 u(k-1) + B_2 u(k-2)$$

$$-A_1 y(k-1) - A_2 y(k-2) \quad (9)$$

Donde $A_{i(p \times p)}, B_{i(p \times q)}$ (p entradas, q salidas) son los parámetros a identificar.

Reescribiendo la ecuación de acuerdo al modelo DARMA se tiene:

$$y(k) = \theta^T \phi(k-1) \quad (10)$$

Donde θ es la matriz de parámetros y $\phi(k-1)$ es el vector de regresiones, m y n el número de regresiones de u y y respectivamente

$$\theta^T = \begin{bmatrix} B_1^T \\ B_m^T \\ -A_1^T \\ -A_n^T \end{bmatrix}, \quad \phi(k-1) = \begin{bmatrix} u(k-1) \\ u(k-m) \\ y(k-1) \\ y(k-n) \end{bmatrix} \quad (11)$$

La identificación del sistema, que consiste en hallar los parámetros de θ que ajustan el modelo se realiza de acuerdo a la siguiente ecuación:

$$\hat{\theta}(k) = \hat{\theta}(k-1)M(k-1)\phi(k-1)e(k) \quad (12)$$

Con $M(k-1)$ que es la matriz de ganancia y se calcula como sigue:

$$M(k-1) = \frac{P(k-1)}{1 + \phi(k-1)^T P(k-1) \phi(k-1)} \quad (13)$$

Siendo P la matriz de covarianza definida por:

$$P(k) = P(k-1) - M(k-1)\phi(k-1)\phi(k-1)^T P(k-1) \quad (14)$$

(14)

Cabe aclarar que las matrices $P_{((m \times q + n \times p), (m \times q + n \times p))}$ y $\theta_{((m \times p + n \times q), p)}$ son iniciadas con valores aleatorios [11].

2.4 Algoritmos de control

2.4.1 Controlador por realimentación de estados

Se linealizó la planta encontrada en [8] alrededor de un punto de operación con la herramienta "Linear Analysis" de Matlab; esta arroja el sistema en espacio de estados en tiempo continuo, por lo que posteriormente se discretizó con un tiempo de muestreo de 10ms; el sistema resultante es de 10 estados, 3 entradas, 3 salidas.

La Figura 2 muestra el diagrama de un regulador de espacio de estados implementado; la ganancia de realimentación K se halló con asignación de estructuras propias según los siguientes polos:

$$Pd = [-0.2 \quad -0.2 \quad -0.2 \quad -0.2 \quad 0.38 \quad 0.38 \quad 0.38 \quad 0.2 \quad 0.2 \quad 0.2]$$

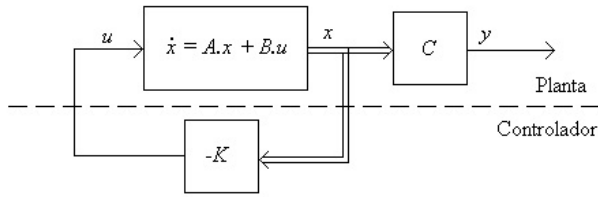


Figura 2. Regulador de Estados

Para lograr que el sistema siga una referencia es necesario introducir la ganancia de realimentación Kg que se halla mediante la ecuación (14).

$$Kg = (C * (I_{10 \times 10} - A + B * K)^{-1} * B + D)^{-1} \quad (15)$$

2.4.2 Modelo de espacio de estados extendido

A partir de la matriz de parámetros θ obtenida de la identificación por mínimos cuadrados, se puede obtener un modelo de espacio de estados extendido para el sistema donde las matrices F , G , C se obtienen según las ecuaciones.

$$F = \begin{bmatrix} \theta^T_{3 \times 12} & & & \\ I_{3 \times 3} & \text{ceros}_{3 \times 3} & \text{ceros}_{3 \times 3} & \text{ceros}_{3 \times 3} \\ \text{ceros}_{3 \times 3} & \text{ceros}_{3 \times 3} & \text{ceros}_{3 \times 3} & \text{ceros}_{3 \times 3} \\ \text{ceros}_{3 \times 3} & \text{ceros}_{3 \times 3} & I_{3 \times 3} & \text{ceros}_{3 \times 3} \end{bmatrix}$$

$$G = \begin{bmatrix} \text{ceros}_{6 \times 3} \\ I_{3 \times 3} \\ \text{ceros}_{3 \times 3} \end{bmatrix}$$

$$C = [\theta^T_{3 \times 12}] \quad [11].$$

2.4.3 Controlador por reubicación de polos

Se diseñó y simuló un controlador (sobre el modelo no lineal) por reubicación de polos usando un LQR (*Linear Quadratic Regulator*); este calcula la matriz de ganancia K de forma óptima minimizando

$$J = \sum_{n=0}^{\infty} \{x^T Q x + u^T R u + 2x^T N u\} \quad (16)$$

Sujeto a

$$x[n + 1] = Ax[n] + Bu[n] \quad (17)$$

Donde R es una matriz identidad de $p \times p$ multiplicada por una ganancia; para este caso la matriz R seleccionada es:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Q es una matriz diagonal del número de estados del sistema donde se da peso a los estados que se desean controlar, para el caso de estudio la matriz Q seleccionada es la siguiente

$$Q = \begin{bmatrix} 1200 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1200 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1200 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

La ganancia óptima de regulación K se calculó mediante el comando `dlqr` de Matlab de acuerdo a (18)

$$K = \text{dlqr}(F, G, Q, R) \quad (18)$$

Para que el sistema siga una referencia es necesario introducir la ganancia Kg que se calcula según (19)

$$Kg = (C * (I_{12 \times 12} - F + G * K)^{-1} * G)^{-1} \quad (19)[12].$$

Este controlador se implementó para controlar las velocidades angulares del vehículo para posteriormente lograr su estabilización con un controlador proporcional.

3. Resultados y discusión

3.1 Identificación por mínimos cuadrados

Se realizó la simulación de la identificación del pitch con una señal cuadrada de 2Hz y amplitud 0.1745 Radianes como referencia; esta simulación se corrió por 50 segundos y se indexó un vector que contiene los valores de salida de la planta y los valores de la identificación, con un tiempo de muestreo de 10ms.

En estadística, el error medio cuadrático mide el promedio de los errores al cuadrado entre un estimador y el valor que se está estimando, el RMSE se define como:

$$RMSE = \frac{1}{n} \sum_{i=1}^n (\hat{Y} - Y)^2$$

Donde \hat{Y} son los datos arrojados por el algoritmo de mínimos cuadrados y Y son las mediciones realizadas directamente a la planta; aplicando este criterio se obtuvo un error medio cuadrático de 0.2562° entre el pitch actual y el estimado por medio de mínimos cuadrados. En general, la operación normal de un QuadCopter no sobrepasa los 45° [7] por tanto si el rango de operación es de $[-45^\circ, 45^\circ]$ el error de estimación sería del 0.28%.

En la planta simulada, se realizó el cambio de masa y distancia de cada uno de los rotores al plato central del QuadCopter y se corrió nuevamente la simulación con las mismas condiciones del primer experimento, se aplicó nuevamente el criterio RMSE obteniendo un error medio de 0.2234° , esto quiere decir que la identificación por mínimos cuadrados efectivamente logra adaptarse a cambios en la planta.

En Figura 3, Figura 4 y Figura 5 se evidencia el funcionamiento del algoritmo de identificación al graficarse la salida de la planta que está representada por la línea sólida vs su identificación, representada por la línea a trozos; en una ampliación de las gráficas se puede observar el error de estimación que rápidamente se reduce en los siguientes instantes de tiempo, representando cada vez con mayor fidelidad el sistema.

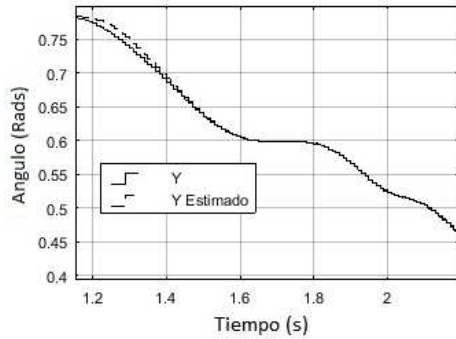


Figura 3. Identificación Roll por mínimos cuadrados

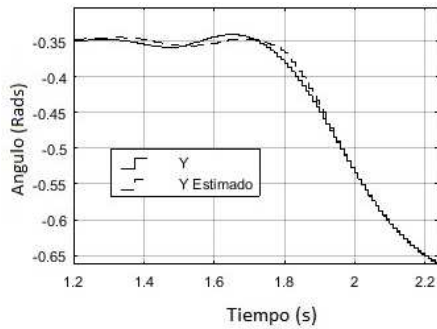


Figura 4. Identificación Pitch por mínimos cuadrados

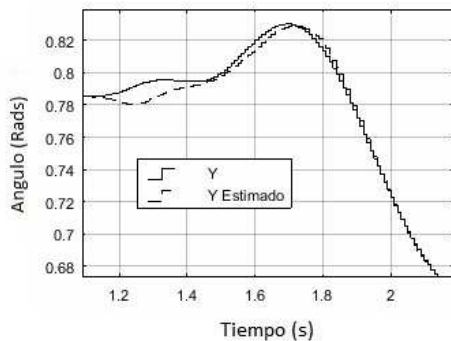


Figura 5. Identificación Yaw por mínimos cuadrados

3.2 Control por realimentación de estados

En la Figura 6 se muestra cómo el controlador por realimentación de estados logra llevar el sistema linealizado a un estado estable en aproximadamente 100ms.

En la Figura 8 se muestra la identificación y seguimiento de referencia del sistema, se puede ver como el controlador logra identificar y llevar la planta a un estado estable en 3.7s.

3.3 Control LQR

En la Figura 7 se muestra la identificación y regulación del sistema, se puede observar que se logra identificar y regular los ángulos de Euler en un tiempo de 3s.

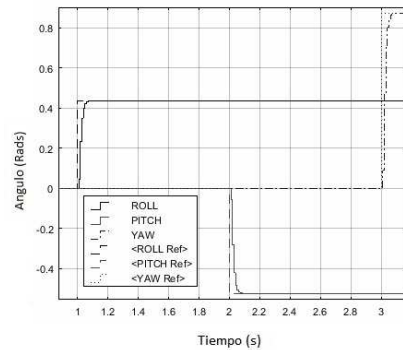


Figura 6 Seguimiento de Referencia Realimentación de Estados

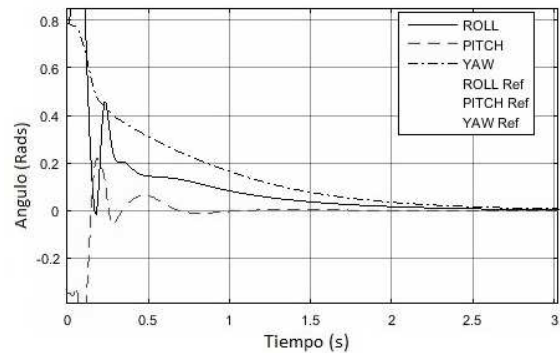


Figura 7. Regulación Ángulos de Euler

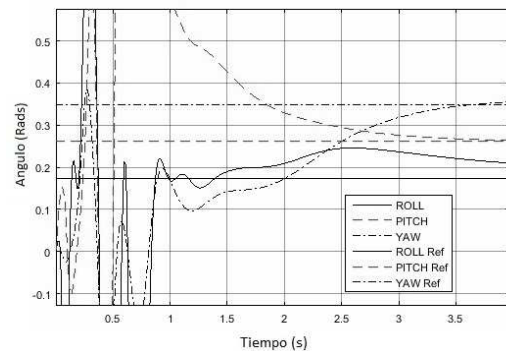


Figura 8. Seguimiento de Referencia

3.4 Comparación controlador PID vs controlador por reubicación de polos

En Figura 9 y Figura 10 se muestra la respuesta de los controladores PID implementado en [8] y controlador LQR después de la identificación inicial, ante una señal cuadrada de amplitud 0.17 Rads y frecuencia 2Hz. Se puede observar la superioridad del controlador LQR frente al PID en el seguimiento de referencia del mismo modelo no lineal.

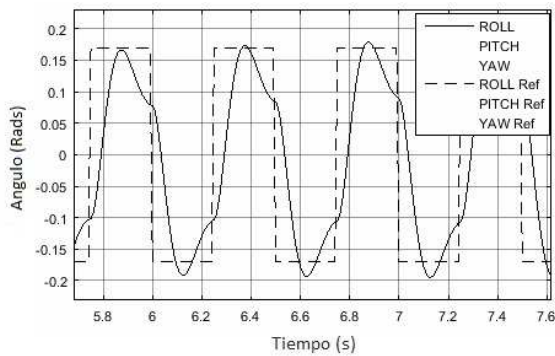


Figura 9. Seguimiento de Referencia Variable Roll

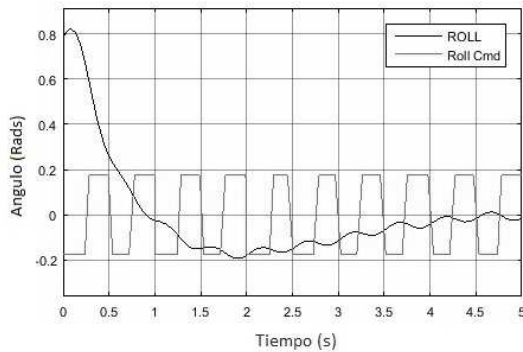


Figura 10. Seguimiento de Referencia PID Variable Roll

4. Conclusiones

El uso de la identificación multivariable por método de mínimos cuadrados combinado con un regulador óptimo cuadrático demostró controlar el sistema de una mejor manera que un controlador PID convencional desacoplado al tener menor tiempo de establecimiento y mejor seguimiento de referencia como se puede observar en *Figura 9* y *Figura 10*. Se comprobó que el método de identificación por mínimos cuadrados representa adecuadamente el sistema al tener un error de 0.2562° por el criterio del error medio cuadrático, lo que significa que si el rango de operación del QuadCopter es de $[-45^\circ - 45^\circ]$ en dos de sus ejes (roll, pitch) se tendría un error de estimación del 0.28%, lo que es suficiente para que el controlador funcione adecuadamente; esto no aplicaría al 3 eje (yaw) ya que este tendría rango de operación infinito. Se comprobó que al cambiar los parámetros de la planta como por ejemplo la masa y distancia desde el HUB (centro) hasta cada uno de los motores, el algoritmo logra identificar correctamente el sistema con un error medio cuadrático de 0.2234° ; aunque este logra identificar fielmente el sistema, se pueden implementar otros algoritmos como

el filtro de Kalman o redes neuronales para obtener errores más bajos como el mostrado en [4].

5. Trabajo Futuro

Como trabajo futuro se plantea implementar el algoritmo de control LQR sobre el modelo real, de esta manera se pueden hacer pruebas y verificar su efectividad ante perturbaciones, también se propone la identificación del sistema por medio de un filtro de Kalman y la utilización algoritmos genéticos para encontrar los pesos más adecuados para la matriz Q .

6. Agradecimientos

Este artículo se desarrolló bajo el proyecto "Identificación de Sistemas Multivariables Aplicada a Generadores Eólicos" de la Universidad Tecnológica de Pereira 6-14-1.

7. Referencias

- [1] Z. C. Q. L. D. Z. Y. W. Jinpeng Yang, «System Identification of Quadrotor UAV Based on Genetic Algorithm», IEEE, p. 5, 2014.
- [2] D. S. G. A. Vishwas R. Puttige, «Real-Time Neural Network Based Online Identification Technique for a UAV Platform», IEEE, p. 6, 2006.
- [3] M. Younes Al-Younes, «Attitude Stabilization of Quadrotor UAV Using Backstepping Fuzzy Logic & Backstepping Least-Mean-Square Controllers», 5th International Symposium on Mechatronics, 2008.
- [4] J. I. Ahsan Javed Awan, «ANFIS Based Attitude Dynamics Identification», SF7.
- [5] M. G. E. a. R. D'Andrea, «Real-Time Attitude Estimation Techniques Applied to a Four Rotor Helicopter», IEEE, 2004.
- [6] A. G. Y. Felipe Jaramillo, «Sistema de Control para la Estabilidad y Orientación de un Helicóptero Quadrotor», Escuela de Ingeniería de Antioquia, Ingeniería Mecatrónica, 2013.
- [7] Ardupilot, «Ardupilot», [En línea]. Available: <http://copter.ardupilot.com/wiki/tuning/>.
- [8] K. L. M. M. S. M. J. K. D. Hartman, «GiftHub», 2014. [En línea]. Available: <https://github.com/dch33/Quad-Sim>.
- [9] A. Barrientos, Fundamentos de Robótica, McGraw-Hill, 1997.
- [10] T. Luukkonen, «Modelling and control of quadcopter», 2011.
- [11] D. A. Eduardo Giraldo, «Adaptive Multivariable Control of a Wind Energy Conversion System Based on Double Feed Induction Generator», p. 5.
- [12] Mathworks, «Mathworks», [En línea]. Available: <http://www.mathworks.com/help/control/ref/lqr.html>. [Último acceso: 01 06 2015].
- [13] M. R. a. K. Khorasani, «Fault Recovery of an Under-Actuated Quadrotor Aerial Vehicle», IEEE, p. 8, 2010.
- [14] C. A. A. C. C. J. D. T. Youmin Zhang, «Flatness-Based Trajectory», IEEE, p. 17, 2012.
- [15] M. W. M. Raffaello D'Andrea, «Stability and control of a quadcopter despite the complete loss of one, two, or three propellers», IEEE, p. 8, 2014.
- [16] E. S. Philippe Martin, «The True Role of Accelerometer Feedback in Quadrotor Control», IEEE, p. 7, 2010.