

## Integración de X-Plane como entorno Model-In-the-Loop para la validación de esquemas de control en VANTs tipo Quadrotor Integration of X-Plane as a Model-In-the-Loop environment for the validation of control schemes in Quadrotor UAVs

I. G. Álvarez-Ortiz <sup>a</sup>, V. G. Sánchez-Meza <sup>b,\*</sup>, Y. Lozano-Hernández <sup>c</sup>, J. González-Sierra <sup>c</sup>, R. Y. Almazan-Arvizu <sup>b</sup>

<sup>a</sup>Universidad Aeronáutica en Querétaro, 76278, Santiago de Querétaro, Querétaro, México.

<sup>b</sup>Sección de posgrado UPIITA, Instituto Politécnico Nacional, 07340, CDMX, México.

<sup>c</sup>Unidad Profesional Interdisciplinaria de Ingeniería Campus Hidalgo (UPIIH), Instituto Politécnico Nacional, 42162, Pachuca, Hidalgo, México.

### Resumen

La dinámica altamente no lineal y las propiedades de vuelo de los Vehículos Aéreos No Tripulados (VANT) tipo Quadrotor continúan despertando el interés de investigadores de diversas áreas. En particular, en el área de control automático se trabaja en el desarrollo de nuevos esquemas de control, de los cuales, muchos no llegan a la etapa de implementación. En ese sentido, las simulaciones Model-In-the-Loop (MIL) resultan atractivas para el desarrollo y comprobación de sistemas de control. Por lo anterior, en este trabajo se muestra el uso MATLAB/Simulink y X-Plane como herramientas de co-simulación para analizar el comportamiento y control de un Quadrotor. Se utiliza un control prealimentado tipo PID para realizar maniobras de vuelo en estado estacionario ante la presencia de diferentes velocidades de viento. Los datos de retroalimentación del esquema de control se obtienen de X-Plane a través de la comunicación basada en el Protocolo de Datagramas de Usuario (por sus siglas en inglés, UDP), estos son enviados a Simulink para su uso en el cálculo de las acciones de control. Posteriormente, las acciones de control pasan por una matriz de transformación que convierte dichas acciones en velocidades angulares de los rotores (información enviada a X-Plane). Por último, se reportan los resultados del esquema control ante diferentes velocidades de viento, siendo 50 Nudos (kt) la máxima velocidad que puede compensar el controlador propuesto.

**Palabras Clave:** Model-In-the-Loop, Simulink, Vehículo Aéreo no Tripulado, Quadrotor, X-Plane.

### Abstract

The highly non-linear dynamics and flight properties of Quadrotor-type Unmanned Aerial Vehicles (UAVs) arouse the interest of researchers in various fields. In particular, in the area of automatic control, work is being done on the development of new control schemes, many of which do not reach the implementation stage. For this reason, Model-In-the-Loop (MIL) simulations are attractive for the development and testing of control systems. Therefore, this paper shows the use of MATLAB/Simulink and X-Plane as co-simulation tools to analyze the behavior and control of a Quadrotor. A PID-type feedforward control is used to perform steady-state flight maneuvers in the presence of different wind speeds. Control scheme feedback data is obtained from X-Plane via User Datagram Protocol (UDP)-based communication, and is sent to Simulink for use in calculating the control schemes control actions. Then the control actions go through a transformation matrix that converts these actions into angular velocities of the rotors (information sent to X-Plane). Finally, the results of the control scheme are reported for different wind speeds, with 50 knots (kt) being the maximum speed that the proposed controller can compensate.

**Keywords:** Model-In-the-Loop, Simulink, Unmanned Aerial Vehicle, Quadrotor, X-Plane.

\* Autor para correspondencia: [vsanchezm1301@alumno.ipn.mx](mailto:vsanchezm1301@alumno.ipn.mx)

**Correo electrónico:** [5456@soyunaq.mx](mailto:5456@soyunaq.mx) (Iván Guadalupe Álvarez-Ortiz), [vsanchezm1301@alumno.ipn.mx](mailto:vsanchezm1301@alumno.ipn.mx) (Victor Gabriel Sánchez Meza), [ylozanoh@ipn.mx](mailto:ylozanoh@ipn.mx) (Yahir Lozano-Hernández), [jagonzalezsi@ipn.mx](mailto:jagonzalezsi@ipn.mx) (Jaime González-Sierra), [ralmazana1700@alumno.ipn.mx](mailto:ralmazana1700@alumno.ipn.mx) (Ricardo Yahir Almazan-Arvizu).

**Historial del manuscrito:** recibido el 01/07/2023, última versión-revisada recibida el 14/09/2023, aceptado el 11/09/2023, publicado el 30/11/2023. **DOI:** <https://doi.org/10.29057/icbi.v11iEspecial4.11406>



## 1. Introducción

Los VANTs son aeronaves sin tripulación a bordo (Wagner y Wolfrum, 2015), que generalmente logran seguir una ruta de vuelo predeterminada. Los VANT surgieron hace años con propósitos militares y hoy en día son ampliamente utilizados, lo cual se debe a su autonomía, ahorro de costos, y en consecuencia, a la ausencia de riesgo por parte del factor humano (Thanh y Hong, 2018).

Actualmente, los VANT tienen aplicaciones en operaciones de vigilancia marítima, detección y extinción de incendios, así como en diversas áreas como fotografía, robótica, agricultura de precisión, entre otros (Sánchez et al., 2020).

Uno de los VANTs con mayor interés es el Quadrotor. Este es propulsado y elevado por cuatro rotores fijos, dichos rotores tienen la misma distancia al centro de masa del Quadrotor. En contraste con un helicóptero de un solo rotor principal, un Quadrotor tiene ciertas ventajas como: tamaño compacto, alta agilidad y maniobrabilidad, suspensión en el aire en un punto fijo, capacidad de Despegue y Aterrizaje Vertical (por sus siglas en inglés, VTOL), y fácil mantenimiento (Li et al., 2015).

Los Quadrotores se utilizan para diversas aplicaciones, desde la recolección de datos de forma remota hasta la entrega de productos/alimentos en áreas locales. Otra aplicación es el mapeo geográfico, que es bastante útil debido a que puede llegar a lugares de difícil acceso, como costas erosionadas o cimas de montañas, y adquirir datos de muy alta resolución para crear mapas en 3D. Además, en la agricultura, se suelen equipar con cámaras para la obtención de imágenes que permitan examinar propiedades del suelo como: nivel de humedad, vegetación y la radiación solar. Esto permite la aplicación de insumos o fertilizantes en áreas específicas, obtenidas del diagnóstico de las imágenes, dando como resultado cultivos más eficientes y de mayor calidad.

Los sistemas de control para VANTs tipo Quadrotor se han desarrollado con gran rapidez y han sido un pilar en el avance de la ciencia y tecnología al lograr el funcionamiento adecuado de sistemas dinámicos (Pérez et al., 2008). Actualmente, la mayoría los esquemas de control que se utilizan son simulados únicamente en MATLAB/Simulink y posteriormente se implementan de manera física. Sin embargo, los resultados físicos suelen ser diferentes a los obtenidos en simulación, lo anterior se debe a la propiedad de subactuación de este vehículo, su dinámica no lineal, la inexactitud o desconocimiento del valor de algunos parámetros como los momentos de inercia y coeficientes aerodinámicos (Zhao et al., 2014).

El uso de MATLAB/Simulink para la simulación de esquemas de control es útil. No obstante, frecuentemente se utilizan modelos matemáticos aproximados del Quadrotor que no contemplan efectos aerodinámicos y/o perturbaciones exógenas, los cuales afectan directamente en el comportamiento del VANT en escenarios reales. Por ello, resulta atractivo integrar simuladores de vuelo utilizados en el sector aeronáutico con MATLAB/Simulink, y de esta forma, tener mayor certeza en el comportamiento esperado por los esquemas de control diseñados. En este sentido, X-Plane es uno de los simuladores con mayor aceptación en el sector aeronáutico y se ha logrado vincular con MATLAB/Simulink.

Referente al uso de X-Plane, se tienen trabajos como (Bittar et al., 2014) en el que se utiliza una ley de control de altitud para

aeronaves pequeñas no tripuladas de ala fija la cual se construye con base en dos fases de un vuelo: vuelo estable y vuelo de maniobra, la simulación experimental se realiza en el software X-Plane. Por otro lado (Aláez et al., 2022), analizan el comportamiento de un VANT tipo VTOL, el cual se realiza a través de integración Hardware in the Loop con todas las funciones de un VANT VTOL comercial, donde el avión se modela en X-Plane y las condiciones climáticas son obtenidas mediante integración propia de datos. En (García y Barnes, 2010) se detalla la creación de un entorno que permita simular de forma simultánea varios VANTs, así como proporcionar datos de estado y entrada de control para los vehículos individuales utilizando X-Plane. Adicionalmente, en (Kaviyarasu y Kumar, 2014) se reporta la simulación de un modelo de VANT de alas batientes utilizando X-Plane y MATLAB/Simulink. El modelo de alas batientes es desarrollado en un software de fabricación de aviones y se ejecuta en el entorno X-Plane.

De lo anterior, (Figueiredo y Saotome, 2012) realiza la simulación de control de vuelo estacionario de un Quadrotor manteniendo una altura deseada y la orientación del vehículo; aunque, los autores no abundan en el diseño del controlador o de las variables utilizadas en la simulación. Por otra parte, (Cho et al., 2015) desarrolla un esquema de control PID convencional para el vuelo estacionario de un Quadrotor en sus 6 ejes de libertad ( $x$ ,  $y$ ,  $z$ ,  $\phi$ ,  $\theta$  y  $\psi$ ). De manera similar, (He et al., 2020) realiza la simulación de despegue de un tiltrotor, generando una transición de modo de vuelo de la aeronave y manteniendo la orientación ante condiciones ideales climáticas. Sin embargo, los artículos anteriores no consideran la comparación entre condiciones de viento para validar la robustez del esquema de control ante perturbaciones climáticas.

Los antecedentes descritos resaltan la necesidad de contar con herramientas de simulación que ayuden a validar esquemas de control propuestos sin necesidad de contar con un prototipo físico o experimental. En ese sentido, resulta atractivo el uso un entorno MIL especializado como X-Plane, el cual es un simulador de vuelo, basado en el cálculo del efecto del flujo de aire sobre las superficies de los aviones, es decir, no requiere de un modelo matemático que describa la dinámica de vuelo del VANT. Así, en este artículo se describe el uso MATLAB/Simulink y X-Plane como herramientas de co-simulación para visualizar el comportamiento de esquemas de control diseñados para Quadrotores ante perturbaciones climáticas que afectan directamente las condiciones de vuelo de los vehículos.

La organización de este trabajo esta dada de la siguiente forma: la Sección 2 describe la configuración requerida para la vinculación y envío de datos a través del protocolo de comunicación basado en UDP entre MATLAB/Simulink y X-Plane; la Sección 3 contiene el modelo matemático utilizado para el diseño del esquema de control, cabe destacar dicho modelo no se ingresa al simulador X-Plane; en la Sección 4 se explica el diseño de un control prealimentado tipo Proporcional Integral Derivativo (PID) para realizar maniobras de vuelo en estado estacionario, lo anterior debido al frecuente uso de este controlador en los Quadrotores; la Sección 5 muestra los resultados del esquema control ante diferentes velocidades de viento, lo anterior para comprobar la funcionalidad de X-Plane y MATLAB como MIL para VANTs tipo Quadrotor. Finalmente, la sección

6 presenta las conclusiones obtenidas.

## 2. Vinculación y envío de datos

X-Plane es uno de los simuladores de vuelo más realistas en el mundo, cuenta con varios modelos de aviones los cuales se pueden simular en diferentes escenarios de vuelo y condiciones ambientales, permite incluso, transmitir y manipular datos de dichas aeronaves. Adicionalmente, X-Plane es capaz de proporcionar una colaboración de hasta 10 vehículos (García y Barnes, 2010) simultáneamente, pero es importante tener presente que con una alta cantidad de vehículos simulados, el ancho de banda disponible en la red disminuirá, dependiendo de la cantidad de datos transferidos. Por otra parte, número de vehículos simulados puede aumentar hasta 20 de acuerdo con el manual de X-Plane 10. También, el software integra Plane Maker, mediante el cual es posible construir y/o modificar aeronaves para propósitos específicos (He *et al.*, 2020; Cerezo-Pacheco *et al.*, 2021).

### 2.1. Plane Maker

Plane-Maker permite a los usuarios diseñar diferentes aeronaves. Los aviones se construyen mediante especificaciones físicas que el diseñador ingresa, por ejemplo: peso, envergadura, potencia del motor, secciones aerodinámicas, etc. Una vez concluido el diseño, el simulador de X-Plane procede a predecir la capacidad de vuelo de ese avión bajo condiciones que emulan el mundo real (Bryan, 2021). El uso de Plane Maker durante el diseño de las aeronaves garantiza el conocimiento de sus parámetros físicos, como el tipo de motor de la aeronave, dimensiones, tipo de hélice, propulsión, relación de motores, entre otros. La Figura 1 muestra el Quadrotor utilizado en este trabajo.

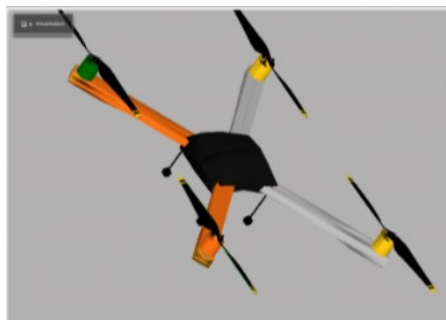


Figura 1: Quadrotor utilizado.

### 2.2. Integración del modelo de Plane Maker a X-Plane

Para la integración del Quadrotor a X-Plane se requiere crear una nueva subcarpeta dentro la carpeta llamada “Aircraft” ubicada en el directorio donde se instaló X-Plane (Este quipo/Escritorio/X-Plane 10/Aircraft/Quadrotor), esta carpeta contiene todas las aeronaves disponibles en el simulador de vuelo, clasificadas en subcarpetas de acuerdo con su función. El usuario puede elegir cualquier subcarpeta dentro de “Aircraft”, y nombrar la carpeta de la aeronave a instalar como desee.

Posteriormente, se ejecuta X-Plane. Al inicio, aparecerá una ventana como la mostrada en la Figura 2, en la cual se selecciona la aeronave a utilizar. Después, se busca la carpeta que

contiene los archivos de la aeronave diseñada, dentro de esta aparece un archivo .acf, el cual corresponde al Quadrotor. La Figura 3 muestra al Quadrotor volando dentro del ambiente de X-Plane.

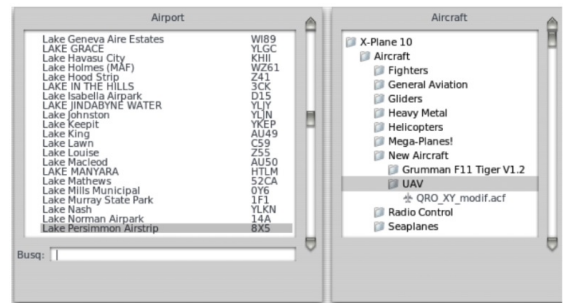


Figura 2: Incorporación del Quadrotor a la base de datos de X-Plane.



Figura 3: Quadrotor operando en el escenario de X-Plane.

### 2.3. Comunicación con MATLAB/Simulink

La transmisión de datos entre X-Plane y MATLAB/Simulink se lleva a cabo mediante comunicación UDP, la cual hace uso de puertos y dirección locales. Lo anterior es posible gracias a que MATLAB/Simulink cuenta con una biblioteca de comunicación UDP, la cual se puede utilizar a través de programación a bloques. Para realizar la comunicación, se debe configurar una dirección IP y los puertos específicos del envío/recepción de los datos, para lo cual es importante tener en cuenta las siguientes consideraciones:

- Dentro de X-Plane, en la sección Ajustes→Conexiones de red→Datos, se introduce la dirección IP (ver Figura 4).
- En la misma sección de X-Plane, se introducen los puertos del envío y recepción de datos (ver Figura 5).

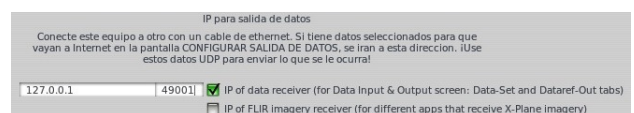


Figura 4: Dirección IP del envío/recepción de datos en X-Plane.

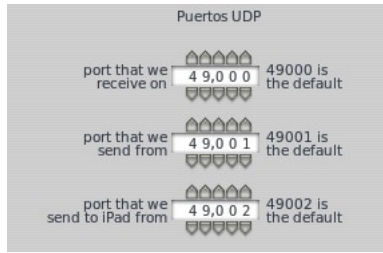


Figura 5: Puertos de comunicación UDP del envío/recepción en X-Plane.

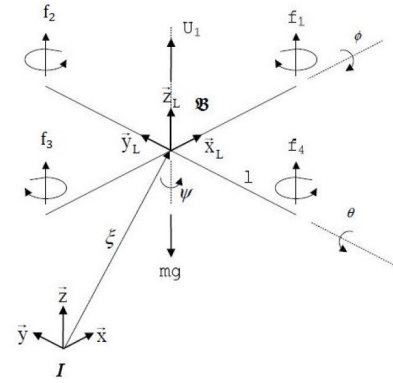


Figura 8: Marcos de referencia del Quadrotor.

Una vez configurada tanto la dirección como los puertos de comunicación, es posible utilizar los bloques UDP de MATLAB/Simulink (ver Figuras 6 y 7) para enviar y recibir datos entre MATLAB/Simulink y X-Plane.

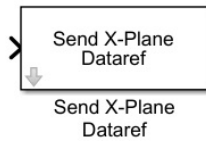


Figura 6: Bloque para envío de datos en MATLAB/Simulink.

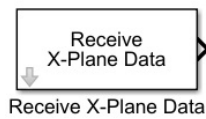


Figura 7: Bloque para recepción de datos en MATLAB/Simulink.

- $f_1, f_2, f_3, f_4$  son las fuerzas de empuje generadas por los rotores.
- $U_1$  es la acción de control sobre el eje  $z$ , resultado de la sumatoria  $f_1 + f_2 + f_3 + f_4$ .
- $l$  corresponde a la longitud del centro del Quadrotor al centro del rotor.
- $\vec{x}_L, \vec{y}_L, \vec{z}_L$  indican la posición del Quadrotor con respecto a un marco de referencia fijo al mismo.
- $\vec{x}, \vec{y}, \vec{z}$  corresponde a la posición debida a un marco de referencia inercial (la tierra).
- $\phi, \theta, \psi$  son los ángulos de orientación del alabeo, cabeceo y guiñada, respectivamente.

### 3.1. Modelo matemático

La dinámica del Quadrotor puede obtenerse mediante la metodología de Euler-Lagrange. Normalmente se considera de dos formas: por su propio marco de referencia, o por un marco de referencia inercial (Lozano Hernández *et al.*, 2019). En este trabajo se contempla un marco de referencia fijo al Quadrotor, y se representa de la siguiente manera:

$$\begin{aligned}
 m\ddot{x} &= -U_1 \sin \theta + a_x \dot{x}, \\
 m\ddot{y} &= U_1 \cos \theta \sin \phi + a_y \dot{y}, \\
 m\ddot{z} &= U_1 \cos \theta \cos \phi - g + a_z \dot{z}, \\
 I_{xx}\ddot{\phi} &= (I_{yy} - I_{zz})\dot{\theta}\dot{\psi} - J_R\Omega_n\dot{\theta} + lU_2 + b_\phi\dot{\phi}, \\
 I_{yy}\ddot{\theta} &= (I_{zz} - I_{xx})\dot{\phi}\dot{\psi} + J_R\Omega_n\dot{\phi} + lU_3 + b_\theta\dot{\theta}, \\
 I_{zz}\ddot{\psi} &= (I_{xx} - I_{yy})\dot{\theta}\dot{\phi} + U_4 + b_\psi\dot{\psi},
 \end{aligned} \tag{1}$$

donde  $a_x, a_y$  y  $a_z$  representan fuerzas aerodinámicas que actúan sobre el Quadrotor, en su respectivo eje;  $J_R$  es el momento de inercia propio del motor alrededor de su eje;  $b_\phi, b_\theta, b_\psi$  son los pares aerodinámicos que actúan sobre el Quadrotor;  $\Omega$  es la suma de la velocidad angular de cada motor;  $I_{xx}, I_{yy}, I_{zz}$  son momentos de inercia del centro de masa del Quadrotor;  $\theta, \phi, \psi, x, y$  y  $z$  son las coordenadas de posición de rotación y traslación respecto a un marco inercial, respectivamente;  $l$  corresponde a la distancia entre los rotores y el centro de gravedad;  $U_i$  representa la entrada de control, para  $i = 1, 2, 3, 4$ .

## 3. Descripción del sistema

Un Quadrotor es un sistema subactuado, es decir, tiene seis grados de libertad ( $x, y, z, \phi, \theta, \psi$ ) y cuatro entradas de control (Cotte Corredor y Moreno Pineda, 2010). Las fuerzas y momentos principales que actúan en el Quadrotor son producidas por sus rotores, de los cuales, dos pares de motores rotan en direcciones opuestas para balancear el torque total del sistema. En la Figura 8 se muestra un diagrama general de las fuerzas que actúan sobre el Quadrotor, la dirección de los motores y los marcos de referencia móvil ( $I$ ) y fijo al vehículo ( $B$ ) que suelen utilizarse para su análisis. La diferencia entre estos radica en que el marco de referencia móvil describe la cinemática del Quadrotor desde un punto de referencia terrestre, mientras que el marco fijo describe la cinemática del VANT respecto su centro de gravedad.

La Figura 9 muestra la relación de fuerzas del Quadrotor correspondiente a sus movimientos de rotación. Como se muestra en la Figura 9(a), para realizar cambios en la altura se debe aumentar o disminuir la velocidad de los 4 rotores, esto se debe hacer manteniendo el empuje constante. El movimiento de alabeo ( $\phi$ ) es obtenido aumentando la velocidad del motor 2 y reduciendo la velocidad del rotor 4 (Ver Figura 9(b)). El movimiento de cabeceo ( $\theta$ ) se observa en la Figura 9(c), en el cual se aumenta la velocidad del rotor 1, mientras que se reduce velocidad del rotor 3. Por último, el movimiento de orientación de guiñada ( $\psi$ ) resulta del aumento de velocidad en los rotores 1 y 3, mientras que se disminuye la velocidad de los rotores 2 y 4 (Ver Figura 9(d)) (Raffo, 2007).

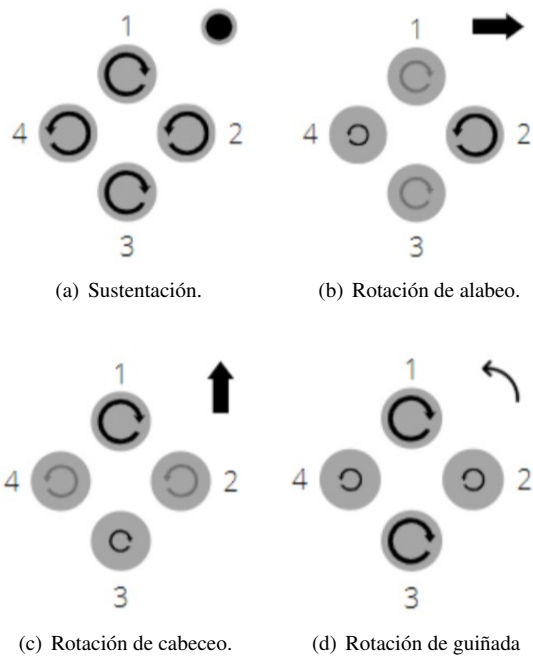


Figura 9: Relación de rotación y velocidades de los rotores.

Lo anterior resalta la importancia de relacionar las acciones de control  $U_i$  con las velocidades angulares requeridas en cada uno de los rotores para manipular los ejes de posición y orientación del Quadrotor en función del objetivo de control. De acuerdo con (Çetinsoy et al., 2012; Mellinger y Kumar, 2011) las ecuaciones que relacionan las entradas de control con las velocidades angulares de los rotores son:

$$\begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} k_f & k_f & k_f & k_f \\ 0 & k_f l & 0 & k_f l \\ -k_f l & 0 & k_f l & 0 \\ k_m & -k_m & k_m & -k_m \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix}, \quad (2)$$

donde:  $l$  es la distancia del rotor al centro de gravedad en la estructura del Quadrotor,  $k_f$  y  $k_m$  son los coeficientes de levantamiento y de arrastre, respectivamente. Finalmente,  $\omega_i^2$  se despeja de la siguiente ecuación:

$$\begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = \begin{bmatrix} \frac{1}{4k_f} & 0 & -\frac{1}{2k_f l} & \frac{1}{4k_m} \\ \frac{1}{4k_f} & \frac{1}{2k_f l} & 0 & -\frac{1}{4k_m} \\ \frac{1}{4k_f} & 0 & -\frac{1}{2k_f l} & \frac{1}{4k_m} \\ \frac{1}{4k_f} & -\frac{1}{2k_f l} & 0 & -\frac{1}{4k_m} \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix}. \quad (3)$$

#### 4. Esquema de control

El sistema (1) presenta dinámicas no lineales, las cuales suelen complicar el cumplimiento de los objetivos de control al emplear esquemas de control lineales tipo PID. Por lo anterior, se propone utilizar un controlador tipo PID prealimentado, dando prioridad al control de altura y orientación del Quadrotor.

La prealimentación se obtiene a partir de (1) y permite linealizar parcialmente al sistema en las coordenadas  $z$ ,  $\theta$ ,  $\phi$  y  $\psi$ . Se debe resaltar que únicamente se considera la dinámica conocida del Quadrotor para el diseño del esquema de control, considerando la velocidad de los rotores y los coeficientes de fricción como perturbaciones. Mientras que, a nivel de simulación se hace uso de la dinámica completa del Quadrotor así como perturbaciones de viento. Así, se proponen las siguientes acciones de control prealimentadas:

$$\begin{aligned} U_1 &= \frac{m(v_z + g)}{\cos \phi \cos \theta}, \\ U_2 &= \frac{I_{xx} v_\phi - (I_{yy} - I_{zz}) \dot{\theta} \dot{\psi}}{l}, \\ U_3 &= \frac{I_{yy} v_\theta - (I_{zz} - I_{xx}) \dot{\phi} \dot{\psi}}{l}, \\ U_4 &= I_{zz} v_\psi - (I_{xx} - I_{yy}) \dot{\theta} \dot{\phi}, \end{aligned} \quad (4)$$

$v_\gamma$  es un controlador auxiliar, donde  $\gamma$  genera la asociación de los subíndices  $z$ ,  $\phi$ ,  $\theta$  y  $\psi$ . Los controladores auxiliares utilizados son tipo PID y están dados por la siguiente estructura:

$$v_\gamma = k_{p_\gamma} e_\gamma(t) + k_{i_\gamma} \int_0^t e_\gamma(\tau) d\tau + k_{d_\gamma} \frac{d}{dt} e_\gamma(t), \quad (5)$$

con  $k_{p_\gamma}$  igual a la ganancia proporcional,  $k_{i_\gamma}$  es la ganancia integral y  $k_{d_\gamma}$  corresponde a la ganancia derivativa. Por último,  $e(t)$  corresponde al error de posición u orientación, el cual se define como la diferencia entre el valor de referencia y la variable de proceso (valor medido).

La sustitución de las acciones de control prealimentadas descritas por (4) en (1) da como resultado el siguiente sistema de ecuaciones:

$$\begin{aligned}
\ddot{x} &= -(v_z + g) \frac{\tan \theta}{\cos \phi} + \frac{a_x}{m} \dot{x}, \\
\ddot{y} &= (v_z + g) \tan \phi + \frac{a_y}{m} \dot{y}, \\
\ddot{z} &= v_z + \frac{a_z}{m} \dot{z}, \\
\ddot{\phi} &= v_\phi - \frac{J_R \Omega_n}{I_{xx}} \dot{\theta} + \frac{b_\phi}{I_{xx}} \dot{\phi}, \\
\ddot{\theta} &= v_\theta + \frac{J_R \Omega_n}{I_{yy}} \dot{\phi} + \frac{b_\theta}{I_{yy}} \dot{\theta}, \\
\ddot{\psi} &= v_\psi + \frac{b_\psi}{I_{zz}} \dot{\psi}.
\end{aligned} \tag{6}$$

De (6),  $\ddot{z}$ ,  $\ddot{\phi}$ ,  $\ddot{\theta}$  y  $\ddot{\psi}$  se pueden describir como:

$$\begin{aligned}
\ddot{z} &= v_z + \xi_z, \\
\ddot{\phi} &= v_\phi + \xi_\phi, \\
\ddot{\theta} &= v_\theta + \xi_\theta, \\
\ddot{\psi} &= v_\psi + \xi_\psi,
\end{aligned} \tag{7}$$

donde,  $\xi_\gamma$  representa el conjunto de perturbaciones de la dinámica interna del Quadrotor y las perturbaciones externas.

#### 4.1. Análisis de estabilidad

Al sustituir las acciones de control  $v_\gamma$  en (7), derivar el resultado y después de algunas manipulaciones algebraicas; se obtiene la dinámica del error, descrita como sigue:

$$e_\gamma^{(3)} + k_{d_\gamma} \ddot{e}_\gamma + k_{p_\gamma} \dot{e}_\gamma + k_{i_\gamma} e_\gamma = \dot{\xi}_\gamma. \tag{8}$$

La selección de los parámetros  $k_{d_\gamma}$ ,  $k_{p_\gamma}$  y  $k_{i_\gamma}$  deben ser tal que el lazo cerrado del polinomio característico

$$s^3 + k_{d_\gamma} s^2 + k_{p_\gamma} s + k_{i_\gamma}, \tag{9}$$

sea Hurwitz (Sánchez-Meza *et al.*, 2020). Así, el controlador conduce la dinámica del seguimiento del error dentro de una vecindad sobre del origen del error. A partir de (7) se puede proponer la siguiente función candidata de Liapunov:

$$\begin{aligned}
V_\gamma(z) &= z_\gamma^\top P_\gamma z_\gamma, \\
z_\gamma &\in \mathbb{R}^3, P_\gamma = P_\gamma^\top > 0, P_\gamma \in \mathbb{R}^{(3) \times (3)}, \\
z_\gamma &= [e_\gamma \quad \dot{e}_\gamma \quad \ddot{e}_\gamma]^\top.
\end{aligned} \tag{10}$$

La realización del espacio de estados de  $z_\gamma$  esta dado por

$$\dot{z}_\gamma = A_\gamma z_\gamma + b_\gamma \xi_\gamma, \tag{11}$$

con

$$\begin{aligned}
A_\gamma &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -k_{i_\gamma} & -k_{p_\gamma} & -k_{d_\gamma} \end{bmatrix}, \\
b_\gamma &= [0 \quad 0 \quad 1]^\top.
\end{aligned}$$

La derivada temporal de  $V_\gamma(z)$  esta dada por

$$\begin{aligned}
\frac{dV_\gamma(z)}{dt} &= 2z_\gamma^\top P_\gamma \dot{z}_\gamma \\
&= z_\gamma^\top (A_\gamma^\top P_\gamma + P_\gamma A_\gamma) z_\gamma + 2z_\gamma^\top P_\gamma \tilde{\xi}_\gamma.
\end{aligned} \tag{12}$$

De la elección de (9) y la construcción de  $A_\gamma$ ,  $A_\gamma$  es Hurwitz, entonces para cualquier matriz  $P_\gamma$  definida positiva, existe  $Q_\gamma = Q_\gamma^\top > 0$ ,  $Q_\gamma \in \mathbb{R}^{(m+3) \times (m+3)}$  tal que  $A_\gamma^\top P_\gamma + P_\gamma A_\gamma = -Q_\gamma$  Sira-Ramírez *et al.* (2017). Entonces, la derivada temporal de  $V_\gamma(z)$  se transforma:

$$\frac{dV_\gamma(z)}{dt} = -z_\gamma^\top Q_\gamma \dot{z}_\gamma + 2z_\gamma^\top P_\gamma \tilde{\xi}_\gamma. \tag{13}$$

Dado  $|\xi_\gamma| \leq \xi_{\gamma, \max}$ ,

$$\begin{aligned}
\frac{dV_\gamma(z)}{dt} &\leq -z_\gamma^\top Q_\gamma \dot{z}_\gamma + 2\|z_\gamma\| \|P_\gamma\| \|\tilde{\xi}_\gamma\| \leq \\
&\leq -z_\gamma^\top Q_\gamma \dot{z}_\gamma + 2\xi_{\gamma, \max} \|z_\gamma\| \|P_\gamma\|.
\end{aligned} \tag{14}$$

Usando la desigualdad de Rayleigh en esta última y simplificando

$$\frac{dV_\gamma(z)}{dt} \leq \lambda_{\min}(Q_\gamma) \|z_\gamma\| \left[ -\|z_\gamma\| + 2\xi_{\gamma, \max} \frac{\lambda_{\max}(P_\gamma)}{\lambda_{\min}(Q_\gamma)} \right]. \tag{15}$$

De este modo, la derivada temporal de  $V(z)$  es definida negativa fuera del conjunto  $\|z_\gamma\| \leq 2\xi_{\gamma, \max} \frac{\lambda_{\max}(P_\gamma)}{\lambda_{\min}(Q_\gamma)}$ , forzando el seguimiento del vector de error  $z$  a ser uniformemente última-mente acotado, garantizando un comportamiento acotado del seguimiento del error en lazo cerrado.

#### 4.2. Asignación de polos

Con respecto a la sintonización de los controladores, esta se realizó a partir del siguiente polinomio (8), el cual debe ser Hurwitz (Sánchez-Meza *et al.*, 2020).

La Tabla 1 muestra las ganancias utilizadas para el control la altura y las orientaciones.

Tabla 1: Ganancias utilizadas en los controladores auxiliares.

Controlador	$k_p$	$k_i$	$k_d$
$v_z$	27	8.4	36.68
$v_\phi$	1425	170	97
$v_\theta$	4000	148	90
$v_\psi$	15	4	8



## 5. Resultados

A continuación, se muestran los resultados obtenidos al realizar 3 simulaciones, los cuales se realizaron bajo la influencia de diferentes velocidad de viento: 0, 20 y 50 kt (nudos), la velocidad UDP utilizada para el envío y recepción de datos entre X-Plane y MATLAB/simulink es de 50 FPS, el cual equivale a un paso de simulación fijo de 0.02 segundos, además, el método numérico establecido en simulink es Runge-Kutta. Cada simulación tiene como objetivo de control realizar el seguimiento de trayectoria de un polinomio de Bezier en el eje  $z$  con un valor inicial de  $3m$  y valor final de  $40m$ , los tiempos de inicio y termino son de 20 y 50s, respectivamente;  $\phi$  y  $\theta$  deben alcanzar el punto equilibrio de  $0^\circ$ ; mientras que en  $\psi$  se pretende realizar el seguimiento de la función  $35 + 15 \sin(0,1t)$ . Respecto a los parámetros físicos del Quadrotor proporcionados por X-Plane son descritos en la Tabla 2 (Ramírez-Rodríguez *et al.*, 2021). (se recomienda visitar <https://youtu.be/QyiR4sD9-g4> para visualizar los resultados).

Tabla 2: Parámetros físicos del Quadrotor.

Parámetro	Símbolo	Magnitud	Unidades
Masa	$m$	1.133	kg
Distancia del rotor al centro de masa	$l$	0.1981	m
Momento de inercia sobre de $x$	$I_{xx}$	0.1801	$kgm^2$
Momento de inercia sobre de $y$	$I_{yy}$	0.1801	$kgm^2$
Momento de inercia sobre de $z$	$I_{zz}$	0.3561	$kgm^2$
Fuerza de gravedad	$g$	9.81	$m^2/s$

La Figura 10 se muestra el esquema de control utilizado y la interacción entre MATLAB/Simulink y X-Plane, resaltando los datos que son enviados y recibidos por cada programa.

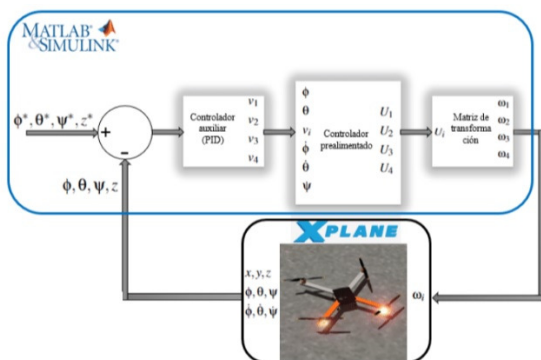


Figura 10: Esquema de control.

En la Figura 11, se tiene una simulación del sistema completo, en la parte izquierda se tienen los resultados obtenidos, mientras que en la parte derecha se tiene tanto la simulación en Simulink como el Quadrotor visualizado en X-Plane.

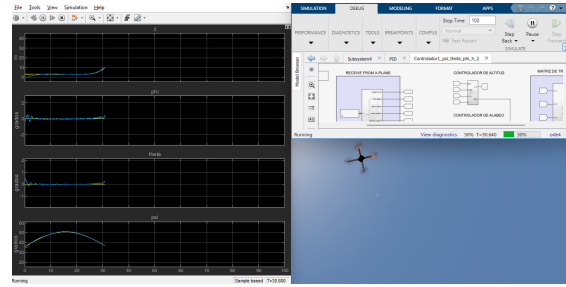


Figura 11: Simulación completa del sistema.

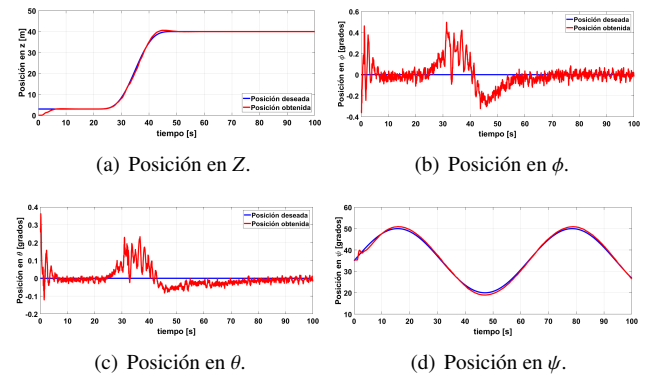


Figura 12: Respuesta del Quadrotor ante una velocidad de viento de 0 kt.

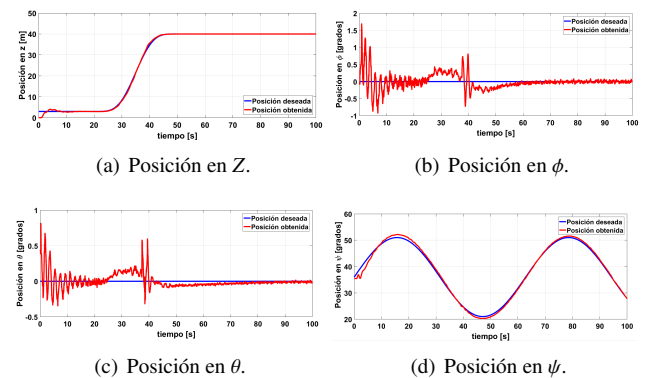


Figura 13: Respuesta del Quadrotor ante una velocidad de viento de 20 kt.

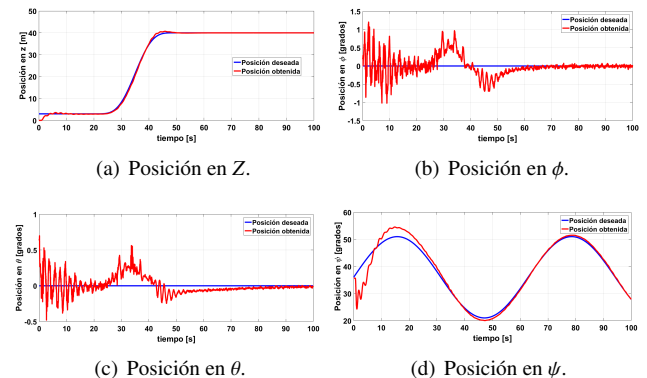


Figura 14: Posiciones bajo velocidad de viento a 50 kt.

La Fig. 12 muestra los resultados obtenidos en  $z$ ,  $\phi$ ,  $\theta$  y  $\psi$ , respectivamente. Cuando se tiene una velocidad de viento de 0 kt, se puede observar que los controles convergen al valor deseado, logrando seguir las trayectorias deseadas en  $x$  y  $\psi$  (ver 12a y 12c); en cuanto a las coordenadas  $\phi$  y  $\theta$ , se aprecian oscilaciones significativas en el intervalo de 20 a 40s, las cuales se deben a los cambios generados en  $z$  (ver 12b y 12c). Una vez la trayectoria deseada de  $z$  se mantiene constante,  $\theta$  y  $\phi$  se mantienen oscilando sobre el punto de equilibrio, lo cual se debe a que el controlador propuesto no garantiza la convergencia a cero del error (Lozano Hernández et al., 2019).

De manera similar, en las Figuras 13 y 14 se muestra el comportamiento del Quadrotor al ser sometido a velocidades de viento de 20 y 50 kt, respectivamente. Se observa un comportamiento similar al obtenido con 0 kt. Sin embargo,  $\psi$  tarda mas tiempo en converger al objetivo de control (aproximadamente 30s). Lo anterior se debe a que el viento incide de manera directa sobre  $\psi$ , afectando en menor medida a los demás grados de libertad.

En adición, las Figuras 15, 16 y 17 muestran el error de seguimiento de  $z$ ,  $\phi$ ,  $\theta$  y  $\psi$ . Donde se puede observar que el error en las cuatro coordenadas para los 3 casos de simulación tiende a ser 0, o bien presenta pequeñas oscilaciones alrededor del 0, lo que permite al Quadrotor poder realizar el seguimiento de trayectorias y mantener la estabilidad.

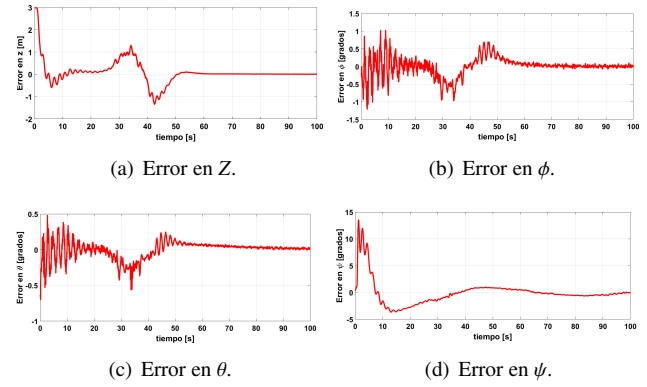


Figura 17: Error del Quadrotor ante una velocidad de viento de 50 kt.

Con respecto a las acciones de control de  $U_1$ ,  $U_2$ ,  $U_3$  y  $U_4$ , las Figuras 18, 19 y 20 muestran el comportamiento obtenido en las tres simulaciones. En los 3 casos  $U_1$  tiende a un valor cercano a 11N, lo cual corresponde a la fuerza requerida para que el Quadrotor se mantenga en vuelo estacionario, mientras que las  $U_2$ ,  $U_3$  y  $U_4$  se mantienen oscilando alrededor de cero, lo cual representa que el VANT trata de mantenerse en vuelo estacionario. Es conveniente mencionar que los valores de los controladores están acotadas a márgenes de funcionamiento posibles.

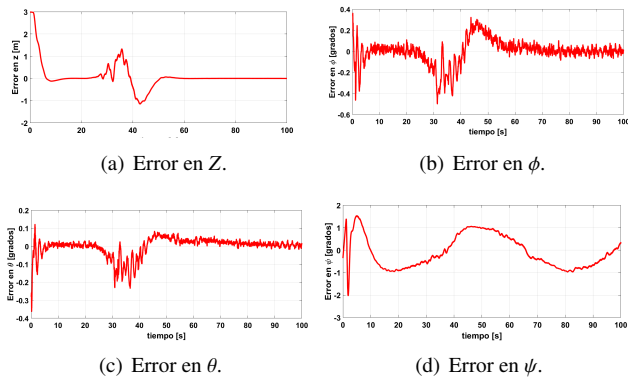


Figura 15: Error del Quadrotor ante una velocidad de viento de 0 kt.

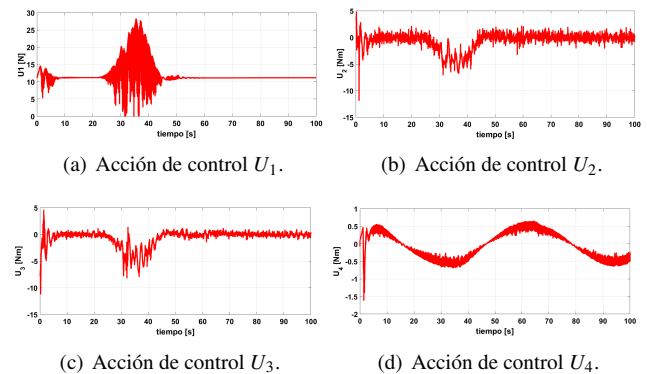


Figura 18: Acciones de control a velocidad de viento a 0 kt.

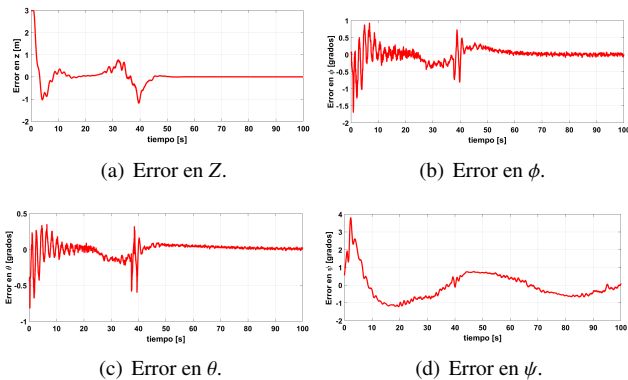


Figura 16: Error del Quadrotor ante una velocidad de viento de 20 kt.

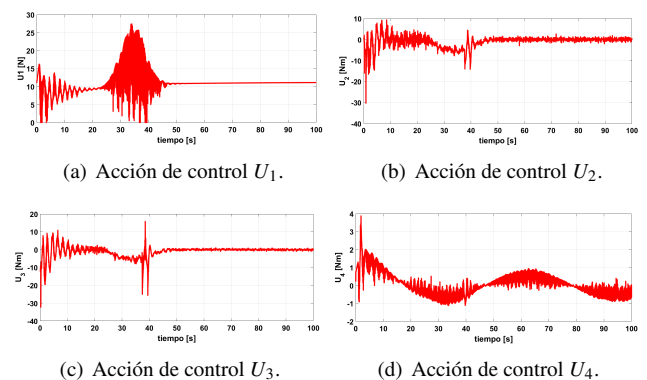


Figura 19: Acciones de control a velocidad de viento a 20 kt.



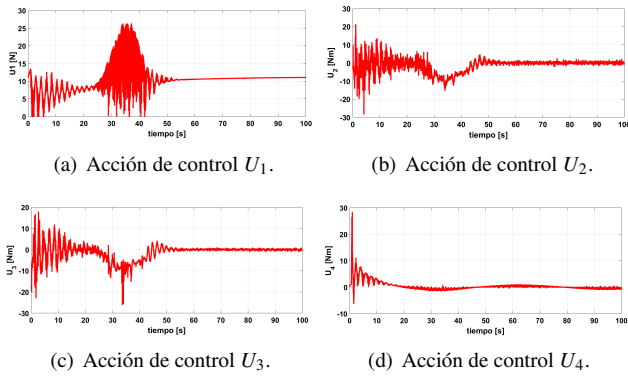


Figura 20: Acciones de control a velocidad de viento a 50 kt.

De las acciones de control resultantes se calcula las velocidades requeridas de los 4 rotores en función de 3, las cuales están acotadas de 0 a 10,000 rpm. Estas velocidades deben ser enviadas a X-Plane a través de la variable “throttle command” en un rango normalizado de 0 a 1, donde 1 representa las 10,000 rpm. En la Figura 21, se observan los valores normalizados representados  $\sigma_i$  de cada  $\omega_i$  ante cada caso de prueba. También se puede observar, que las referencias de los ángulos  $\phi$  y  $\theta$  son cero, las velocidades de los rotores tienen un comportamiento similar.

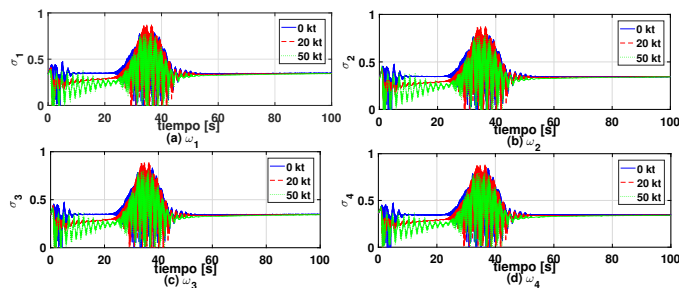


Figura 21: Proporción de las velocidades enviadas a X-Plane para cada los 4 rotores, con velocidades de viento de 0, 20 y 50 kt.

## 6. Conclusiones

El uso de XPlane y MATLAB/Simulink como herramientas de co-simulación para analizar el comportamiento y control de un VANTs resulta atractivo para validar esquemas y leyes de control sin necesidad de contar con el prototipo físico, lo anterior repercute en un ahorro tanto de recursos financieros como de tiempo. En este trabajo fue posible integrar ambas herramientas y conocer la viabilidad de su uso, se mostró el comportamiento de un esquema de control clásico ante una de las perturbaciones ambientales mas frecuentes, el viento. Actualmente, la mayoría de los esquemas de control que se utilizan son simulados únicamente en MATLAB/Simulink y posteriormente se implementan. Sin embargo, los resultados físicos suelen ser diferentes a los obtenidos en la simulación, lo anterior se debe al uso modelos matemáticos aproximados del Quadrotor que no contemplan efectos aerodinámicos y/o perturbaciones exógenas. Así, el presente trabajo resalta la importancia del uso de estas MATLAB/Simulink y X-Plane como herramientas para la validación de algoritmos de control para Quadrotores.

También, se validó el envío y recepción de datos haciendo uso del protocolo UDP, el cual permite retroalimentar los ángulos, altura y velocidades angulares del Quadrotor (en X-Plane) para ser utilizados en el calculo de la acción de control (en MATLAB/Simulink). Los resultados obtenidos son interesantes, ya que, muestran un comportamiento similar al reportado en aplicaciones reales de este controlador Ramírez-Rodríguez et al. (2021); Mellinger y Kumar (2011); Thanh y Hong (2018); Lozano y Gutiérrez (2016), resaltando la viabilidad de utilizar X-Plane (el cual se basa en el cálculo del efecto del flujo de aire sobre las superficies de los aviones) antes de llevar los algoritmos a una aplicación final.

En lo que al esquema de control se refiere, este permitió realizar maniobras de seguimiento de trayectoria en la deseada, incluso ante velocidades del viento de 50 kt. Por su parte, en la guiñada se presentan mayores sobretiros a mayores velocidades de viento, lo cual se debe a la injerencia directa del viento sobre esta coordenadas; mientras que en alabeo y cabeceo se obtienen mayores oscilaciones durante el tiempo en el que  $z$  no es constante, lo cual debido a su naturaleza subactuada, afecta de manera directa a  $\psi$ . En las 3 simulaciones, se logra estabilizar al Quadrotor y obtener un error cercano a cero. Es importante mencionar que en este trabajo se utilizó un periodo de muestreo de 20ms, el cual influye en las oscilaciones presentes y el tiempo de convergencia al valor deseado.

El presente trabajo muestra áreas de oportunidad interesantes, entre las que resaltan el diseño y validación de nuevos esquemas de control, así como la integración y desarrollo de aeronaves en el software X-Plane para abordar el análisis de vuelo en los vehículos aéreos de interés. También, abre la puerta a explorar el diseño de esquemas de control para diferentes tipos de VANTs, como lo son los híbridos; en este sentido, X-Plane permite construir diseños propios de aeronaves, e incluso, modificar modelos comerciales como el Bell Boeing V-22 Osprey (tal como se reporta en Cerezo-Pacheco et al. (2021)). Actualmente, este grupo de trabajo se encuentra desarrollando esquemas de control para VANTs tipo SUAVI, el cual puede ser diseñado en X-Plane y con ello validar los esquemas propuestos para vuelos que involucran la transición de un modo ala fija a uno de ala rotatoria (y viceversa).

## Agradecimientos

Los autores están en el Instituto Politécnico Nacional y ellos son quienes expresan su gratitud por el apoyo recibido. Este artículo fue parcialmente apoyado por el financiamiento de un proyecto de innovación asignado a V. G. Sánchez-Meza y la Secretaría de Investigación y Posgrado IPN bajo subsidios 20230569 y 20230121. V. G. Sánchez-Meza y R. Y. Almazan-Arvizu son estudiantes becarios del CONAHCYT (964590) y (889449) respectivamente y están agradecidos por el apoyo recibido.

## Referencias

- Aláez, D., Olaz, X., Prieto, M., Porcellinis, P., y Villadangos, J. (2022). Hil flight simulator for vtol-uav pilot training using x-plane. *Information*, 13(12):585.

- Bittar, A., De Oliveira, N. M. F., y De Figueiredo, H. V. (2014). Hardware-in-the-loop simulation with x-plane of attitude control of a suav exploring atmospheric conditions. *Journal of Intelligent & Robotic Systems*, 73:271–287.
- Bryan, J. D. (2021). The University of Alabama in Huntsville.
- Cerezo-Pacheco, A. D., Pérez-Velasco, C. A., Lozano-Hernández, Y., Rodríguez-Cortés, H., y Sánchez-Meza, V. G. (2021). Integration of x-plane and matlab for modeling and simulation of a tiltrotor uav. En *2021 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)*, pp. 39–44.
- Çetinsoy, E., Dikyar, S., Hançer, C., Oner, K., Sirimoglu, E., Unel, M., y Aksit, M. (2012). Design and construction of a novel quad tilt-wing uav. *Mechatronics*, 22(6):723–745.
- Cho, K., Shin, J., y Kuc, T. (2015). Design of quadrotor controller for emergency situation using xplane. En *2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, pp. 311–314.
- Cotte Corredor, J. M. y Moreno Pineda, A. F. (2010). Diseño de control robusto de velocidad de motores brushless para robótica aérea. *Departamento de Ingeniería Eléctrica y Electrónica*.
- Figueiredo, H. V. y Saotome, O. (2012). Simulation platform for quadricopter: Using matlab/simulink and x-plane. En *2012 Brazilian Robotics Symposium and Latin American Robotics Symposium*, pp. 51–55.
- García, R. y Barnes, L. (2010). Multi-uav simulator utilizing x-plane. En *Selected papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009*, pp. 393–406. Springer.
- He, G., Yu, L., Jia, S., y Wang, X. (2020). Simulation verification of flight control of a tilt tri-rotor uav using x-plane. *2020 39th Chinese Control Conference (CCC)*, pp. 7008–7013.
- Kaviyarasu, A. y Kumar, K. S. (2014). Simulation of flapping-wing unmanned aerial vehicle using x-plane and matlab/simulink. *Defence Science Journal*, 64(4):327.
- Li, L., Sun, L., y Jin, J. (2015). *Survey of advances in control algorithms of quadrotor unmanned aerial vehicle*.
- Lozano, Y. y Gutiérrez, O. (2016). Design and control of a four-rotary-wing aircraft. *IEEE Latin America Transactions*, 14(11):4433–4438.
- Lozano Hernández, Y., Gutiérrez Frías, O., Lozada-Castillo, N., y Luviano Juárez, A. (2019). Control algorithm for taking off and landing manoeuvres of quadrotors in open navigation environments. *International Journal of Control, Automation and Systems*, 17(9):2331–2342.
- Mellinger, D. y Kumar, V. (2011). Minimum snap trajectory generation and control for quadrotors. En *2011 IEEE international conference on robotics and automation*, pp. 2520–2525. IEEE.
- Pérez, M., Pérez, A., y Pérez, E. (2008). Introducción a los sistemas de control y modelo matemático para sistemas lineales invariantes en el tiempo. *Universidad Nacional de San Juan*, pp. 1–69.
- Raffo, G. V. (2007). Modelado y control de un helicóptero quadrotor. *Universidad de Sevilla*.
- Ramírez-Rodríguez, J., Tlatelpa-Osorio, Y. E., y Rodríguez-Cortés, H. (2021). Low level controller for quadrotors. En *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 1155–1161. IEEE.
- Sánchez, P., Casado, R., y Bermúdez, A. (2020). Real-time collision-free navigation of multiple uavs based on bounding boxes. *Electronics*, 9(10):1632.
- Sánchez-Meza, V. G., Lozano-Hernández, Y., y Gutiérrez-Frías, O. O. (2020). Modeling and control of a two dof helicopter with tail rotor disturbances. *2020 International Conference on Mechatronics, Electronics and Automotive Engineering (ICMEAE)*, pp. 79–84.
- Sira-Ramírez, H., Luviano-Juárez, A., Ramírez-Neria, M., y Zurita-Bustamante, E. W. (2017). *Active disturbance rejection control of dynamic systems: a flatness based approach*. Butterworth-Heinemann.
- Thanh, H. L. N. N. y Hong, S. K. (2018). Quadcopter robust adaptive second order sliding mode control based on pid sliding surface. *IEEE Access*, 6:66850–66860.
- Wagner, M. y Wolfrum, R. (2015). Unmanned aerial vehicles. *Max Planck Encyclopedia of Public International Law*, (15–12).
- Zhao, B., Xian, B., Zhang, Y., y Zhang, X. (2014). Nonlinear robust adaptive tracking control of a quadrotor uav via immersion and invariance methodology. *IEEE Transactions on Industrial Electronics*, 62(5):2891–2902.