

# **ESCUELA POLITÉCNICA NACIONAL**

## **FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA**

### **DISEÑO Y SIMULACIÓN DEL CONTROL PARA EL CUADRIRROTOR DISPONIBLE EN LA HERRAMIENTA UAV TOOLBOX DE SIMULINK PARA EFECTUAR EL SEGUIMIENTO DE UN OBJETO MÓVIL TERRESTRE USANDO CIRCUNNAVEGACIÓN**

**TRABAJO DE TITULACIÓN PREVIO A LA OBTENCIÓN DEL TÍTULO DE  
INGENIERO EN ELECTRÓNICA Y CONTROL**

**VIVIANA ESTEFANÍA VALENZUELA VITERI**

**DIRECTOR: ING. PATRICIO JAVIER CRUZ DÁVALOS, PhD.**

**Quito, Noviembre 2022**

## **AVAL**

Certifico que el presente trabajo fue desarrollado por Viviana Estefanía Valenzuela Viteri, bajo mi supervisión.

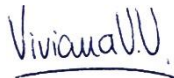
A handwritten signature in blue ink, reading "Patricio Cruz Dávalos", written over a horizontal line.

**ING. PATRICIO JAVIER CRUZ DÁVALOS, PhD.  
DIRECTOR DEL TRABAJO DE TITULACIÓN**

## DECLARACIÓN DE AUTORÍA

Yo, Viviana Estefanía Valenzuela Viteri, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración dejo constancia de que la Escuela Politécnica Nacional podrá hacer uso del presente trabajo según los términos estipulados en la Ley, Reglamentos y Normas vigentes.



---

Viviana Estefanía Valenzuela Viteri

## DEDICATORIA

*A mis padres, porque la educación comienza desde casa; su apoyo me impulsa a seguir mis sueños y conseguir mis metas. A mí, para que este trabajo sea un recordatorio de que todo el esfuerzo y corazón que pones en lo que haces vale la pena.*

## AGRADECIMIENTO

Doy gracias primero a Dios porque en cada paso me sostiene de su mano y cuida mi vida.

A mis padres, por todo su cariño y paciencia; porque, a pesar de su preocupación, saben confiar en mi proceso. Mami, gracias por ser incondicional y enseñarme a hacerme escuchar y tener sentido del humor. Papa, gracias por aguantar tantas molestadas y enseñarme a pensar antes de actuar. Me lo han dado todo, los amo y espero honrar su vida con cada logro alcanzado. A mi ñaño, porque me enseñas más de lo que crees. Son lo más valioso que tengo.

A mi familia, los Valenzuela y los Viteri, por enseñarme sobre unidad y apoyo, porque la distancia no reduce el amor.

A mis amigos, hay que rodearse de gente que sume y sin duda he encontrado eso en cada uno de ustedes. A Momo, por siempre estar, aun estando lejos. A Chelo, por los consejos y dejarme ser tu amiga. A Jorge, por seguir escribiéndome y estar pendiente de mí, por el amor de padre. A Jenny, Carmen, Maife, Meli e Inés, por ser mis hermanas y apoyarme siempre. A los Chachos, Controlitos, Electritos, Teletubbies y Redes, que tienen gran parte de mi corazón, porque llenaron mi vida de mil recuerdos que no cambio por nada. A Juanito, por llevarme de la mano en cada proyecto y demostrarme que cualquier idea se puede hacer realidad. Al mejor asesor y eminencia, Edgarín, porque sin tu apoyo no encontraba la salida ni el impulso para continuar. A mi amiguito Laleins, porque me llenas de admiración e ilusión sobre el futuro. A mi amiguito SonEdiSon, gracias por cada palabra de aliento y presión para acabar este trabajo. A mi amiguita Kimberly, gracias por entenderme, aun cuando no diga nada. A mi amiguito pantaloncín, gracias por alegrarme la vida. A Monilu, gracias por brindarme tu amistad más pura. Al gordo, por tu cariño inmenso.

A Javi, porque estuviste a mi lado en todo momento. En los bajones más difíciles supiste ser mi mejor amigo y mi confidente, y eres parte de los días más felices en mi vida. Gracias por celebrar cada pequeña victoria conmigo y recordarme que puedo con todo lo que me propongo. Entre dos, la carga se vuelve más ligera.

A mi director de tesis, Dr. Patricio Cruz, por su paciencia durante tanto tiempo y su dedicación, por compartir su conocimiento, darme la oportunidad de aprender sobre drones y lograr este proyecto con su ayuda.

# ÍNDICE DE CONTENIDO

AVAL .....	I
DECLARACIÓN DE AUTORÍA.....	II
DEDICATORIA .....	III
AGRADECIMIENTO .....	IV
ÍNDICE DE CONTENIDO .....	V
RESUMEN.....	VIII
ABSTRACT .....	IX
1. INTRODUCCIÓN.....	1
1.1 OBJETIVOS .....	1
1.2 ALCANCE .....	2
1.3 MARCO TEÓRICO.....	3
1.3.1 VEHÍCULOS AÉREOS NO TRIPULADOS .....	3
1.3.1.1 Clasificación .....	4
1.3.1.1.1 Drones de ala fija.....	4
1.3.1.1.2 Drones de ala rotatoria .....	5
1.3.1.1.3 Drones auto-sustentados.....	6
1.3.1.1.4 Drones de ala flexible .....	6
1.3.2 CUADRICÓPTERO O DRON DE CUATRO HÉLICES .....	7
1.3.2.1 Configuración .....	8
1.3.2.2 Movimientos .....	8
1.3.2.2.1 Thrust, altitud o empuje .....	9
1.3.2.2.2 Pitch o cabeceo .....	9
1.3.2.2.3 Roll, balanceo o inclinación.....	9
1.3.2.2.4 Yaw o guiñada.....	9
1.3.2.3 Modelo del cuadricóptero .....	10
1.3.3 SISTEMAS DE CONTROL DE UAVS.....	14
1.3.3.1 Controlador tipo PID .....	15
1.3.3.2 Controlador basado en Lyapunov .....	16
1.3.4 CIRCUNNAVEGACIÓN.....	18
1.3.5 UAV TOOLBOX.....	19
1.3.5.1 UAV Guidance Model .....	21

2.	METODOLOGÍA.....	23
2.1	DESARROLLO DE CONTROLADORES .....	23
2.1.1	LINEALIZACIÓN.....	23
2.1.1.1	Subsistema vertical .....	24
2.1.1.2	Subsistema Direccional .....	25
2.1.1.3	Subsistema Lateral.....	25
2.1.1.4	Subsistema Longitudinal.....	25
2.1.2	CONTROLADOR TIPO PID.....	26
2.1.2.1	Control PD de altura .....	26
2.1.2.2	Control PD de x.....	29
2.1.2.3	Control PD de y.....	31
2.1.3	CONTROLADOR BASADO EN LYAPUNOV .....	33
2.1.3.1	Control de altura basado en Lyapunov .....	34
2.1.3.2	Control de x,y basado en Lyapunov.....	39
2.1.4	INCORPORACIÓN DEL ÁNGULO YAW .....	44
2.1.4.1	Control tipo PID con incorporación del ángulo yaw .....	45
2.1.4.2	Control Basado en Lyapunov con incorporación del ángulo yaw .....	47
2.1.5	CIRCUNNAVEGACIÓN.....	48
2.2	IMPLEMENTACIÓN DEL ESCENARIO DE SIMULACIÓN.....	49
2.2.1	MODELO DEL UAV .....	49
2.2.2	VISUALIZACIÓN DEL UAV .....	50
2.2.3	SIMULACIÓN 3D.....	51
2.2.4	HMI.....	53
2.2.4.1	Diagrama de flujo de la interfaz .....	56
3.	RESULTADOS Y DISCUSIÓN.....	58
3.1	PRUEBAS DE TRAYECTORIAS DEL UAV .....	58
3.1.1	TRAYECTORIA CUADRADA .....	58
3.1.1.1	Prueba con controlador tipo PID.....	59
3.1.1.2	Prueba con controlador basado en Lyapunov.....	61
3.1.1.3	Comparación de resultados .....	63
3.1.2	TRAYECTORIA CIRCULAR .....	63
3.1.2.1	Prueba con controlador tipo PID.....	64
3.1.2.2	Prueba con controlador basado en Lyapunov.....	66
3.1.2.3	Comparación de resultados .....	68
3.2	PRUEBAS DE CIRCUNNAVEGACIÓN A UN OBJETO MÓVIL TERRESTRE .....	69

3.2.1	TRAYECTORIA CUADRADO DE OBJETO MÓVIL .....	69
3.2.1.1	Prueba con controlador tipo PID .....	69
3.2.1.2	Prueba con controlador basado en Lyapunov.....	72
3.2.1.3	Comparación de resultados .....	75
3.2.2	TRAYECTORIA CÍRCULO DE OBJETO MÓVIL .....	75
3.2.2.1	Prueba con controlador tipo PID .....	76
3.2.2.2	Prueba con controlador basado en Lyapunov.....	78
3.2.2.3	Comparación de resultados .....	81
3.2.2.4	Comparación de resultados entre circunnavegación a la trayectoria de un cuadrado y un círculo .....	81
3.2.3	TRAYECTORIA LIBRE DE OBJETO MÓVIL .....	82
3.2.3.1	Prueba con controlador tipo PID .....	82
3.2.3.2	Prueba con controlador basado en Lyapunov.....	85
3.2.3.3	Comparación de resultados .....	88
4.	CONCLUSIONES Y RECOMENDACIONES.....	90
4.1	CONCLUSIONES.....	90
4.2	RECOMENDACIONES .....	91
	REFERENCIAS BIBLIOGRÁFICAS .....	93
	ANEXOS.....	97
	ANEXO A.....	97
	ANEXO B.....	100



## RESUMEN

En este trabajo de titulación se presenta el diseño y simulación del control para el cuadricóptero disponible en la herramienta UAV Toolbox de Simulink para efectuar el seguimiento de un objeto móvil terrestre usando circunnavegación. Para lo cual, se realiza el estudio del manejo de drones tipo cuadricópteros y de diferentes controladores para el desplazamiento del mismo. Se realiza la implementación de un controlador tipo PID y uno basado en Lyapunov, que permite el manejo del robot aéreo disponible en el UAV Toolbox de Simulink, con la finalidad de realizar la circunnavegación de un objeto móvil terrestre; es decir, tratando de mantener el frente del UAV apuntando hacia el objeto en seguimiento. Para lograr esto, se controla tanto la posición como la orientación del dron y se realiza la comparación de los controladores propuestos mediante el uso de índices de desempeño. Los resultados obtenidos a través de la simulación realizada se muestran mediante una interfaz gráfica desarrollada en App Designer de Matlab, en la cual se visualiza el comportamiento del cuadricóptero al realizar el seguimiento de trayectorias predefinidas y la circunnavegación a un objeto móvil terrestre. También se observan señales de control, errores e índices de desempeño obtenidos en las distintas pruebas realizadas.

**PALABRAS CLAVE:** Cuadricóptero, Circunnavegación, UAV Toolbox, Controlador Tipo PID, Controlador basado en Lyapunov

## **ABSTRACT**

In this final degree project, we present the design and simulation of the control of a quadcopter available in the UAV Toolbox of Simulink, so it can track a terrestrial mobile object using circumnavigation. To achieve this goal, a study about handling quadcopter drones and about different controllers for its movement are carried out. The implementation of a PID type controller and a Lyapunov-based controller is developed. Both allow to drive the quadcopter available in the UAV Toolbox of Simulink and to circumnavigate a terrestrial mobile object. Therefore, the front of the UAV has to be pointing towards the object being tracked at all times. To achieve this task, the drone's position and orientation are controlled. The comparison of the proposed controllers is performed by using performance indices. The simulation results are shown through a graphical interface developed in MATLAB App Designer, in which the behavior of the quadcopter is visualized when following predefined trajectories and circumnavigating a mobile terrestrial object. Control signals, errors, and performance indexes from the different tests are also shown in the interface.

**KEYWORDS:** Quadcopter, Circumnavigation, UAV Toolbox, PID-type controller, Lyapunov-based controller

# 1. INTRODUCCIÓN

En los últimos años el uso de drones ha crecido exponencialmente y en el mercado se pueden encontrar cada vez mayor cantidad de estas aeronaves [1]. Los drones, vehículos aéreos no tripulados (UAVs, por sus siglas en inglés) o robots voladores, en un inicio, se utilizaron con fines principalmente militares; sin embargo, en la actualidad se utilizan en funciones civiles como son la vigilancia, el control de tráfico, la búsqueda y rescate, la agricultura, etc. [2].

Este tipo de robots móviles son sistemas complejos de seis grados de libertad (6GDL o 6DOF), tres para definir su desplazamiento y tres para su orientación. Se clasifican en dos tipos: multirrotor y de ala fija [3]. Dentro de los multirrotor se encuentran los cuadrirrotores, que son drones con cuatro brazos que sostienen un motor y una hélice cada uno [4]. En comparación con los de ala fija, los multirrotor son más manejables y pueden volar de forma segura dentro de espacios cerrados, lo que los hace adecuados para usarse dentro de laboratorios o como hobby [3].

Un UAV puede ser útil en misiones de reconocimiento, para planear tácticas, observar instalaciones desde el aire o realizar tareas de escolta a un objeto móvil de interés [5]. Una misión de vigilancia y reconocimiento exitosa es capaz de mejorar el conocimiento de la situación en un entorno desconocido mediante la adquisición de información, la cual puede ser obtenida a través de la circunnavegación [6]. Es decir, orbitar alrededor de un objetivo a una distancia deseada tratando de mantener el frente del UAV apuntando hacia el objeto en seguimiento [6].

En función a lo antes mencionado, en este proyecto de titulación se realiza el diseño y simulación de un controlador tipo PID y un controlador basado en Lyapunov que permita el manejo del cuadrirrotor disponible en el UAV Toolbox de Simulink, con la finalidad de realizar la circunnavegación de un objeto móvil terrestre. Para esto, se requiere controlar tanto la posición como la orientación de dirección del dron.

## 1.1 OBJETIVOS

El objetivo general de este Proyecto Técnico es: Diseñar y simular el control para el cuadrirrotor disponible en la herramienta UAV Toolbox de Simulink para el seguimiento de un objeto móvil terrestre usando circunnavegación.

Los objetivos específicos del Proyecto Técnico son:

- Realizar una revisión bibliográfica del funcionamiento y modelado de UAVs cuadricópteros, de la tarea de circunnavegación, de controladores PID y basados en Lyapunov, y del manejo del software Simulink y su complemento UAV Toolbox.
- Diseñar un esquema de control tipo PID y control basado en Lyapunov aplicado al cuadricóptero disponible en el UAV Toolbox para el control de posición y de su ángulo de dirección (heading).
- Diseñar el control del modelo de cuadricóptero del UAV Toolbox para seguimiento de un objeto móvil terrestre mediante circunnavegación en base al control de posición y del ángulo de dirección diseñado.
- Implementar una interfaz gráfica donde se permita el manejo de un objeto móvil terrestre y la visualización del comportamiento del sistema con los diferentes controladores diseñados.
- Realizar pruebas para verificar el funcionamiento de los controladores diseñados.

## **1.2 ALCANCE**

- Se realiza una revisión bibliográfica del funcionamiento y modelado de UAVs cuadricópteros, del manejo del software de Simulink y de su complemento UAV Toolbox, para efectuar la tarea de circunnavegación de un objeto móvil.
- Se realiza una revisión bibliográfica de controladores PID y basados en Lyapunov para el manejo de cuadricópteros para analizar cómo se puede emplear los mismos para la tarea propuesta en el presente plan.
- Se realiza el trabajo con enfoque en el modelo guía del cuadricóptero disponible como complemento UAV Toolbox (no se emplea el modelo del dron de ala fija disponible en el Toolbox).
- Se diseña un esquema de control con un controlador lineal tipo PID y uno basado en Lyapunov para el control de posición y del ángulo de dirección (heading).
- Se diseña un algoritmo de control del cuadricóptero para realizar seguimiento de un objeto móvil terrestre mediante circunnavegación y se lo integra con los controladores PID y basado en Lyapunov.
- Se desarrolla el diseño e implementación de una interfaz gráfica que permita visualizar los resultados de las simulaciones realizadas en el software de Simulink.

- Se realizan pruebas iniciales a diferentes trayectorias preestablecidas para calibrar los controladores de posición y orientación del ángulo de dirección.
- Se realizan pruebas mediante simulación y con la interfaz gráfica del seguimiento del objeto móvil para lograr hacer circunnavegación cuando se realiza el movimiento de manera manual y con trayectorias preestablecidas. Estas pruebas con el objeto móvil terrestre se realizan en un ambiente controlado, sin obstáculos.
- Se realizan el análisis y comparación del desempeño de los controladores a través del cálculo de índices de rendimiento IAE y TVU.

### **1.3 MARCO TEÓRICO**

En el siguiente apartado se detallan los conceptos relacionados con UAVs, los fundamentos de los sistemas de control implementados en este trabajo, el concepto de circunnavegación y la información relacionada al UAV Toolbox de Simulink, utilizado en el presente proyecto.

#### **1.3.1 VEHÍCULOS AÉREOS NO TRIPULADOS**

Los robots aéreos, vehículos aéreos no tripulados (Unmanned Aerial Vehicles, UAVs) o drones han formado gran parte de la investigación de robótica móvil en los últimos años debido a su versatilidad. Estas naves pueden llevar a cabo una gran gama de tareas y propósitos, por ejemplo, de manera recreacional, para fotografía, de manera comercial y militar, pero sus funciones más básicas son el vuelo y la navegación. Los UAVs fueron desarrollados originalmente para la milicia y la industria aeroespacial; sin embargo, se los usa de manera general debido a sus actuales niveles de seguridad y eficiencia [2], [7].

Estos vehículos aéreos no tripulados funcionan sin piloto a bordo y con diferentes niveles de autonomía. El nivel de autonomía de un dron puede ir desde el pilotaje a distancia (un operador controla sus movimientos) hasta la autonomía avanzada, que significa que se basa en un sistema de sensores y detectores, por ejemplo, LIDARs, para calcular su movimiento [7]. Al darle a un dron el poder de despegar y aterrizar por su cuenta y sensar objetos, este llega a asemejarse más a la idea de lo que implica ser un robot autónomo [8].

Los robots aéreos difieren de los robots terrestres en algunas características importantes. Estos tienen seis grados de libertad (6DoF, 6 Degrees of Freedom) y son accionados por fuerzas. Por esto su modelo se expresa en términos de fuerza, torques y aceleraciones, en lugar de velocidades como en los vehículos terrestres; por lo que, generalmente se usa un modelo dinámico en lugar de uno cinemático [3].

### 1.3.1.1 Clasificación

Los drones pueden ser tan grandes como un avión o tan pequeños como la palma de la mano. Poseen alas fijas o un rotor o multirrotores para el vuelo. También existen vehículos aéreos no tripulados auto-sustentados, como dirigibles y globos, y pequeños vehículos aéreos no tripulados de ala flexible [7].

Los UAVs tienen una amplia forma de clasificarse, pero la más común es la clasificación en función de su tipo de sustentación, en base a la cual se tienen los siguientes tipos: drones de ala fija, drones de ala rotatoria, drones ligeros o auto-sustentados, y drones de ala flexible [7].

#### 1.3.1.1.1 Drones de ala fija

Los drones de ala fija tienen forma aerodinámica como la de los aviones normales, en los que las alas proporcionan la sustentación y facilitan el mantenerse en el aire durante horas, lo que los hace muy eficientes. Suelen utilizar combustible en lugar de electricidad, lo que les permite planear en el aire durante más de 16 horas, por lo cual son utilizados, por ejemplo, para hacer mapeado [7], [9].

Como estos drones suelen ser mucho más grandes, y debido a su diseño, necesitan despegar y aterrizar en pistas de aterrizaje al igual que los aviones [7]. No pueden despegar ni aterrizar en vertical y para despegar necesitan ser lanzados con la mano o con una catapulta [1], véase Figura 1.1. Su diseño les impide realizar vuelo estacionario, por lo cual no permite realizar trabajos en los que sea necesario una imagen fija [9]. Son utilizados para transportar grandes cantidades de equipo e incluso por organizaciones sin ánimo de lucro para entregar alimentos y otros bienes en zonas de difícil acceso [7].



**Figura 1.1.** Despegue de un dron de ala fija [1]

### 1.3.1.1.2 Drones de ala rotatoria

Los drones de ala rotatoria pueden ser de un solo rotor o multirrotores, siendo los últimos los más conocidos, utilizados y comunes del mercado. La principal diferencia entre los drones de ala rotatoria y los de ala fija es que los segundos generan sustentación a través de las fuerzas que generan las hélices de sus rotores [1], [7].

Los drones de un solo rotor tienen la forma de helicópteros pequeños que pueden ser de combustión o eléctricos. El tener solo un rotor y la capacidad de funcionar a combustión, ayudan a su estabilidad y les permite volar largas distancias. Suelen utilizarse para transportar objetos más pesados, como sistemas LIDAR [7].

Los drones multirrotores suelen ser algunos de los más pequeños y ligeros del mercado, algunos ejemplos se pueden visualizar en la Figura 1.2. Tienen una distancia, velocidad y altura limitadas, pero son el vehículo de vuelo perfecto para los aficionados y los fotógrafos aéreos [7]. Este tipo de drones permite realizar trabajos estacionarios gracias a sus motores, por lo que permiten capturar una imagen fija [9]. Son capaces de despegar y aterrizar de forma vertical, por lo cual no necesitan una superficie de terreno específica. Los multirrotores tienen un elevado consumo de energía debido a que todos sus rotores deben estar en funcionamiento para que el robot se mantenga en el aire [1], por lo que pueden estar entre 20 y 30 minutos en el aire transportando una carga ligera, como una cámara [9]. Entre más motores tenga un dron multirrotor, mayor será su capacidad de estabilización, pero también, mayor será su tamaño, lo cual aumenta la dificultad de ser manejados en espacios cerrados [9].

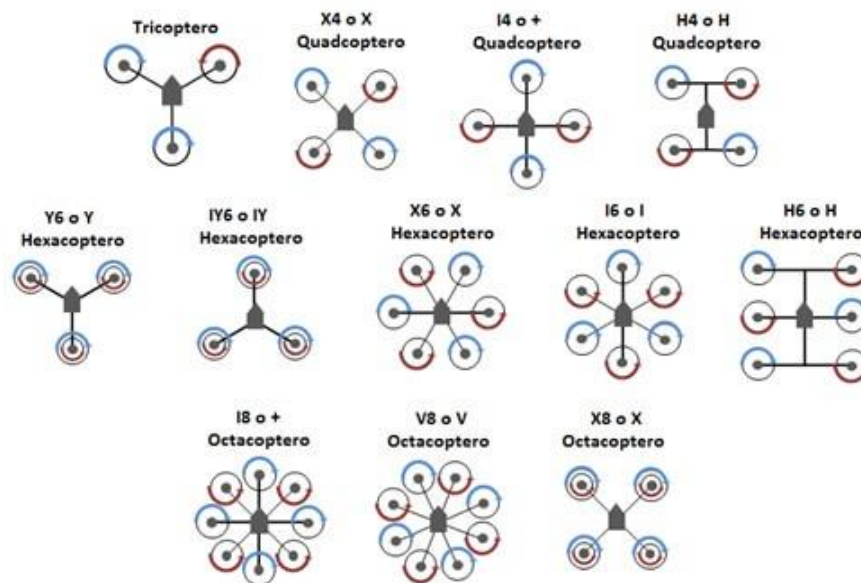


Figura 1.2. Drones de ala rotatoria multirrotores [10]

#### 1.3.1.1.3 Drones auto-sustentados

También llamados Lighter-than-air UAVs en inglés, son drones como dirigibles y globos. Tienen un funcionamiento silencioso y de gran resistencia, y gracias a sus vuelos de larga duración pueden utilizarse para vigilancia y fotografía aérea. Estos drones no pueden realizar maniobras, por lo que suelen estar atados para permitir al usuario mantener el control y que el viento no los desvíe, véase Figura 1.3. El ancla también puede actuar como sistema de comunicación para carga y descarga de información del vehículo [11].

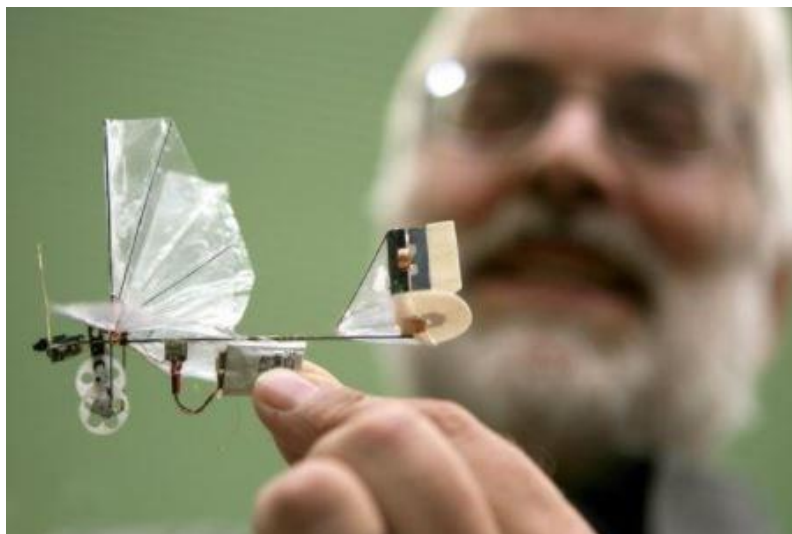


**Figura 1.3.** Dron auto-sustentado como globo atado [11]

#### 1.3.1.1.4 Drones de ala flexible

Los drones de ala flexible (“flapping wing”) u ornitóptero (Figura 1.4) utilizan la mecánica del vuelo de las aves y han sido utilizados por militares para vigilancia. Esta tecnología genera inconvenientes como la falta de maniobrabilidad y de tiempo de vuelo por la cantidad de energía requerida para el mecanismo de aleteo. Sin embargo, gana más popularidad a medida que avanza su tecnología [12].





**Figura 1.4.** Dron de ala batiente [12]

Una vez definidos los diferentes tipos de drones, para este proyecto se va a hacer énfasis en el multirroto conocido como cuadricóptero. Este tipo de dron se encuentra en la herramienta que se implementará en los siguientes capítulos y del cual se ha realizado un estudio más extenso, mostrado en las siguientes secciones.

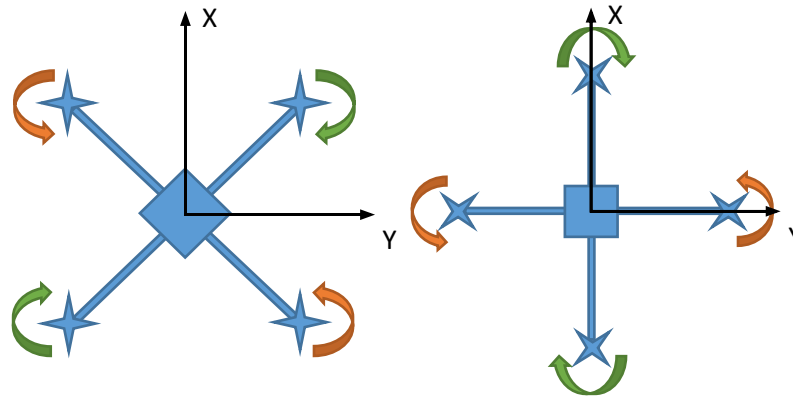
### **1.3.2 CUADRICÓPTERO O DRON DE CUATRO HÉLICES**

Un cuadricóptero, también llamado quadcopter, quadrotor, quad-copter, o cuadrirrotor, es un dron multirroto con cuatro brazos, y en la parte final de cada brazo, un motor y una hélice. Dos de sus hélices giran en el sentido horario y las otras dos hélices, en sentido antihorario, consiguiendo de esta forma estabilidad y creando la fuerza de empuje necesaria para elevarse [4], [13].

Los cuadricópteros ahora están ampliamente disponibles, en productos comerciales como en proyectos de código abierto (open-source). En comparación con los drones de ala fija, son muy maniobrables y pueden volarse con seguridad en interiores, lo que los hace muy adecuados para su uso en laboratorios o como aficionados. En comparación con los helicópteros convencionales, el cuadricóptero es más fácil de volar, no tiene el complejo mecanismo de plato oscilante y es más fácil de modelar y controlar [3]. Los cuadricópteros se pueden usar como juguetes, para realizar fotografías o grabar vídeos desde un plano aéreo, en operaciones de la policía, vigilancia, gestión de desastres naturales, operaciones militares y aplicaciones en el sector agrícola [4]. Como la tecnología ha avanzado y los costes se han reducido, muchas industrias han visto en los cuadricópteros una buena oportunidad de innovación para solucionar determinados problemas [14].

### 1.3.2.1 Configuración

El modelo de un cuadirrotor puede estar dispuesto en diferentes configuraciones como puede ser la configuración en X y configuración en cruz, véase Figura 1.5.



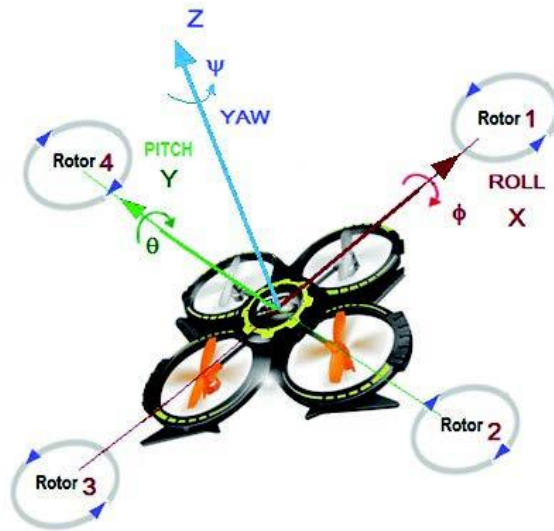
**Figura 1.5.** Configuración en X y en cruz

La operación del cuadirrotor cambia dependiendo de la configuración con la que se trabaje. Actualmente, se prefiere la configuración en X principalmente porque es más fácil añadir una funcionalidad de cámara, ya que los brazos del cuadirrotor no interferirán con las imágenes capturadas [15].

### 1.3.2.2 Movimientos

Solo se cuentan con cuatro actuadores en este UAV, esto a pesar de tener 6 GDL [3], por lo cual se dice que son sistemas subactuados. Los seis grados de libertad están compuestos por los ejes cartesianos (traslacionales) y los ángulos de Euler (rotacionales) [16]. Las variables de los ejes cartesianos,  $x$ ,  $y$ ,  $z$ , representan la distancia del centro de masa del cuadirrotor reflejada en el marco de referencia, mientras que, las otras tres son los ángulos de roll ( $\phi$ ), pitch ( $\theta$ ) y yaw ( $\psi$ ), que representan la orientación del cuadirrotor [17]. Los ángulos de roll y pitch se denominan las actitudes del cuadirrotor, y el ángulo de yaw se define como el rumbo del cuadirrotor [17].

En base a esto, los principales movimientos que puede realizar un dron son: Roll, Pitch, Yaw y Thrust. Dichos movimientos se pueden visualizar en la Figura 1.6 y se detalla en qué consisten a continuación.



**Figura 1.6.** Esquema de un cuadricóptero [18]

#### 1.3.2.2.1 Thrust, altitud o empuje

Este movimiento permite el desplazamiento vertical del cuadricóptero y logra que este realice un despegue y llegue a una altura deseada. Para ello, se genera una fuerza de empuje igual en cada rotor que aumenta a medida que se quiera despegar y elevar el dron, y disminuye cuando se lo quiere aterrizar o disminuir su altura [16].

#### 1.3.2.2.2 Pitch o cabeceo

Es el ángulo que realiza su movimiento alrededor del eje  $y$ . Se realiza al variar la velocidad de los rotores del frente y atrás [17], dependiendo de hacia dónde se quiera realizar el movimiento. Si se quiere mover el dron hacia adelante, la parte trasera del dron se levanta haciendo que el aire de las hélices empuje el dron en la dirección deseada, o viceversa [19].

#### 1.3.2.2.3 Roll, balanceo o inclinación

Es el ángulo que realiza su movimiento alrededor del eje  $x$ . Se realiza al variar la velocidad de los rotores de los lados [17], dependiendo de la orientación deseada. Si se quiere mover el dron hacia la derecha, las hélices deben mover mayor cantidad de aire hacia la izquierda, o viceversa [19].

#### 1.3.2.2.4 Yaw o guiñada

Es el ángulo que se mueve alrededor del eje  $z$ . Se refiere a este ángulo como el que dirige al cuadricóptero [17] y se obtiene un giro en sentido horario o antihorario. Este movimiento permite seguir objetos que puedan cambiar de dirección [19].

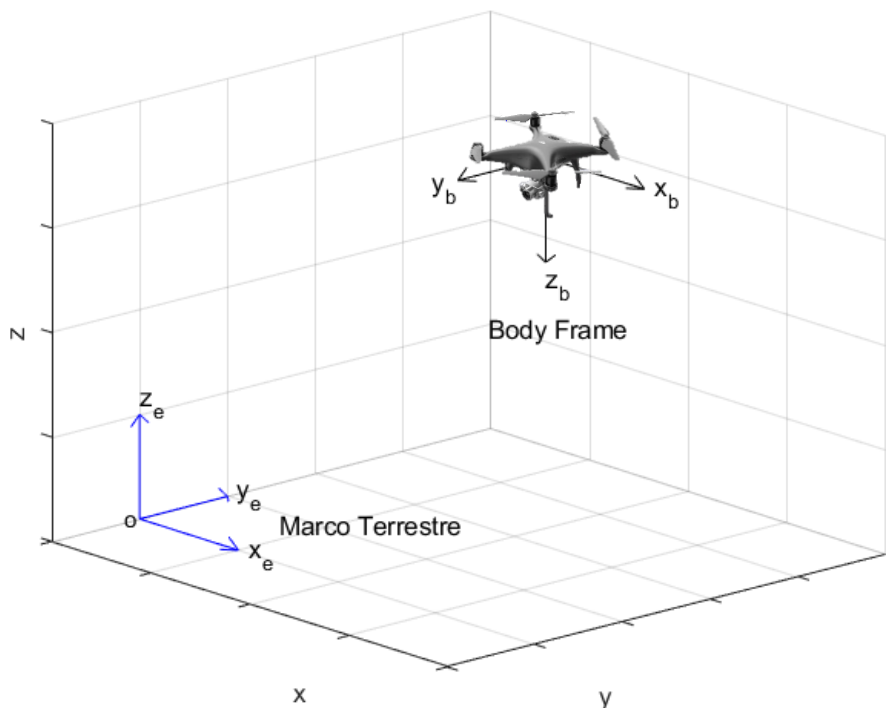
Un dron no puede realizar sus movimientos en las direcciones  $x$  o  $y$ , pero se puede mover en la dirección de pitch o roll, lo que resulta en un movimiento en la dirección  $x$  o  $y$ . Los

ángulos de pitch y roll son un medio para lograr el control de la traslación y no pueden ser fijados independientemente [3].

En relación a las configuraciones en X y en cruz, presentadas anteriormente, se tendrán similitudes en la dinámica de los movimientos de thrust o empuje y yaw o guiñada; sin embargo, en la configuración en cruz, solo dos rotores generan el momento de roll o balanceo y el de pitch o cabeceo, mientras que, en la configuración en x, los cuatro rotores se ven involucrados [17].

### 1.3.2.3 Modelo del cuadricóptero

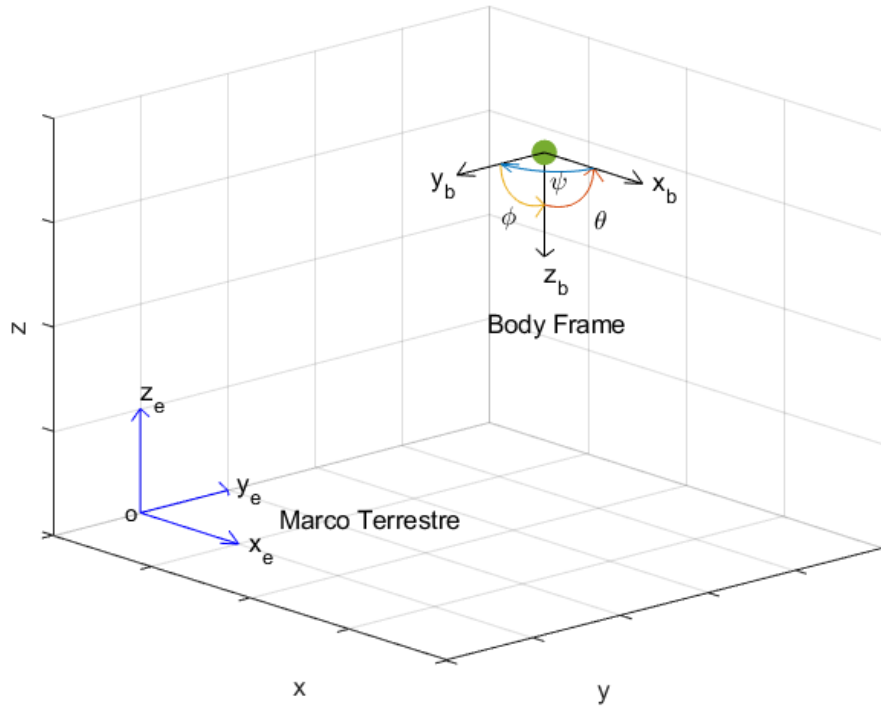
Para definir el modelo de guiado del cuadricóptero, se utilizan diferentes ecuaciones de la dinámica del sistema que se detallan en esta sección. Primero, es importante conocer que existen dos marcos de referencia para generar las coordenadas de un cuadricóptero, el marco terrestre o inercial y el marco del cuerpo o body frame. El origen del marco terrestre es un punto fijo en la Tierra, mientras que el origen del body frame del UAV corresponde con su centro de gravedad, mismo que estaría localizado en el centro de la carrocería debido a su forma simétrica [17]. En la Figura 1.7, se puede identificar los marcos de referencia mencionados.



**Figura 1.7.** Representación de un cuadricóptero con sus marcos de referencia

Para tener una mejor visualización, en la Figura 1.8 se representa al dron como una partícula verde, donde la posición del cuadricóptero en el marco terrestre está representada

por  $[x_e, y_e, z_e]$  y su body frame utiliza las coordenadas  $[x_b, y_b, z_b]$ , con orientación de los ángulos de Euler ZYX,  $[\psi, \theta, \phi]$  en radianes. Las velocidades angulares son  $[p, q, r]$  en radianes por segundo [20].



**Figura 1.8.** Representación de los ejes de referencia del cuadricóptero

Para las siguientes explicaciones, sobre todo en lo referente a las ecuaciones, se usarán las variables antes mencionadas, mismas que se encuentran definidas en la Tabla 1.1.

**Tabla 1.1.** Variables de estado

Variable	Descripción
$x_e$	Posición del dron en el eje $x$ del marco terrestre
$y_e$	Posición del dron en el eje $y$ del marco terrestre
$z_e$	Posición del dron en el eje $z$ del marco terrestre
$x_b$	Posición del dron en el eje $x$ del body frame
$y_b$	Posición del dron en el eje $y$ del body frame
$z_b$	Posición del dron en el eje $z$ del body frame
$u$	Velocidad lineal de $x$ en el body frame
$v$	Velocidad lineal de $y$ en el body frame
$w$	Velocidad lineal de $z$ en el body frame
$\psi$	Ángulo de Euler, yaw
$\theta$	Ángulo de Euler, pitch
$\phi$	Ángulo de Euler, roll
$p$	Velocidad angular de roll en el body frame
$q$	Velocidad angular de pitch en el body frame
$r$	Velocidad angular de yaw en el body frame

Se requiere un medio de transformación para llevar las ecuaciones de un sistema a otro y poder trabajar sobre un marco único de referencia [17]. Además, es necesario determinar fuerzas y momentos en el sistema de referencia inercial para lograr obtener las ecuaciones de la dinámica del sistema [18]. Para ello, se requieren tres matrices de rotación en una secuencia específica. Primero, se rota el body frame la cantidad de  $\phi$  para igualar los ejes  $x_e$  y  $x_b$ , obteniendo una matriz de rotación de  $x$ . Después, se rota la misma referencia la cantidad de  $\theta$  en  $y$ , igualando  $y_e$  y  $y_b$ , y resultando en una matriz de rotación de  $y$ . Por último, se rota la referencia del cuerpo la cantidad de  $\psi$  en  $z$ , hasta que  $z_e$  sea igual a  $z_b$ , para obtener la matriz de rotación de  $z$  [17]. Una vez obtenidas las 3 matrices primarias, se multiplican para obtener la matriz de rotación que rota del mundo al marco del cuerpo de la siguiente manera:

$$R_b^e = R_x \times R_y \times R_z \quad (1.1)$$

$$R_b^e = \begin{bmatrix} c\theta \cdot c\psi & s\phi \cdot s\theta \cdot c\psi - c\phi \cdot s\psi & c\phi \cdot s\theta \cdot c\psi + s\phi \cdot s\psi \\ c\theta \cdot s\psi & s\phi \cdot s\theta \cdot s\psi + c\phi \cdot c\psi & c\phi \cdot s\theta \cdot s\psi - s\phi \cdot c\psi \\ -s\theta & s\phi \cdot c\theta & c\phi \cdot c\theta \end{bmatrix} \quad (1.2)$$

Donde,  $c\alpha$  representa el coseno del ángulo  $\alpha$  y  $s\alpha$ , el seno.

Aplicando las ecuaciones de Newton Euler, se determina el movimiento lineal y angular del cuadricóptero. Se define la fuerza que actúa sobre el cuadricóptero como:

$$\Sigma F = m \cdot \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} \quad (1.3)$$

La sumatoria de fuerzas que actúan sobre el dron viene dada por la fuerza de empuje generada por las hélices y el peso del cuadricóptero, en sentido contrario, de la siguiente forma:

$$\begin{bmatrix} 0 \\ 0 \\ m \cdot g \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -F_{thrust} \end{bmatrix} = m \cdot \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} \quad (1.4)$$

Donde  $m$  es la masa del dron,  $g$  es la gravedad y  $F_{thrust}$  es la fuerza total en el eje  $-z_b$ .

Tomando en cuenta que, para obtener la posición del cuadricóptero con referencia al sistema inercial se utiliza la siguiente proyección:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = R_b^e \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (1.5)$$

La aceleración del dron con respecto al entorno viene dada por la siguiente ecuación:

$$m \begin{bmatrix} \ddot{x}_e \\ \ddot{y}_e \\ \ddot{z}_e \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ m \cdot g \end{bmatrix} + R_b^e \begin{bmatrix} 0 \\ 0 \\ -F_{thrust} \end{bmatrix} \quad (1.6)$$

Para encontrar las ecuaciones de movimiento rotacional del cuadricóptero, es necesario determinar la relación entre las velocidades angulares con referencia al cuerpo y la variación con respecto al tiempo de los ángulos de roll, pitch y yaw con referencia en el marco terrestre [18]. Para ello se emplea la siguiente matriz:

$$J = \begin{bmatrix} 1 & \sin \phi \cdot \tan \theta & \cos \phi \cdot \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & \frac{\sin \phi}{\cos \theta} & \frac{\cos \phi}{\cos \theta} \end{bmatrix} \quad (1.7)$$

Donde,  $J$  representa la matriz de transferencia. Para este caso no se usa la matriz de rotación  $R$  [17].

La velocidad y aceleración angulares se obtienen a partir de las siguientes ecuaciones:

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = J \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (1.8)$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cdot \cos \theta \\ 0 & -\sin \phi & \cos \phi \cdot \cos \theta \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (1.9)$$

Donde la Ecuación (1.8) define el modelo cinemático de rotación del dron, mientras que la Ecuación (1.6), el modelo dinámico de traslación. En base a estas ecuaciones, se definen las variables de estado del modelo:

$$[x_e \ y_e \ z_e \ \dot{x}_e \ \dot{y}_e \ \dot{z}_e \ \psi \ \theta \ \phi \ p \ q \ r]^T \quad (1.10)$$

El momento angular del modelo viene representado por la siguiente ecuación:

$$M = I \cdot \omega \quad (1.11)$$

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = I \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (1.12)$$

Donde,  $I$  es la matriz que expresa la inercia del cuadricóptero alrededor de sus ejes y diferentes movimientos de rotación [17].

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \quad (1.13)$$

Los primeros subíndices de los elementos de  $I$  corresponden al momento angular  $M$ , mientras que los segundos subíndices, a la velocidad angular  $\omega$  referente al dron [17]. Al tener  $M$  y  $\omega$  la misma dirección, algunos elementos de la matriz de inercia se reducen a cero [17] y se obtiene una matriz diagonal de la siguiente forma:

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \quad (1.14)$$

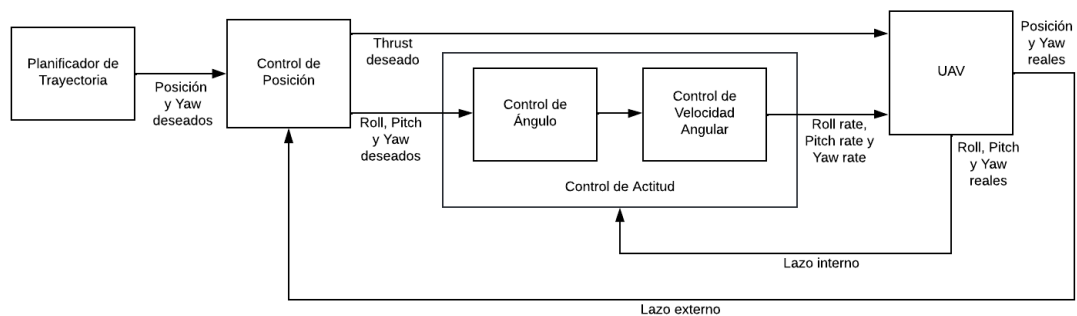
Entonces, se expresa el momento angular así:

$$\begin{bmatrix} M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (1.15)$$

### 1.3.3 SISTEMAS DE CONTROL DE UAVS

El diseño de los sistemas de control de los UAVs es crucial debido a la ausencia de piloto en la aeronave y es de vital importancia debido a la dificultad que implica, ya que su carácter es sub-actuado, multivariable, no lineal e inestable [21].

El sistema de control para los drones consiste, generalmente, de dos lazos. El lazo interno corresponde al control de orientación o actitud, donde se manejan los ángulos de roll, pitch y yaw; y el lazo externo, al control de posición. Dentro del lazo interno, a su vez, se tiene un control en cascada, con un lazo que realiza el seguimiento de la velocidad angular y uno para el seguimiento del ángulo. Mientras que, dentro del lazo externo, la posición real del sistema se compara con la posición deseada [17] (Figura 1.9).



**Figura 1.9.** Diagrama simplificado del sistema de control, adaptado de [17]

Se utilizará el control interno de actitud implementado dentro del bloque del UAV Toolbox, el mismo que no constará de modificaciones dentro de este proyecto. Se propone desarrollar únicamente un control externo de posición, para el cual se consideran: un controlador tipo PID y un controlador basado en Lyapunov.



### 1.3.3.1 Controlador tipo PID

Aunque existen varios métodos de control, los controladores PID se han convertido en la técnica más usada en diferentes aplicaciones debido a su sencilla implementación, ajuste, y que normalmente ofrecen un rendimiento satisfactorio [22]. Son aplicables a sistemas de una entrada, una salida, por lo cual cada variable se debe tratar por separado [23]. Justamente, se puede encontrar este controlador en cuadricópteros, ya que estabilizan el sistema.

El objetivo del controlador PID es corregir la diferencia entre un valor medido y un punto de ajuste deseado. El error puede reducirse ajustando las entradas de control en cada lazo, es decir cada ganancia: proporcional P, integral I y derivativa D. Un sistema PID en un dron básicamente trata de minimizar el error, indicando la velocidad a la que debe girar cada motor [24]. A continuación, se describe cada ganancia y cómo afecta a un cuadricóptero.

- La ganancia proporcional, P, determina la intensidad con la que el controlador de vuelo trabaja para corregir el error y lograr la trayectoria de vuelo deseada. En general, una ganancia P más alta significa un control más agudo, mientras que una ganancia P baja significa un control más suave. Si P es demasiado alta, el cuadricóptero se vuelve demasiado sensible y tiende a sobre corregir, eventualmente causará sobrepicos, y se tendrá oscilaciones de alta frecuencia [24].
- La ganancia integral, I, determina la fuerza con la que el control de vuelo trabaja para mantener la orientación del dron contra fuerzas externas, como el viento. Cuando la ganancia I es demasiado alta, el cuadricóptero estará demasiado limitado y comenzará a sentirse rígido y sin respuesta. En el mundo real, los rotores no son idénticos, por lo que proporcionan diferentes niveles de empuje incluso cuando giran bajo las mismas condiciones, lo que puede causar un movimiento no deseado al disminuir su aceleración; por ello, se puede aumentar la ganancia integral para mejorar el rendimiento de vuelo [24]. En teoría, si se usa un modelo ideal, sin perturbaciones, no es necesario aumentar esta ganancia.
- La ganancia derivativa, D, funciona como un amortiguador y reduce la sobre-corrección y los excesos causados por la ganancia P. La adición de la ganancia D puede "suavizar" y contrarrestar las oscilaciones causadas por la excesiva ganancia P, así como minimizar las oscilaciones causadas por las hélices. Cuando la ganancia D es demasiado baja, el dron tendrá rebotes al final de una voltereta o giro, y también se experimentará peores oscilaciones en los descensos verticales. Aumentar la ganancia de D puede mejorar estos problemas, sin embargo, un valor excesivo de D

puede introducir vibraciones en el cuadricóptero porque amplifica el ruido en el sistema. Eventualmente esto llevará a un sobrecalentamiento del motor y a la oscilación del dron [24].

### 1.3.3.2 Controlador basado en Lyapunov

La estabilidad en el estudio de sistemas de control es importante ya que demuestra continuidad en el comportamiento dinámico de un sistema, por lo que la teoría de estabilidad de Lyapunov es una de las herramientas más relevantes al analizar sistemas no lineales [25], [26]. Estas técnicas determinan si un sistema es estable o asintóticamente estable sin requerir la resolución de ecuaciones [27]. En el siguiente apartado se estudiará el método directo de Lyapunov, ya que puede servir para realizar controladores [25].

Dicho método indica que, si la energía de un sistema es disipada de manera continua, el sistema termina convergiendo hacia un punto de equilibrio [27]. Por lo que, las funciones de Lyapunov demuestran la estabilidad de cierto punto fijo en un sistema dinámico o en las ecuaciones diferenciales autónomas [28]. El objetivo de Lyapunov es definir una función  $V(x)$  que sea de energía positiva, tenga derivadas parciales continuas y su derivada a través de una trayectoria de estado sea  $\dot{V}(x) \leq 0$  [25], [27]. No existe un método sistemático para hallar una función de Lyapunov, por lo que es necesario proponer una función candidata y probar si cumple con los requisitos de estabilidad [28].

En base a lo mencionado, el teorema de Lyapunov dice: “Sea  $x$  un punto de equilibrio del sistema  $\dot{x} = f(x)$  y sea  $V: D \rightarrow \mathfrak{R}$  un campo escalar continuamente diferenciable definido en un dominio  $D \subset \mathfrak{R}^n$  que contiene al origen, entonces [28]:

- Si  $V(x)$  es definida positiva y  $\dot{V}(x)$  es semidefinida negativa, el origen es un punto de equilibrio estable.
- Si  $V(x)$  es definida positiva y  $\dot{V}(x)$  es definida negativa, el origen es un punto de equilibrio asintóticamente estable.”

Habiendo definido el teorema, se realizará un ejemplo tomado de [28] para entender cómo se puede realizar el control basado en Lyapunov de un sistema dinámico. Se propone un sistema inestable  $\dot{x} = x$ . Para controlar este sistema, se debe incluir una variable de control  $\mu$ , tal que:

$$\dot{x} = x + \mu \tag{1.16}$$

Se quiere que  $x$  tienda a un determinado valor de referencia, para esto se toma la variable de error  $e = x - x_{ref}$ , y se supone un valor de  $x$  constante y la derivada temporal de la referencia nula, por lo tanto,

$$\dot{e} = \dot{x} = e + x_{ref} + \mu \quad (1.17)$$

La variable de control  $\mu$  se define usando una función de Lyapunov candidata. Esta función debe tener las siguientes características para que el sistema sea estable:

$$V(x) \geq 0, \quad \forall x \quad (1.18)$$

$$\dot{V}(x) = \frac{dV}{dt} < 0, \quad \forall x \quad (1.19)$$

Entonces se propone la siguiente función candidata:

$$V = \frac{1}{2} e^2 \quad (1.20)$$

Misma que tiene la siguiente derivada temporal:

$$\frac{dV}{dt} = e \cdot \dot{e} = e \cdot (e + x_{ref} + \mu) = e^2 + e \cdot (x_{ref} + \mu) \quad (1.21)$$

Se determina la variable de control  $\mu$  tal que se cumpla  $\dot{V} < 0$ .

$$e \cdot (x_{ref} + \mu) = -k \cdot e^2 \quad (1.22)$$

$$\mu = -k \cdot e - x_{ref} \quad (1.23)$$

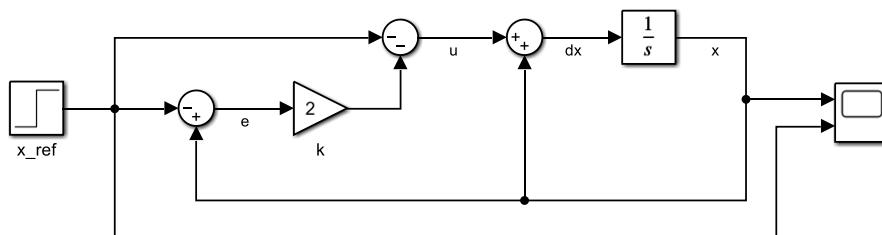
Reemplazando en la Ecuación (1.21), se acota el valor de  $k$ , así:

$$e^2 + e \cdot (x_{ref} - k \cdot e - x_{ref}) \leq 0 \quad (1.24)$$

$$e^2 - k \cdot e^2 \leq 0 \quad (1.25)$$

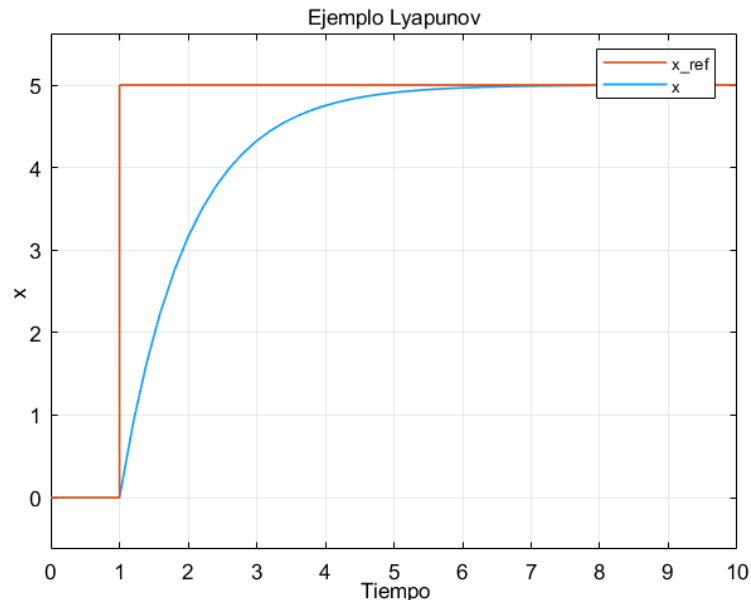
$$k \geq 1 \quad (1.26)$$

A continuación, se presenta el diagrama de bloques del sistema en la Figura 1.10 donde se eligió usar un valor de  $k = 2$  y un valor de referencia  $x_{ref} = 5$ .



**Figura 1.10.** Modelo de ejemplo de control basado en Lyapunov

El sistema tiene como resultado una respuesta suave que puede disminuir su tiempo de establecimiento a medida que se aumenta el valor de  $k$ . Se puede ver el resultado en la Figura 1.11.



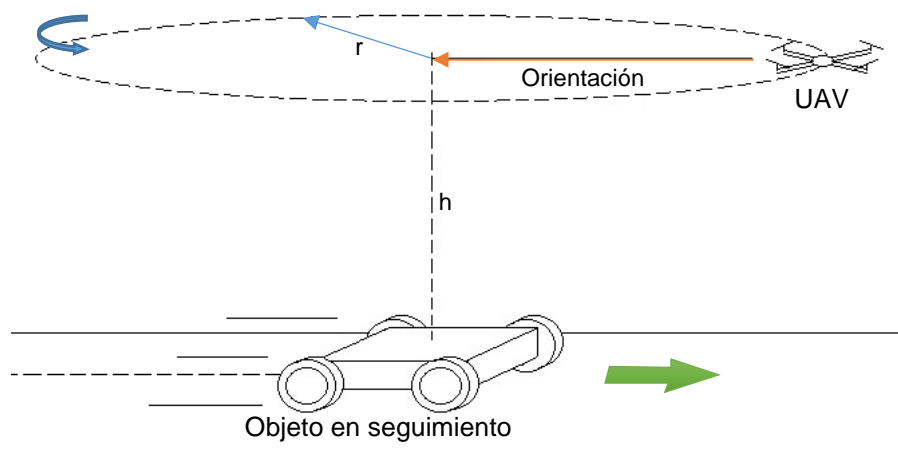
**Figura 1.11.** Respuesta del sistema ante una entrada paso

Encontrar una función de Lyapunov para el análisis de estabilidad a un sistema no lineal puede ser un problema, por lo cual es importante conocer la física del mismo [27]. Para el control de un UAV, es necesario usar las ecuaciones de la dinámica del robot y realizar inicialmente un control de velocidad para después proceder a realizar el control de posición. Esto se podrá ver desarrollado en el segundo capítulo.

#### 1.3.4 CIRCUNNAVEGACIÓN

Para cumplir con el objetivo del trabajo realizado, se requiere definir el movimiento que tendrá el dron a través de un algoritmo de control aparte de los expuestos en la sección anterior. La circunnavegación se refiere al viaje alrededor de un lugar. En general, este término se usa en navegación marítima, sin embargo, también se lo usa para referirse a vuelos, viajes en cualquier vehículo a motor o a pie [29].

Como se ha mencionado anteriormente, unas de las aplicaciones del uso de drones son la vigilancia, reconocimiento y seguimiento de objetos en movimiento. Un UAV puede ser útil en la obtención de información de un objetivo si puede orbitar alrededor del mismo a una distancia deseada. Como se indicaba anteriormente, este movimiento circular alrededor de un objetivo se denomina circunnavegación [6], mismo que se ejemplifica en la Figura 1.12.



**Figura 1.12.** Ejemplificación de la circunnavegación de un UAV sobre un objeto móvil terrestre

En este trabajo se realiza un algoritmo de control con el fin de realizar la circunnavegación de un objeto móvil terrestre por parte de un cuadricóptero; es decir, tratando de mantener el frente del UAV apuntando hacia el objeto en seguimiento. Esto conlleva tanto a un control de posición como del ángulo de *heading* o de dirección.

### 1.3.5 UAV TOOLBOX

Debido al modelo dinámico que presentan los drones, se debe crear un control de orientación y posición para poder manejarlos. Al ser un dispositivo que se eleva, se vuelven peligrosos si no se tiene un correcto control de estos o las respectivas medidas de seguridad. Por ello, una simulación de los mismos ayuda a analizar de mejor manera su funcionamiento para su uso adecuado, presentando como ventaja la conservación del hardware y seguridad para los usuarios. Debido a que muchas aplicaciones de UAVs incorporan tecnologías para el funcionamiento autónomo, se necesita de la simulación para garantizar un funcionamiento adecuado. El diseño basado en modelos permite desarrollar sistemas y aplicaciones, evaluarlos mediante pruebas virtuales y posteriormente implementarlos en hardware [30].

Justamente para este fin, Simulink, que es una extensión de MATLAB, permite realizar programación mediante el uso de bloques y simular diferentes sistemas dinámicos. Cada bloque define una parte del sistema y puede representar una parte física, un sistema pequeño interno o una función [31]. Con esto, el usuario obtiene un sistema interactivo e intuitivo que permite crear modelos de sistemas de manera rápida con el beneficio de poder probar el funcionamiento del sistema sin el uso de hardware. Simulink y MATLAB también

permiten añadir complementos para ampliar sus capacidades, los mismos que pueden ser apps, toolboxes o herramientas, paquetes de soporte y funcionalidades opcionales [32].

Actualmente, dentro de MATLAB y Simulink se encuentra disponible el UAV Toolbox, el cual es un complemento creado por MathWorks y lanzado en el 2020. Este provee herramientas para diseñar, simular sistemas y probar aplicaciones de UAVs autónomos en Simulink [33]. Con esta herramienta se pueden realizar metodologías de modelado para sistemas UAVs, diseño de algoritmo, simulación de aplicaciones autónomas con modelos de sensores, generación automática de software e implementación en el hardware, entre otros [30].

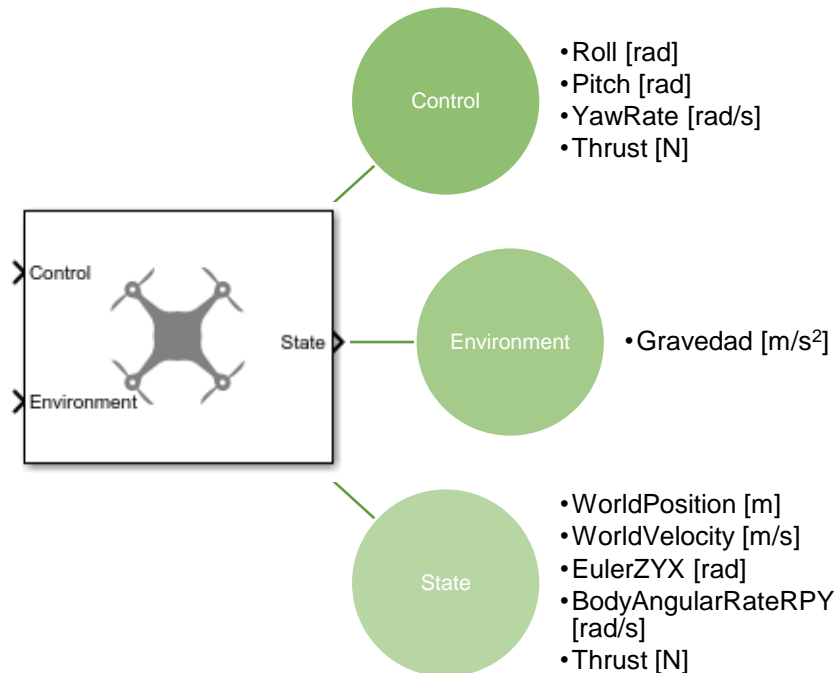
El UAV Toolbox proporciona herramientas y aplicaciones de referencia. Este toolbox contiene varios bloques con el modelado de un UAV de ala fija y cuadrirrotor, entre ellos UAV Guidance Model, Fixed-Wing UAV Point Mass, Waypoint Follower, etc. El usado en este trabajo es el UAV Guidance Model debido a que es un bloque con el modelo de orden reducido para UAVs. A su vez, contiene bloques que permiten la visualización del movimiento del dron en 2.5D y 3D [33]. Para este trabajo también se utiliza el bloque UAV Animation, donde se puede visualizar el vuelo animado del dron usando traslación y rotación en las entradas del bloque.

Para el uso de esta herramienta es necesario tomar en cuenta las diferentes especificaciones en cuanto a los requerimientos para usar ciertos bloques del Toolbox, así como el hecho de que el bloque a usar en este trabajo con el modelo de UAV es aproximado, es decir, no se pueden incluir todas las características del dron que se quiere modelar. Un modelo aproximado será capaz de procesar los datos de manera más rápida, pero no tendrá de manera detallada todos sus componentes. Si se desea realizar un programa autónomo de un UAV, es suficiente empezar con un modelo aproximado pues con ello se puede realizar un mayor enfoque en el algoritmo autónomo del dron. Adicionalmente, de ser necesario, se puede hacer una transición de un modelo de baja a alta fidelidad utilizando los bloques del UAV Toolbox apropiados [30].

El UAV Toolbox utiliza la convención del sistema de coordenadas Norte-Este-Abajo (NED, North-East-Down), tal como se planteó en la Sección 1.3.2.3. El vector de posición del UAV consiste en tres números para la posición a lo largo del eje norte, el eje este y la posición vertical. El elemento descendente cumple con la regla de la mano derecha y resulta en valores negativos para la ganancia de altitud [20].

### 1.3.5.1 UAV Guidance Model

Este bloque representa el modelo de un UAV pequeño que estima su estado basándose en las entradas de control y del entorno. El modelo se aproxima al comportamiento de un sistema de lazo cerrado que consiste en un controlador de piloto automático y un modelo cinemático de ala fija o multirrotor para el movimiento 3D [20].



**Figura 1.13.** Bloque UAV Guidance Model, tipo de modelo MultirrotorGuidance, adaptado de [20]

Como se requiere el modelo de un cuadrirrotor, véase Figura 1.13, las entradas de control son:

- Roll, en radianes;
- Pitch, en radianes;
- YawRate, en radianes por segundo; y
- Thrust, en Newtons.

La entrada del entorno para el multirrotor, necesaria para conseguir los controles ordenados, es la gravedad dada en metros por segundo al cuadrado [20]. A la salida se obtiene el estado del UAV en cinco elementos de un bus:

- WorldPosition, la posición de  $x$ ,  $y$ ,  $z$  en metros con respecto al mundo;
- WorldVelocity, la velocidad de  $x$ ,  $y$ ,  $z$  en metros por segundo con respecto al mundo;
- EulerZYX, los ángulos de Euler  $\psi$ ,  $\phi$ ,  $\theta$  en radianes;

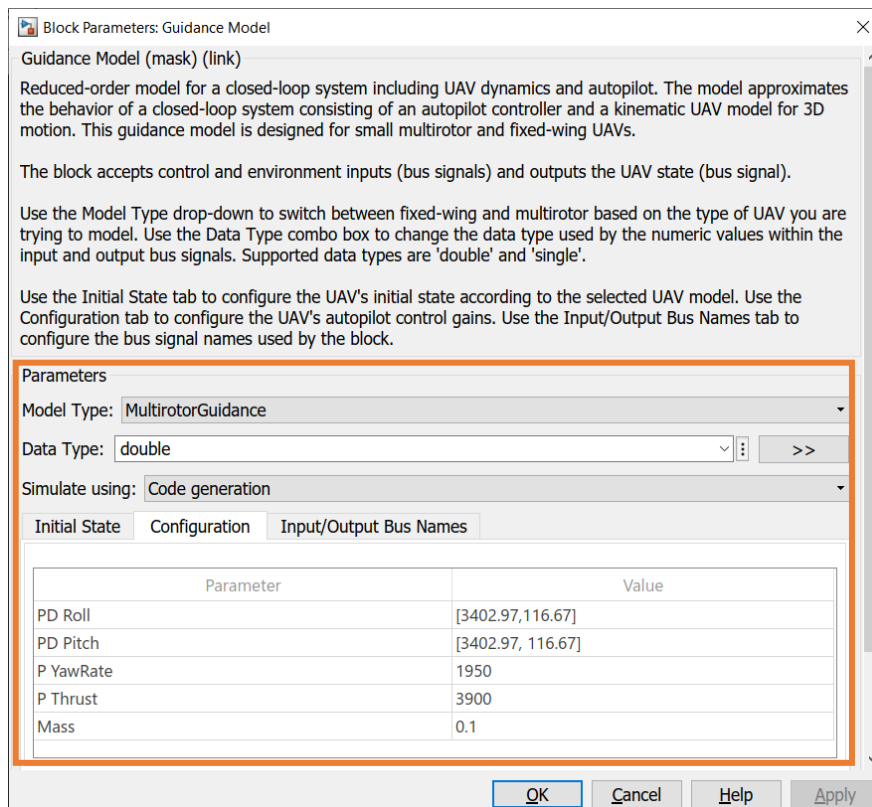
- BodyAngularRateRPY, la velocidad angular del UAV r, p, q en radianes por segundo a lo largo de los ejes x, y, z del UAV; y
- Thrust, la fuerza resultante de empuje en Newtons [20].

Como se mencionó en la Sección 1.3.3, el modelo del bloque Guidance Model tiene un control interno cerrado de actitud de roll-pitch. Este control tiene 2 controladores PD independientes para los dos ángulos de rotación y 2 controladores P para Yaw Rate y Thrust [20]. De manera que se tiene las siguientes ecuaciones para la velocidad angular y el Thrust:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cdot \cos \theta \\ 0 & -\sin \phi & \cos \phi \cdot \cos \theta \end{bmatrix} \begin{bmatrix} K_{P,\phi}(\dot{\phi}_d - \dot{\phi}) + K_{D,\phi}(-\dot{\phi}) \\ K_{P,\theta}(\dot{\theta}_d - \dot{\theta}) + K_{D,\theta}(-\dot{\theta}) \\ K_{P,\psi}(\dot{\psi}_d - \dot{\psi}) \end{bmatrix} \quad (1.27)$$

$$\dot{F}_{thrust} = K_{P,F}(F_{thrust_d} - F_{thrust}) \quad (1.28)$$

Las ganancias P y D para las entradas de control se especifican como  $K_{P,\alpha}$  y  $K_{D,\alpha}$ , donde  $\alpha$  es el ángulo de rotación o el thrust [20]. Estas ganancias, junto con la masa del UAV,  $m$ , se especifican en la propiedad de configuración del objeto multirrotor, como se puede ver en la Figura 1.14. Se decidió mantener los valores predefinidos para este proyecto, incluido el valor de 0.1 [kg] para la masa del UAV.



**Figura 1.14.** Parámetros del bloque UAV Guidance Model



## **2. METODOLOGÍA**

En esta sección, se detalla la metodología aplicada en este trabajo de titulación donde se describe el procedimiento desarrollado durante la investigación. En la recolección de información, se utilizan fuentes primarias como libros, revistas técnicas y literatura científica, y fuentes secundarias como trabajos de titulación, artículos académicos, informes relacionados a robótica móvil, etc.

En el primer capítulo, se realizó una revisión de los conceptos que componen el marco teórico de este proyecto. Con la información provista, en este segundo capítulo se procede a trabajar en el desarrollo teórico del control de posición de un cuadricóptero a través de los controladores tipo PID y basados en Lyapunov, y se presenta la calibración de los diferentes controladores usando un índice de desempeño. Además, se determina el algoritmo de control a usarse para realizar la circunnavegación del dron sobre un objeto móvil terrestre. A partir de esto, se procede a implementar el escenario de simulación del proyecto y una interfaz gráfica que trabaja como herramienta para la visualización y obtención de resultados. Finalmente, se realiza la implementación de los controladores y algoritmos desarrollados.

En el capítulo tres, se presentan los resultados de una serie de pruebas simuladas de los controles propuestos y su correspondiente análisis comparativo considerando indicadores de desempeño. Por último, en el capítulo cuatro se comparte el análisis de resultados a manera de conclusiones y recomendaciones de todo el trabajo realizado.

### **2.1 DESARROLLO DE CONTROLADORES**

Uno de los objetivos del presente trabajo de titulación es el diseño de control tipo PID y control basado en Lyapunov de un UAV para el manejo de su posición y direccionamiento. En esta sección se presenta el desarrollo de dichos controladores y su implementación.

#### **2.1.1 LINEALIZACIÓN**

Antes de presentar los controladores, es importante mencionar que, al tener un sistema no lineal con el cuadricóptero, comúnmente se considera un modelo lineal invariante en el tiempo para su representación en espacios de estado, por lo cual se realiza una linealización del modelo matemático [15]. Para ello, se tomó en consideración el punto de equilibrio del sistema cuando el dron está en un estado estacionario a una cierta altura, lo cual se conoce como hover [15]. El vector que se obtiene de la Ecuación (1.10) en el punto de equilibrio es:

$$[x_e \ y_e \ z_e \ 0 \ 0 \ 0 \ \psi \ 0 \ 0 \ 0 \ 0 \ 0]^T \quad (2.1)$$

Aplicando la expansión de primer orden de Taylor, alrededor del punto de equilibrio, se obtienen las siguientes ecuaciones linealizadas [15]:

$$\begin{cases} \Delta F_z = -m \cdot \Delta \dot{w} + m \cdot g \\ \Delta M_x = I_{xx} \cdot \Delta \dot{p} \\ \Delta M_y = I_{yy} \cdot \Delta \dot{q} \\ \Delta M_z = I_{zz} \cdot \Delta \dot{r} \end{cases} \quad (2.2)$$

En el estado de hover, el body frame y el marco terrestre coinciden [15], lo que significa que:

$$\begin{cases} \Delta \dot{x} = \Delta u \\ \Delta \dot{y} = \Delta v \\ \Delta \dot{z} = \Delta w \end{cases} \quad \begin{cases} \Delta \phi = \Delta p \\ \Delta \theta = \Delta q \\ \Delta \psi = \Delta r \end{cases} \quad (2.3)$$

Se representa el modelo en variables de estado de la siguiente manera:

$$\begin{bmatrix} \Delta \dot{x} \\ \Delta \dot{y} \\ \Delta \dot{z} \\ \Delta \dot{\psi} \\ \Delta \dot{\theta} \\ \Delta \dot{\phi} \\ \Delta \dot{u} \\ \Delta \dot{v} \\ \Delta \dot{w} \\ \Delta \dot{r} \\ \Delta \dot{q} \\ \Delta \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \\ \Delta \psi \\ \Delta \theta \\ \Delta \phi \\ \Delta u \\ \Delta v \\ \Delta w \\ \Delta r \\ \Delta q \\ \Delta p \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -\frac{1}{m} & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{I_{zz}} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{I_{yy}} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{I_{xx}} & 0 \end{bmatrix} \begin{bmatrix} \Delta F_z \\ g \\ \Delta M_z \\ \Delta M_y \\ \Delta M_x \end{bmatrix} \quad (2.4)$$

Las fuerzas verticales, laterales, longitudinales y direccionales actúan independientemente una de la otra en el modelo matemático del cuadricóptero y, por lo tanto, la dinámica del robot se puede desacoplar en subsistemas [15]. Se obtienen cuatro subsistemas, lo que implica que se puede descomponer el sistema en sus distintos ejes para realizar el control de manera independiente [34].

### 2.1.1.1 Subsistema vertical

El subsistema vertical describe el movimiento ascendente y descendente del cuadricóptero, es decir, el Thrust; y se obtiene el siguiente espacio de estado:

$$\begin{bmatrix} \Delta \dot{z} \\ \Delta \dot{w} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta z \\ \Delta w \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -1/m & 1 \end{bmatrix} \begin{bmatrix} \Delta F_z \\ g \end{bmatrix} \quad (2.5)$$

Con lo cual se puede obtener su función de transferencia de la siguiente manera:

$$\Delta \ddot{z} = -\frac{\Delta F_z}{m} + g \quad (2.6)$$

$$G_h(s) = \frac{\Delta z}{m \cdot g - \Delta F_z} = \frac{1}{m \cdot s^2} \quad (2.7)$$

### 2.1.1.2 Subsistema Direccional

El movimiento de heading del dron viene dado por el ángulo de yaw y su velocidad, por lo que el subsistema de dirección se denota de la siguiente manera:

$$\begin{bmatrix} \Delta \dot{\psi} \\ \Delta \dot{r} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta \psi \\ \Delta r \end{bmatrix} + \begin{bmatrix} 0 \\ 1/I_{zz} \end{bmatrix} \Delta M_z \quad (2.8)$$

Obteniendo su función de transferencia a partir de las siguientes ecuaciones:

$$\Delta \ddot{\psi} = \frac{1}{I_{zz}} \Delta M_z \quad (2.9)$$

$$G_\psi(s) = \frac{\Delta \psi}{\Delta M_z} = \frac{1}{I_{zz} \cdot s^2} \quad (2.10)$$

### 2.1.1.3 Subsistema Lateral

Para el subsistema del movimiento lateral, se usa el ángulo de pitch y su posición y en el marco inercial de referencia, con lo que se obtiene el siguiente espacio de estado:

$$\begin{bmatrix} \Delta \dot{y} \\ \Delta \dot{v} \\ \Delta \dot{\phi} \\ \Delta \dot{p} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta y \\ \Delta v \\ \Delta \phi \\ \Delta p \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/I_{xx} \end{bmatrix} \Delta M_x \quad (2.11)$$

Basándose en el espacio de estado, se obtiene la tercera función de transferencia así:

$$\Delta \ddot{\phi} = \frac{1}{I_{xx}} \Delta M_x \quad (2.12)$$

$$G_\phi(s) = \frac{\Delta \phi}{\Delta M_x} = \frac{1}{I_{xx} \cdot s^2} \quad (2.13)$$

Para este subsistema también es necesario tomar en cuenta que se tiene una relación entre  $y$  y  $\phi$  que no se puede evaluar con este espacio de estado formado.

### 2.1.1.4 Subsistema Longitudinal

De manera similar al anterior subsistema, se obtiene el espacio de estado para el movimiento alrededor del eje  $x$  del cuadricóptero, y su posición y velocidad en el marco terrestre en el subsistema longitudinal.

$$\begin{bmatrix} \Delta \dot{x} \\ \Delta \dot{u} \\ \Delta \dot{\theta} \\ \Delta \dot{q} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta u \\ \Delta \theta \\ \Delta q \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1/I_{yy} \end{bmatrix} \Delta M_y \quad (2.14)$$

Para lo cual, se obtiene la última función de transferencia del sistema.

$$\Delta \ddot{\theta} = \frac{1}{I_{yy}} \Delta M_y \quad (2.15)$$

$$G_{\theta}(s) = \frac{\Delta \theta}{\Delta M_y} = \frac{1}{I_{yy} \cdot s^2} \quad (2.16)$$

Al igual que en el subsistema lateral, es importante mencionar que existe una relación entre  $x$  y  $\theta$  que no se muestra en este espacio de estado.

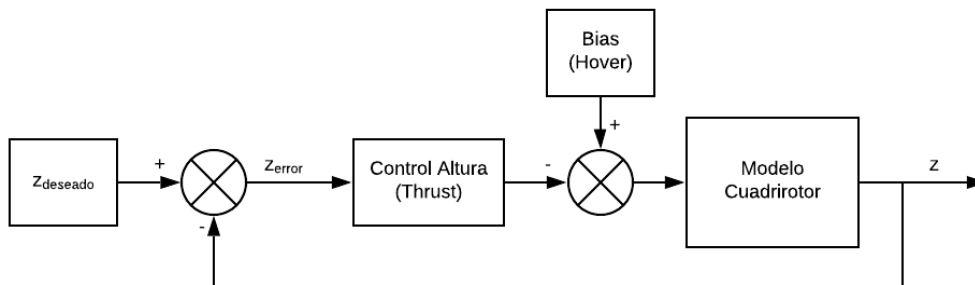
## 2.1.2 CONTROLADOR TIPO PID

Tomando en cuenta que la forma de las funciones de transferencia obtenidas para cada subsistema es de segundo orden sin amortiguamiento, se puede definir al sistema como inestable, ya que tiene dos polos en el origen. Por lo que, se decidió implementar un control PD para cada eje de movimiento.

Para realizar un control total de posición del UAV, se han desarrollado los controladores siguiendo un orden en específico. Primero, se diseña el control de altura del dron; segundo, se implementa el control en  $x$ ; y, por último, el control en  $y$ .

### 2.1.2.1 Control PD de altura

Se toma en consideración al dron en su estado estacionario de hovering, donde solo va a realizar un movimiento vertical y se mantendrá en la altura deseada. La forma del sistema controlado viene representada como se ve en la Figura 2.1, donde resaltan cuatro bloques: la altura deseada, el control de altura o de Thrust, un valor Bias, y el modelo del cuadrirrotor.



**Figura 2.1.** Diagrama de bloques del control de altura

Para que el dron se encuentre en modo hovering, se debe compensar el peso del cuadricóptero con un valor específico de Thrust, mismo que en este trabajo se conoce

como Bias. Este valor se obtiene de manera teórica, tomando en cuenta que los ángulos de rotación roll, pitch, yaw, y los valores de posición  $x$ ,  $y$ ,  $z$  son iguales a cero. Entonces, en la Ecuación (1.6) se reemplazan las variables mencionadas y se obtiene el siguiente modelo en el espacio de estados:

$$m \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ m \cdot g \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ -F_{thrust} \end{bmatrix} \quad (2.17)$$

$$\begin{bmatrix} 0 \\ 0 \\ m \cdot g \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ F_{thrust} \end{bmatrix} \quad (2.18)$$

Se obtiene entonces,

$$F_{thrust} = m \cdot g = 0.1 \cdot 9.8066 = 0.98066 [N] \quad (2.19)$$

Este valor se añade al sistema para, a partir de eso, poder realizar el control del cuadrirrotor.

Para el control de altura se tomó en cuenta el modelo simplificado del UAV de la Ecuación (2.7). Conociendo que la masa  $m$  es de 0.1 [kg], obtenida por default en el UAV Guidance Model, se tiene:

$$G_h(s) = \frac{1}{0.1 \cdot s^2} \quad (2.20)$$

Dado que la función de transferencia de la planta es de segundo orden, se usa un controlador PD:

$$G_{c_h}(s) = K_{p_h}(1 + T_{D_h} \cdot s) \quad (2.21)$$

Por lo que, la función de transferencia en lazo cerrado tiene la siguiente forma:

$$G(s) = \frac{G_h(s) \cdot G_{c_h}(s)}{1 + G_h(s) \cdot G_{c_h}(s)} \quad (2.22)$$

Reemplazando la Ecuación (2.20) y Ecuación (2.21) en la Ecuación (2.22), se tiene:

$$G(s) = \frac{\frac{K_{p_h}(1 + T_{D_h} \cdot s)}{0.1 \cdot s^2}}{1 + \frac{K_{p_h}(1 + T_{D_h} \cdot s)}{0.1 \cdot s^2}} \quad (2.23)$$

$$G(s) = \frac{K_{p_h}(1 + T_{D_h} \cdot s)}{0.1 \cdot s^2 + K_{p_h}(1 + T_{D_h} \cdot s)} = \frac{10 \cdot K_{p_h}(1 + T_{D_h} \cdot s)}{s^2 + 10 \cdot K_{p_h} \cdot T_{D_h} \cdot s + 10 \cdot K_{p_h}} \quad (2.24)$$

Obteniendo una función de transferencia de segundo orden con amortiguamiento, y tomando en cuenta que:

$$2 \cdot \xi \cdot \omega_n = 10 \cdot K_{P_h} \cdot T_{D_h} \quad (2.25)$$

$$\omega_n^2 = 10 \cdot K_{P_h} \quad (2.26)$$

Según lo visto en la documentación de [35] y [36], se ha podido observar que para drones con un peso cercano a 0.1 [kg], semejante a la característica del modelo del UAV Toolbox, se puede considerar que la velocidad máxima de ascenso es de 5 [m/s], por lo cual, para la aplicación requerida en este trabajo de titulación, se define una velocidad aproximada de 1 [m/s]. La altura máxima con la que se realizarán las pruebas para este trabajo es de 10 [m], por lo tanto, se determina un valor de tiempo de establecimiento igual a 10 segundos para realizar el diseño del controlador.

$$T_s = \frac{4}{\xi \cdot \omega_n} = 10 \text{ [seg]} \quad (2.27)$$

Se asume un valor de factor de amortiguamiento  $\xi$  de 0.65, valor que define una respuesta de un sistema subamortiguado [37]. Reemplazando en la Ecuación (2.27), se obtiene un valor de frecuencia natural  $\omega_n$ .

$$T_s = \frac{4}{0.65 \cdot \omega_n} = 10 \text{ [seg]} \quad (2.28)$$

$$\omega_n = 0.61538 \text{ [Hz]} \quad (2.29)$$

Usando este valor se obtiene  $K_{P_h}$  al reemplazarlo en la Ecuación (2.26).

$$(0.61538)^2 = 10 \cdot K_{P_h} \quad (2.30)$$

$$K_{P_h} = \frac{0.61538^2}{10} = 0.03787 \quad (2.31)$$

Y, usando los valores obtenidos, se usa la Ecuación (2.25) para encontrar  $T_{D_h}$ .

$$2(0.65)(0.61538) = 10(0.03787)T_{D_h} \quad (2.32)$$

$$T_{D_h} = \frac{2(0.65)(0.61538)}{10(0.03787)} = 2.11247 \quad (2.33)$$

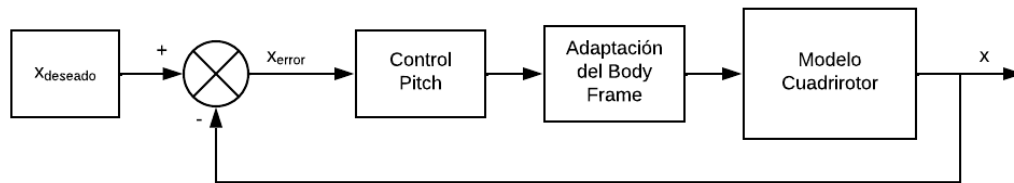
Es necesario mencionar que valores altos en Thrust llegarán a ser un problema para un UAV físico ya que esos valores implican que se requiere mucha velocidad de las hélices para realizar el cambio de referencia. Debido a que el controlador puede llegar a enviar valores muy altos en su acción de control, se decidió realizar una saturación al mismo. El valor de saturación se debe tomar de la capacidad máxima de velocidad de los rotores del dron a usar. En este caso, al ser un cuadricóptero de simulación sin características físicas específicas, se decidió usar la mitad del valor de Thrust del dron en modo hovering, es decir,

$$\text{Límite máximo de Thrust} = \frac{0.98066 [N]}{2} = 0.49033 [N] \quad (2.34)$$

$$\text{Límite mínimo de Thrust} = -\frac{0.98066 [N]}{2} = -0.49033 [N] \quad (2.35)$$

### 2.1.2.2 Control PD de x

Una vez realizado el control de altura, se procede a realizar el control del movimiento en x del cuadricóptero. Para ello, se debe realizar un control al ángulo de pitch, como se ve en la Figura 2.2.



**Figura 2.2.** Diagrama de bloques del control de x

Así como en la altura, se tiene un subsistema desacoplado para realizar esta acción. La función de transferencia del modelo viene dada de la Ecuación (2.13); sin embargo, debido a su forma y a que se desconoce el valor de inercia del modelo, se realizó un control PD de forma experimental, con la siguiente forma:

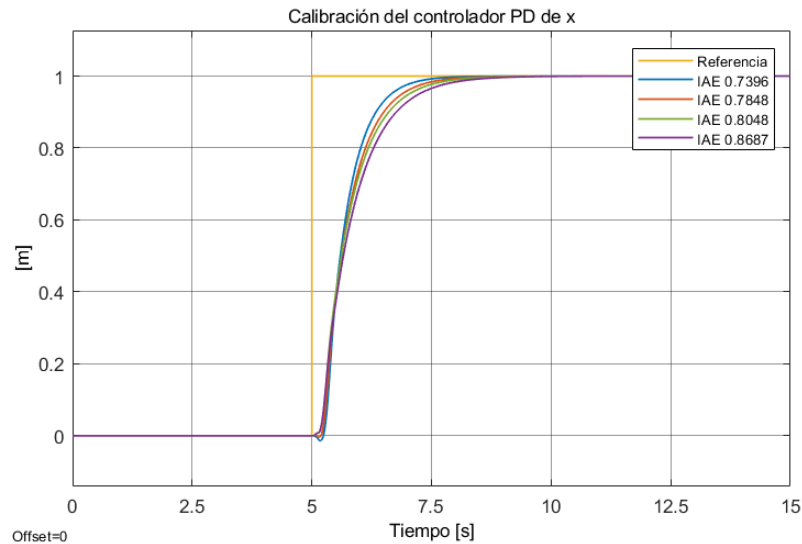
$$G_{c_x}(s) = K_{P_x}(1 + T_{D_x} \cdot s) \quad (2.36)$$

Para realizar la calibración de las ganancias del controlador, se determinó un valor aproximado usando la herramienta de Matlab, PID Tuner, donde se obtuvieron valores de referencia iniciales. Se realizó una tabla donde se ubicaron los valores incrementados en un 5%, tanto para la ganancia proporcional, como para la derivativa. Con cada par de valores, a una entrada paso de 1m, se determinó el valor del error usando el índice de desempeño IAE, integral del error absoluto, con la finalidad de analizar cuáles son los valores que permiten que la respuesta se aproxime más a la referencia (Tabla 2.1).

**Tabla 2.1.** Valores experimentales para calibración del controlador PD de x

$T_D/K_P$	<b>2.016</b>	<b>2.128</b>	<b>2.24</b>	<b>2.352</b>	<b>2.464</b>
<b>1.215</b>	0.7687	0.7521	0.7422	0.7396	0.7486
<b>1.2825</b>	0.8208	0.8029	0.7918	0.7897	0.8015
<b>1.35</b>	0.8289	0.804	0.7848	0.7708	0.7628
<b>1.4175</b>	0.8327	0.8048	0.7813	0.7626	0.7488
<b>1.485</b>	0.8687	0.8402	0.818	0.8003	0.7897

En la Figura 2.3, se presentan cuatro respuestas que corresponden a diferentes pares de ganancias de la calibración realizada para el controlador PD de  $x$ . En la gráfica se identifican las salidas con los valores de IAE de la tabla anterior.



**Figura 2.3.** Diagrama de una muestra de valores de la calibración del controlador PD de  $x$

Al realizar el análisis de la tabla, se eligió el menor valor de IAE, el cual se encuentra definido por un recuadro naranja, y con ello, los valores obtenidos para las ganancias del controlador fueron los siguientes:

$$K_{P_x} = 2.352 \quad (2.37)$$

$$T_{D_x} = 1.215 \quad (2.38)$$

Para la implementación del controlador en el modelo del UAV Toolbox, es necesario tomar en cuenta el cambio de referencia del body frame hacia el marco terrestre; con esta consideración, se estaría haciendo una rotación en el eje  $y$ . Tomando como referencia el eje  $y$  del body frame como positivo, se define de acuerdo a la regla de la mano derecha el desplazamiento del dron en el eje  $x$  con respecto al movimiento del ángulo de pitch ( $\theta$ ), como se puede observar en la Figura 1.8. Con esto, el cuadricóptero se desplaza en el eje  $x$  negativo cuando el ángulo de pitch gira en sentido antihorario. Por lo cual, al querer realizar un movimiento positivo en  $x$ , el giro del ángulo de pitch debe ser horario y con acción del controlador inversa. Debido a lo explicado, se aumenta el bloque de Adaptación del Body Frame (Figura 2.2), el cual hace que la acción total del controlador sea negativa.

A su vez, así como en el control de altura, es necesario aumentar una saturación debido a las características de los drones, donde se debe tomar en consideración un valor máximo



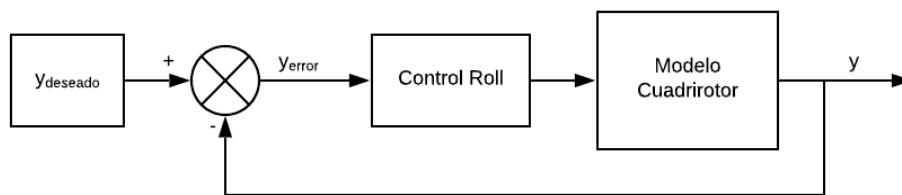
de ángulo de pitch para evitar que se descompense y caiga. Dicho valor se definió en  $10^\circ$ , y transformándolo a radianes, ya que esa es la unidad que requiere el modelo, se obtiene:

$$\text{Límite máximo de pitch} = 10^\circ \cdot \frac{\pi}{180^\circ} = 0.1745 \text{ [rad]} \quad (2.39)$$

$$\text{Límite mínimo de pitch} = -10^\circ \cdot \frac{\pi}{180^\circ} = -0.1745 \text{ [rad]} \quad (2.40)$$

### 2.1.2.3 Control PD de y

De igual manera, el control del movimiento en y del cuadricóptero se define con el control del ángulo de roll (Figura 2.4).



**Figura 2.4.** Diagrama de bloques del control de y

La función de transferencia tomada en consideración para realizar el control de y se obtiene de la Ecuación (2.16). Dado que el modelo se asume como simétrico, los valores de inercia deberían ser iguales; sin embargo, al sintonizar el controlador, se obtienen valores distintos a los encontrados para x. Por esto, los valores encontrados para el control PD de roll son los siguientes:

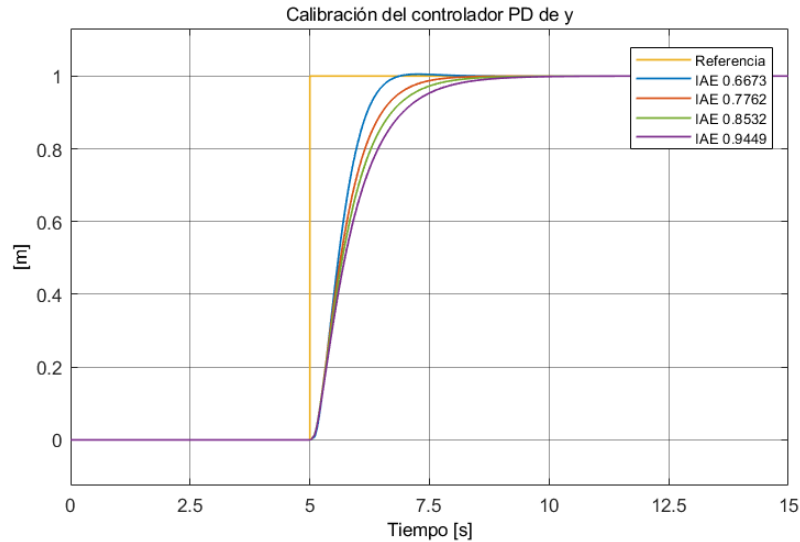
$$G_{c_y}(s) = K_{P_y} (1 + T_{D_y} \cdot s) \quad (2.41)$$

Así como en el controlador de x, para el eje y se realizó una calibración mediante el índice de rendimiento IAE, con una entrada paso de 1m, resultando en una tabla de valores de cada par de ganancias, tanto proporcional como derivativa, con variación del 5% (Tabla 2.2).

**Tabla 2.2.** Valores experimentales para calibración del controlador PD de y

<b><math>T_D/T_P</math></b>	<b>0.495</b>	<b>0.5225</b>	<b>0.55</b>	<b>0.5775</b>	<b>0.605</b>
<b>0.396</b>	0.7852	0.7492	0.7178	0.6907	0.6673
<b>0.418</b>	0.8161	0.7777	0.7433	0.7126	0.6855
<b>0.44</b>	0.8525	0.8123	0.7762	0.7436	0.7142
<b>0.462</b>	0.8954	0.8532	0.8152	0.7811	0.75
<b>0.484</b>	0.9449	0.9003	0.8605	0.8244	0.7917

En la Figura 2.5, se presentan cuatro respuestas que corresponden a diferentes pares de ganancias de la calibración realizada para el controlador PD de  $y$ . En la gráfica se identifican las salidas con los valores de IAE de la tabla anterior.



**Figura 2.5.** Diagrama de una muestra de valores de la calibración del controlador PD de  $y$

Realizando un análisis de la tabla anterior, se eligió el menor valor de IAE, el cual se encuentra definido por un recuadro naranja, y se determinaron los siguientes valores para las ganancias del controlador PD de  $y$ :

$$K_{P_y} = 0.605 \quad (2.42)$$

$$T_{D_y} = 0.396 \quad (2.43)$$

En este caso, los valores se mantienen positivos debido a que el sentido de giro antihorario del ángulo de roll coincide con el desplazamiento del dron en el eje  $y$  positivo, ver Figura 1.8.

Se realizó también una saturación a este controlador, donde se toma en consideración un valor máximo de ángulo de roll para evitar que se descompense. De manera similar a lo anterior, este valor es:

$$\text{Límite máximo de roll} = 10^\circ \cdot \frac{\pi}{180^\circ} = 0.1745 \text{ [rad]} \quad (2.44)$$

$$\text{Límite mínimo de roll} = -10^\circ \cdot \frac{\pi}{180^\circ} = -0.1745 \text{ [rad]} \quad (2.45)$$

### 2.1.3 CONTROLADOR BASADO EN LYAPUNOV

Así como en el control tipo PID, para este caso se desarrollan los controladores de cada eje de movimiento en el orden específico mencionado en el anterior apartado. La diferencia en este controlador es que resulta necesario implementar dos controladores basados en Lyapunov para cada subsistema; uno para controlar la velocidad y otro para el control de posición. La razón de esto viene descrita a continuación, en base a la ecuación del modelo cinemático del dron con un solo lazo de control.

Tomando la variable  $\alpha$  como una variable de posición genérica, se tiene la siguiente ecuación del modelo cinemático de una partícula:

$$F_{\alpha} = m \cdot a_{\alpha} \quad (2.46)$$

Se busca encontrar una ecuación en función de la posición, a partir de la ecuación anterior. Para ello, se realiza el siguiente artificio matemático:

$$\frac{F_{\alpha}}{m} = \frac{dv_{\alpha}}{dt} \cdot \frac{d\alpha}{d\alpha} \quad (2.47)$$

$$\frac{F_{\alpha}}{m} = v_{\alpha} \cdot \frac{dv_{\alpha}}{d\alpha} \quad (2.48)$$

Al realizar la integral de ambas partes, se obtiene la siguiente ecuación:

$$\frac{v_{\alpha}^2}{2} = \frac{F_{\alpha}}{m} \cdot \alpha ; v_{\alpha} = \dot{\alpha} \quad (2.49)$$

$$\dot{\alpha}^2 = \frac{2 \cdot F_{\alpha}}{m} \cdot \alpha \quad (2.50)$$

$$\dot{\alpha}^2 = K \cdot \alpha \quad (2.51)$$

Donde,  $K$  es una constante.

La Ecuación (2.51), que describe el movimiento de la partícula en función de la posición, es una ecuación inestable, por lo cual, se sigue el procedimiento descrito en la Sección 1.3.3.2 donde:

$$e = \alpha - \alpha_{ref} \quad (2.52)$$

$$\dot{e} = \dot{\alpha} \quad (2.53)$$

Se propone la siguiente función candidata de Lyapunov que cumple la primera condición de estabilidad descrita en la Ecuación (1.18):

$$V(\alpha) = \frac{1}{2} \cdot e^2 \quad (2.54)$$

Misma que tiene la siguiente derivada temporal:

$$\dot{V}(\alpha) = e \cdot \dot{e} = (\alpha - \alpha_{ref}) \cdot \dot{\alpha} \quad (2.55)$$

De la Ecuación (2.51), se obtiene la siguiente expresión:

$$\dot{\alpha} = \sqrt{K \cdot \alpha} \quad (2.56)$$

Siendo  $\dot{\alpha}$  igual a la raíz cuadrada de una constante multiplicada por la posición de la partícula, existe la posibilidad de tener dos resultados: uno positivo y uno negativo. Por ende, al reemplazar la Ecuación (2.56) en la Ecuación (2.55), no se podría determinar una expresión, tal que se cumpla la segunda condición de estabilidad de Lyapunov (Ecuación (1.19)).

Para lograr realizar un control basado en Lyapunov a este modelo matemático específico, se propone separar la Ecuación (2.51) de segundo grado en dos secciones, realizando el siguiente artificio matemático:

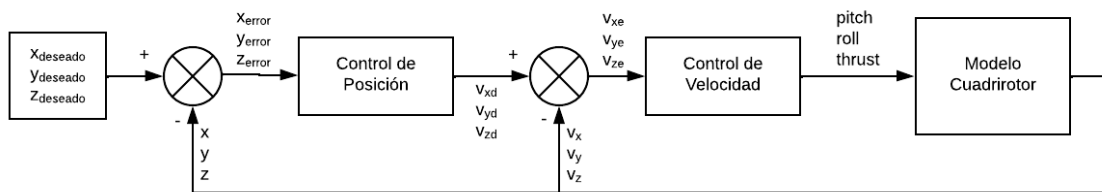
$$K_1 = \frac{\alpha}{\dot{\alpha}} = K_2 \cdot \dot{\alpha} \quad (2.57)$$

$$\alpha = K_1 \cdot \dot{\alpha} \quad (2.58)$$

$$K_1 = K_2 \cdot \dot{\alpha} \quad (2.59)$$

Dando como resultado dos subsistemas con dos leyes de control, que buscan relacionar posición, con velocidad y con la acción final de control del modelo.

Por ello, en la Figura 2.6, se puede visualizar de manera general un diagrama que muestra la forma en que se implementa este control.



**Figura 2.6.** Diagrama de control basado en Lyapunov

Para realizar los controladores, se toma en consideración los pasos realizados en el ejemplo de la Sección 1.3.3.2.

### 2.1.3.1 Control de altura basado en Lyapunov

Dado que se asume que el cuadriróptero tiene un cuerpo rígido y simétrico, se define el modelo dinámico del dron visto como una partícula, tomando en cuenta el movimiento en el eje  $z$  únicamente, de la siguiente manera:

$$F_z = m \cdot a_z \quad (2.60)$$

Tomando en cuenta que se tiene una fuerza de empuje  $F_T$  para elevar el dron, y también su peso como fuerzas que actúan en el eje z, se obtiene las siguientes expresiones:

$$m \cdot g - F_T = m \cdot a_z \quad (2.61)$$

$$a_z = \frac{m \cdot g - F_T}{m} \quad (2.62)$$

$$\frac{dv_z}{dt} = \dot{v}_z = \frac{m \cdot g - F_T}{m} \quad (2.63)$$

Sabiendo que es necesario que la velocidad en z tienda a un valor de referencia, se define su error como:

$$e_{v_z} = v_z - v_{zref}, \quad (2.64)$$

y su derivada como:

$$\dot{e}_{v_z} = \dot{v}_z - \dot{v}_{zref} \quad (2.65)$$

Se propone una función de Lyapunov  $V(v_z)$  para conseguir controlar la velocidad, misma que asegura que se cumpla la Ecuación (1.18) para asegurar estabilidad.

$$V(v_z) = \frac{1}{2} (e_{v_z})^2 \quad (2.66)$$

La derivada de dicha función candidata es:

$$\dot{V}(v_z) = e_{v_z} \cdot \dot{e}_{v_z} \quad (2.67)$$

Sustituyendo la Ecuación (2.65) en la Ecuación (2.67) se tiene que

$$\dot{V}(v_z) = e_{v_z} (\dot{v}_z - \dot{v}_{zref}) \quad (2.68)$$

Y, sustituyendo la Ecuación (2.63) en la anterior expresión se tiene:

$$\dot{V}(v_z) = e_{v_z} \left( \frac{m \cdot g - F_T}{m} - \dot{v}_{zref} \right) \quad (2.69)$$

Se propone la siguiente ecuación para asegurar que se cumpla el segundo criterio de estabilidad de Lyapunov, ver Ecuación (1.19), donde se saca la ley de control que garantice que  $\dot{V}(v_z)$  sea no positivo. Para ello, es necesario forzar que el segundo término sea negativo, así:

$$-k_{v_z} \cdot e_{v_z}^2 = e_{v_z} \left( \frac{m \cdot g - F_T}{m} - \dot{v}_{zref} \right) \quad (2.70)$$

Entonces,

$$-k_{v_z} \cdot e_{v_z} = \frac{m \cdot g - F_T}{m} - \dot{v}_{zref} \quad (2.71)$$

$$\frac{m \cdot g - F_T}{m} = -k_{v_z} \cdot e_{v_z} + \dot{v}_{zref} \quad (2.72)$$

Así, se obtiene la ley de control para la velocidad en z que resulta de la siguiente manera:

$$F_T = m \cdot g - m(\dot{v}_{zref} - k_{v_z} \cdot e_{v_z}) \quad (2.73)$$

Reemplazando esta ecuación en la Ecuación (2.69), se acota el valor de  $k$ , obteniendo las siguientes expresiones:

$$e_{v_z} \left( \frac{m \cdot g - m \cdot g + m(\dot{v}_{zref} - k_{v_z} \cdot e_{v_z})}{m} - \dot{v}_{zref} \right) \leq 0 \quad (2.74)$$

$$e_{v_z} (\dot{v}_{zref} - k_{v_z} \cdot e_{v_z} - \dot{v}_{zref}) \leq 0 \quad (2.75)$$

$$-k_{v_z} \cdot e_{v_z}^2 \leq 0 \quad (2.76)$$

$$k_{v_z} \geq 0 \quad (2.77)$$

Una vez controlada la velocidad en z, se procede a realizar el control de posición. Esto implica que la ecuación del sistema se vea de la siguiente manera:

$$z = \dot{z} \quad (2.78)$$

En este caso, se aumenta una variable de control  $\mu$  para determinar cual es el valor de referencia de velocidad para la ley de control de velocidad en z.

$$z + \mu = \dot{z} \quad (2.79)$$

El error de posición viene dado por la siguiente ecuación:

$$e_z = z - z_{ref} \quad (2.80)$$

Y, su derivada es:

$$\dot{e}_z = \dot{z} \quad (2.81)$$

Reemplazando la anterior expresión y la Ecuación (2.80) en la Ecuación (2.79), se obtiene lo siguiente:

$$\dot{e}_z = e_z + z_{ref} + \mu \quad (2.82)$$

Se propone la siguiente función de Lyapunov para conseguir controlar la posición, misma que asegura que se cumpla el primer criterio de estabilidad de Lyapunov (Ecuación (1.18)):

$$V(z) = \frac{1}{2} (e_z)^2 \quad (2.83)$$

Donde, la derivada de la función candidata es:

$$\dot{V}(z) = e_z \cdot \dot{e}_z \quad (2.84)$$

Sustituyendo la Ecuación (2.82) en la anterior expresión, se tiene:

$$\dot{V}(z) = e_z(e_z + z_{ref} + \mu) \quad (2.85)$$

$$\dot{V}(z) = e_z^2 + e_z(z_{ref} + \mu) \quad (2.86)$$

Para cumplir con la segunda condición de estabilidad de Lyapunov, se elige la variable de control  $\mu$  tal que se cumpla la Ecuación (1.19), de la siguiente manera:

$$e_z(z_{ref} + \mu) = -k_z \cdot e_z^2 \quad (2.87)$$

Obteniendo la ley de control para la posición en  $z$ .

$$\mu = -k_z \cdot e_z - z_{ref} \quad (2.88)$$

Para acotar el valor de  $k$ , se reemplaza la ley de control en la Ecuación (2.86). Entonces,

$$e_z^2 + e_z(z_{ref} - k_z \cdot e_z - z_{ref}) \leq 0 \quad (2.89)$$

$$e_z^2 - k_z \cdot e_z^2 \leq 0 \quad (2.90)$$

$$k_z \geq 1 \quad (2.91)$$

Entonces, reemplazando la ley de control en la Ecuación (2.79), se tiene la referencia de velocidad del sistema en el eje  $z$ .

$$z - k_z \cdot e_z - z_{ref} = \dot{z} = v_{zref} \quad (2.92)$$

Una vez obtenidas las leyes de control de velocidad, Ecuación (2.73), y posición de la altura, Ecuación (2.88), y acotadas las ganancias de cada controlador, es necesario realizar una calibración. Sin embargo, dado que se trata de dos controladores, tanto de velocidad como posición, es necesario aumentar un saturador de la velocidad de referencia, a la salida del controlador de posición, de la siguiente manera:

$$\text{Límite máximo de } v_{ref} = 30 \text{ [m/s]} \quad (2.93)$$

$$\text{Límite mínimo de } v_{ref} = -30 \text{ [m/s]} \quad (2.94)$$

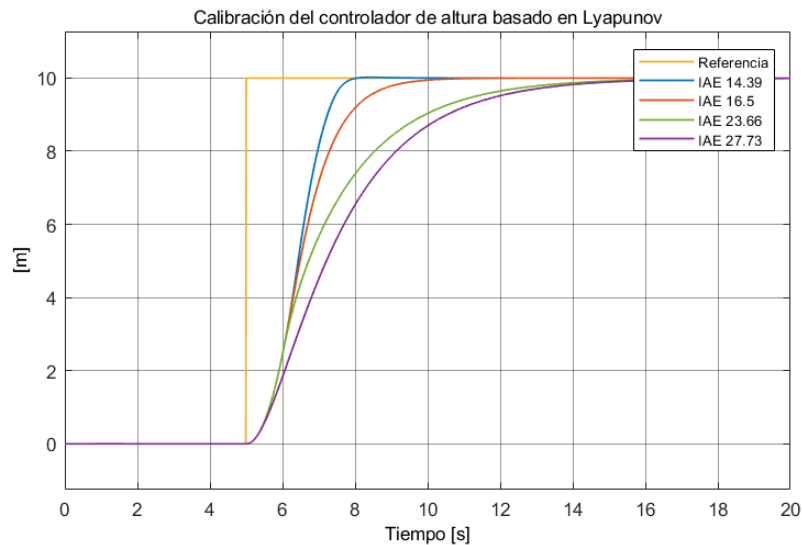
Habiendo desarrollado el control basado en Lyapunov, se tiene las constantes  $k_z$  y  $k_{v_z}$  que se rigen a las desigualdades matemáticas vistas en las Ecuaciones (2.77) y (2.91), lo que implica que no tienen un límite superior. Por ello, es necesario integrar otro saturador a la salida del control de velocidad para limitar el Thrust, como se determinó en el controlador PD de altura en las Ecuaciones (2.34) y (2.35).

Con esto, se puede realizar una calibración conjunta, considerando como referencia del sistema una entrada paso de 10m. Para esto, se utiliza el índice de desempeño IAE y se registran los valores para determinar los más adecuados para este sistema (Tabla 2.3).

**Tabla 2.3.** Valores experimentales para calibración del controlador de z basado en Lyapunov

$k_{vz}/k_z$	1.5	2.375	3.25	4.125	5
0.5	40.23	27.71	25.81	25.24	23.66
1.375	27.73	17.87	16.51	16.07	15.9
2.25	25.87	16.5	15.2	14.94	14.5
3.125	25.27	16.08	14.39	14.94	15.78
4	25	15.88	14.51	15.78	17

En la Figura 2.7, se presentan cuatro respuestas que corresponden a diferentes pares de ganancias de la calibración realizada para el controlador basado en Lyapunov de altura. En la gráfica se identifican las salidas con los valores de IAE de la tabla anterior.



**Figura 2.7.** Diagrama de una muestra de valores de la calibración del controlador basado en Lyapunov de z

Al realizar el análisis de la tabla anterior, se eligió el menor valor de IAE, el cual se encuentra definido por un recuadro naranja, y se determinaron los siguientes valores. Para la ganancia del control de velocidad del eje z,

$$k_{vz} = 3.125 \quad (2.95)$$

y para la ganancia del control de posición de z,

$$k_z = 3.25 \quad (2.96)$$



### 2.1.3.2 Control de x,y basado en Lyapunov

En esta sección se mostrará el proceso de diseño de los controladores basados en Lyapunov, tanto para el eje x como para el eje y, semejante al procedimiento realizado para el control de altura mencionado en la Sección 2.1.3.1.

Según el modelo cinemático del dron visto como una partícula, evaluando el movimiento en los ejes x y y, se obtiene la ecuación que describe el sistema:

$$F_{x,y} = m \cdot a_{x,y} \quad (2.97)$$

Tomando en cuenta que solo se tiene una fuerza de empuje en x o y ( $T_{x,y}$ ), se reemplaza esta variable para obtener las siguientes ecuaciones:

$$T_{x,y} = m \cdot a_{x,y} \quad (2.98)$$

$$\frac{dv_{x,y}}{dt} = \frac{T_{x,y}}{m} \quad (2.99)$$

Para encontrar la función candidata de Lyapunov, primero se define el error de velocidad en x,y como:

$$e_{v_{x,y}} = v_{x,y} - v_{x,yref}, \quad (2.100)$$

y su respectiva derivada,

$$\dot{e}_{v_{x,y}} = \dot{v}_{x,y} - \dot{v}_{x,yref} \quad (2.101)$$

Se propone la siguiente función candidata de Lyapunov para controlar la velocidad en x,y, misma que asegura que se cumpla la Ecuación (1.18):

$$V(v_{x,y}) = \frac{1}{2} (e_{v_{x,y}})^2 \quad (2.102)$$

La derivada de la función candidata quedaría como:

$$\dot{V}(v_{x,y}) = e_{v_{x,y}} \cdot \dot{e}_{v_{x,y}} \quad (2.103)$$

Sustituyendo la Ecuación (2.101) de la derivada del error de velocidad en x,y, se obtiene la siguiente función:

$$\dot{V}(v_{x,y}) = e_{v_{x,y}} (\dot{v}_{x,y} - \dot{v}_{x,yref}) \quad (2.104)$$

Y reemplazando la Ecuación (2.99 de la derivada de la velocidad en la Ecuación (2.104), se obtiene:

$$\dot{V}(v_{x,y}) = e_{v_{x,y}} \left( \frac{T_{x,y}}{m} - \dot{v}_{x,yref} \right) \quad (2.105)$$

Con ello, se propone la siguiente ecuación para asegurar que se cumpla el segundo criterio de estabilidad de Lyapunov (Ecuación (1.19)):

$$-k_{v_{x,y}} \cdot e_{v_{x,y}}^2 = e_{v_{x,y}} \left( \frac{T_{x,y}}{m} - \dot{v}_{x,yref} \right) \quad (2.106)$$

Entonces, se obtiene la ley de control para la velocidad en x,y de la siguiente manera:

$$T_{x,y} = m \left( -k_{v_{x,y}} \cdot e_{v_{x,y}} + \dot{v}_{x,yref} \right) \quad (2.107)$$

Para acotar el valor de  $k$ , se reemplaza la ley de control en la Ecuación (2.105), así:

$$e_{v_{x,y}} \left( -k_{v_{x,y}} \cdot e_{v_{x,y}} + \dot{v}_{x,yref} - \dot{v}_{x,yref} \right) \leq 0 \quad (2.108)$$

$$e_{v_{x,y}} \left( -k_{v_{x,y}} \cdot e_{v_{x,y}} \right) \leq 0 \quad (2.109)$$

$$-k_{v_{x,y}} \cdot e_{v_{x,y}}^2 \leq 0 \quad (2.110)$$

$$k_{v_{x,y}} \geq 0 \quad (2.111)$$

Una vez controlada la velocidad de x,y, se procede a realizar el diseño del control de posición. Para ello, se utilizará la variable  $a$  en reemplazo de  $x$  o  $y$ .

La ecuación de cada sistema se vea de la siguiente manera:

$$a = \dot{a} \quad (2.112)$$

Para determinar el valor de referencia de velocidad de x,y, se aumenta una variable de control  $\mu$ .

$$a + \mu = \dot{a} \quad (2.113)$$

El error de posición viene dado por la siguiente ecuación:

$$e_a = a - a_{ref} \quad (2.114)$$

Y, su derivada es:

$$\dot{e}_a = \dot{a} \quad (2.115)$$

Reemplazando la Ecuación (2.114) y la Ecuación (2.115) en la Ecuación (2.113), se obtiene lo siguiente:

$$\dot{e}_a = e_a + a_{ref} + \mu \quad (2.116)$$

Se propone la siguiente función de Lyapunov para conseguir controlar la posición, misma que asegura que se cumpla el primer criterio de estabilidad de Lyapunov (Ecuación (2.18)):

$$V(a) = \frac{1}{2}(e_a)^2 \quad (2.117)$$

Donde, la derivada de la función candidata es:

$$\dot{V}(a) = e_a \cdot \dot{e}_a \quad (2.118)$$

Sustituyendo la Ecuación (2.116) en la anterior expresión, se tiene:

$$\dot{V}(a) = e_a(e_a + a_{ref} + \mu) \quad (2.119)$$

$$\dot{V}(a) = e_a^2 + e_a(a_{ref} + \mu) \quad (2.120)$$

Para cumplir con la segunda condición de estabilidad de Lyapunov, se elige la variable de control  $\mu$  tal que se cumpla la Ecuación (1.19), de la siguiente manera:

$$e_a(a_{ref} + \mu) = -k_a \cdot e_a^2 \quad (2.121)$$

Obteniendo la ley de control para la posición en  $x, y$ .

$$\mu = -k_a \cdot e_a - a_{ref} \quad (2.122)$$

Acotando el valor de  $k$ , se reemplaza la ley de control en la Ecuación (2.120). Entonces, se tiene:

$$e_a^2 + e_a(a_{ref} - k_a \cdot e_a - a_{ref}) \leq 0 \quad (2.123)$$

$$e_a^2 - k_a \cdot e_a^2 \leq 0 \quad (2.124)$$

$$k_a \geq 1 \quad (2.125)$$

Entonces, reemplazando la ley de control en la Ecuación (2.113), se tiene la referencia de velocidad del sistema en el eje  $x$  y en el eje  $y$ .

$$x - k_x \cdot e_x - x_{ref} = \dot{x} = v_{xref} \quad (2.126)$$

$$y - k_y \cdot e_y - y_{ref} = \dot{y} = v_{yref} \quad (2.127)$$

Habiendo obtenido las leyes de control para velocidad en la Ecuación (2.107) y para posición en la Ecuación (2.122), de forma similar al control de altura basado en Lyapunov, se vuelve necesario delimitar las salidas de cada controlador. En el caso de la salida del controlador de posición, se coloca el mismo saturador visto en la Ecuación (2.93) y la Ecuación (2.94). En la salida del control de velocidad, para el control de  $x$ , se incluye un saturador del ángulo de pitch igual al visto en la Ecuación (2.39) y la Ecuación (2.40); y para el control de  $y$ , se utiliza el saturador del ángulo de roll mostrado en la Ecuación (2.63) y la Ecuación (2.64).

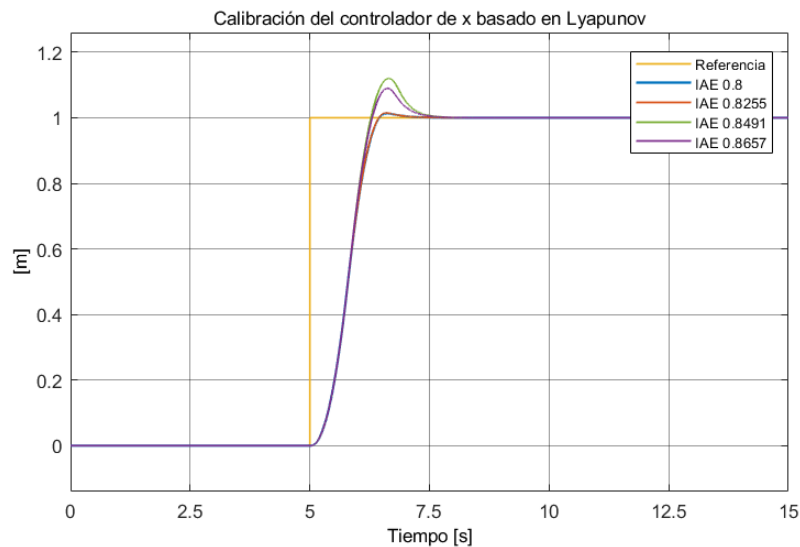
Una vez colocados los saturadores, se realiza la calibración de los controladores con una entrada paso de 1m, y se ubican los valores del índice de desempeño IAE en una tabla

para los pares de ganancias tanto del control de velocidad, como el de posición, basados en Lyapunov. En la Tabla 2.4, se pueden visualizar los valores para el control del eje x.

**Tabla 2.4.** Valores experimentales para calibración del controlador de x basado en Lyapunov

$k_{vx}/k_x$	5.5	6	6.5	7	7.5
4	0.824	0.8052	0.8	0.8055	0.8131
4.5	0.8004	0.8033	0.8134	0.8255	0.8382
5	0.8033	0.8164	0.8323	0.8491	0.8657
5.5	0.8134	0.8323	0.8528	0.8733	0.8932
6	0.8255	0.8491	0.8733	0.897	0.9196

En la Figura 2.8, se presentan cuatro respuestas que corresponden a diferentes pares de ganancias de la calibración realizada para el controlador basado en Lyapunov del eje x. En la gráfica se identifican las salidas con los valores de IAE de la tabla anterior.



**Figura 2.8.** Diagrama de una muestra de valores de la calibración del controlador basado en Lyapunov de x

Al realizar el análisis de la tabla, se eligió el menor valor de IAE, el cual se encuentra definido por un recuadro naranja, y el valor obtenido de ganancia de velocidad en x fue de:

$$k_{vx} = 4, \quad (2.128)$$

y el de posición,

$$k_x = 6.5 \quad (2.129)$$

Al igual que en el controlador PD de x, la acción de control debe ser inversa, debido al sentido de giro del ángulo de pitch. Por lo cual, se aumenta un bloque de Adaptación del

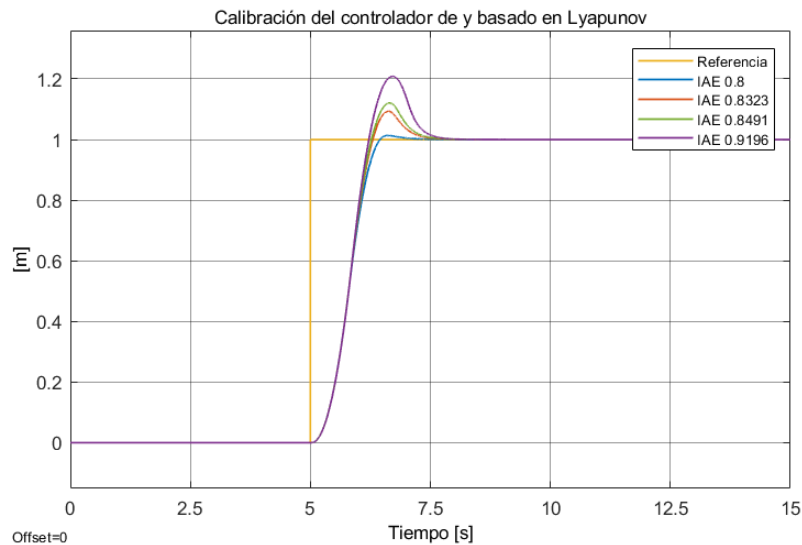
Body Frame, véase Figura 2.2, el cual no afecta las ganancias obtenidas, sino la acción resultante del control de  $x$ .

Así mismo, en la Tabla 2.5 se pueden visualizar los valores para el control del eje  $y$ .

**Tabla 2.5.** Valores experimentales para calibración del controlador de  $y$  basado en Lyapunov

$k_{vy}/k_y$	5.5	6	6.5	7	7.5
4	0.824	0.8052	0.8	0.8055	0.8131
4.5	0.8004	0.8033	0.8134	0.8255	0.8382
5	0.8033	0.8164	0.8323	0.8491	0.8657
5.5	0.8134	0.8323	0.8528	0.8733	0.8932
6	0.8255	0.8491	0.8733	0.897	0.9196

En la Figura 2.9, se presentan cuatro respuestas que corresponden a diferentes pares de ganancias de la calibración realizada para el controlador basado en Lyapunov del eje  $y$ . En la gráfica se identifican las salidas con los valores de IAE de la tabla anterior.



**Figura 2.9.** Diagrama de una muestra de valores de la calibración del controlador basado en Lyapunov de  $y$

Al realizar el análisis de la tabla, se puede notar que los valores obtenidos para el eje  $y$  son los mismos que se obtuvieron en el eje  $x$ , se eligió el menor valor de IAE, el cual se encuentra definido por un recuadro naranja; así, el valor obtenido de ganancia de velocidad en  $y$  fue de:

$$k_{vy} = 4, \tag{2.130}$$

y el de posición,

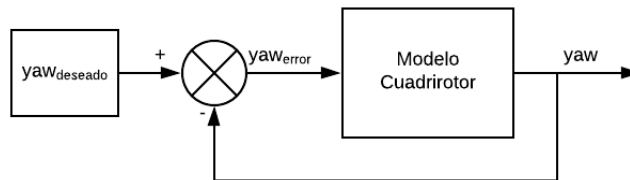
$$k_y = 6.5 \quad (2.131)$$

Esto, debido al desarrollo matemático en el diseño de los controladores, tomando en cuenta que al dron se lo observa como una partícula, como una masa uniforme.

### 2.1.4 INCORPORACIÓN DEL ÁNGULO YAW

En un inicio, se asume que el sistema de coordenadas del entorno se encuentra alineado con el del dron, es decir,  $yaw = 0$ . Sin embargo, si se realiza un cambio en este ángulo, se debe tomar en consideración los marcos de referencia del dron y del marco terrestre. Para ello, es necesario el control del ángulo yaw.

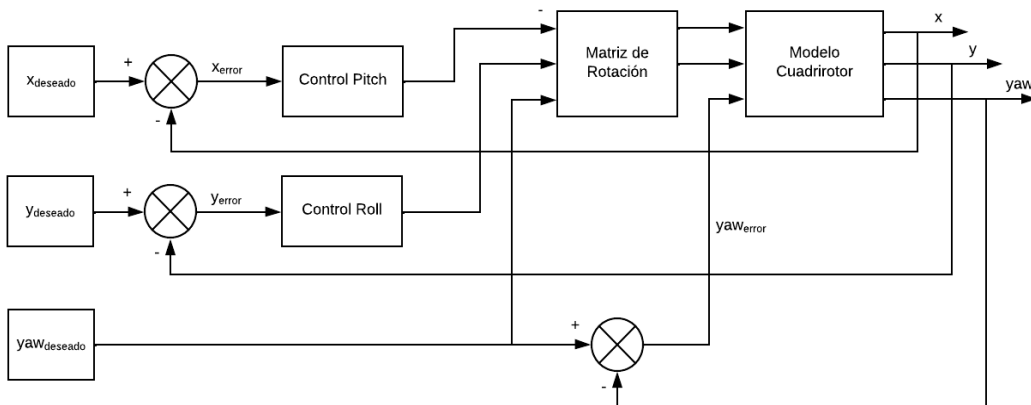
Debido al control interno incluido en el modelo del UAV usado (Figura 1.14), simplemente se estableció un control en lazo cerrado del ángulo yaw, como se puede ver en la Figura 2.10.



**Figura 2.10.** Diagrama de bloques de yaw

Una vez definido el control del ángulo de yaw, se requiere realizar una transformación de coordenadas para la incorporación de este ángulo con el eje x y el eje y del marco terrestre. Esto se realiza mediante una matriz de rotación, como la vista en las Ecuaciones (1.1) y (1.2); sin embargo, en esta ocasión se toman solo los ejes x,y, como se puede ver en la Figura 2.11. Por lo cual, la matriz de rotación en dos dimensiones sería la siguiente:

$$R = \begin{bmatrix} \cos \psi & \sin \psi \\ -\sin \psi & \cos \psi \end{bmatrix} \quad (2.132)$$



**Figura 2.11.** Diagrama de bloques de la incorporación de yaw al control de x y y

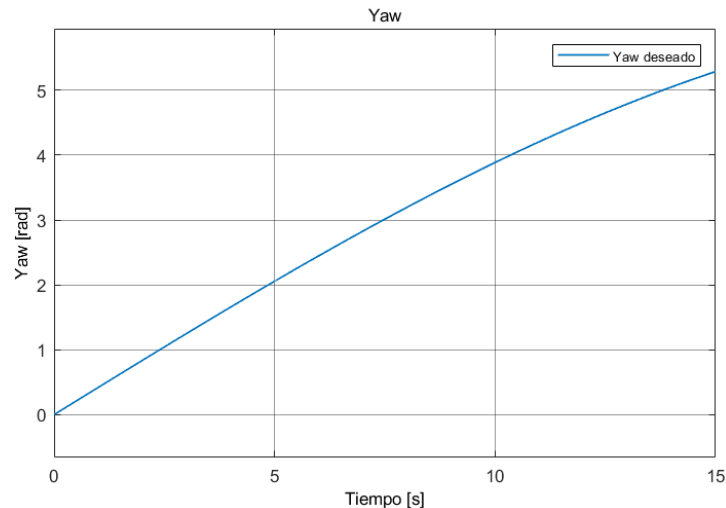
Entonces, la ecuación queda de la siguiente forma:

$$\begin{bmatrix} \theta_{d\_yaw} \\ \phi_{d\_yaw} \end{bmatrix} = R \begin{bmatrix} \theta_c \\ \phi_c \end{bmatrix} \quad (2.133)$$

$$\begin{bmatrix} \theta_{d\_yaw} \\ \phi_{d\_yaw} \end{bmatrix} = \begin{bmatrix} \cos \psi_d & \sin \psi_d \\ -\sin \psi_d & \cos \psi_d \end{bmatrix} \begin{bmatrix} \theta_c \\ \phi_c \end{bmatrix} \quad (2.134)$$

Donde,  $\psi_d$  es el ángulo de yaw deseado,  $\theta_c$  es el valor que sale del control en x,  $\phi_c$  es el valor que se obtiene del control en y,  $\theta_{d\_yaw}$  es el valor deseado de pitch y  $\phi_{d\_yaw}$  es el valor deseado de roll que ingresan al modelo del cuadrirrotor, tomando en consideración el ángulo de heading.

Una vez implementada la matriz de rotación, se verifica que los controladores x,y funcionen correctamente al cambiar el ángulo de yaw, por lo cual se mostrará el resultado obtenido para el control tipo PID y control basado en Lyapunov con un ángulo de yaw variable que se puede ver en la Figura 2.12.



**Figura 2.12.** Variación del ángulo de yaw

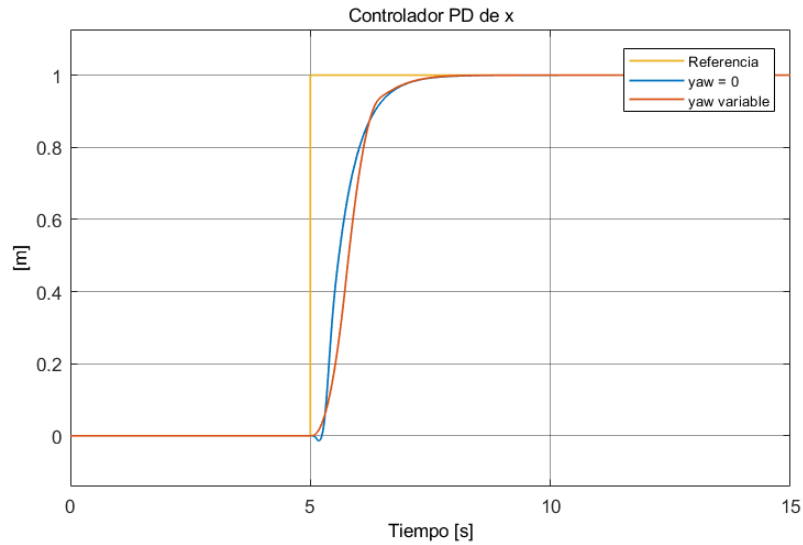
En las siguientes pruebas se verifica el correcto funcionamiento de la incorporación de la matriz rotacional para el manejo del ángulo de yaw en cada uno de los controladores sintonizados.

#### 2.1.4.1 Control tipo PID con incorporación del ángulo yaw

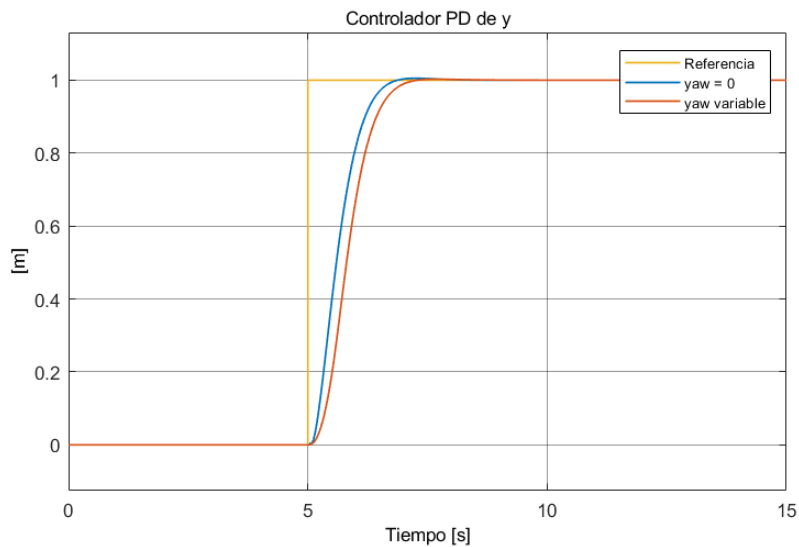
Según los parámetros obtenidos para el control PD del eje x vistos en las Ecuaciones (2.37) y (2.38), y del eje y vistos en las Ecuaciones (2.42) y (2.43), se realiza una comparación entre los resultados obtenidos con y sin la matriz rotacional de la Ecuación (2.134).

A continuación, se presentan los resultados obtenidos del control PD de x en la Figura 2.13 y del control PD de y en la Figura 2.14, tomando en cuenta que, para la variación del ángulo

yaw, es necesario incorporar los saturadores mencionados en las Ecuaciones (2.39) y (2.40) para el ángulo de roll, y las Ecuaciones (2.44) y (2.45) para el ángulo de pitch.



**Figura 2.13.** Comparación de respuestas del controlador PD de x



**Figura 2.14.** Comparación de respuestas del controlador PD de y

Se determinó el valor del error usando el índice de desempeño IAE, con la finalidad de comparar los valores obtenidos una vez incorporada la matriz rotacional y la variación de yaw (Tabla 2.6).

**Tabla 2.6.** Valores de IAE del controlador tipo PID

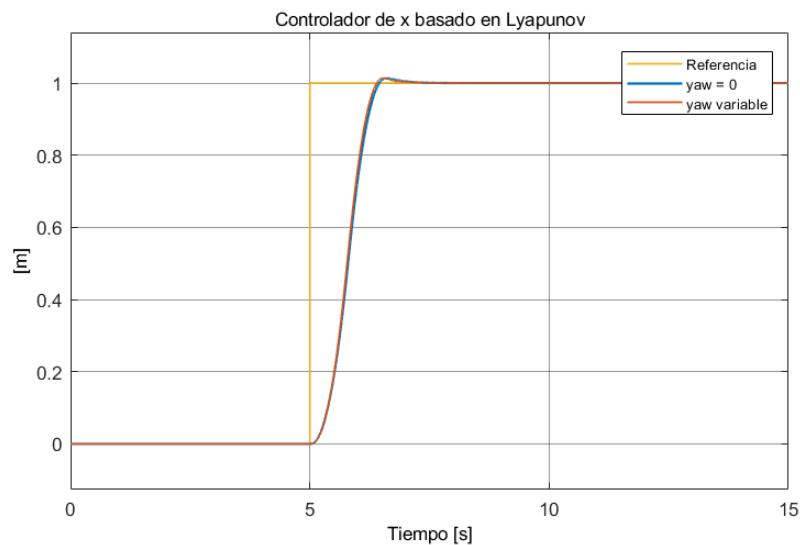
IAE	X	Y
<b>Yaw = 0</b>	0.7396	0.6673
<b>Yaw variable</b>	0.8473	0.8663



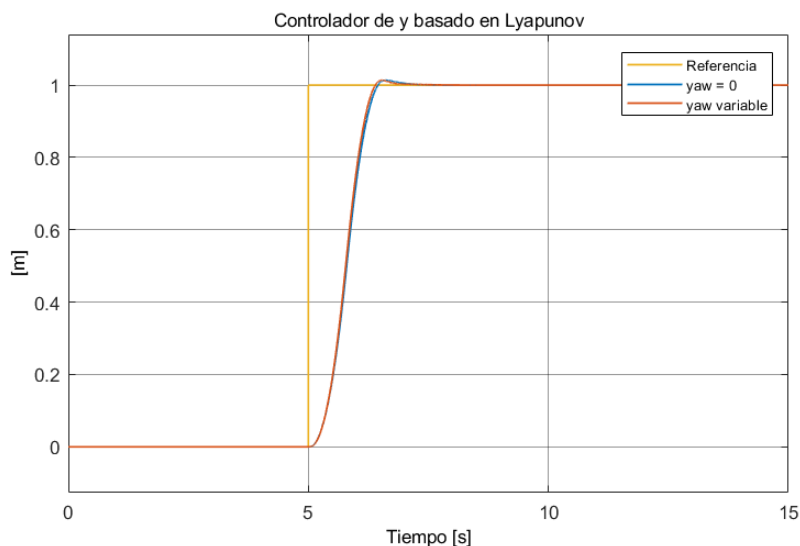
### 2.1.4.2 Control Basado en Lyapunov con incorporación del ángulo yaw

Según los parámetros obtenidos para el control basado en Lyapunov del eje x vistos en las Ecuaciones (2.128) y (2.129), y del eje y vistos en las Ecuaciones (2.130) y (2.131), se realiza una comparación entre los resultados obtenidos con y sin la matriz rotacional de la Ecuación (2.134).

A continuación, se presentan los resultados obtenidos del control basado en Lyapunov de x en la Figura 2.15 y del control basado en Lyapunov de y en la Figura 2.16.



**Figura 2.15.** Comparación de respuestas del controlador basado en Lyapunov de x



**Figura 2.16.** Comparación de respuestas del controlador basado en Lyapunov de y

Se determinó el valor del error usando el índice de desempeño IAE, con la finalidad de comparar los valores obtenidos una vez incorporada la matriz rotacional y la variación de yaw (Tabla 2.7).

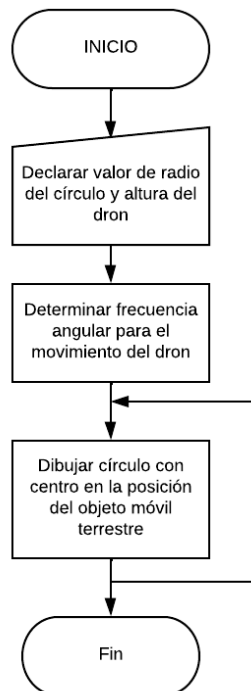
**Tabla 2.7.** Valores de IAE del controlador basado en Lyapunov

IAE	X	Y
Yaw = 0	0.8	0.8
Yaw variable	0.7796	0.7791

### 2.1.5 CIRCUNNAVEGACIÓN

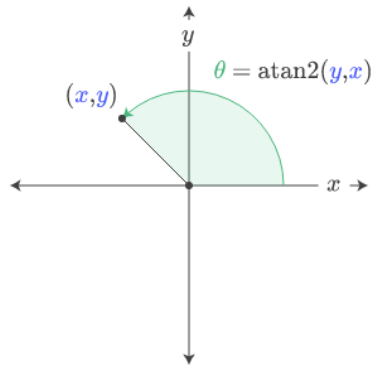
Una vez desarrollados los controladores para el movimiento y dirección del cuadricóptero, se procede a realizar un algoritmo de control para que el dron pueda circunnavegar alrededor de un objeto móvil terrestre. Dicho algoritmo debe tomar en consideración tres funciones importantes: la primera, realizar movimientos circulares continuos; la segunda, hacer el seguimiento a un objeto móvil; y la tercera, que el frente del dron apunte al objeto en seguimiento.

Partiendo de esto, se siguió la lógica de programación explicada en el diagrama de flujo de la Figura 2.17 para el algoritmo de control para realizar circunnavegación.



**Figura 2.17.** Diagrama de flujo de algoritmo de control para circunnavegación

Resueltas las dos primeras consideraciones antes mencionadas, se debe conseguir mover el ángulo de heading del cuadricóptero para que mantenga su frente hacia el objeto móvil terrestre. Para esto, se utiliza la función *atan2* para determinar el ángulo en radianes que traza el dron, siendo el objeto móvil el centro de las coordenadas, tal como se muestra en la Figura 2.18.



**Figura 2.18.** Función arco tangente 2 [38]

Al tener un marco de referencia diferente al marco terrestre, es necesario restar la posición del objeto móvil de la posición del UAV e implementar estos valores en la función *atan2* de la siguiente manera:

$$angulo = atan2(y_{uav} - y_{obj}, x_{uav} - x_{obj}) \quad (2.135)$$

Una vez obtenido el ángulo de posición, es necesario sumar  $\pi$  radianes puesto que, con esta función, el cuadricóptero tendría su frente en dirección contraria a lo establecido.

Ahora, la función *atan2* genera un ángulo entre 0 y  $2\pi$  radianes, sin embargo, el cuadricóptero estará moviéndose en círculos continuos, lo que implica que habrá un corte en cada cambio de  $2\pi$  radianes a 0. Para resolver este problema, se añadió el bloque Unwrap en la simulación, mismo que permite realizar un desenvolvimiento de fase, lo que significa que se añaden múltiplos de  $2\pi$  radianes a cada entrada de fase, y así eliminar la discontinuidad [39].

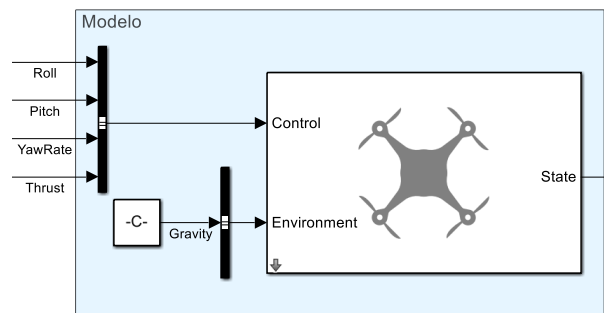
## 2.2 IMPLEMENTACIÓN DEL ESCENARIO DE SIMULACIÓN

Para el presente trabajo, como se indicó anteriormente, se utilizó Simulink de MATLAB y su herramienta UAV Toolbox. En este apartado se podrán visualizar los elementos usados en la simulación, y además se presentan algunos parámetros seleccionados en los bloques para realizar este trabajo de titulación. Aquí se podrá conocer cómo se deben realizar las conexiones de los bloques del UAV Toolbox, su visualización e implementación final.

### 2.2.1 MODELO DEL UAV

Como se mencionó anteriormente, el modelo a usar en la simulación viene dado por el bloque UAV Guidance Model, el cual tiene como entrada de control un bus de cuatro elementos, mismos que son definidos como la actitud del UAV; y en su entrada de entorno, un bus de un elemento, que sería el valor de gravedad  $9.8066 \text{ [m/s}^2\text{]}$ . Esto se puede

visualizar en la Figura 2.19, donde para ambos casos es necesario el uso de un creador de buses, ya que es lo que requiere el bloque mencionado.

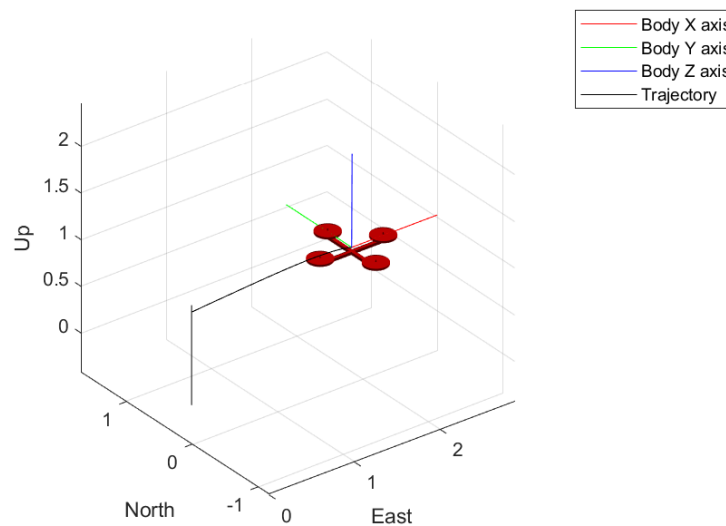


**Figura 2.19.** Bloques que conforman el modelo de la simulación

A la salida del bloque UAV Guidance Model, se tiene el bus con el estado del modelo del cual se toman tres elementos para el trabajo realizado, que son WorldPosition, EulerZYX y Thrust.

### 2.2.2 VISUALIZACIÓN DEL UAV

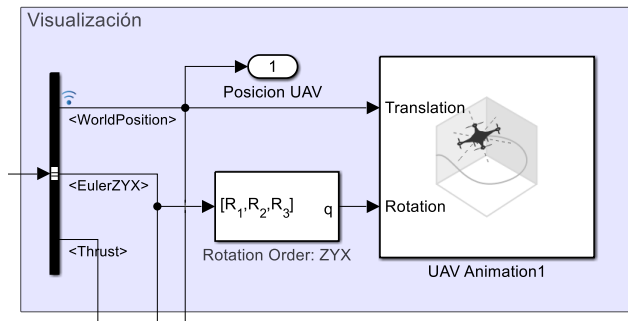
Para la visualización del UAV, se usa otro bloque del UAV Toolbox llamado UAV Animation. Este bloque muestra el camino que ha seguido el dron en la simulación en una animación 3D, como se puede ver en el ejemplo de la Figura 2.20.



**Figura 2.20.** UAV Animation

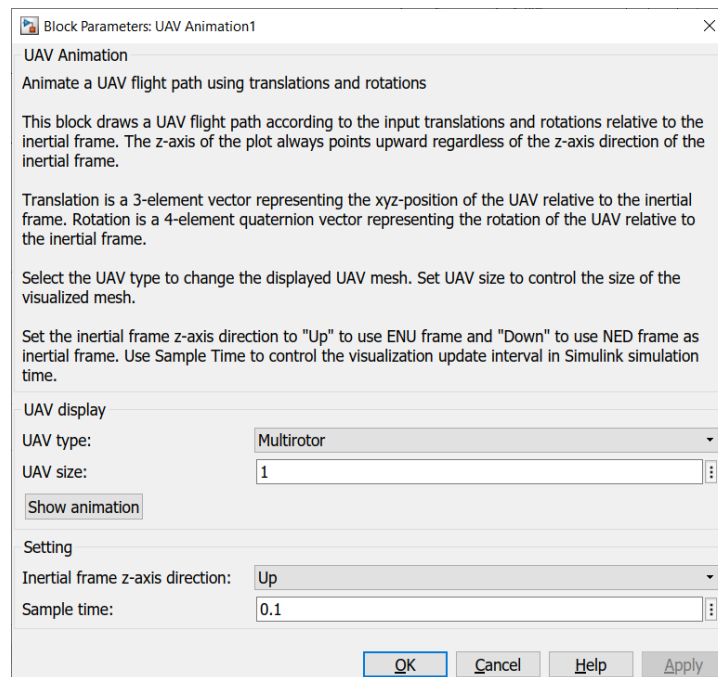
Tiene dos entradas, la primera es un vector de traslación, donde se especifica la posición del UAV; y la segunda es un vector cuaternión de rotación relativo a la referencia inercial, es decir, al marco terrestre. Del bus de salida del UAV Guidance Model se toman los vectores WorldPosition, para la entrada de traslación; y Euler ZYX, para la entrada de

rotación, usando un bloque intermedio de transformación de vector de rotación de orden ZYX a cuaternión (Figura 2.21).



**Figura 2.21.** Bloques para visualización del UAV

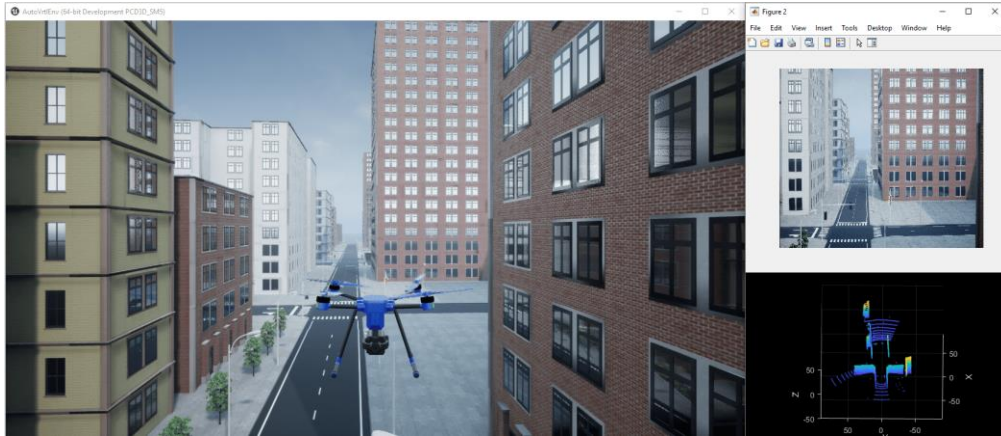
En los parámetros del bloque UAV Animation, se seleccionó el tipo de UAV a mostrar, en este caso un multirroto; y se cambió la dirección de la referencia del cuadricóptero para que el eje z sea positivo hacia arriba (Figura 2.22), con la finalidad de que la visualización se asemeje a lo propuesto en la Figura 1.7 y se relacione con los controladores diseñados.



**Figura 2.22.** Parámetros de bloque UAV Animation

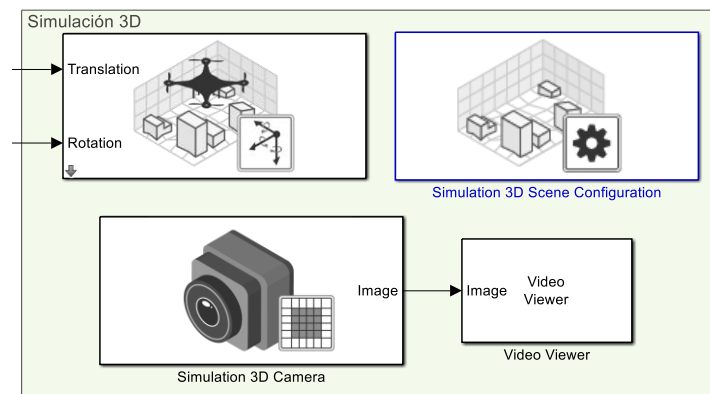
### 2.2.3 SIMULACIÓN 3D

Además del bloque de animación del UAV Toolbox, se tienen otros bloques que permiten realizar una simulación 3D que permite una visualización más dinámica del UAV en un entorno 3D modificable. Un ejemplo de lo mencionado se puede ver en la Figura 2.23, donde se muestran diferentes opciones de visualización.



**Figura 2.23.** Visualización del entorno 3D generado por los bloques de Simulation 3D [40] Estas herramientas son útiles para verificar el comportamiento del dron de manera interactiva, sin embargo, requieren de un procesamiento alto del computador y se sugiere revisar primero los requerimientos de dichos bloques para saber si pueden ser utilizados en futuros trabajos. Para este trabajo se implementaron algunos de estos bloques como un añadido, con la finalidad de mostrar a breves rasgos parte importante de los usos del UAV Toolbox. Estos bloques son Simulation 3D Scene Configuration, Simulation 3D UAV Vehicle y Simulation 3D Camera.

El bloque principal es el primero mencionado ya que configura el entorno que se presenta en la simulación 3D, como son la escena y los cambios climáticos que se podrían presentar en la vida real; y une los bloques de sensores y visualización. Este bloque además permite seleccionar la vista de la simulación 3D, que puede ser la vista en el origen de la escena o la vista desde el dron.



**Figura 2.24.** Bloques para la simulación 3D

El bloque Simulation 3D UAV Vehicle incluye un UAV en el entorno visualización 3D. Este bloque requiere dos entradas, la primera es un vector de traslación, tal como en el bloque UAV Animation; y la segunda es un vector de rotación, donde se puede unir el vector de

salida EulerZYX del bloque UAV Guidance Model de manera directa. Dentro de los parámetros de este bloque, es necesario definir el tipo de UAV a visualizar, en este caso, Quadrotor; y se debe incluir un nombre específico para poder identificarlo con los sensores que se quieran implementar y, de ser el caso, la vista que se quiera presentar.

El bloque Simulation 3D Camera proporciona la imagen de una cámara para visualizar el entorno 3D y se podría usar para ejemplificar las cámaras que vienen adaptadas a los UAVs comerciales. Dentro de sus parámetros se puede elegir el vehículo del que será parte la cámara o si será parte de la escena, su posición y rotación relativa, así como los píxeles y distorsión de la imagen que se requiera. Este bloque debe ir acompañado de un bloque de video (Video Viewer) del Computer Vision Toolbox, el cual permite realizar transmisiones de video o mostrar imágenes.

Todos los bloques mencionados de la simulación 3D se pueden ver en la Figura 2.24. En este caso, se elige la vista en el origen de la escena en el bloque Simulation 3D Scene Configuration y la cámara con el visualizador de video montada en el dron, como una alternativa para tener dos perspectivas distintas y poder tener una visualización más amplia. Esto además ayudaría a ver el objeto móvil terrestre al momento de realizar la circunnavegación propuesta en este trabajo.

#### **2.2.4 HMI**

Para unir todos los procesos de la simulación, facilitar el manejo de la misma y tener una visualización más clara de las respuestas obtenidas, se realiza una interfaz gráfica con ayuda de la herramienta interactiva de MATLAB, App Designer, la cual es un entorno para diseñar una aplicación y programar su comportamiento, que ofrece un amplio conjunto de componentes de interfaz de usuario interactivos.

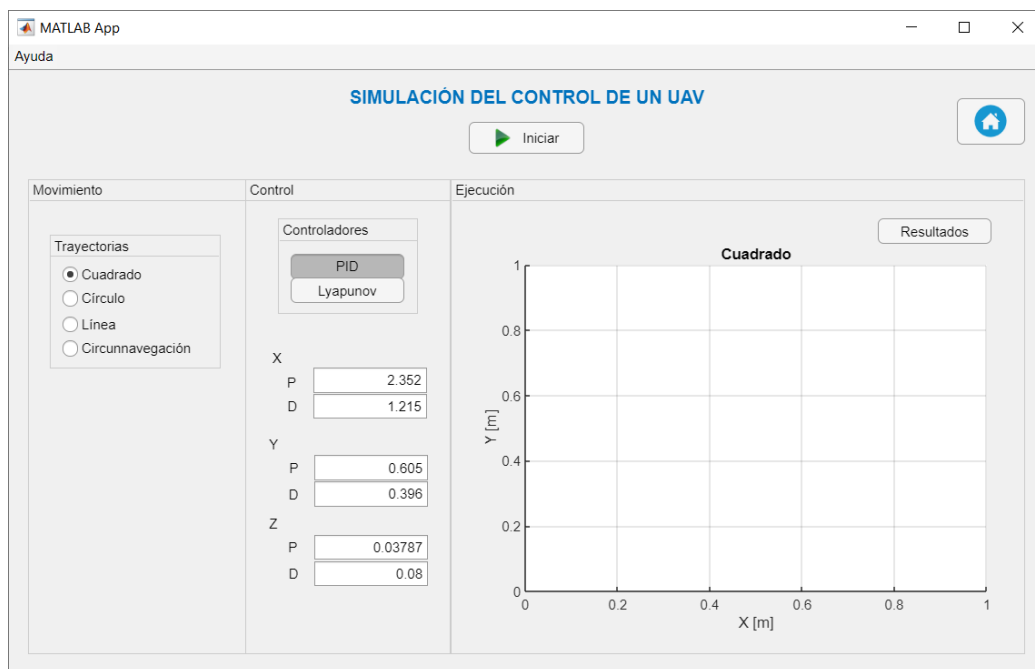
La HMI desarrollada es de tres niveles, con una determinada funcionalidad de cada elemento que se explica a detalle en el ANEXO B.

En el primer nivel se presenta la portada del trabajo donde se identifica la información principal como se muestra en la Figura 2.25. En esta pantalla se tienen dos botones, uno para el comando de salida o cierre de la pantalla, y otro para el paso a la pantalla principal.



**Figura 2.25.** Portada de la interfaz gráfica

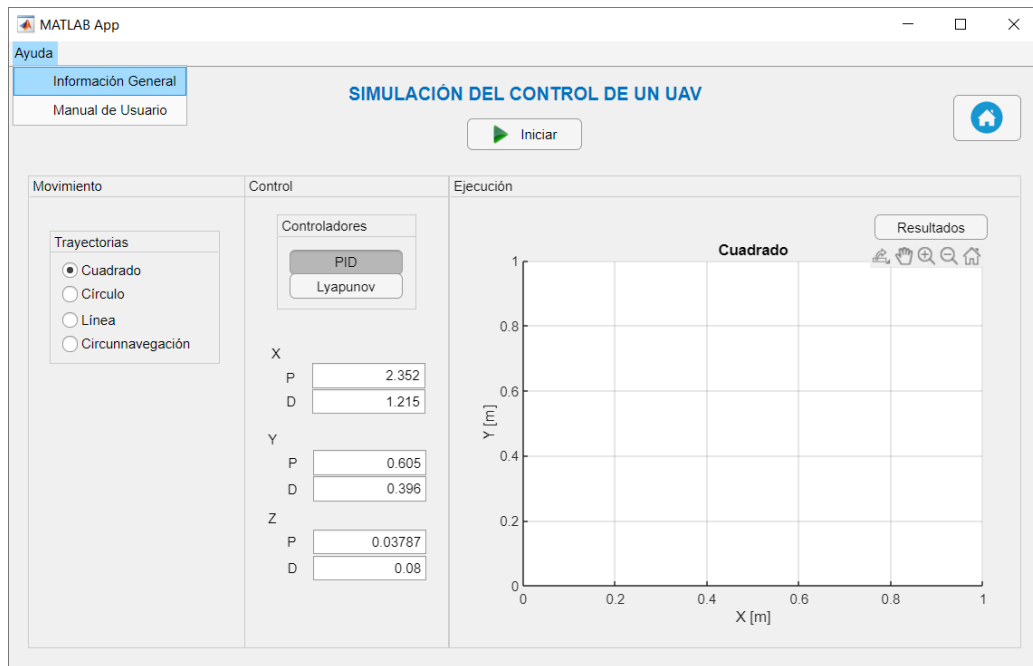
En el segundo nivel se tiene una pantalla de control, ver Figura 2.26, la cual se divide en tres secciones. En la primera sección, se tiene la posibilidad de seleccionar el tipo de movimiento que tendrá el UAV, siendo este un movimiento siguiendo una trayectoria específica o haciendo circunnavegación a un objeto móvil terrestre. En la segunda sección, se elige el tipo de controlador a usar en la ejecución del movimiento elegido, y se puede modificar las constantes de cada controlador. Finalmente, en la tercera sección se encuentran los resultados obtenidos en la simulación.



**Figura 2.26.** Pantalla principal de control del UAV

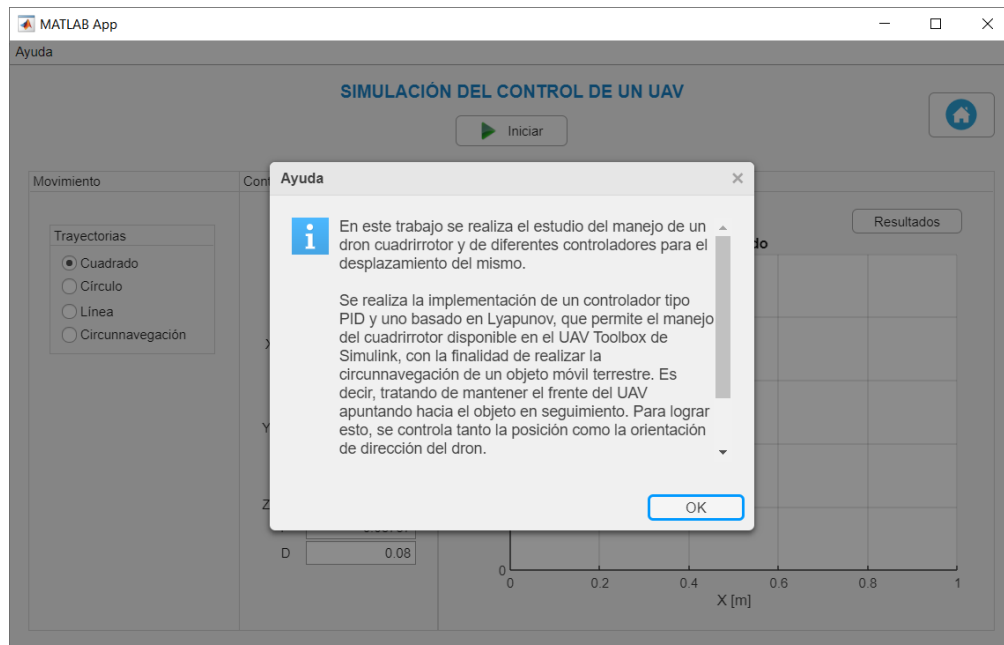


El tercer nivel corresponde a las ventanas de ayuda al usuario, que se seleccionan como se puede ver en la Figura 2.27.



**Figura 2.27.** Selección de Ayuda al usuario

En estas opciones se tendrá información relevante para manejar la interfaz realizada. La ventana que se abre al presionar la opción “Información General” se puede ver en la Figura 2.28, y la opción “Manual de Usuario” presenta el documento con la información detallada en el ANEXO B.



**Figura 2.28.** Ventana emergente de Información General

### 2.2.4.1 Diagrama de flujo de la interfaz

En la Figura 2.29 y la Figura 2.30 se indican los diagramas de flujo de la interfaz gráfica mostrada anteriormente. La descripción a detalle de cada componente se encuentra en el ANEXO B.

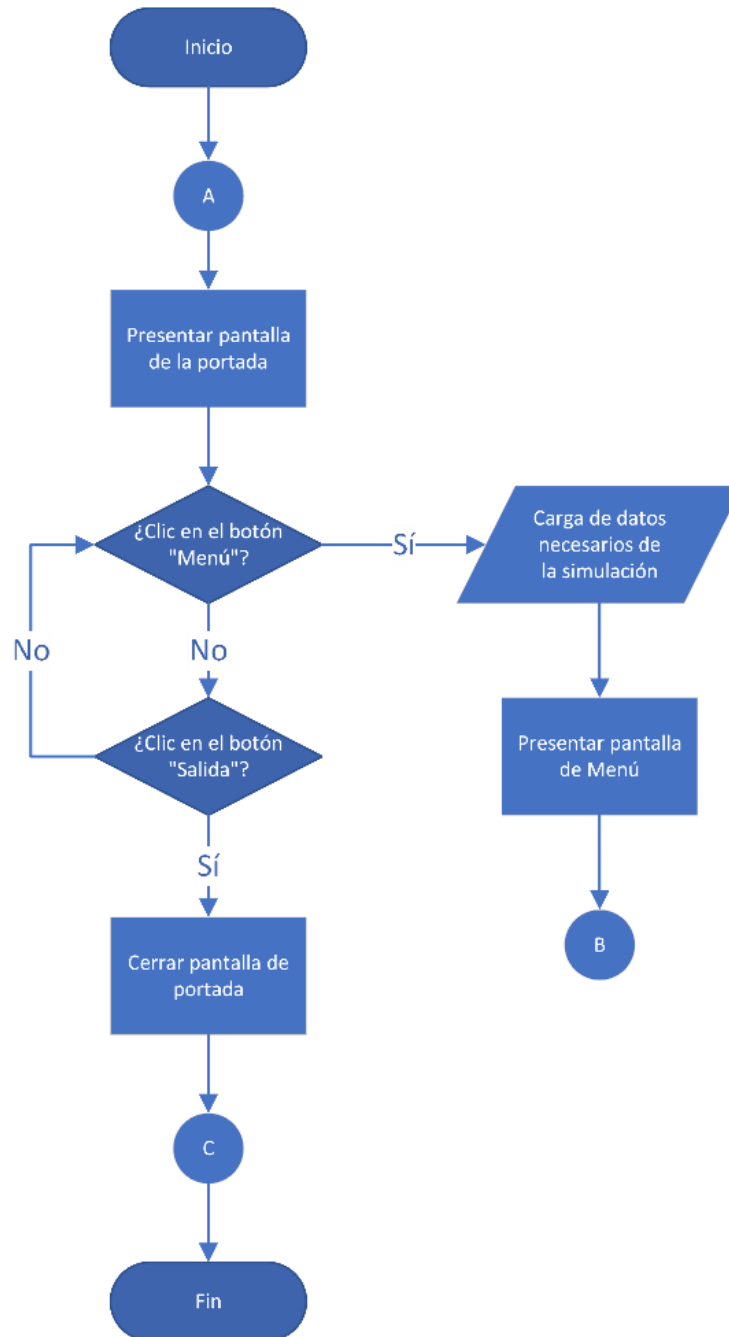
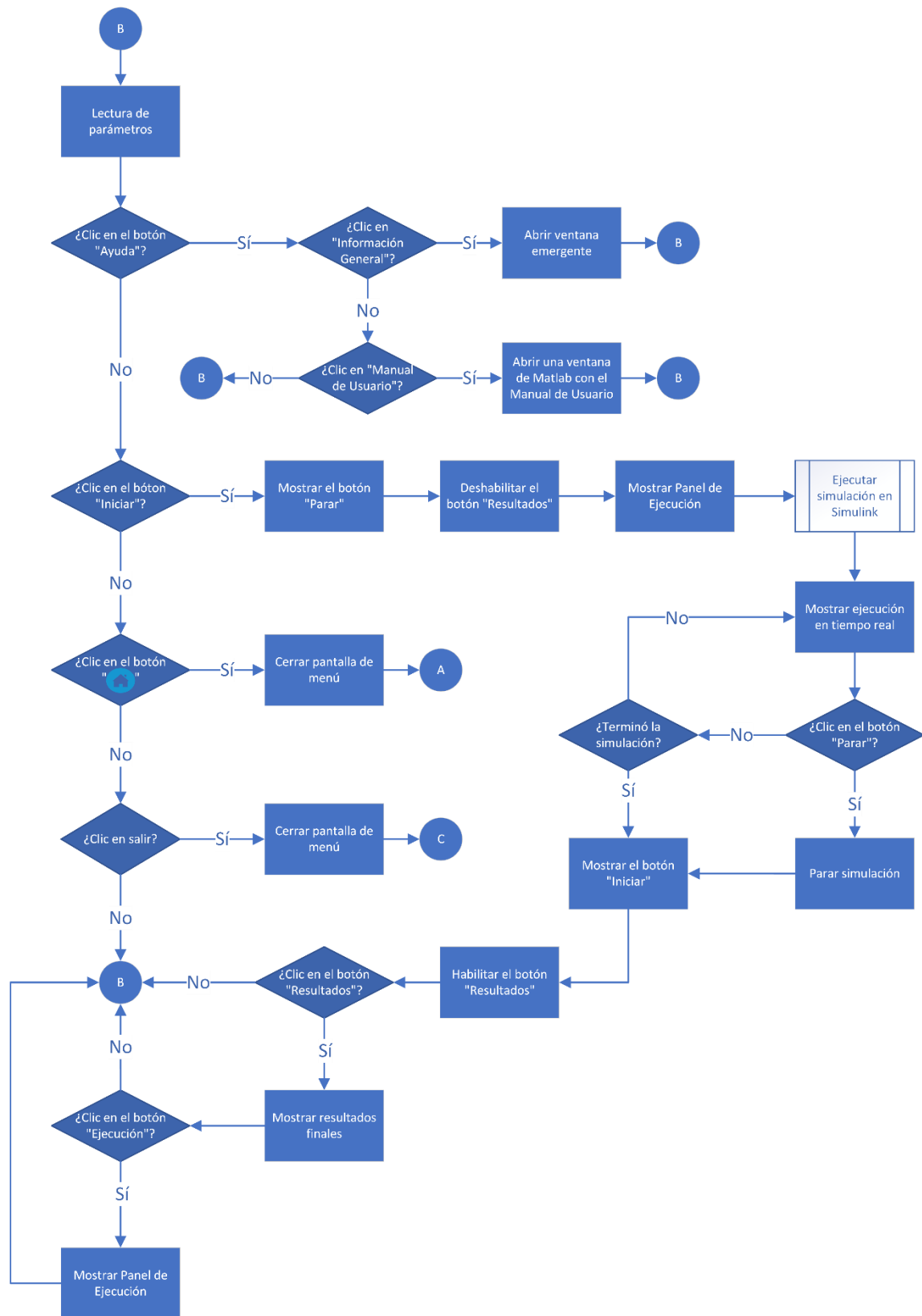


Figura 2.29. Diagrama de flujo de la HMI



**Figura 2.30.** Diagrama de flujo del nivel 2 y 3 de la HMI

### **3. RESULTADOS Y DISCUSIÓN**

En este capítulo se presentan los resultados obtenidos al poner a prueba los esquemas de control propuestos, tanto el controlador tipo PID y el controlador basado en Lyapunov, para el manejo de un cuadricóptero simulado. Las pruebas planteadas permiten comparar, mediante el análisis de los resultados obtenidos durante el seguimiento de trayectorias seleccionadas, el funcionamiento del UAV Toolbox de Simulink con los diferentes controladores mencionados.

Se realizan dos diferentes tipos de pruebas, las cuales se describen a continuación:

- Prueba 1: Seguimiento de dos trayectorias predefinidas de un cuadricóptero empleando los controladores tipo PID y basado en Lyapunov.
- Prueba 2: Seguimiento de dos trayectorias predefinidas y una libre del objeto móvil terrestre mientras el cuadricóptero realiza un seguimiento mediante circunnavegación empleando los controladores tipo PID y basado en Lyapunov.

En todas las pruebas se obtendrán los errores de posición de cada eje, las acciones de control de cada controlador y una vista general del resultado obtenido del seguimiento de trayectoria del dron. En base a los datos obtenidos se calcularán índices de desempeño en el intervalo de tiempo que dura la trayectoria.

En ambos casos, los resultados se analizan mediante la cuantificación de los índices IAE, para definir el desempeño de la respuesta de los procesos; y TVU, para conocer el esfuerzo que aplica la acción de control sobre el UAV. De esta manera, se podrá comparar los resultados obtenidos para evaluar la efectividad de la aplicación de los controladores.

#### **3.1 PRUEBAS DE TRAYECTORIAS DEL UAV**

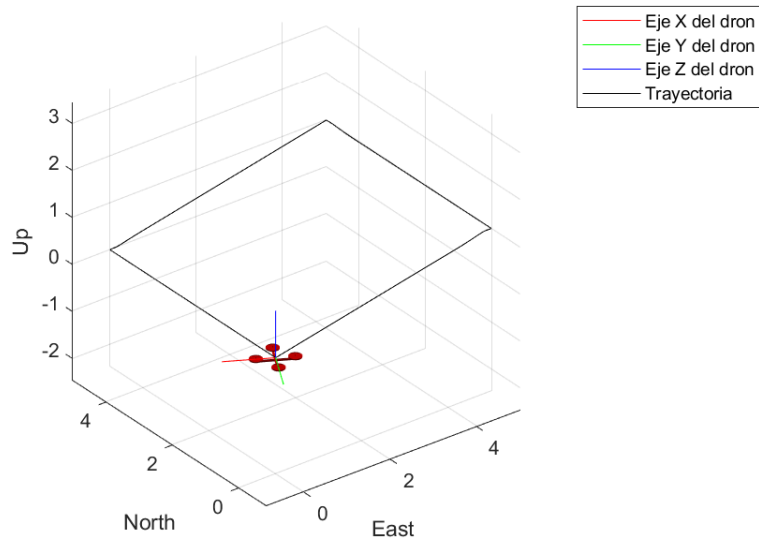
A continuación, se presentan los resultados obtenidos por ambos controladores, tipo PID y basado en Lyapunov, al definir dos trayectorias para el movimiento del cuadrirrotor. En cada prueba, se muestran la gráfica de la trayectoria, la gráfica de los errores de control, las gráficas de las acciones de control y los índices de desempeño obtenidos. Para estas pruebas se ha definido un tiempo de simulación de 200 [s] para realizar cada trayectoria predefinida.

##### **3.1.1 TRAYECTORIA CUADRADA**

La trayectoria que sigue el dron del UAV Toolbox en esta prueba es la de un cuadrado de 5x5 [m], con variación de altura de 1 [m] y con yaw variable con referencia de una señal sinusoidal.

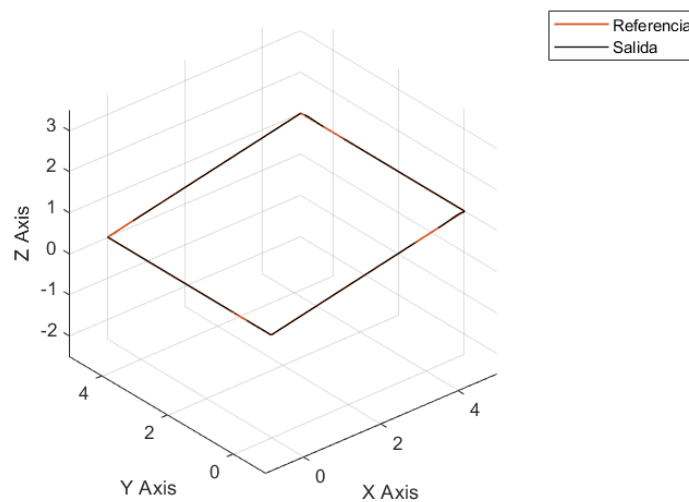
### 3.1.1.1 Prueba con controlador tipo PID

En la Figura 3.1 se muestra la trayectoria realizada por el dron que inicia y termina en las coordenadas [0,0,0] implementando controladores PD para los ejes x,y,z, donde se puede apreciar el cambio de referencia en altura y los cuatro cambios de referencias que conforman una figura cuadrada. A su vez, se puede apreciar los ejes del body frame del cuadricóptero.

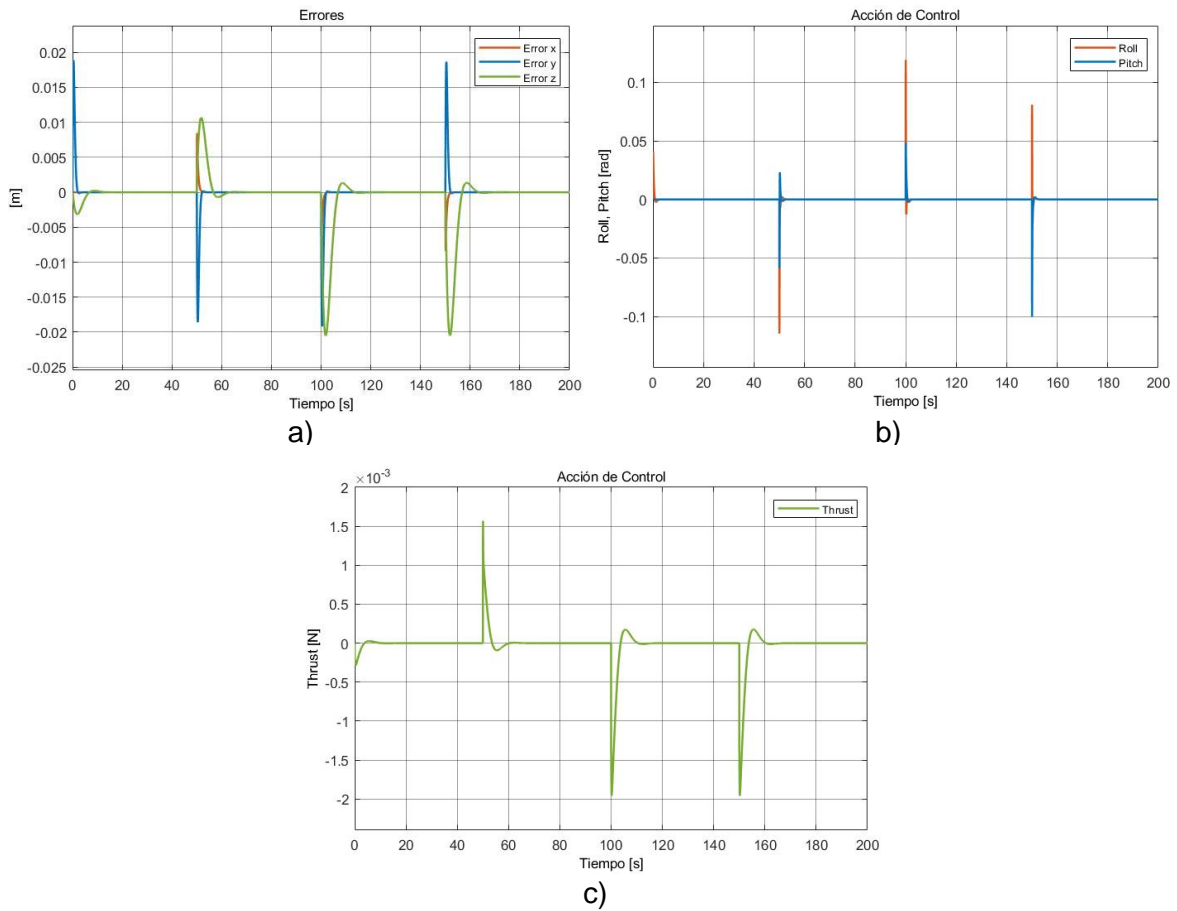


**Figura 3.1.** Gráfica 3D de la trayectoria del dron con control tipo PID - Cuadrado

En la Figura 3.2 se puede apreciar de manera clara la trayectoria de referencia y la realizada por el cuadricóptero. Existen pequeñas variaciones en la gráfica resultante debido a los cambios de referencia dados en cada eje.



**Figura 3.2.** Gráfica 3D del seguimiento de la trayectoria cuadrada del dron con controlador tipo PID

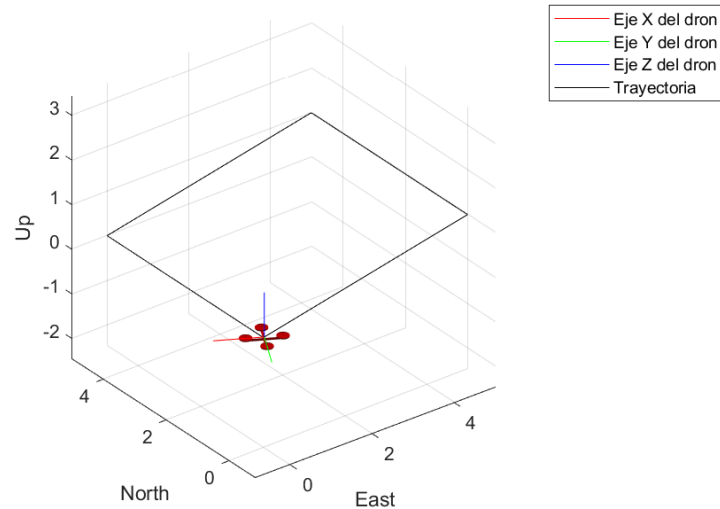


**Figura 3.3.** Resultados de trayectoria de un cuadrado con control tipo PID: a) Gráfica de errores  $x,y,z$ ; b) Gráfica de acciones de control  $x,y$ ; c) Gráfica de acción de control de  $z$

Como se puede observar en la Figura 3.3, se ha obtenido las diferentes gráficas de los resultados de la simulación. En la Figura 3.3.a, se muestran los errores de posición de  $x,y,z$  ante los cuatro cambios de referencia pronunciados que conforman el cuadrado; donde, una vez generado cada cambio de referencia, el controlador PD genera una acción de control que contrarresta el error de posición en cada eje para que este tienda a cero. Las acciones de control generadas para el movimiento lateral y longitudinal de roll y pitch, se pueden observar en la Figura 3.3.b, y la acción de control de Thrust para el cambio de altura se encuentra en la Figura 3.3.c; esto, debido a las diferentes unidades de medidas que tienen estas variables. Analizando las gráficas, se puede observar que no existen picos significativos en ninguno de los ejes, por lo que se identifica que el trabajo realizado por el controlador PD no presenta un cambio brusco; esto también, debido a que, en la ejecución de la trayectoria, no se presentan sobrepicos que deformen la figura de referencia.

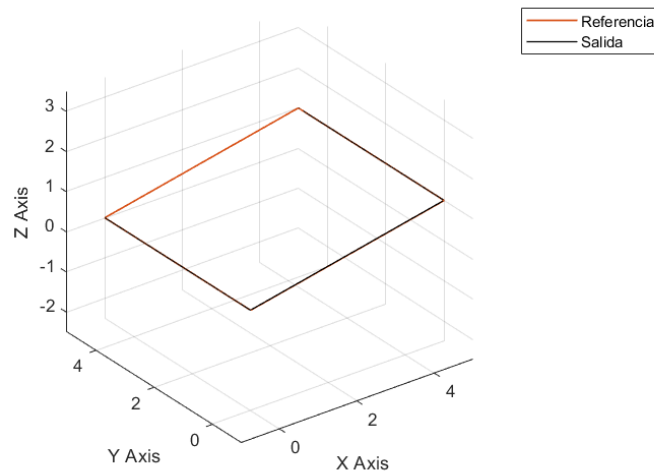
### 3.1.1.2 Prueba con controlador basado en Lyapunov

Como en la prueba anterior, se muestra en la Figura 3.4 la trayectoria realizada por el cuadricóptero con controladores basados en Lyapunov en los ejes  $x,y,z$ , que inicia y termina en las coordenadas  $[0,0,0]$ , donde se puede apreciar el cambio de referencia en altura y los cuatro cambios de referencias de un cuadrado.



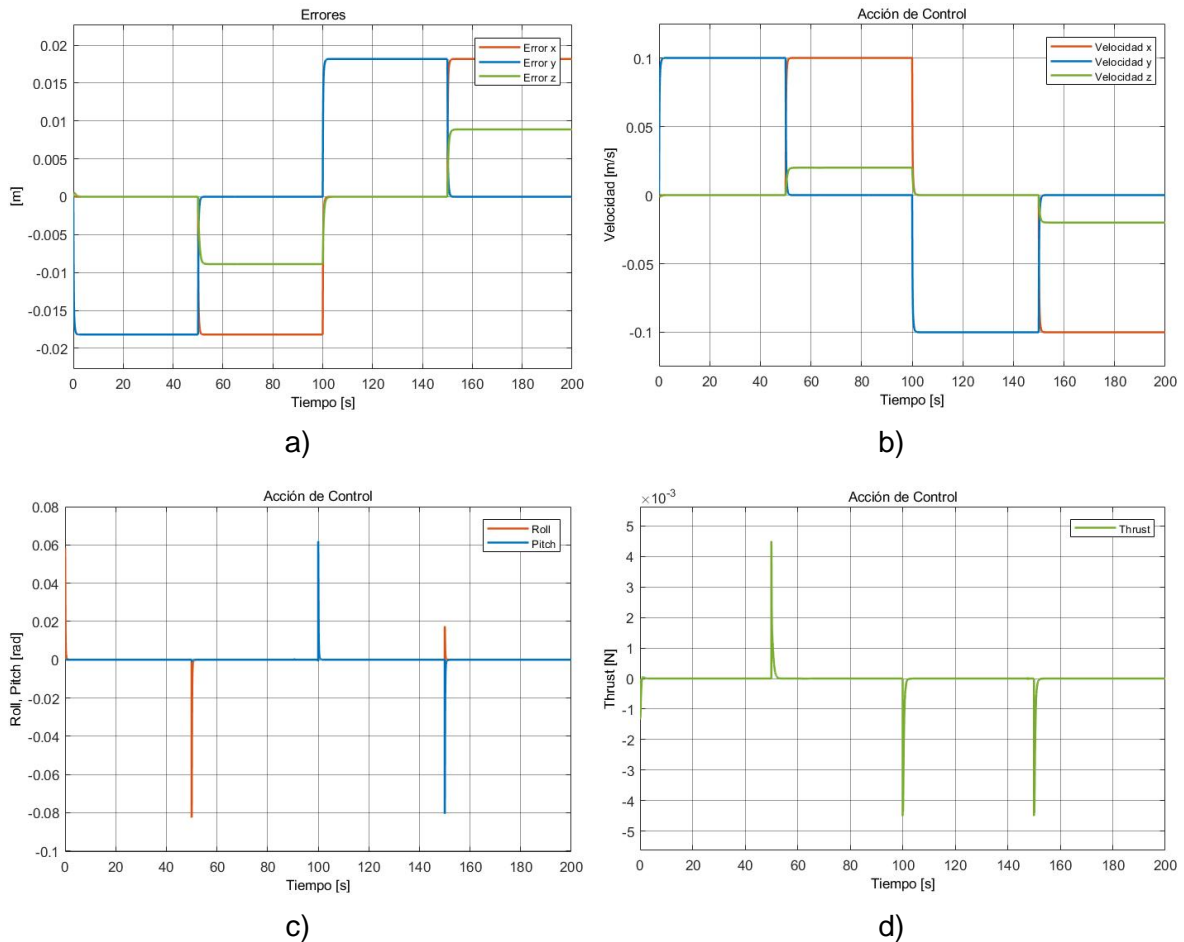
**Figura 3.4.** Gráfica 3D de la trayectoria del dron con controlador basado en Lyapunov - Cuadrado

En la Figura 3.5 se puede apreciar de manera clara las trayectoria de referencia y realizada por el cuadricóptero. En esta gráfica se puede ver que no hay tantas variaciones como en la Figura 3.2; sin embargo, se puede notar que en el primer cambio de referencia del eje  $x$ , el dron pierde un poco de altura.



**Figura 3.5.** Gráfica 3D del seguimiento de la trayectoria cuadrada del dron con controlador basado en Lyapunov

Así como en la anterior prueba, en la Figura 3.6 se han obtenido las diferentes gráficas de los resultados de la simulación. En este caso, se presenta una gráfica más, debido a que en el control basado en Lyapunov se obtuvieron un lazo de control interno y uno externo.



**Figura 3.6.** Resultados de trayectoria de un cuadrado con control basado en Lyapunov:

- a) Gráfica de errores  $x,y,z$ ; b) Gráfica de acciones de control de velocidad de  $x,y,z$ ;
- c) Gráfica de acciones de control  $x,y$ ; d) Gráfica de acción de control de  $z$

En la Figura 3.6.a, se muestran los errores de posición de  $x,y,z$  ante los cuatro cambios de referencia del cuadrado. Al tener un lazo interno de velocidad, se muestra un error de posición constante debido a que la referencia del control de velocidad debe ser diferente de cero para que el dron continúe desplazándose; la velocidad en cada eje no será igual a cero si la trayectoria sigue presentando un cambio de referencia, respectivamente. Una vez generado cada cambio de referencia, el controlador basado en Lyapunov genera una acción de control que contrarresta el error de posición en cada eje. En la Figura 3.6.b, se presenta la acción de control de velocidad respectiva para desplazarse en cada uno de los ejes, siendo esta la referencia para el lazo interno de control que genera las acciones de



control para el movimiento lateral y longitudinal de roll y pitch, ver la Figura 3.6.c, y la acción de control de Thrust para el cambio de altura, Figura 3.6.d.

De las gráficas se puede observar que el controlador basado en Lyapunov funciona efectivamente debido a que los valores de errores en cada eje están cercanos a cero, y las acciones finales de control no presentan picos significativos en ninguno de los ejes. También, en la ejecución de la trayectoria, no se presenta ninguna deformación la figura de referencia.

### 3.1.1.3 Comparación de resultados

Como se pudo observar en las pruebas realizadas, cada controlador logró el resultado esperado, siguiendo la trayectoria de un cuadrado propuesta. Sin embargo, con la Tabla 3.1 se puede realizar una comparación específica tomando en consideración el desempeño de cada controlador.

**Tabla 3.1.** Comparación de índices de desempeño - Cuadrado

Controlador	IAE			TVU		
	X	Y	Z	X	Y	Z
Tipo PID	0.01309	0.06844	0.2124	0.7971	0.347	0.01255
Basado en Lyapunov	1.815	1.818	0.885	0.3511	0.4719	0.0309

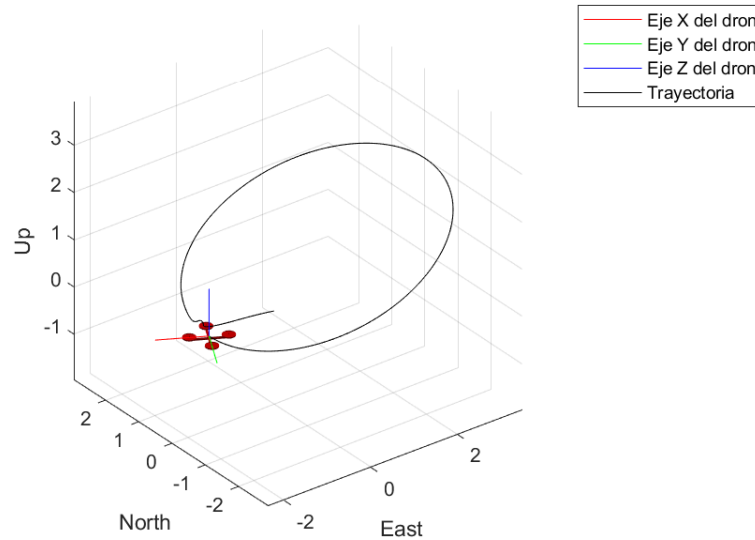
Empezando por el índice de desempeño IAE, el controlador tipo PID muestra valores menores en cada uno de sus ejes a los obtenidos con el controlador basado en Lyapunov. Lo cual implica que se realizó un mejor seguimiento de la trayectoria con el controlador tipo PID. En cuanto al índice TVU, el controlador basado en Lyapunov presenta menor esfuerzo en el eje x, sin embargo, el controlador tipo PID presenta esta característica en los otros ejes. Es necesario mencionar que estos últimos dos valores no representan un cambio significativo, por lo cual se puede decir que ambos controladores generan un esfuerzo similar.

### 3.1.2 TRAYECTORIA CIRCULAR

En esta prueba, la trayectoria determinada para el dron del UAV Toolbox es la de un círculo de 2.5 [m] de radio, con yaw variable con referencia de una señal sinusoidal. La variación de altura inicial es de 1 [m] antes del comienzo de la trayectoria del círculo, y después la altura del dron varía 1 [m] más. Esta prueba es fundamental para definir el movimiento del dron cuando se realice la implementación del seguimiento de un objeto móvil terrestre mediante circunnavegación.

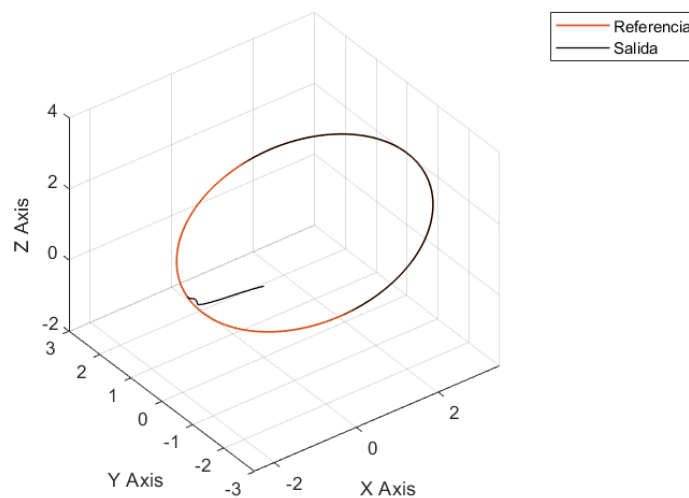
### 3.1.2.1 Prueba con controlador tipo PID

En la Figura 3.7 se muestra la trayectoria realizada por el dron para dibujar un círculo implementando controladores PD para los ejes  $x, y, z$ , donde se puede apreciar el cambio de referencia en altura. A diferencia de la prueba de trayectoria cuadrada, esta prueba requiere de señales sinusoidales para todos los ejes, los cuales no son tan bruscos.



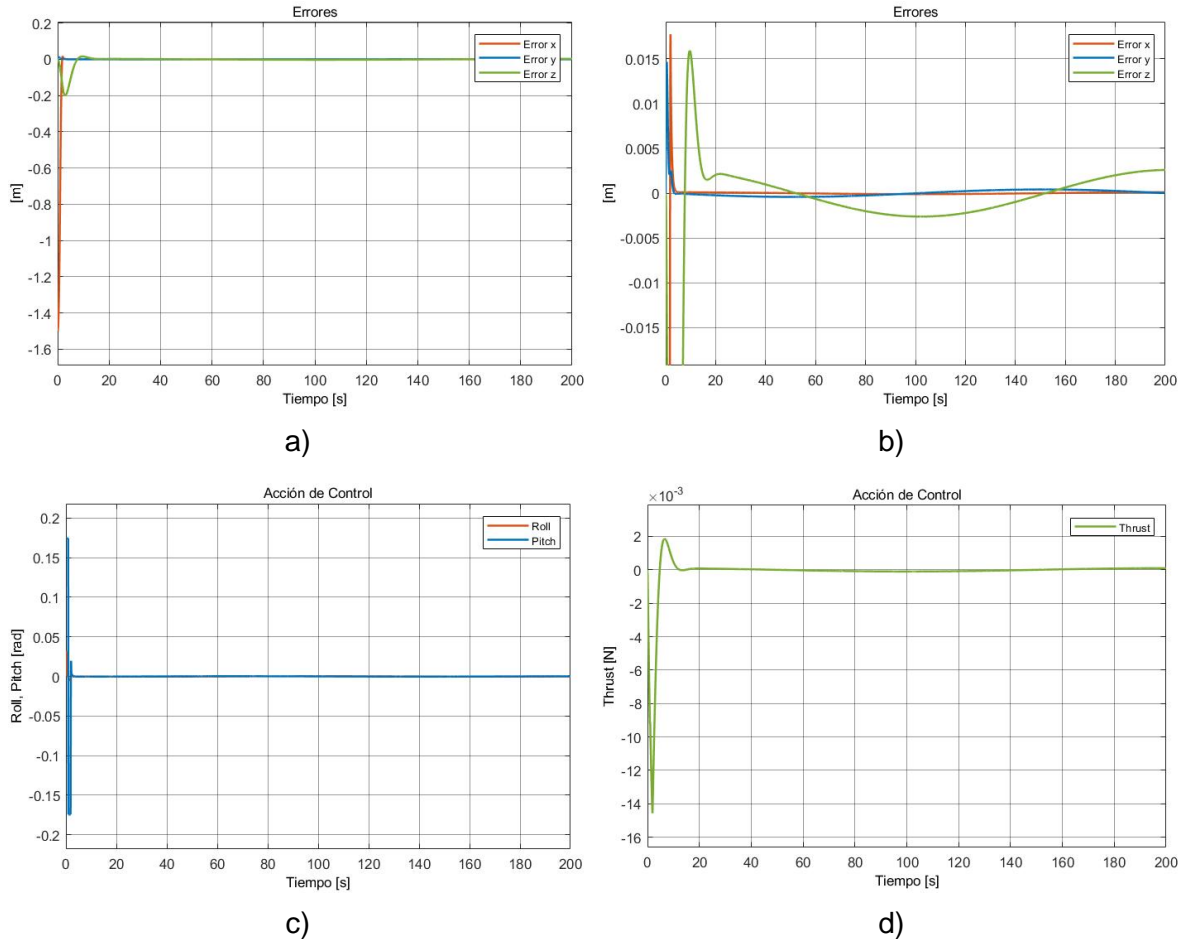
**Figura 3.7.** Gráfica 3D de la trayectoria del dron con control tipo PID - Círculo

En la Figura 3.8 se puede apreciar las trayectorias de referencia y la realizada por el cuadricóptero. En la gráfica se observa que el dron sigue la trayectoria correctamente para los ejes  $x$  y  $y$ , pero se muestra un cambio en el seguimiento de la referencia del eje  $z$ .



**Figura 3.8.** Gráfica 3D del seguimiento de la trayectoria circular del dron con controlador tipo PID

En la Figura 3.9 se pueden observar las diferentes gráficas de los resultados de la simulación de esta prueba. En todas las imágenes se puede resaltar que hay un pico al inicio en cada señal obtenida, esto se debe a que el dron no empieza en el origen del marco terrestre de referencia, por lo que debe moverse al punto inicial del círculo.



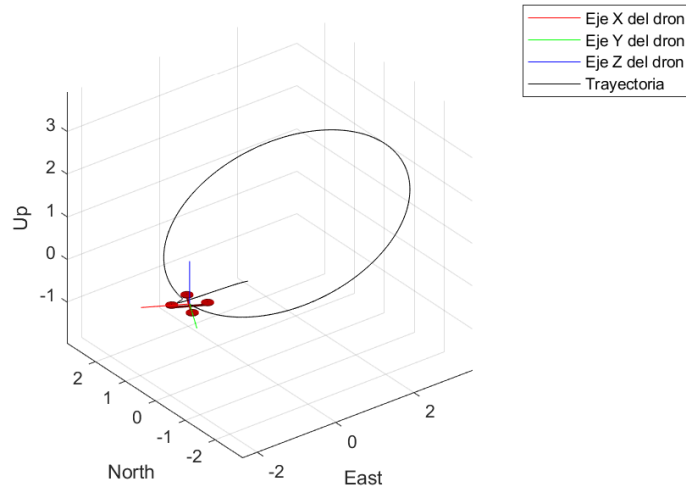
**Figura 3.9.** Resultados de trayectoria de un círculo con control tipo PID: a) Gráfica de errores x,y,z; b) Gráfica de errores x,y,z ampliada; c) Gráfica de acciones de control de x,y; d) Gráfica de acción de control de z

En la Figura 3.9.a, se muestran los errores de posición de x,y,z obtenidos a partir del seguimiento a las señales de referencia sinusoidales en cada eje. Al obtener errores tan pequeños en el seguimiento de la trayectoria y, teniendo en cuenta los picos iniciales de cada eje, no se puede observar de manera clara las gráficas, por lo que se decidió ampliar la imagen en la Figura 3.9.b. En esta figura se puede observar que no hay sobrepicos y los errores son muy cercanos a cero. En la Figura 3.9.c, se muestra la acción de control que genera el controlador PD para roll y pitch; y en la Figura 3.9.d, la acción de control de Thrust.

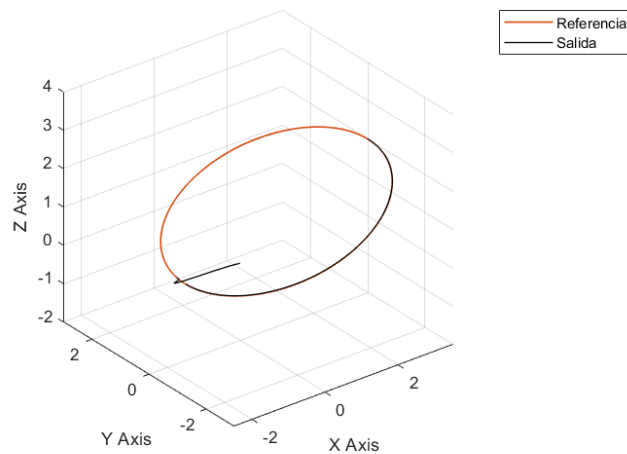
Al realizar un análisis de las gráficas, se determina que el control PD implementado muestra un desempeño favorable en la trayectoria definida, a pesar del pico inicial de traslado al punto de partida y el cambio sinusoidal del ángulo de heading del dron.

### 3.1.2.2 Prueba con controlador basado en Lyapunov

En la Figura 3.10, se muestra la trayectoria circular realizada por el cuadricóptero con controladores basados en Lyapunov en los ejes x,y,z.



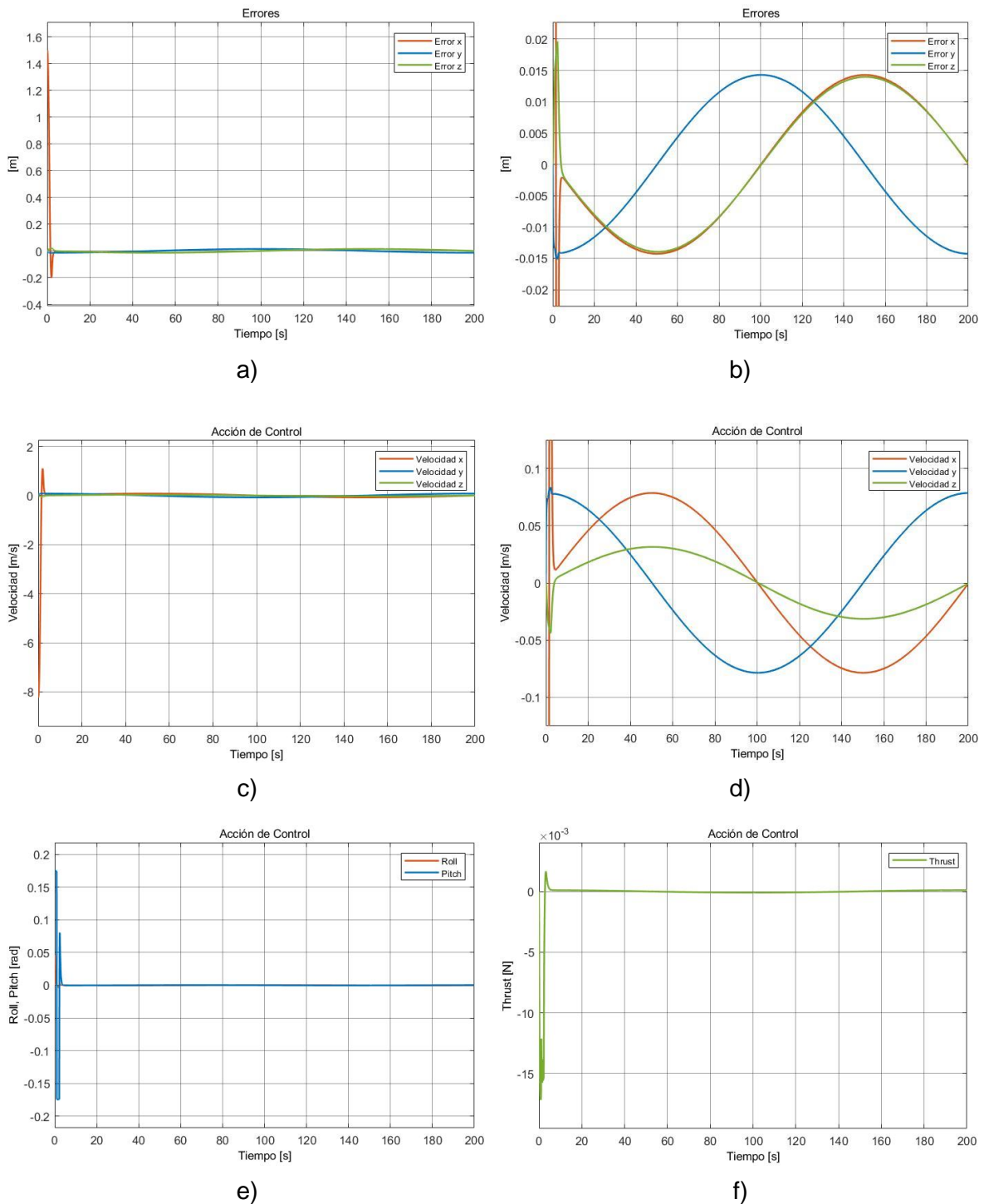
**Figura 3.10.** Gráfica 3D de la trayectoria del dron con control basado en Lyapunov - Círculo



**Figura 3.11.** Gráfica 3D del seguimiento de la trayectoria circular del dron con controlador basado en Lyapunov

En la Figura 3.11 se pueden apreciar las trayectorias de referencia y realizada por el cuadricóptero. Se observa un cambio en el seguimiento de la referencia del eje z, donde se puede percibir que hay un ligero desfase entre la referencia y la salida. Esto puede

derivarse de la forma en que se implementaron dos controladores, debido a que los mismos se encuentran en cascada (Figura 2.6).



**Figura 3.12.** Resultados de trayectoria de un círculo con control basado en Lyapunov:

- a) Gráfica de errores  $x,y,z$ ; b) Gráfica de errores  $x,y,z$  ampliada; c) Gráfica de acciones de control de velocidad de  $x,y,z$ ; d) Gráfica de acciones de control de velocidad de  $x,y,z$  ampliada; e) Gráfica de acciones de control de  $x,y$ ; f) Gráfica de acción de control de  $z$

Así como en la anterior prueba, en la Figura 3.12 se han obtenido las diferentes gráficas de los resultados de la simulación.

En la Figura 3.12.a, se muestran los errores de posición de  $x,y,z$  con referencia de señales sinusoidales y su ampliación se evidencia en la Figura 3.12.b. Como se explicó en la prueba de trayectoria cuadrada con el mismo controlador, el lazo interno de velocidad muestra una acción de control opuesta a las señales del error producido, las cuales son variables debido a que existe un cambio de referencia simultáneo en los tres ejes. Así como en la gráfica del error de posición, en la Figura 3.12.c, se presenta la acción de control de velocidad respectiva para desplazarse en cada uno de los ejes, y en la Figura 3.12.d, una ampliación. Estas gráficas son la referencia para el lazo interno de control que genera las acciones de control para el movimiento de roll y pitch, ver la Figura 3.12.e, y la acción de control de Thrust para el cambio de altura, Figura 3.12.f. En estas gráficas se evidencian acciones de control con valores bajos una vez que el dron se ha estabilizado.

Con esto, se puede mencionar que el controlador basado en Lyapunov funciona efectivamente.

### 3.1.2.3 Comparación de resultados

En la Tabla 3.2 se registraron los datos de las pruebas ejecutadas con ambos controladores para realizar una comparación específica tomando en consideración el desempeño de cada una. En ambos controladores, se puede verificar que hay un valor mayor de error y esfuerzo del controlador del eje  $x$ , respecto a los otros ejes, debido al traslado del dron a la posición inicial.

**Tabla 3.2.** Comparación de índices de desempeño - Círculo

Controlador	IAE			TVU		
	X	Y	Z	X	Y	Z
Tipo PID	1.471	0.06773	1.115	0.7448	0.07904	0.0336
Basado en Lyapunov	3.381	1.816	1.817	0.8898	0.1086	0.05093

El controlador tipo PID tuvo un mejor desempeño al seguir la trayectoria del círculo, sin embargo, los esfuerzos registrado en ambos controladores presentan magnitudes similares. En cuanto a los índices de desempeño del IAE, se puede concluir que a pesar de que el dron presenta valores menores en cada uno de sus ejes a los obtenidos con el controlador basado en Lyapunov, se debe resaltar que tanto en el eje  $x$  como en el  $z$ , los valores son similares, lo cual implica que hubo una variación significativa del error en

ambos ejes al inicio de la trayectoria. Mientras que, en el controlador basado en Lyapunov la variación se registra únicamente en el eje  $x$ .

## 3.2 PRUEBAS DE CIRCUNNAVEGACIÓN A UN OBJETO MÓVIL TERRESTRE

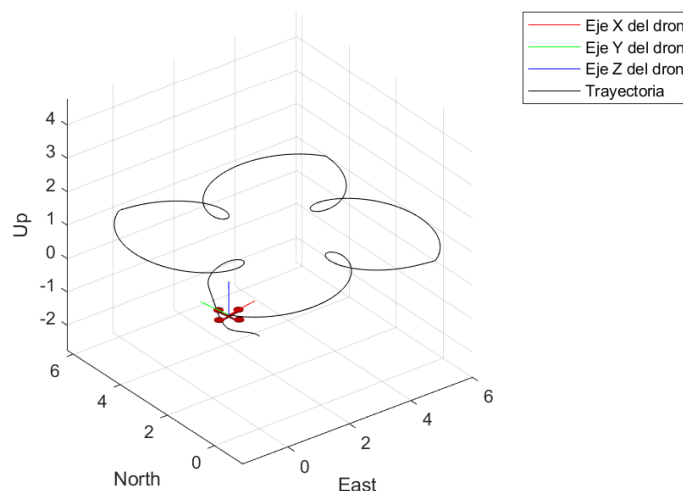
A continuación, se mostrarán las pruebas resultantes de la trayectoria del cuadricóptero al realizar el seguimiento de un objeto móvil terrestre mediante circunnavegación. Para estas pruebas, el objeto móvil terrestre se desplazará de tres formas, realizando la trayectoria de un cuadrado, la de un círculo y una libre; y, en cada una de ellas, el UAV realizará la circunnavegación con el controlador tipo PID y el controlador basado en Lyapunov para realizar una comparación de desempeño. Como se explicó en la Sección 2.1.5, el dron realizará una búsqueda del ángulo de referencia para que su dirección siempre esté orientada hacia el objeto móvil. Para cada prueba la circunnavegación del dron será con un radio de 2.5 [m] y velocidad angular de  $\pi/20$  [rad/s].

### 3.2.1 TRAYECTORIA CUADRADO DE OBJETO MÓVIL

El objeto móvil terrestre recorre una trayectoria de un cuadrado de 5x5 [m], mientras que el dron volará a una altura de 1 [m]. Está determinado un tiempo de simulación de 200 [s].

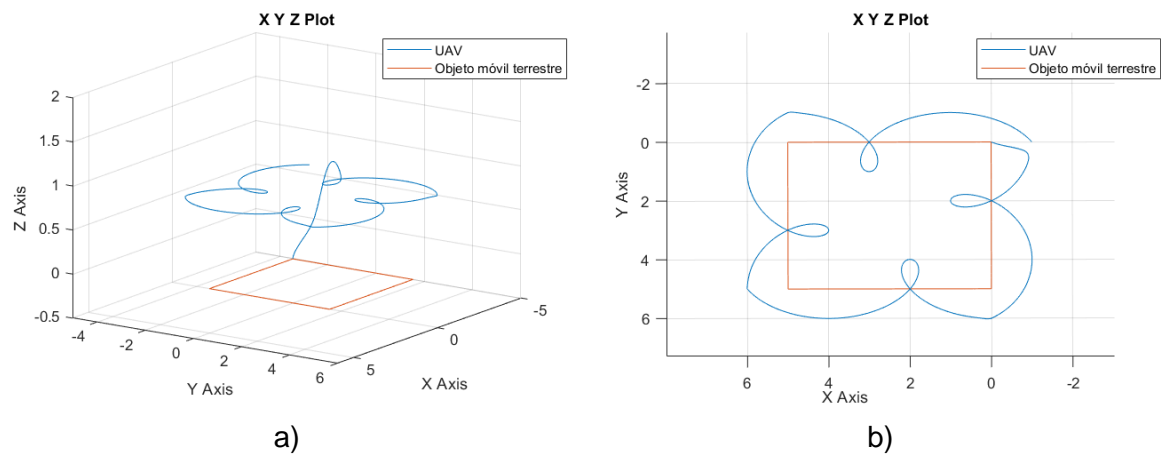
#### 3.2.1.1 Prueba con controlador tipo PID

En la Figura 3.13 se observa la figura resultante de la trayectoria seguida por el dron, con controladores PD para los ejes  $x,y,z$ , al realizar la circunnavegación del objeto móvil que sigue una trayectoria cuadrada. En esta figura se puede apreciar los cuatro cambios de referencias que conforman la figura cuadrada.

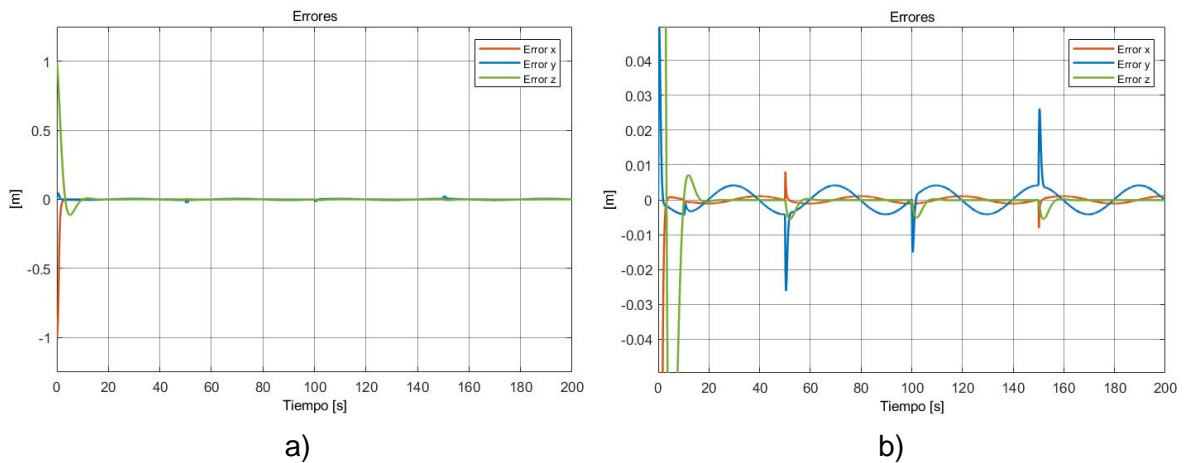


**Figura 3.13.** Gráfica 3D de la trayectoria del dron con control tipo PID – Circunnavegación a una trayectoria cuadrada

Para una mejor visualización del seguimiento del dron al objeto móvil, se presenta la Figura 3.14, mostrando una vista isométrica y una vista aérea del desplazamiento de ambos.



**Figura 3.14.** Gráficas de las trayectorias del dron con control tipo PID y el objeto móvil terrestre – Circunnavegación a una trayectoria cuadrada: a) Gráfica 3D; b) Vista aérea



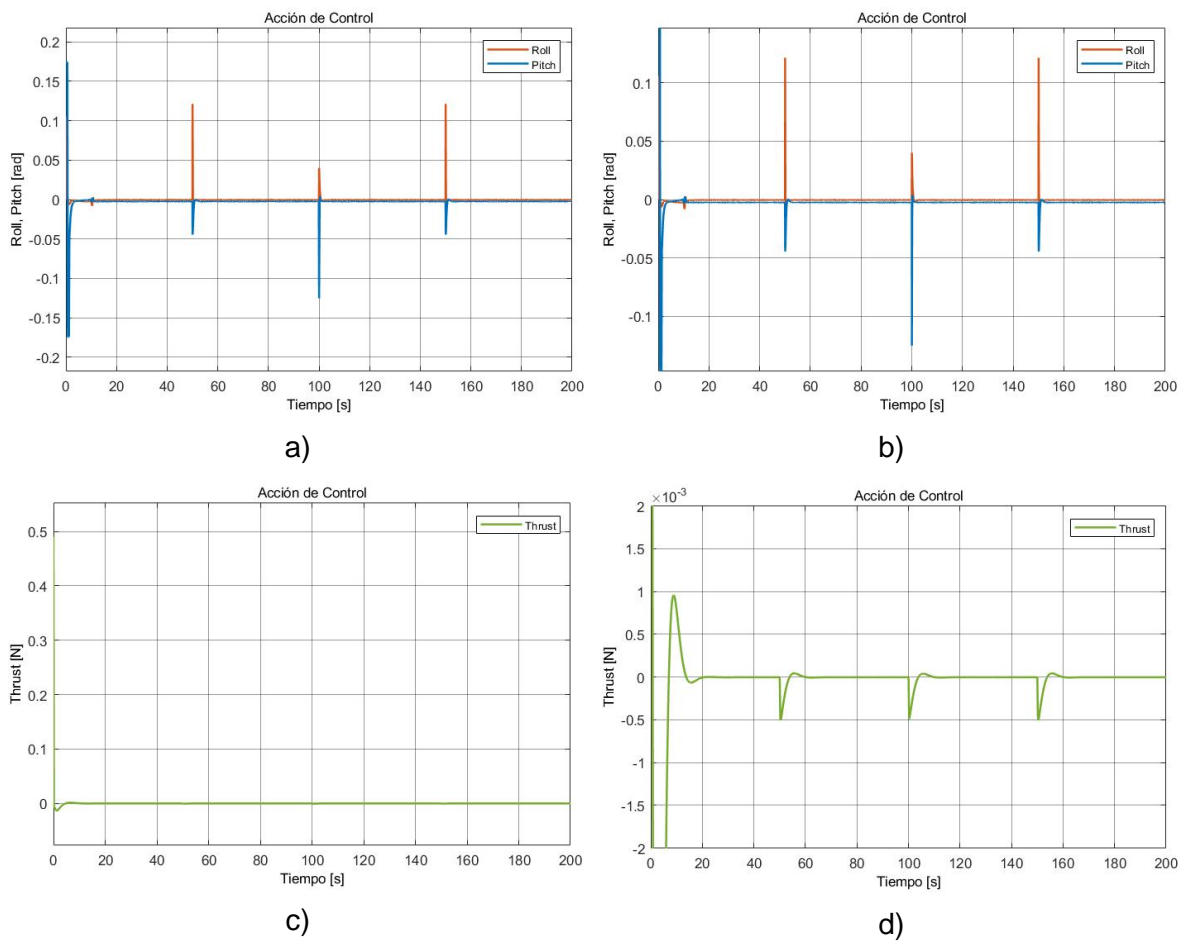
**Figura 3.15.** Resultados de trayectoria de circunnavegación a un cuadrado con control tipo PID: a) Gráfica de errores x,y,z; b) Gráfica de errores x,y,z ampliada

En la Figura 3.15 y la Figura 3.16 se observan los resultados obtenidos en la simulación. En estas figuras se podrá ver una mezcla de los resultados obtenidos en las pruebas anteriores debido a la combinación de trayectorias de ambos objetos.

En la Figura 3.15, se muestran los errores de posición de x,y,z, donde se puede visualizar los cuatro cambios de referencia de la trayectoria de un cuadrado que sigue el objeto móvil, y las señales sinusoidales de la circunferencia que sigue el dron mientras se desplaza. Los valores de los errores son muy bajos, por lo que se requirió realizar una ampliación a la



imagen. Al inicio se puede observar los picos más significativos debido al desplazamiento del dron al punto de partida.



**Figura 3.16.** Resultados de trayectoria de circunnavegación a un cuadrado con control tipo PID: a) Gráfica de acciones de control de x,y; b) Gráfica de acciones de control de x,y ampliada; c) Gráfica de acción de control de z; d) Gráfica de acción de control de z ampliada

En la Figura 3.16.a y b, se presentan las acciones de control para el movimiento lateral y longitudinal de roll y pitch, y la acción de control de Thrust para el cambio de altura, en la Figura 3.16.c y d.

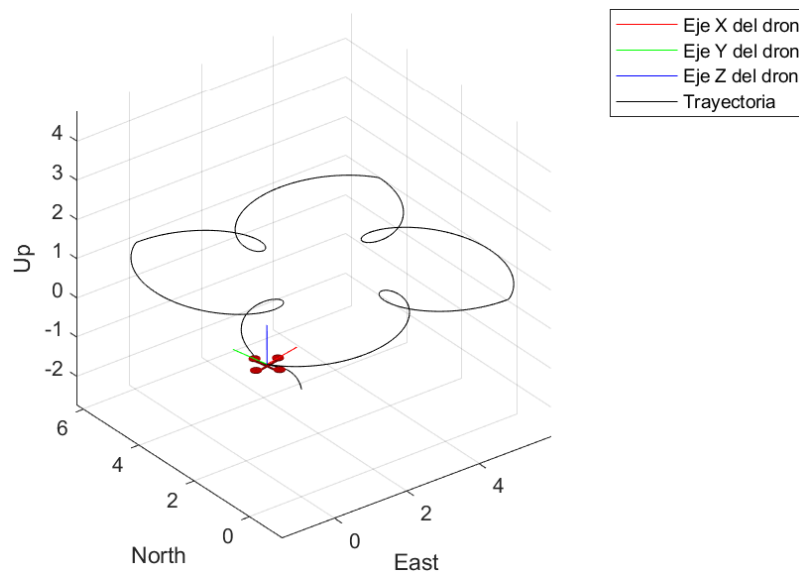
La acción de control generada para el Thrust muestra una variación en cada cambio de referencia del cuadrado, considerando así que el dron se mantiene estable en el resto de la trayectoria. En cuanto a las acciones de control de roll y pitch, se puede notar que los cambios que presentan, igualmente, corresponden a la variación de referencia del cuadrado que sigue el objeto móvil; sin embargo, existe una diferenciación entre la acción

de control de roll cuando el objeto se moviliza en el eje x, y en pitch, al desplazarse en el eje y.

Analizando las gráficas en conjunto, se puede determinar que el control PD implementado funciona de manera correcta. A diferencia de las pruebas anteriores donde el dron seguía la trayectoria de un cuadrado o un círculo, ahora se aumenta un algoritmo de control que determina la posición que debe seguir en el momento de ejecución; y, a pesar de eso, se obtienen resultados favorables.

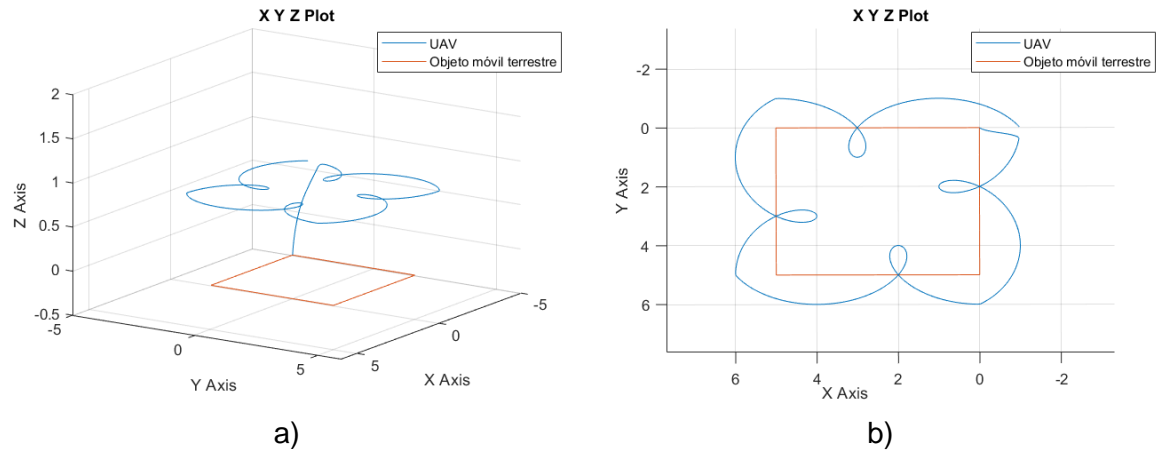
### 3.2.1.2 Prueba con controlador basado en Lyapunov

En la Figura 3.17 se observa el resultado de la trayectoria seguida por el dron, con controladores basados en Lyapunov para los ejes x,y,z, al realizar la circunnavegación del objeto móvil que sigue una trayectoria cuadrada. Como en la anterior prueba, en esta figura se puede apreciar los cuatro cambios de referencias que conforman la figura cuadrada; sin embargo, se puede apreciar la diferencia entre ambos controladores al inicio de la trayectoria, donde en esta prueba se tiene un movimiento menos brusco.



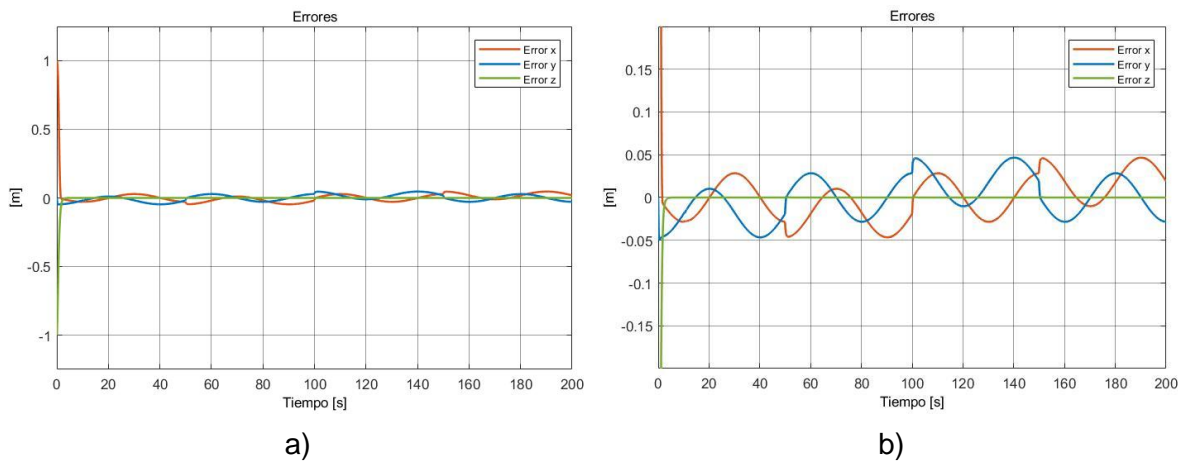
**Figura 3.17.** Gráfica 3D de la trayectoria del dron con control basado en Lyapunov – Circunnavegación a una trayectoria cuadrada

Se presenta en la Figura 3.18, una vista isométrica y una vista aérea del desplazamiento del dron y el objeto móvil para tener una mejor visualización del seguimiento.

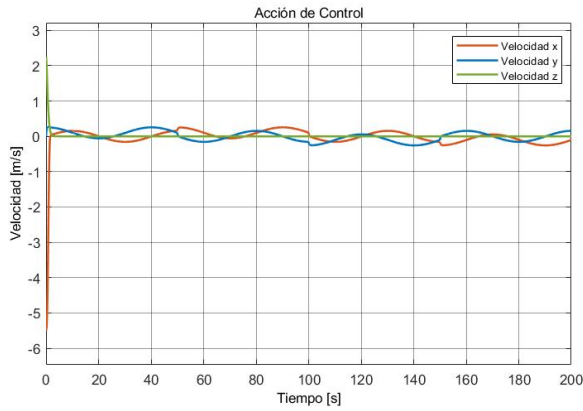


**Figura 3.18.** Gráficas de las trayectorias del dron con control basado en Lyapunov y el objeto móvil terrestre – Circunnavegación a una trayectoria cuadrada: a) Gráfica 3D; b) Vista aérea

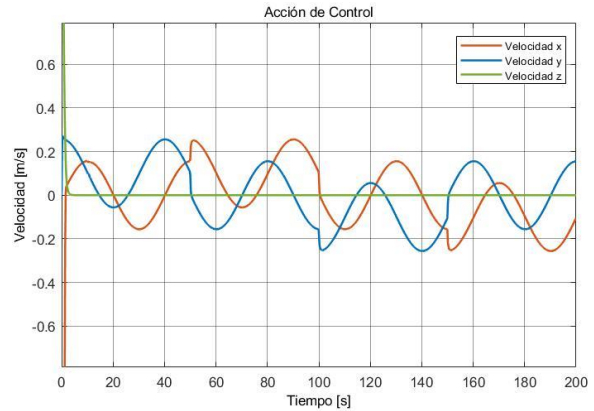
En la Figura 3.19, se muestran los errores de posición de  $x,y,z$ . En el eje  $z$  se puede apreciar que el error tiende a cero cuando se estabiliza después de generar un pico inicial que implica el cambio de posición al solicitado. Los errores para los otros dos ejes muestran una sinusoidal con cuatro cambios en su offset, mismos que muestran el cambio de referencia de la trayectoria del cuadrado que sigue el objeto móvil. A diferencia del anterior controlador, en esta prueba no se ven picos en la trayectoria. Los valores de los errores son muy bajos, por lo que se requirió realizar una ampliación a la imagen. Al inicio se puede observar los picos más significativos debido al desplazamiento del dron al punto de partida.



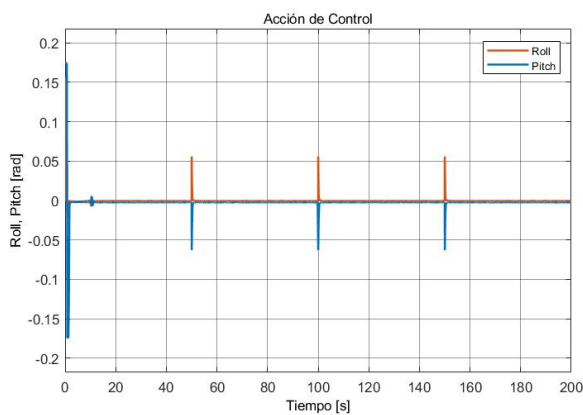
**Figura 3.19.** Resultados de trayectoria de circunnavegación a un cuadrado con control basado en Lyapunov: a) Gráfica de errores  $x,y,z$ ; b) Gráfica de errores  $x,y,z$  ampliada



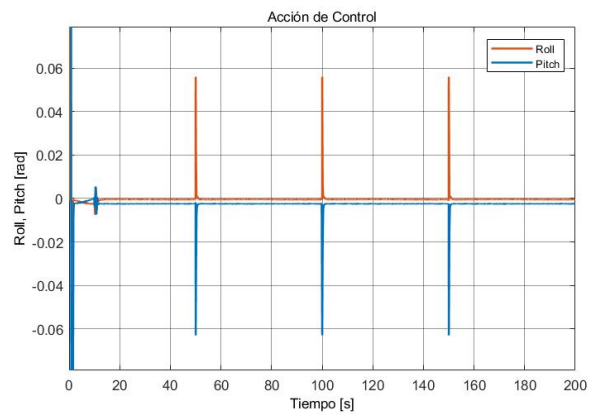
a)



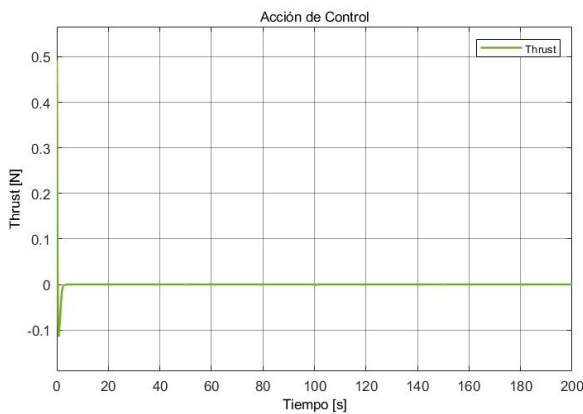
b)



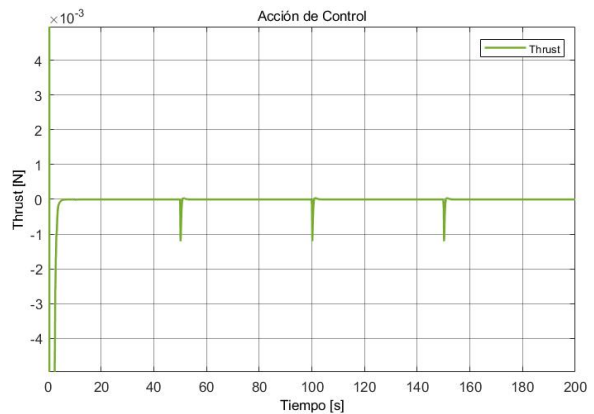
c)



d)



e)



f)

**Figura 3.20.** Resultados de trayectoria de circunnavegación a un cuadrado con control basado en Lyapunov: a) Gráfica de acciones de control de velocidad de  $x,y,z$ ; b) Gráfica de acciones de control de velocidad de  $x,y,z$  ampliada; c) Gráfica de acciones de control  $x,y$ ; d) Gráfica de acciones de control de  $x,y$  ampliada; e) Gráfica de acción de control de  $z$ ; f) Gráfica de acción de control de  $z$

En la Figura 3.20 se presentan las acciones de control tanto de velocidad como de posición. Las acciones de velocidad de  $x,y,z$  se pueden ver en la Figura 3.20.a y b. En estas gráficas se encuentran las señales sinusoidales contrarias a lo visto en el error de posición de  $x,y,z$ ; es decir, señales continuas con cambios de offset, esto debido a que se requiere de una velocidad continua para poder desplazarse. Las acciones de control generadas para roll y pitch, ver Figura 3.20.c y Figura 3.20.d, muestran una similitud en sus magnitudes como un espejo. Esto se obtiene por la sintonización realizada en la Sección 2.1.3.2 donde se determina al dron como una partícula y, por lo tanto, se utilizan las mismas constantes para el control de posición y velocidad de cada eje. En la Figura 3.20.e y f, el Thrust muestra una variación pequeña e instantánea en cada cambio de referencia del cuadrado, considerando así que el dron se mantiene estable en el resto de la trayectoria.

Analizando las gráficas en conjunto, se puede determinar que el control basado en Lyapunov implementado funciona de manera correcta.

### 3.2.1.3 Comparación de resultados

En la Tabla 3.3 se registraron los datos de las pruebas ejecutadas con ambos controladores para realizar una comparación específica tomando en consideración el desempeño de cada una. Se puede observar que los valores de los índices de desempeño son más altos que en las anteriores pruebas realizadas a una trayectoria cuadrada sin circunnavegación.

**Tabla 3.3.** Comparación de índices de desempeño – Circunnavegación a un cuadrado

Controlador	IAE			TVU		
	X	Y	Z	X	Y	Z
Tipo PID	1.02	0.6155	2.034	2.21	0.7168	1.016
Basado en Lyapunov	5.125	4.282	0.751	1.488	1.556	1.308

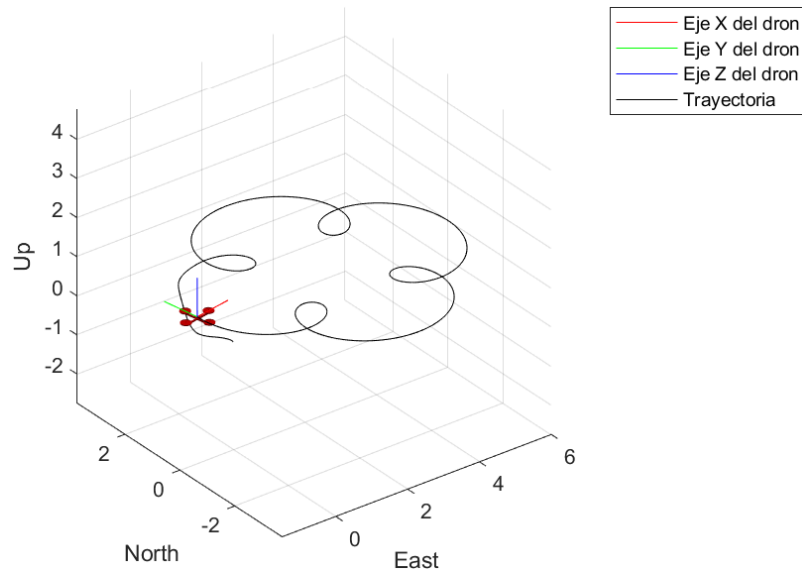
En esta prueba se obtiene como resultado que el controlador tipo PID tuvo un mejor desempeño al hacer circunnavegación en seguimiento a la trayectoria cuadrada de un objeto móvil; sin embargo, el valor de TVU en el eje x presenta la mayor magnitud de todas en este índice. En el controlador basado en Lyapunov se tiene un valor menor de IAE en el eje z que en el controlador tipo PID y sus valores de TVU son muy similares entre todos los ejes.

### 3.2.2 TRAYECTORIA CÍRCULO DE OBJETO MÓVIL

El objeto móvil terrestre recorre una trayectoria de un círculo de 2.5 [m], mientras que el dron volará a una altura de 1 [m]. Está determinado un tiempo de simulación de 200 [s].

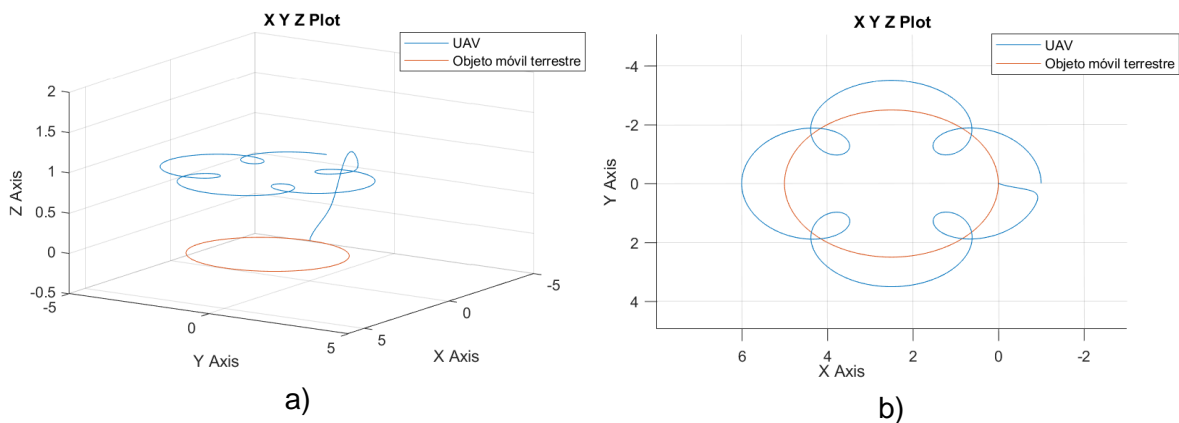
### 3.2.2.1 Prueba con controlador tipo PID

En la Figura 3.21 se observa el resultado de la trayectoria seguida por el dron, con controladores PD para los ejes  $x,y,z$ , al realizar la circunnavegación del objeto móvil que sigue una trayectoria circular. A diferencia de la trayectoria cuadrada del objeto móvil, en esta figura se puede ver un camino más uniforme y continuo.

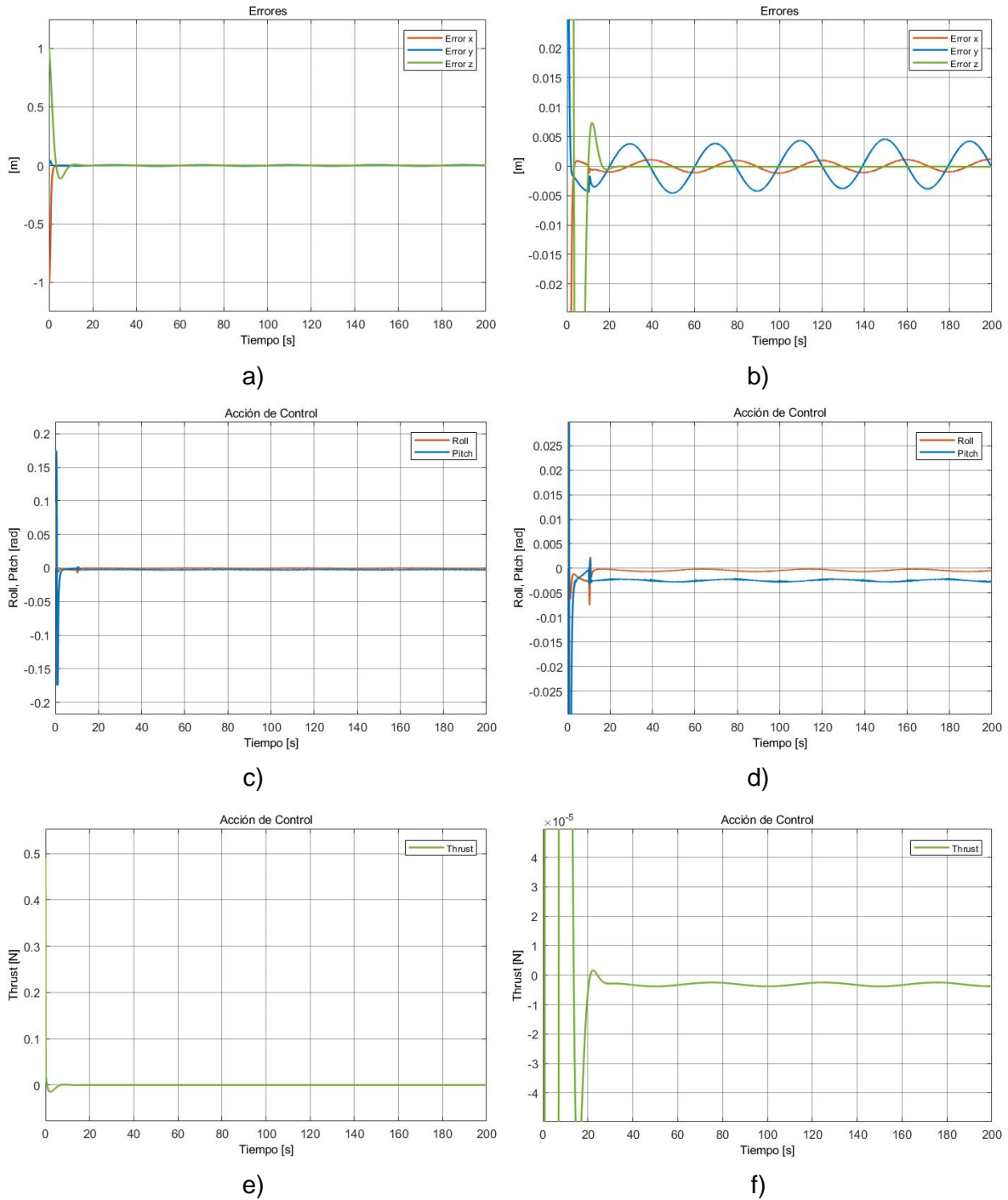


**Figura 3.21.** Gráfica 3D de la trayectoria del dron con control tipo PID – Circunnavegación a una trayectoria circular

Para una mejor visualización del seguimiento del dron al objeto móvil, se presenta la Figura 3.22, mostrando una vista isométrica y una vista aérea del desplazamiento de ambos.



**Figura 3.22.** Gráficas de las trayectorias del dron con control tipo PID y el objeto móvil terrestre – Circunnavegación a una trayectoria circular: a) Gráfica 3D; b) Vista aérea



**Figura 3.23.** Resultados de trayectoria de circunnavegación a un círculo con control tipo PID: a) Gráfica de errores x,y,z; b) Gráfica de errores x,y,z ampliada; c) Gráfica de acciones de control de x,y; d) Gráfica de acciones de control de x,y ampliada; e) Gráfica de acción de control de z; f) Gráfica de acción de control de z ampliada

En la Figura 3.23 se observan los resultados obtenidos en la simulación. Las dos primeras gráficas muestran los errores de posición de x,y,z, donde se puede visualizar las señales sinusoidales de la circunferencia que sigue el dron mientras se desplaza. Además, existe

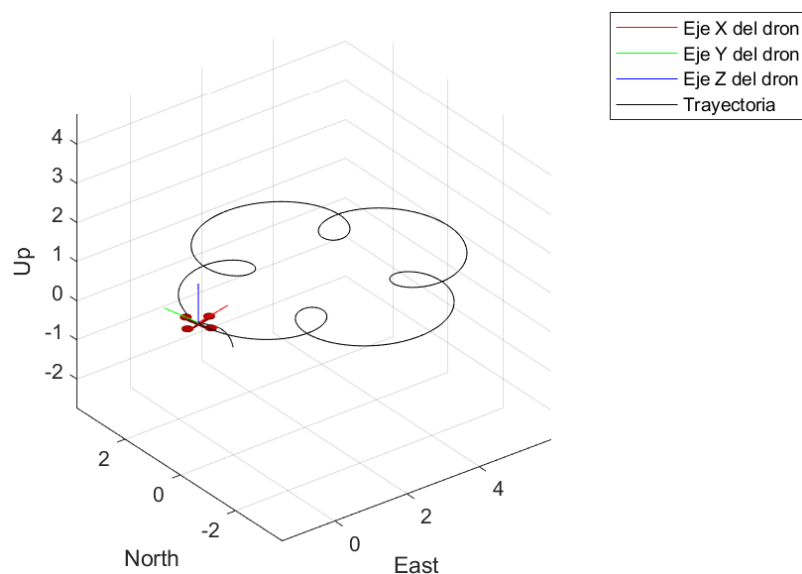
un movimiento sinusoidal del offset de la señal debido a la trayectoria circular que sigue el objeto móvil, esto se puede percibir de mejor manera en la visualización del error del eje y en la Figura 3.23.b. Los valores de los errores son muy bajos, por lo que se requirió realizar una ampliación a la imagen. Al inicio se puede observar los picos más significativos debido al desplazamiento del dron al punto de partida.

En la Figura 3.23.c y d, se presentan las acciones de control para el movimiento lateral y longitudinal de roll y pitch, y la acción de control de Thrust para el cambio de altura, en la Figura 3.23.e y f. Las acciones de control de roll, pitch y Thrust no muestran cambios significativos, siendo estas similares a las que se obtuvo en la prueba de seguimiento de la trayectoria de un círculo, Figura 3.9. En la Figura 3.23.c y f, se puede visualizar un cambio mínimo en las señales debido a las ondas sinusoidales que sigue la circunnavegación.

Analizando las gráficas en conjunto, se puede determinar que el control PD implementado funciona de manera correcta sin cambios repentinos.

### 3.2.2.2 Prueba con controlador basado en Lyapunov

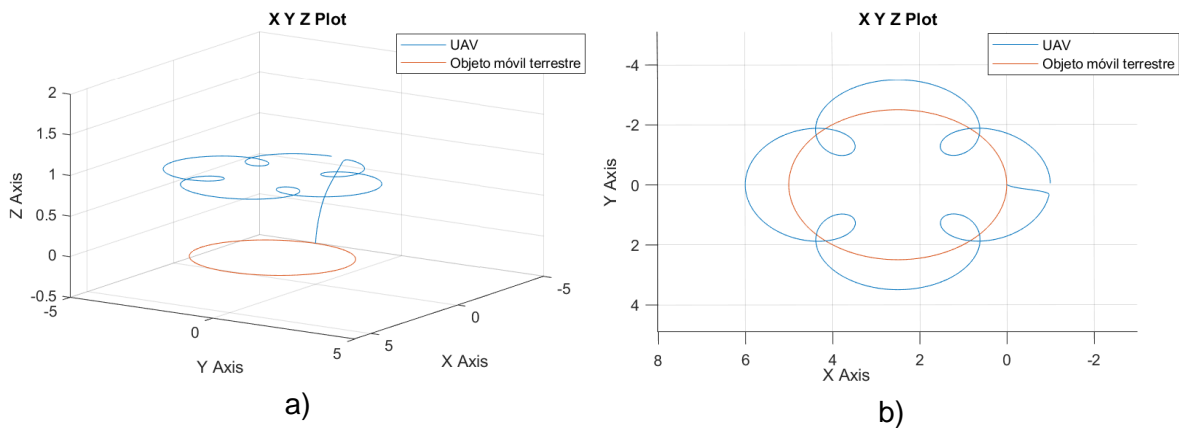
En la Figura 3.24 se observa el resultado de la trayectoria seguida por el dron, con controladores basados en Lyapunov para los ejes x,y,z, al realizar la circunnavegación del objeto móvil que sigue una trayectoria circular. Como en la anterior prueba, en esta figura se puede apreciar cambios más homogéneos; sin embargo, se puede apreciar la diferencia entre ambos controladores al inicio de la trayectoria, donde en esta prueba se tiene un movimiento menos brusco.



**Figura 3.24.** Gráfica 3D de la trayectoria del dron con control basado en Lyapunov – Circunnavegación a una trayectoria circular



Se presenta en la Figura 3.25, una vista isométrica y una vista aérea del desplazamiento del dron y el objeto móvil para tener una mejor visualización del seguimiento.

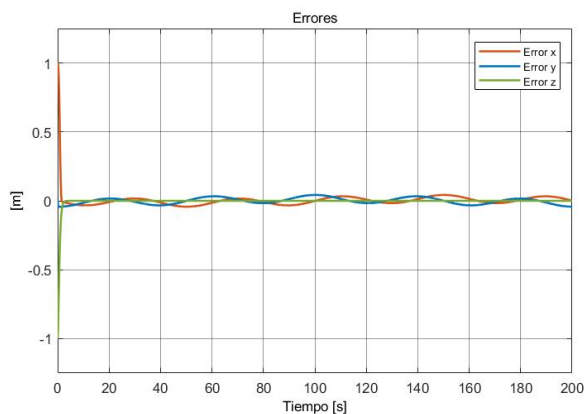


**Figura 3.25.** Gráficas de las trayectorias del dron con control basado en Lyapunov y el objeto móvil terrestre – Circunnavegación a una trayectoria circular: a) Gráfica 3D; b) Vista aérea

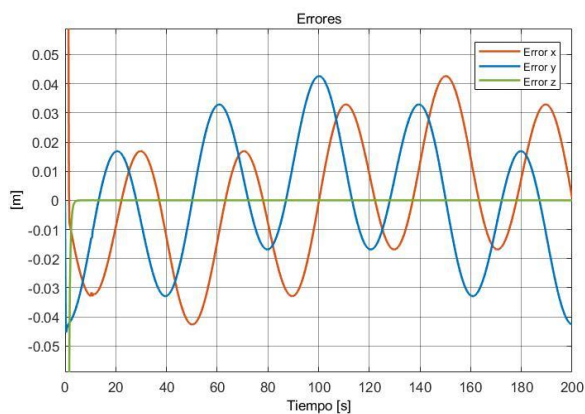
En la Figura 3.26.a y b, se muestran los errores de posición de  $x,y,z$ . A diferencia del controlador tipo PID previamente implementado, en esta prueba se pueden divisar de manera más notoria los errores producidos en el desplazamiento del dron mientras circunnavega al objeto móvil. En la Figura 3.26.b se puede diferenciar que las señales se mueven sobre la sinusoidal que pertenece al círculo que sigue el objeto móvil terrestre.

En la Figura 3.26.c y d, se presentan las acciones de control de velocidad de  $x,y,z$ . En estas gráficas se encuentran las señales sinusoidales contrarias a lo visto en el error de posición de  $x,y,z$ ; es decir, señales continuas con cambios de offset sobre una señal sinusoidal. También se puede notar que los valores de las acciones de control de velocidad son mayores a los vistos anteriormente. Finalmente, en los ejes  $x$  y  $y$  se tienen señales similares debido a la sintonización igualitaria que se propuso en la Sección 2.1.3.2.

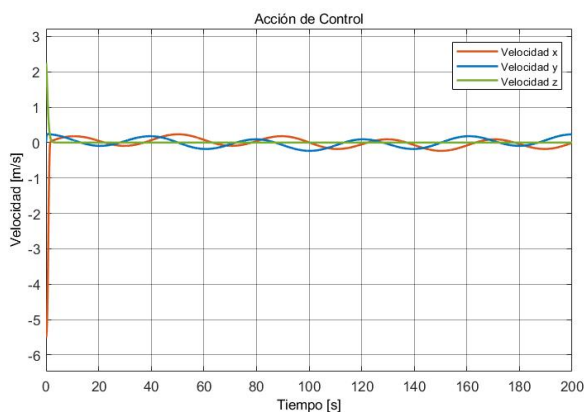
Las acciones de control generadas para roll y pitch se encuentran en la Figura 3.26.e, donde se muestran una similitud en sus magnitudes y se encuentran oscilaciones casi imperceptibles. En la Figura 3.26.f, el Thrust muestra una variación pequeña e instantánea al inicio del seguimiento y con ello se puede determinar que el dron se mantiene estable en el resto de la trayectoria.



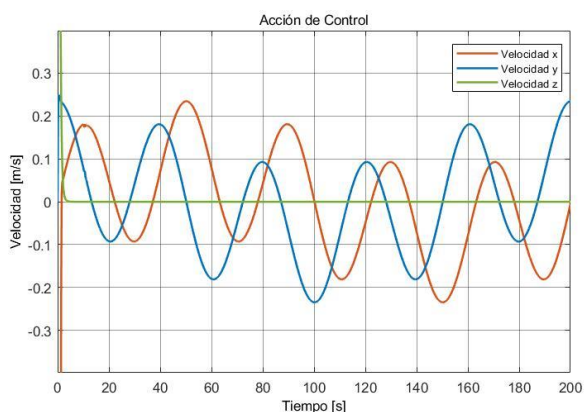
a)



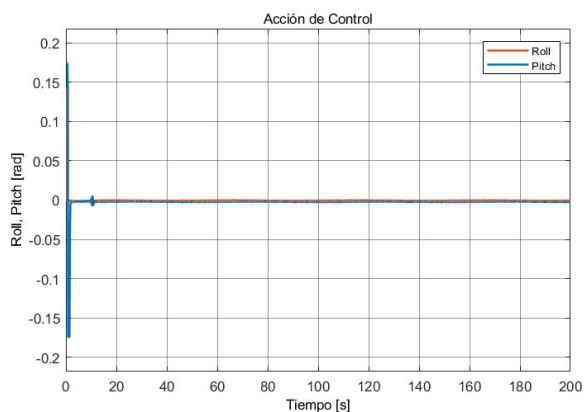
b)



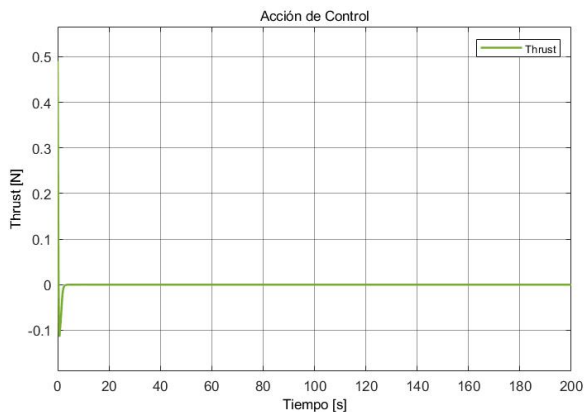
c)



d)



e)



f)

**Figura 3.26.** Resultados de trayectoria de circunnavegación a un círculo con control basado en Lyapunov: a) Gráfica de errores  $x,y,z$ ; b) Gráfica de errores  $x,y,z$  ampliada; c) Gráfica de acciones de control de velocidad de  $x,y,z$ ; d) Gráfica de acciones de control de velocidad de  $x,y,z$  ampliada; e) Gráfica de acciones de control de  $x,y$ ; f) Gráfica de acción de control de  $z$

### 3.2.2.3 Comparación de resultados

En la Tabla 3.4 se registraron los datos de las pruebas ejecutadas con ambos controladores para realizar una comparación específica tomando en consideración el desempeño de cada una.

**Tabla 3.4.** Comparación de índices de desempeño – Circunnavegación a un círculo

Controlador	IAE			TVU		
	X	Y	Z	X	Y	Z
Tipo PID	1.009	0.5631	1.973	1.415	0.4492	1.012
Basado en Lyapunov	4.69	3.843	0.7491	1.133	0.7019	1.243

En esta prueba se obtiene como resultado que el controlador tipo PID tuvo un mejor desempeño al hacer circunnavegación en seguimiento a la trayectoria circular de un objeto móvil. En el controlador basado en Lyapunov, se tiene un valor menor de IAE en el eje z que en el controlador tipo PID, sin embargo, los valores de TVU no son tan altos en ninguno de los ejes.

### 3.2.2.4 Comparación de resultados entre circunnavegación a la trayectoria de un cuadrado y un círculo

A parte de las comparaciones de los índices de rendimiento realizadas en cada prueba, en la Tabla 3.5 se muestran los valores para los ejemplos de circunnavegación para una trayectoria cuadrada y una circular del objeto móvil terrestre.

En esta comparación no se toma en consideración la prueba realizada con la circunnavegación a una trayectoria libre debido a que esta no tiene el mismo tiempo de ejecución y en este caso el usuario tendrá la posibilidad de definir la trayectoria manualmente en un tiempo indeterminado.

**Tabla 3.5.** Comparación de índices de desempeño de circunnavegación

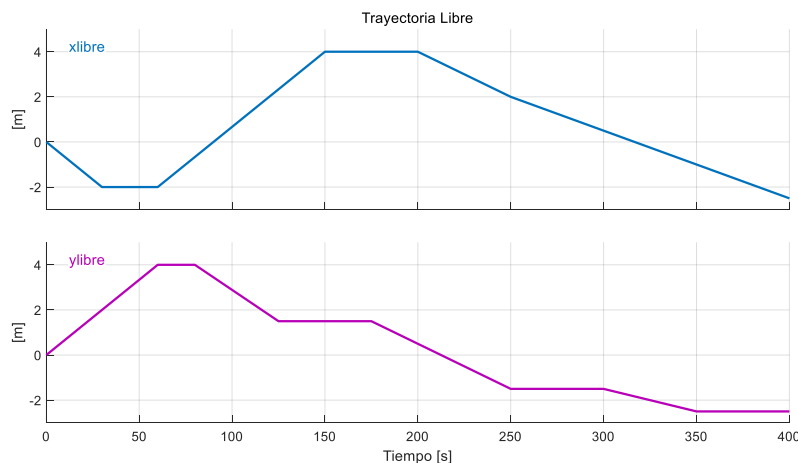
Trayectoria del objeto móvil	Controlador	IAE			TVU		
		X	Y	Z	X	Y	Z
Cuadrado	Tipo PID	1.02	0.6155	2.034	2.21	0.7168	1.016
	Basado en Lyapunov	5.125	4.282	0.751	1.488	1.556	1.308
Círculo	Tipo PID	1.009	0.5631	1.973	1.415	0.4492	1.012
	Basado en Lyapunov	4.69	3.843	0.7491	1.133	0.7019	1.243

En esta tabla se puede notar que los valores en ambas pruebas de circunnavegación resultan similares para cada controlador. Los valores de TVU muestran un esfuerzo mayor

en los controladores al realizar la circunnavegación a un cuadrado, esto debido a los cambios pronunciados que se presentan para realizar la figura. Los valores del IAE del controlador tipo PID no varían tanto entre trayectoria del objeto móvil como lo hacen los del controlador basado en Lyapunov. En el control de altura se puede verificar que el mejor controlador fue el basado en Lyapunov, sin embargo, se tienen mejores resultados con el controlador tipo PID en los otros ejes.

### 3.2.3 TRAYECTORIA LIBRE DE OBJETO MÓVIL

El objeto móvil terrestre recorre una trayectoria que inicia en el punto  $[0,0,0]$ , mientras que el dron volará a una altura de 1 [m]. Está determinado un tiempo de simulación de 400 [s]. La trayectoria, en esta prueba, es predefinida con el propósito de realizar una comparación con ambos controladores implementados. Se usó un bloque Signal Builder de Simulink con las siguientes señales para el eje  $x$  y el eje  $y$ :

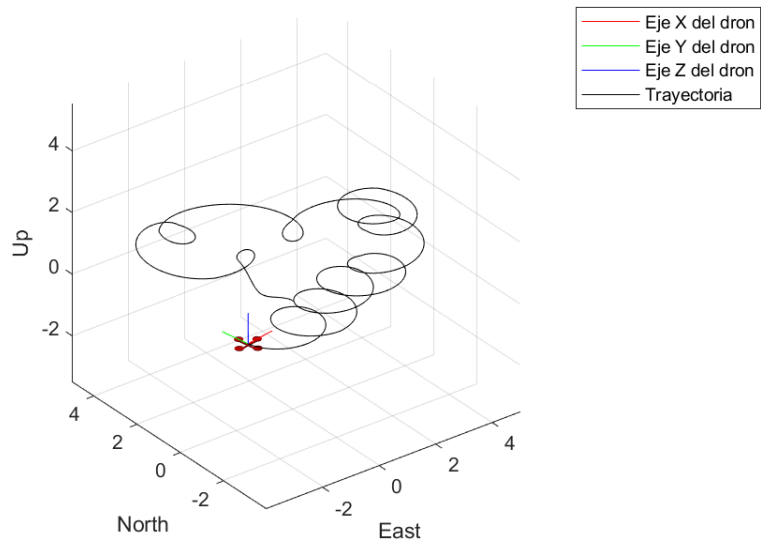


**Figura 3.27.** Trayectorias para los ejes  $x,y$  - Circunnavegación Libre

En la implementación de la HMI, el tiempo definido para el caso de circunnavegación libre es infinito, es decir, la simulación para cuando el usuario lo desee. Además, el movimiento del objeto móvil terrestre se realiza de manera manual, sin el uso del bloque implementado en esta prueba.

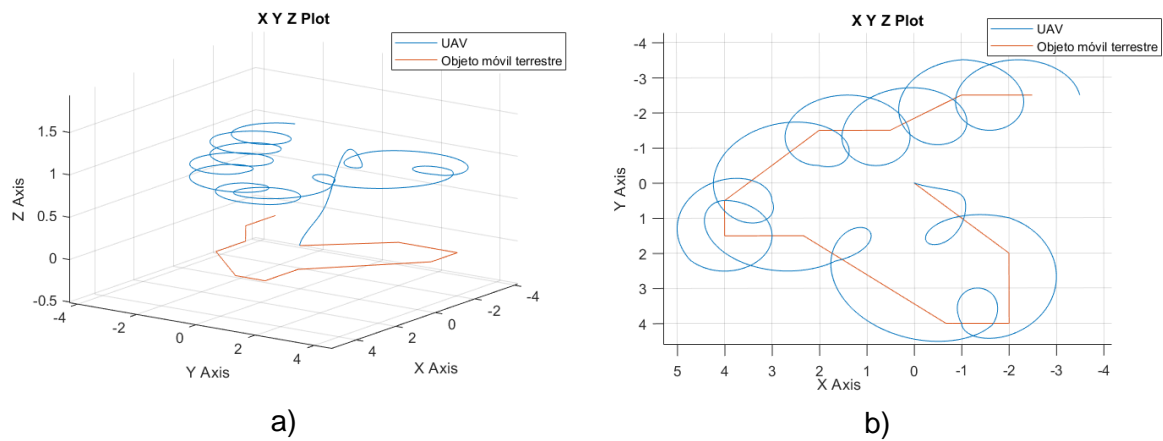
#### 3.2.3.1 Prueba con controlador tipo PID

En la Figura 3.28 se observa la figura resultante de la trayectoria realizada por el dron, con controladores PD para los ejes  $x,y,z$ , al realizar la circunnavegación del objeto móvil que sigue una trayectoria predefinida sin una figura establecida. En esta figura se puede apreciar los cambios de referencia que conforman las señales mostradas en la Figura 3.27.



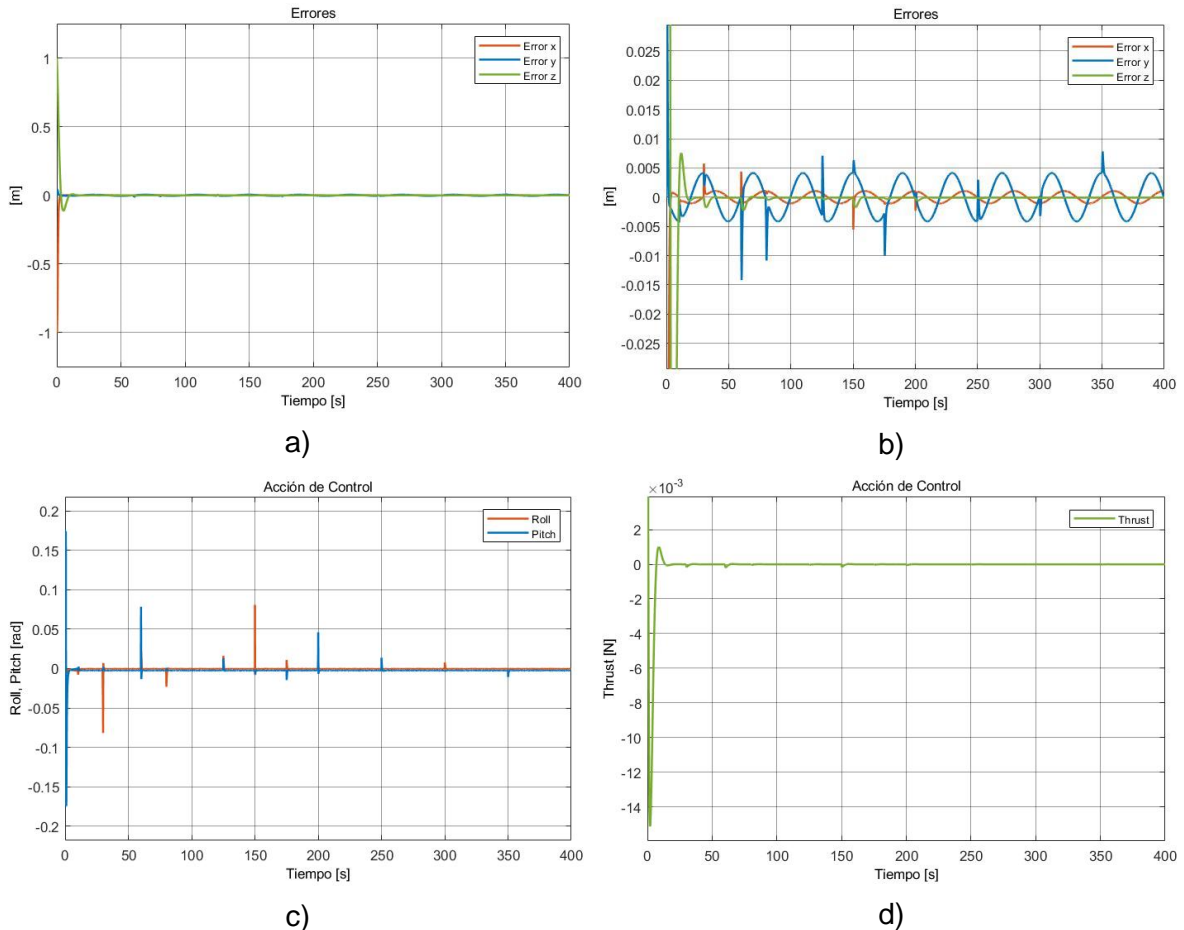
**Figura 3.28.** Gráfica 3D de la trayectoria del dron con control tipo PID – Circunnavegación a una trayectoria libre

Para una mejor visualización del seguimiento del dron al objeto móvil, se presenta la Figura 3.29, mostrando una vista isométrica y una vista aérea del desplazamiento de ambos.



**Figura 3.29.** Gráficas de las trayectorias del dron con control tipo PID y el objeto móvil terrestre – Circunnavegación a una trayectoria libre: a) Gráfica 3D; b) Vista aérea

En la Figura 3.30 se observan los resultados obtenidos en la simulación. En estas figuras se podrá ver una mezcla de los resultados obtenidos en las pruebas anteriores debido a la combinación de trayectorias de ambos objetos.



**Figura 3.30.** Resultados de trayectoria libre de circunnavegación con control tipo PID:

- a) Gráfica de errores  $x,y,z$ ; b) Gráfica de errores  $x,y,z$  ampliada; c) Gráfica de acciones de control de  $x,y$ ; d) Gráfica de acción de control de  $z$

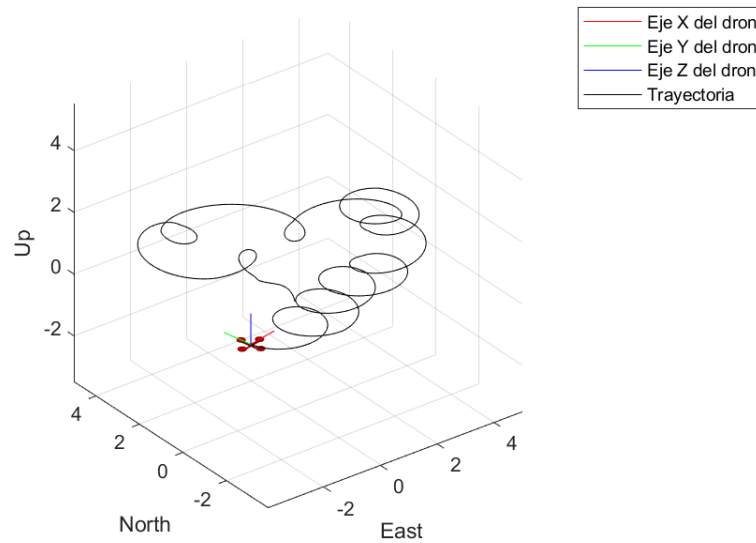
En la Figura 3.30.a y b, se muestran los errores de posición de  $x,y,z$ , donde se puede visualizar los cambios de referencia de la trayectoria libre que sigue el objeto móvil, representados por los picos en las señales sinusoidales, mismas que se ven definidas por la circunferencia que sigue el dron mientras se desplaza. Los valores de los errores son muy bajos, por lo que se requirió realizar una ampliación a la imagen. Al inicio se puede observar los picos más significativos debido al desplazamiento del dron al punto de partida.

En la Figura 3.30.c y d, se presentan las acciones de control para el movimiento lateral y longitudinal de roll y pitch, y la acción de control de Thrust para el cambio de altura. En estas se puede observar que hay variaciones en cada cambio de referencia; sin embargo,

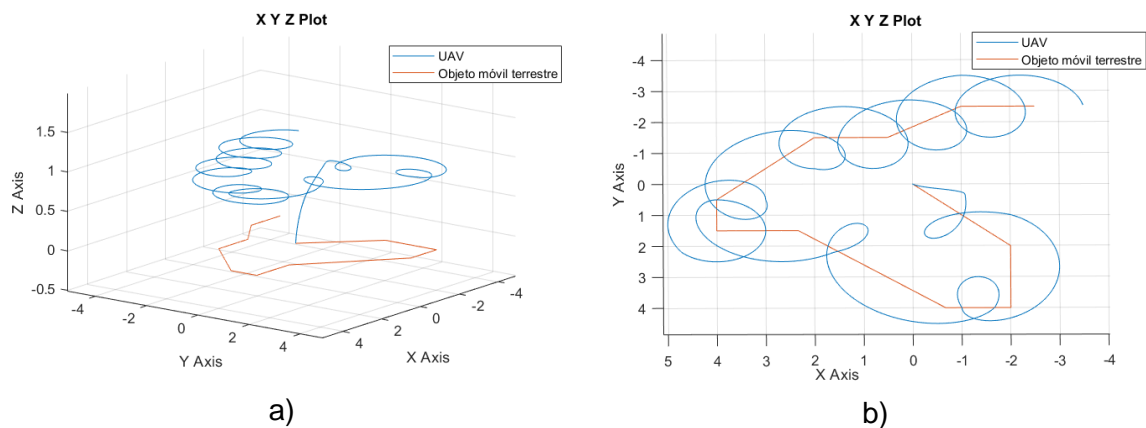
la acción de control generada para el Thrust se mantiene estable en la mayor parte de la trayectoria, y en las acciones de control de roll y pitch es donde más se puede notar los cambios realizados para los ejes  $x,y$ . Analizando las gráficas en conjunto, se puede determinar que el control PD implementado funciona de manera correcta.

### 3.2.3.2 Prueba con controlador basado en Lyapunov

En la Figura 3.31 se observa el resultado de la trayectoria realizada por el dron, con controladores basados en Lyapunov para los ejes  $x,y,z$ , al realizar la circunnavegación del objeto móvil que sigue una trayectoria predefinida por las señales de la Figura 3.27.



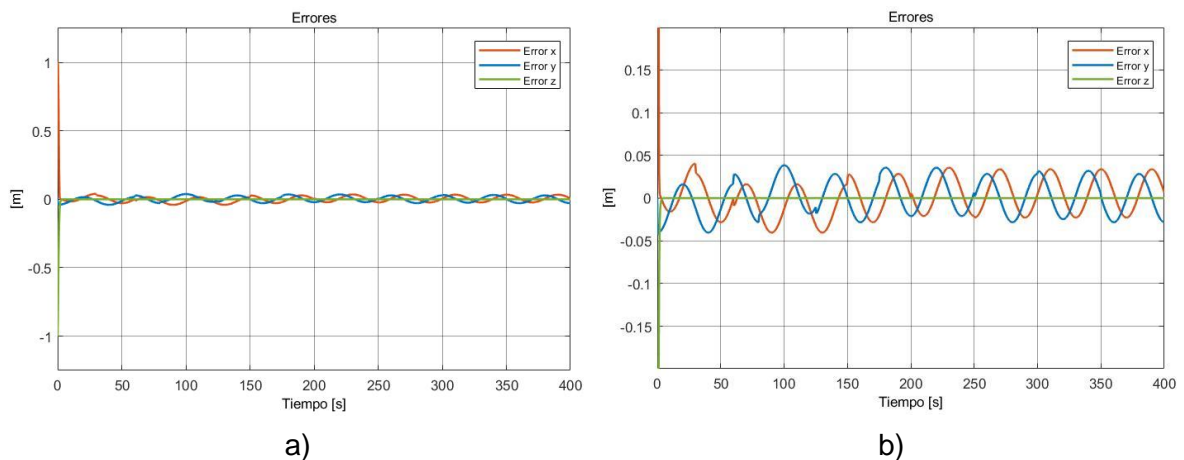
**Figura 3.31.** Gráfica 3D de la trayectoria del dron con control basado en Lyapunov – Circunnavegación a una trayectoria libre



**Figura 3.32.** Gráficas de las trayectorias del dron con control basado en Lyapunov y el objeto móvil terrestre – Circunnavegación a una trayectoria libre: a) Gráfica 3D; b) Vista aérea

Se presenta en la Figura 3.32, una vista isométrica y una vista aérea del desplazamiento del dron y el objeto móvil para tener una mejor visualización del seguimiento.

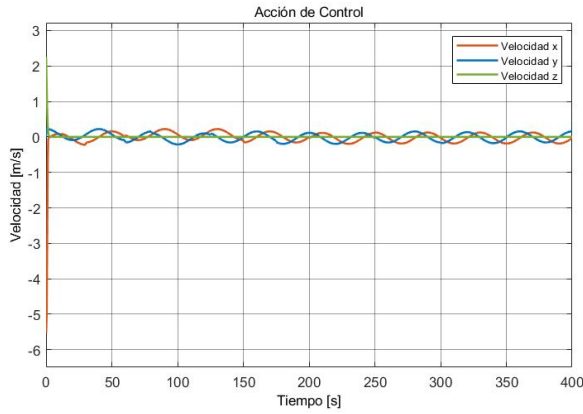
En la Figura 3.33.a y b, se muestran los errores de posición de  $x,y,z$ . En el eje  $z$  se puede apreciar que el error tiende a cero cuando se estabiliza después de generar un pico inicial. Los errores para los otros dos ejes muestran una sinusoidal con cambios en su offset, mismos que muestran el cambio de referencia de la trayectoria predefinida que sigue el objeto móvil. A diferencia del anterior controlador, en esta prueba no se ven picos en la trayectoria.



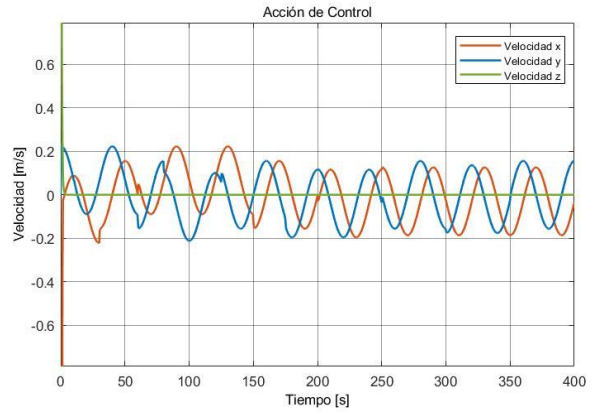
**Figura 3.33.** Resultados de trayectoria libre de circunnavegación con control basado en Lyapunov: a) Gráfica de errores  $x,y,z$ ; b) Gráfica de errores  $x,y,z$  ampliada

En la Figura 3.34 se presentan las acciones de control tanto de velocidad como de posición. En la Figura 3.34.a y b, donde se muestran las acciones de velocidad de  $x,y,z$ , se encuentran las señales sinusoidales contrarias a lo visto en el error de posición de  $x,y,z$ ; es decir, señales continuas con cambios de offset. En la Figura 3.34.c y d, se pueden ver las acciones de control generadas para roll y pitch. En la Figura 3.34.e y f, el Thrust muestra una variación pequeña e instantánea en cada uno de los cambios de referencia, considerando así que el dron se mantiene estable en toda la trayectoria. Estas variaciones son casi imperceptibles incluso con la imagen ampliada.

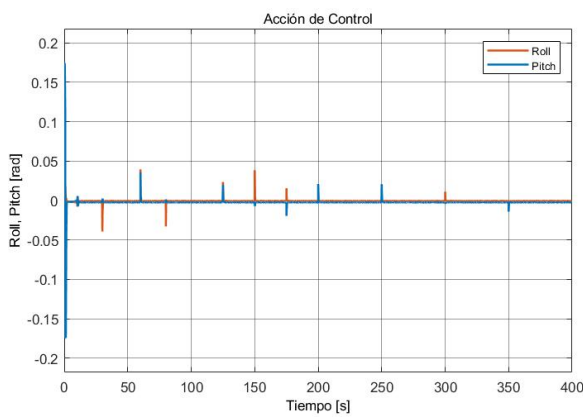




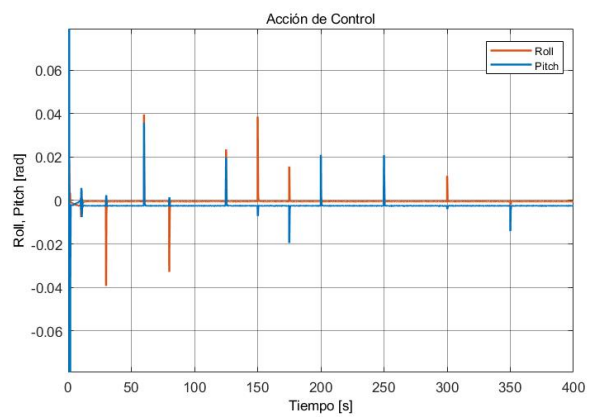
a)



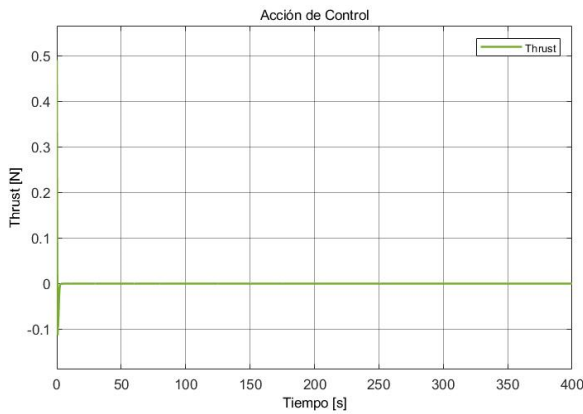
b)



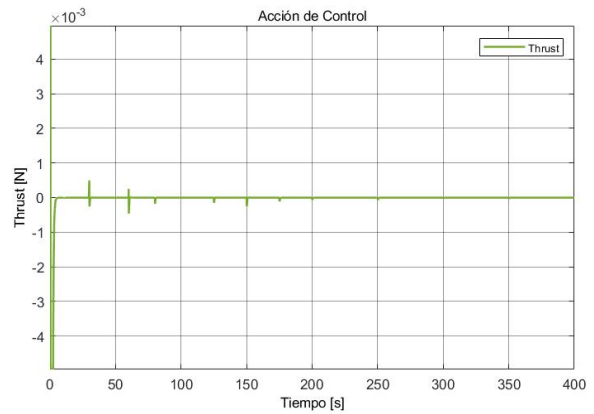
c)



d)



e)



f)

**Figura 3.34.** Resultados de trayectoria libre de circunnavegación con control basado en Lyapunov: a) Gráfica de acciones de control de velocidad de  $x,y,z$ ; b) Gráfica de acciones de control de velocidad de  $x,y,z$  ampliada; c) Gráfica de acciones de control  $x,y$ ; d) Gráfica de acciones de control de  $x,y$  ampliada; e) Gráfica de acción de control de  $z$ ; f) Gráfica de acción de control de  $z$

Analizando las gráficas en conjunto, se puede determinar que el control basado en Lyapunov implementado funciona de manera correcta.

### 3.2.3.3 Comparación de resultados

En la Tabla 3.6 se registraron los datos de las pruebas ejecutadas con ambos controladores para realizar una comparación específica tomando en consideración el desempeño de cada una. A diferencia de las pruebas anteriores donde el objeto móvil terrestre seguía la trayectoria de un cuadrado o un círculo, ahora se ejecuta una trayectoria libre; y, a pesar de eso, se obtienen resultados favorables.

**Tabla 3.6.** Comparación de índices de desempeño – Circunnavegación libre

Controlador	IAE			TVU		
	X	Y	Z	X	Y	Z
Tipo PID	1.195	1.116	2.022	2.854	0.9552	1.015
Basado en Lyapunov	8.169	7.375	0.7506	1.905	1.555	1.321

En esta prueba se obtiene como resultado que el controlador tipo PID tuvo un mejor desempeño al hacer circunnavegación en seguimiento a la trayectoria libre de un objeto móvil. Los resultados obtenidos en la tabla son bastante similares a los obtenidos en las anteriores pruebas de circunnavegación. Con esto, se puede asegurar que el funcionamiento del dron en circunnavegación a una trayectoria libre manejada por el usuario de manera manual será equiparable a los resultados obtenidos en esta Sección 3.2.3.

## 3.3 COMPARACIÓN DE CONTROLADORES

A lo largo de las pruebas realizadas se pudo confirmar que se obtuvieron resultados satisfactorios con ambos controladores, y se volvieron notorias ciertas características diferenciadoras de cada controlador implementado, específicas del desarrollo de este proyecto de titulación, las cuales se detallan a continuación:

El control basado en Lyapunov se realiza de manera ideal buscando la estabilidad del sistema mediante un punto de equilibrio, en cambio el control tipo PID se realiza mediante el establecimiento de parámetros de respuesta transitoria.

El controlador tipo PID propone una implementación más directa ya que logra realizar el control del sistema con un solo lazo, a diferencia del control basado en Lyapunov que requirió dos lazos de control para la misma tarea.

Se tomó en cuenta el modelo cinemático del dron para el control basado en Lyapunov sin necesidad de conocer las características físicas del dron, lo cual se vuelve una ventaja frente al controlador tipo PID, mismo que requirió los valores de inercia del cuadricóptero.

En el controlador tipo PID se tiene un método de calibración más selectivo debido a las diferentes ganancias que lo componen con las que se puede especificar los parámetros de diseño requeridos para la respuesta transitoria del sistema. Mientras que, en el controlador basado en Lyapunov se define únicamente el parámetro de estabilidad del sistema y su calibración requiere de pruebas heurísticas para llegar a establecer valores adecuados.

Al observar las gráficas de los resultados de las trayectorias seguidas por el dron, cada controlador demostró de manera satisfactoria el cumplimiento del objetivo final del seguimiento a un objeto móvil terrestre mediante circunnavegación, sin variaciones notorias en un entorno macro, siendo los índices de rendimiento propuestos los diferenciadores del trabajo efectuado por cada controlador. Donde, el esfuerzo del controlador tipo PID generó valores menores frente al del controlador basado en Lyapunov, lo que implica que tiene una acción de control menos brusca en cada cambio de referencia.

## 4. CONCLUSIONES Y RECOMENDACIONES

### 4.1 CONCLUSIONES

- En el capítulo referente al DESARROLLO DE CONTROLADORES, se evidenció la diferencia en el diseño de los esquemas de control sugeridos aplicados al cuadricóptero disponible en el UAV Toolbox. Para el controlador tipo PID, se realizó de dos formas; en el control de altura, se debió linealizar el modelo para, posteriormente, diseñar el controlador, y en el control de roll y pitch, se obtuvo un diseño de manera heurística. Para el control basado en Lyapunov, se definió el dron como una partícula y se diseñó el controlador tomando en cuenta las fuerzas aplicadas al UAV, buscando una estabilización del sistema. Además, se hizo evidente que el control del ángulo de dirección o heading del cuadricóptero venía incluido internamente en el bloque utilizado, por lo que no fue necesario un control como tal.
- A pesar de que en el controlador tipo PID se utilizó un solo lazo de control, en el controlador basado en Lyapunov fue necesario generar un control en cascada; es decir, un control interno de velocidad y, uno externo de posición. Esto debido a la naturaleza del sistema y las condiciones de estabilidad de este controlador.
- Dado que el modelo de un cuadricóptero es inestable, para el seguimiento de un objeto móvil terrestre mediante circunnavegación se determinó a través de los índices de rendimiento IAE y TVU que se obtiene un mejor funcionamiento cuando los movimientos son menos bruscos, ya que los controladores muestran un menor esfuerzo en el seguimiento de un objeto móvil con la trayectoria de un círculo que la de un cuadrado.
- Se desarrolló una interfaz gráfica que permite el manejo tanto del cuadricóptero como de un objeto móvil terrestre y la visualización del comportamiento del sistema con los diferentes controladores diseñados.
- Obteniendo los índices de rendimiento se puede llegar a definir el controlador más adecuado para un diferente fin, donde se priorice el grado de exactitud o la acción del controlador. En el desarrollo de este trabajo, ambos controladores muestran un resultado satisfactorio en el objetivo del proyecto, sin realizar un enfoque específico en la acción de control de cada subsistema.
- Mediante el índice de desempeño del IAE obtenido al realizar pruebas para verificar el funcionamiento de los controladores diseñados, se corroboró que todas las

trayectorias planteadas fueron posibles de ejecutar con un error bastante cercano a cero con los parámetros obtenidos en las diferentes sintonizaciones.

- Con el fin de tener un acercamiento más realista, se consideró implementar saturadores con valores máximos y mínimos a los controladores desarrollados debido a las características que podría presentar un cuadricóptero real. En ambos controladores se utilizaron saturadores en los ángulos de roll y pitch, y en el Thrust; y en el caso de Lyapunov, uno de velocidad.
- Cada controlador aporta con características diferentes a los resultados de cada una de las pruebas realizadas; en pruebas distintas, un controlador podría presentar mejores resultados debido a su implementación y parametrización dependiendo de la aplicación en la que sean utilizados; sin embargo, ambos controladores presentaron resultados favorables en la aplicación propuesta en este trabajo de titulación.
- Se recopiló información relevante sobre el funcionamiento de cuadricópteros, el manejo del UAV Toolbox de Simulink, la tarea de circunnavegación, y la implementación de los controladores tipo PID y basado en Lyapunov.
- Se logró realizar la tarea de circunnavegar a un objeto móvil terrestre con el cuadricóptero del UAV Toolbox, y se diseñó un algoritmo de control necesario que permite tener una dirección o heading orientado a la visualización del objeto móvil terrestre.

## **4.2 RECOMENDACIONES**

- Antes de utilizar los bloques del UAV Toolbox implementados en este proyecto de titulación, es necesario revisar si la capacidad del computador a usar concuerda con los requerimientos planteados por MATLAB. A su vez, se debe tomar en cuenta que la versión de MATLAB utilizada sea igual o superior a la 2020a.
- Tener en cuenta que el trabajo realizado tiene como objetivo circunnavegar sobre un objeto móvil y en un escenario controlado, por lo cual, si se desea implementar en un cuadricóptero real, se debe revisar los parámetros del dron y sintonizar nuevamente los controladores.
- En futuros trabajos relacionados a este proyecto, se recomienda combinar los controladores tipo PID y basado en Lyapunov dependiendo del movimiento que se quiera controlar. Específicamente, se recomienda implementar el controlador basado

en Lyapunov para la altura, y el controlador tipo PID, en los movimientos de roll y pitch.

- Se recomienda siempre usar los parámetros establecidos en cada uno de los controladores del UAV, ya que se ha realizado un procedimiento de calibración tomando los índices de desempeño, y entender cómo afecta cada parámetro a la simulación.
- Para trabajos futuros se podría implementar la simulación 3D del UAV Toolbox, que muestra un entorno más aproximado a la realidad. Si se tienen las capacidades de la computadora y los requerimientos para el correcto funcionamiento del Toolbox, y no funciona esta aplicación, se recomienda iniciar el programa de MATLAB con permisos de Administrador.

## REFERENCIAS BIBLIOGRÁFICAS

- [1] AerialInsights, «¿Cuántos tipos de drones existen en el mercado?,» 27 Octubre 2017. [En línea]. Available: <https://www.aerial-insights.co/blog/tipos-de-drones/>.
- [2] M. Rouse, «drone (UAV),» 7 July 2019. [En línea]. Available: <https://internetofthingsagenda.techtarget.com/definition/drone>.
- [3] P. Corke, Robotics, Vision and Control, 2nd ed., B. Siciliano y O. Khatib, Edits., Springer International Publishing, 2017.
- [4] J. R. Cherukuri y A. Padmanabhan, «Quadcopter,» 2 Marzo 2022. [En línea]. Available: <https://devopedia.org/quadcopter>.
- [5] Inteligencia DYNAMICS, «Aplicaciones y usos,» 2012. [En línea]. Available: [http://www.iuavs.com/pages/aplicaciones\\_y\\_usos](http://www.iuavs.com/pages/aplicaciones_y_usos).
- [6] Y. Cao, J. Muse, D. Casbeer y D. Kingston, «Circumnavigation of an Unknown Target Using UAVs with Range and Range Rate Measurements,» 28 Agosto 2013. [En línea]. Available: <https://ieeexplore.ieee.org/document/6760439>.
- [7] S. Daley, «Drones. What Is A Drone? What Are Uses For Drones?,» 10 Febrero 2022. [En línea]. Available: <https://builtin.com/drones>.
- [8] M. Simon, «The WIRED Guide to Robots,» 16 Abril 2020. [En línea]. Available: <https://www.wired.com/story/wired-guide-to-robots/>.
- [9] NovoDrone, «Tipos de drones,» Agosto 2018. [En línea]. Available: <https://novodrone.com/tipos-de-drones/>.
- [10] RCTecnic, «¿Qué es un drone? Tipos, nombres y componentes,» 16 Noviembre 2017. [En línea]. Available: <https://juguetecnic.com/blog/que-es-un-drone-tipos-nombres-componentes.html>.
- [11] 911Security, «Types of Drones - Lighter than Air,» Octubre 2017. [En línea]. Available: <https://www.911security.com/learn/airspace-security/drone-fundamentals/types-of-drones-lighter-than-air>.
- [12] 911Security, «Types of Drones - Flapping Wing,» Octubre 2017. [En línea]. Available: <https://www.911security.com/learn/airspace-security/drone-fundamentals/types-of-drones-flapping-wing>.
- [13] ÁreaTecnología, «Drones y tipos de drones,» 30 Julio 2014. [En línea]. Available: <https://www.areatecnologia.com/aparatos-electronicos/drones.html>.

- [14] L. Verdad, «Los drones y su innovación en áreas profesionales,» 12 Noviembre 2018. [En línea]. Available: <https://laverdadnoticias.com/innovacion/Los-drones-y-su-innovacion-en-areas-profesionales-20181108-0058.html>.
- [15] C. Luis, «Design of a Trajectory Tracking Controller for a Nanoquadcopter,» 20 Agosto 2016. [En línea]. Available: <https://arxiv.org/pdf/1608.05786.pdf>.
- [16] E. Cortez, «Control de orientación y altitud de un vehículo aéreo no tripulado, del tipo cuadirrotor,» Octubre 2016. [En línea]. Available: <http://eprints.uanl.mx/15769/1/1080237882.pdf>.
- [17] R. A. Amir Hussein, «Autopilot Design for a Quadcopter,» Octubre 2017. [En línea]. Available: [https://www.researchgate.net/publication/331298873\\_Autopilot\\_Design\\_for\\_a\\_Quadcopter](https://www.researchgate.net/publication/331298873_Autopilot_Design_for_a_Quadcopter).
- [18] F. A. V. D. L. T. Edwin Dimitri Nieto Guerrero, «Desarrollo de un modelo matemático, cinemático y dinámico con la aplicación de software, para modificar el funcionamiento de un dron, para que este realice monitoreo automático,» RECIAMUC, 5 Marzo 2020. [En línea]. Available: <http://recimundo.com/index.php/es/article/view/814>.
- [19] UAVCoach, «How to Fly a Drone - A Beginner's Guide to Multicopter Systems & Flight Proficiency,» 5 Octubre 2021. [En línea]. Available: <https://uavcoach.com/how-to-fly-a-quadcopter-guide/>.
- [20] Mathworks, «Guidance Model,» Febrero 2022. [En línea]. Available: <https://la.mathworks.com/help/uav/ref/guidancemodel.html>.
- [21] R. Rico, P. Maisterra, M. Gil-Martínez, J. Rico-Azagra y S. Nájera, «Identificación experimental de los parámetros de un cuadirrotor,» 2 Septiembre 2015. [En línea]. Available: <https://www.ehu.es/documents/3444171/4484750/131.pdf>.
- [22] N. Xuan-Mung y S.-K. Hong, «Improved Altitude Control Algorithm for Quadcopter Unmanned Aerial Vehicles,» 24 Mayo 2019. [En línea]. Available: <https://doi.org/10.3390/app9102122>.
- [23] L. E. Romero Paredes, «Diseño y construcción de un módulo de control PID para la estabilización de un cuadricóptero,» Septiembre 2014. [En línea]. Available: <https://bibdigital.epn.edu.ec/bitstream/15000/8644/3/CD-5814.pdf>.
- [24] O. Liang, «Quadcopter PID Explained,» 11 Febrero 2021. [En línea]. Available: <https://oscarliang.com/quadcopter-pid-explained-tuning/>.



- [25] P. E. Samaniego Villacrés y E. A. Vaca Cerda, «Diseño y simulación de dos estrategias de control basado en espacio nulo y tipo PID para una formación de cuadricópteros,» 14 Febrero 2017. [En línea]. Available: <https://bibdigital.epn.edu.ec/handle/15000/17058>.
- [26] V. Mazzone, «Control Automático II - Apuntes, Capítulo 5,» Agosto 2018. [En línea]. Available: <https://controlautomaticomazzone.files.wordpress.com/2018/08/cap5.pdf>.
- [27] «DISEÑO DE CONTROLADORES ADAPTATIVOS PARA CONTROLAR LA POSICIÓN DE UN ROBOT: TEORÍA Y SIMULACIÓN.,» 10 Septiembre 2008. [En línea]. Available: [http://sappi.ipn.mx/cgpi/archivos\\_anexo/20080910\\_5675.pdf](http://sappi.ipn.mx/cgpi/archivos_anexo/20080910_5675.pdf).
- [28] S. Gutiérrez Bermejo, «Modelado, simulación y control inteligente de UAVs,» Septiembre 2013. [En línea]. Available: [https://biblus.us.es/bibing/proyectos/abreproy/60178/fichero/PFC\\_completo\\_Sara\\_Gutierrez\\_Bermejo.pdf](https://biblus.us.es/bibing/proyectos/abreproy/60178/fichero/PFC_completo_Sara_Gutierrez_Bermejo.pdf).
- [29] Wikipedia, «Circunnavegación,» 1 Octubre 2021. [En línea]. Available: <https://es.wikipedia.org/wiki/Circunnavegaci%C3%B3n>.
- [30] F. Noto, R. George y J. Antoniou, «Developing UAV Applications with MATLAB and Simulink,» 15 Junio 2021. [En línea]. Available: <https://la.mathworks.com/videos/developing-uav-applications-with-matlab-and-simulink-1624305602400.html>.
- [31] MathWorks, «Simulink Block Diagrams,» 28 Enero 2020. [En línea]. Available: <https://la.mathworks.com/help/simulink/gs/simulink-block-diagrams.html>.
- [32] MathWorks, «Complementos,» 28 Enero 2020. [En línea]. Available: <https://la.mathworks.com/help/matlab/add-ons.html>.
- [33] MathWorks, «UAV Toolbox,» Septiembre 2021. [En línea]. Available: [https://la.mathworks.com/help/uav/index.html?s\\_tid=CRUX\\_topnav](https://la.mathworks.com/help/uav/index.html?s_tid=CRUX_topnav).
- [34] M. T. Calderón Jácome y D. J. Maldonado Andrade, «Control por visión de un cuadricóptero utilizando ROS,» 20 Octubre 2014. [En línea]. Available: <https://bibdigital.epn.edu.ec/bitstream/15000/8723/3/CD-5862.pdf>.
- [35] PowerVision, «PowerEye Product Specifications,» 21 Diciembre 2017. [En línea]. Available: <https://www.powervision.me/en/product/powereye/specs>.
- [36] DJI, «DJI Mini 2,» 5 Noviembre 2020. [En línea]. Available: <https://www.dji.com/mini-2/specs>.

- [37] M. A. Golato, «Análisis de respuestas transitorias, sistemas de segundo orden,» 4 Abril 2016. [En línea]. Available: [https://catedras.facet.unt.edu.ar/sistemasdecontrol/wp-content/uploads/sites/101/2016/04/4\\_An%C3%A1lisis-de-Respuestas-Transitorias\\_Sistemas-de-2do-Orden\\_2016-1.pdf](https://catedras.facet.unt.edu.ar/sistemasdecontrol/wp-content/uploads/sites/101/2016/04/4_An%C3%A1lisis-de-Respuestas-Transitorias_Sistemas-de-2do-Orden_2016-1.pdf).
- [38] K. Bruns, «Arc Tangent 2 Function,» 2022. [En línea]. Available: <https://wumbo.net/function/arc-tangent-2/>.
- [39] Mathworks, «Unwrap,» 28 Enero 2020. [En línea]. Available: [https://la.mathworks.com/help/dsp/ref/unwrap.html#mw\\_6964e809-11b4-4753-af1a-2b805e83102f](https://la.mathworks.com/help/dsp/ref/unwrap.html#mw_6964e809-11b4-4753-af1a-2b805e83102f).
- [40] Mathworks, «UAV Package Delivery,» Febrero 2022. [En línea]. Available: <https://la.mathworks.com/help/uav/ug/uav-package-delivery.html>.
- [41] HeraldDeAragon, «Cinco avances tecnológicos importantes en la historia,» 22 Septiembre 2020. [En línea]. Available: <https://www.heraldo.es/noticias/sociedad/2017/11/25/5-avances-tecnologicos-importantes-historia-1206915-310.html>.

# ANEXOS

## ANEXO A

### A. MANUAL DE INSTALACIÓN DEL UAV TOOLBOX

#### A.1. INTRODUCCIÓN

Para el funcionamiento óptimo del trabajo de titulación desarrollado, es necesario que se revisen los requerimientos previos que necesita el UAV Toolbox y las especificaciones técnicas mínimas que debería tener el computador donde se implemente esta herramienta.

#### A.2. REQUERIMIENTOS GENERALES

Para el uso de la herramienta, se tienen los siguientes requerimientos generales:

- MATLAB: El toolbox fue incorporado a partir de la versión 2020b, por lo cual se sugiere instalar esa versión o una posterior.
- Windows: Si bien el toolbox permite su uso en Mac, Linux y Windows, la simulación y visualización basada en el software Unreal Engine de Epic Games se ejecuta solamente en Windows; para identificar los bloques que requieren el software mencionado, ver la Sección 2.2.3. Para mayor detalle, visitar el siguiente enlace: <https://la.mathworks.com/help/uav/ug/3d-visualization-engine-requirements.html>.

#### A.3. EXTENSIONES DE SIMULINK

Dentro del software de Simulink es necesario instalar diferentes extensiones para poder ejecutar la simulación presentada en este trabajo, las cuales se enlistan a continuación:

- UAV Toolbox: Permite el uso del UAV Guidance Model y las herramientas relacionadas al UAV. (Figura A.1)
- Aerospace Blockset: Se encuentra relacionado con el UAV Toolbox, además, se utilizó uno de sus bloques de conversión. (Figura A.2)
- Control System Toolbox: Es necesario para el control PID utilizado en el proyecto. (Figura A.3)
- Symbolic Math Toolbox: Se usa para algunas funciones utilizadas para las trayectorias implementadas. (Figura A.4)

- DSP System Toolbox: Provee un bloque de Unwrap que permite modificar la señal por periodos. (Figura A.5)

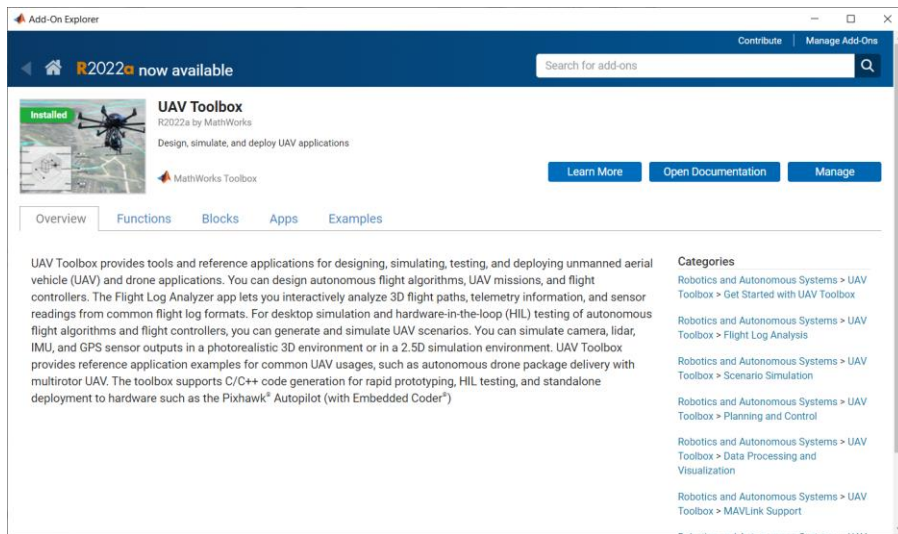


Figura A.1. UAV Toolbox Add-On

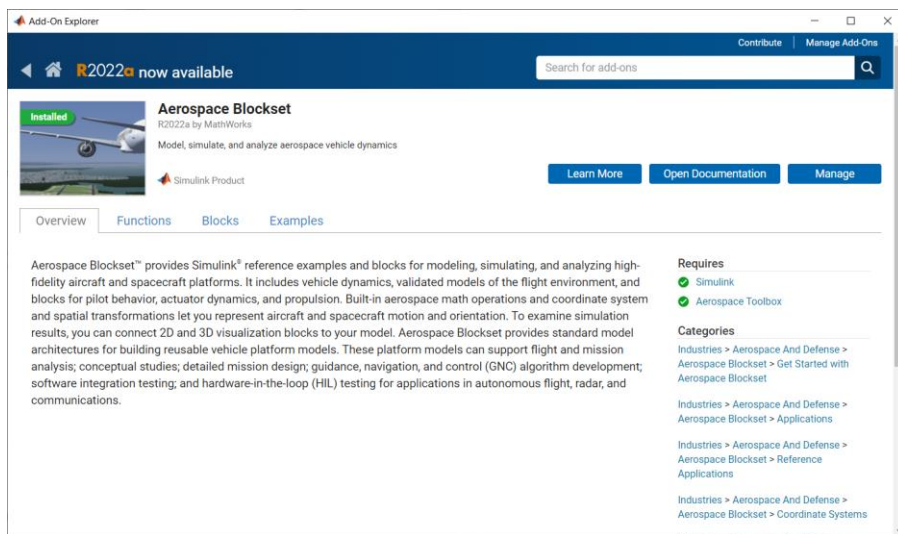


Figura A.2. Aerospace Blockset Add-On

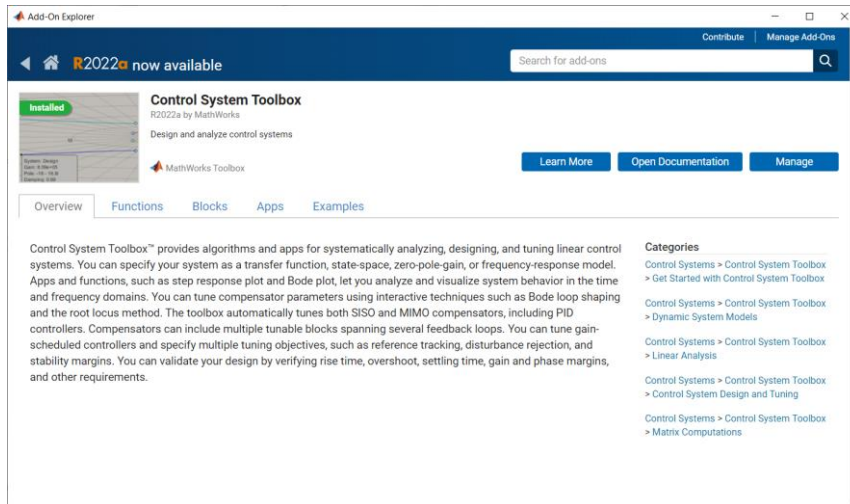


Figura A.3. Control System Toolbox Add-On

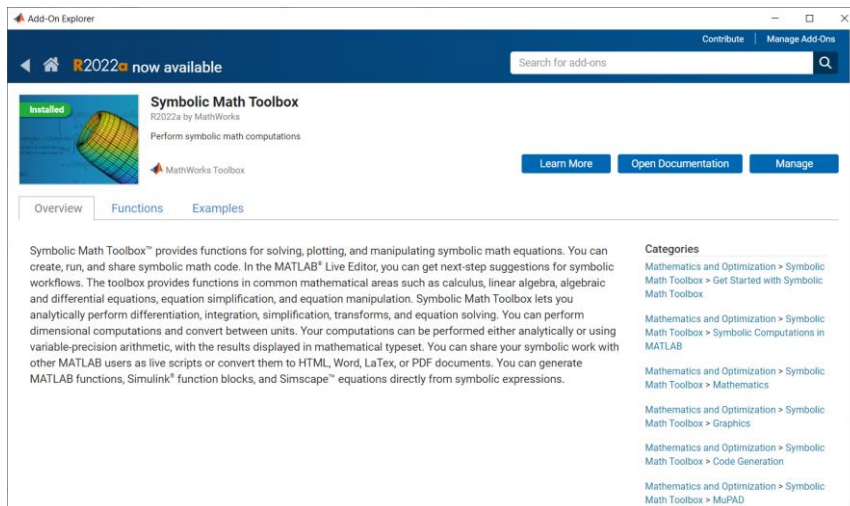


Figura A.4. Symbolic Math Toolbox

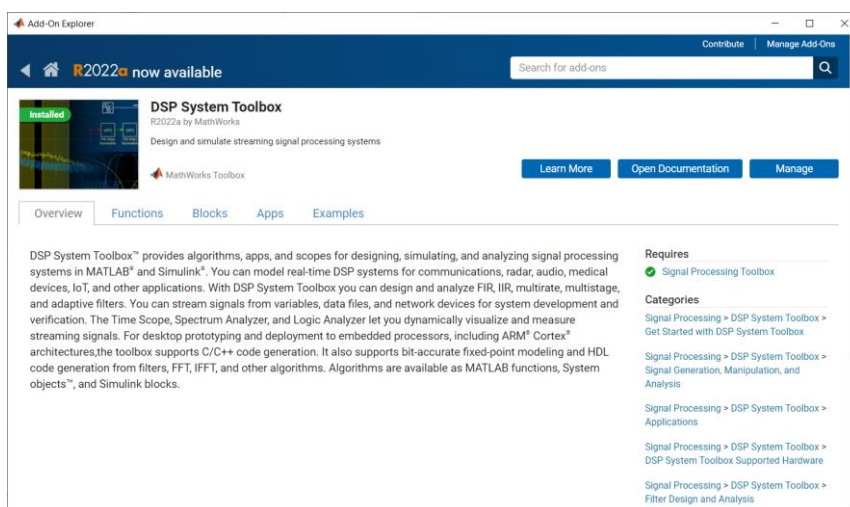


Figura A.5. DSP System Toolbox Add-On

## ANEXO B

### B. MANUAL DE USUARIO DE LA INTERFAZ GRÁFICA

La interfaz gráfica de este trabajo de titulación ha sido desarrollada en App Designer de Matlab, en su versión 2021a, para el control del cuadirrotor disponible en la herramienta UAV Toolbox de Simulink para efectuar el seguimiento de un objeto móvil terrestre usando circunnavegación. La interfaz permite elegir la trayectoria que el dron va a seguir, entre una trayectoria predeterminada o la circunnavegación a un objeto móvil; seleccionar el tipo de controlador a utilizar, al igual que modificar cada uno de los parámetros de cada controlador implementado; visualizar los resultados de la simulación, como son la ejecución, errores, señales de control e índices de desempeño.

#### B.1. INICIALIZACIÓN

Para ejecutar la interfaz desarrollada, ubicar la dirección donde se encuentra la carpeta “Interfaz Tesis”, la cual lleva los archivos necesarios para el funcionamiento de la simulación como se muestra en el recuadro naranja de la Figura B.1. Dentro de la carpeta, dar clic en el archivo “caratula.mlapp” que se encuentra señalado con un recuadro azul en la Figura B.1.

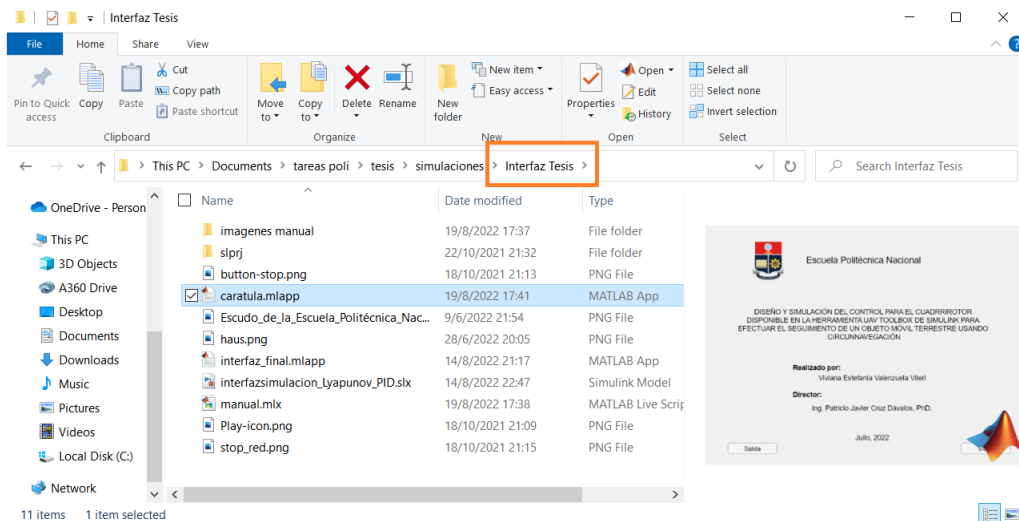


Figura B.1. Inicialización de la HMI desarrollada en App Designer

Una vez dado clic en el archivo mencionado, se abrirá la portada del HMI.

#### B.2. PORTADA DE LA INTERFAZ GRÁFICA

Esta pantalla contiene la presentación del proyecto de titulación con dos botones: botón de “Menú” y botón de “Salida” (Figura B.2).

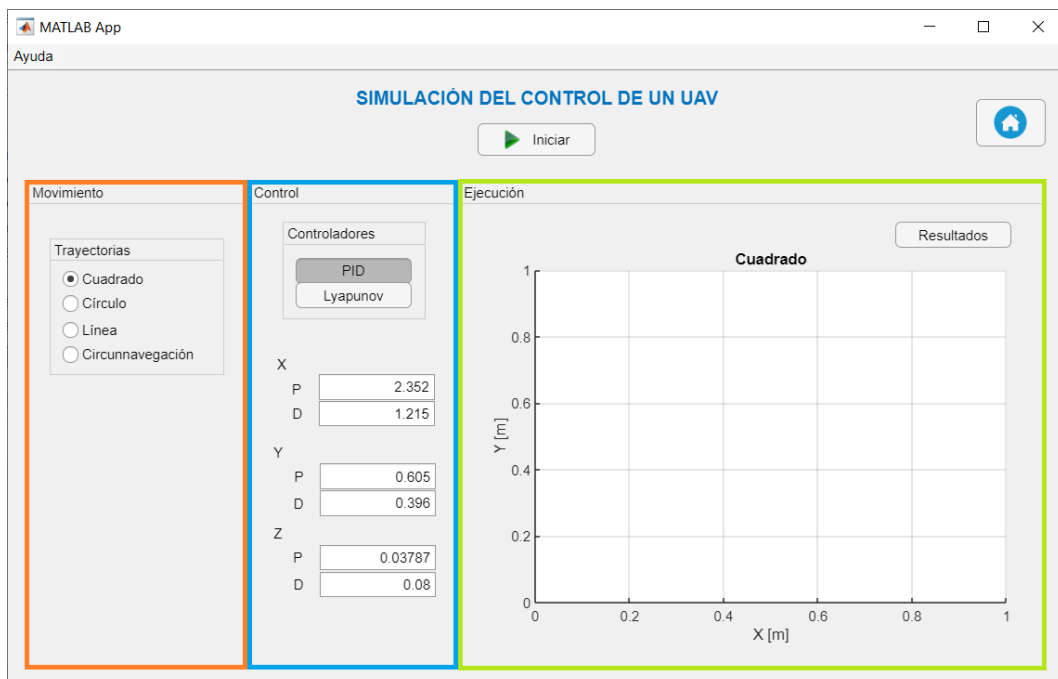


**Figura B.2.** Pantalla de bienvenida a la interfaz

El botón de “Salida” cierra la ventana y termina la ejecución de la interfaz gráfica, mientras que al presionar el botón de “Menú”, se desplegará la pantalla principal del control del UAV.

### B.3. PANTALLA PRINCIPAL

En la Figura B.3 se presenta la pantalla principal de la interfaz gráfica desarrollada, la cual se divide en 3 secciones principales: Movimiento, Control y Resultados.



**Figura B.3.** Pantalla principal de la interfaz gráfica

### B.3.1. MOVIMIENTO

La sección de Movimiento presenta las diferentes opciones que se pueden visualizar en la Figura B.4, donde, si se selecciona la trayectoria de circunnavegación, se desplegará un menú de selección de trayectorias que sigue el objeto móvil terrestre. Además, en esta opción se puede modificar la altura a la que el UAV realizará la circunnavegación.

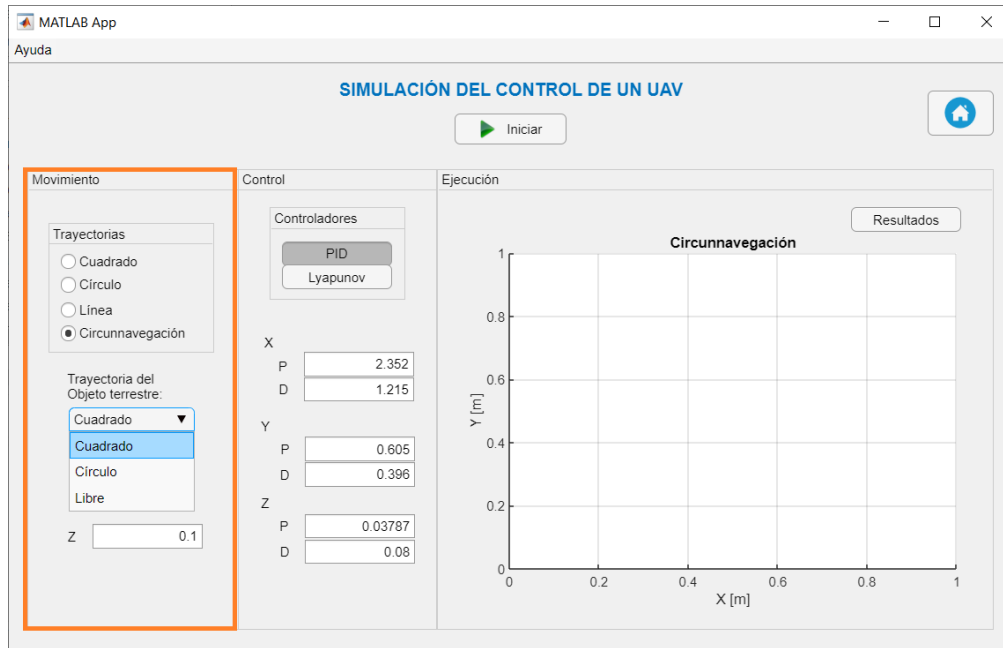


Figura B.4. Despliegue de todos los movimientos que puede realizar el UAV

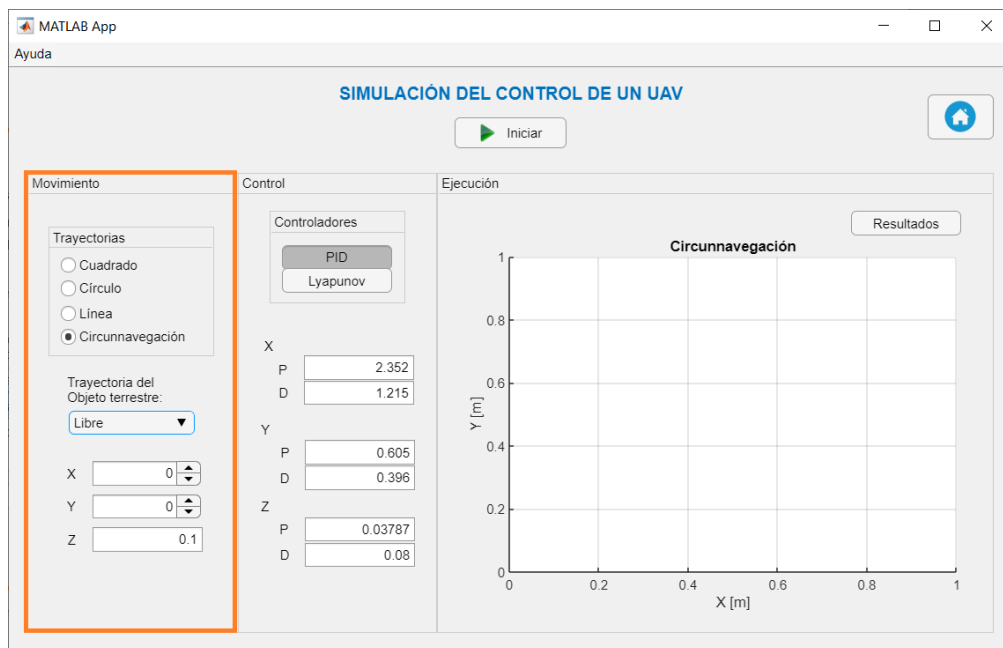


Figura B.5. Pantalla principal – Elección de circunnavegación del UAV siguiendo una trayectoria libre del objeto móvil terrestre

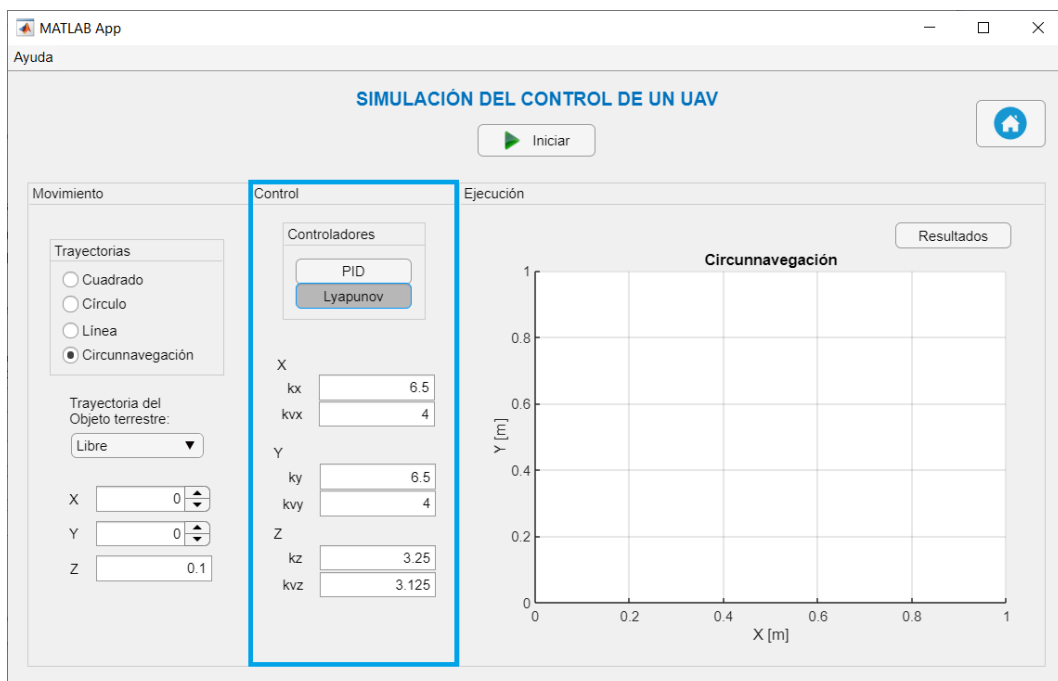


En el caso de seleccionar la circunnavegación a un objeto que se mueve libremente, se presentan dos recuadros adicionales donde se podrá manejar de manera manual el desplazamiento en el eje  $x$  y el eje  $y$  del objeto móvil terrestre, como se puede ver en la Figura B.5.

### B.3.2. CONTROL

En la sección de Control se presenta la selección de los dos controladores implementados: tipo PID y basado en Lyapunov. En cada uno de estos controladores se desglosan los parámetros respectivos sintonizados para las ganancias de cada eje; sin embargo, estos parámetros son modificables en un rango limitado con la finalidad de permitir al usuario entender el funcionamiento del UAV.

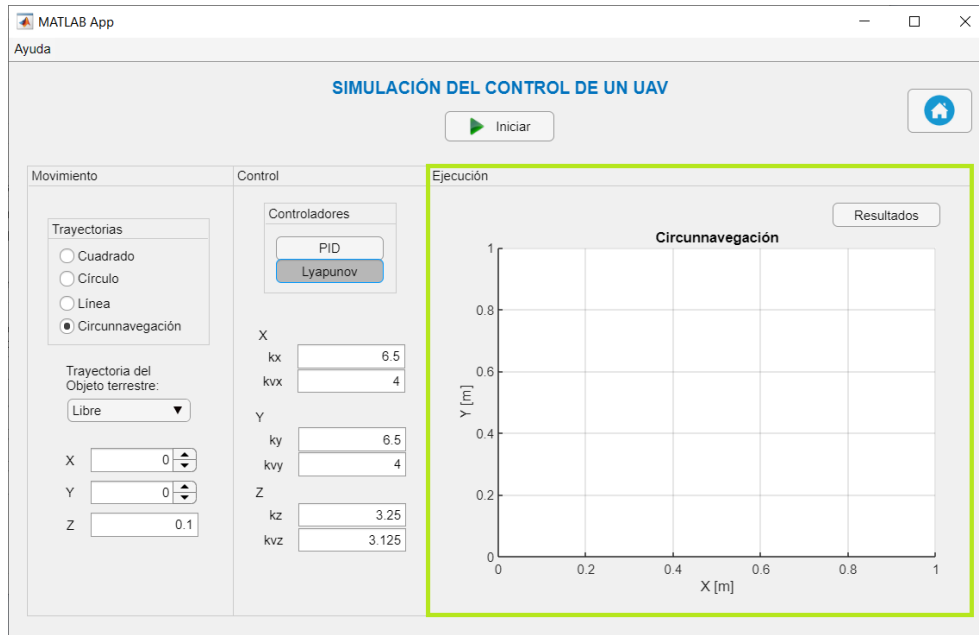
En la Figura B.6, se muestra el ejemplo de la selección del controlador basado en Lyapunov y el cambio a sus respectivos parámetros. En las anteriores figuras se puede observar la selección por default del controlador tipo PID.



**Figura B.6.** Pantalla principal – Selección del controlador basado en Lyapunov

### B.3.3. RESULTADOS

Al realizar la ejecución de la simulación presionando el botón “Iniciar”, se muestra la trayectoria en tiempo real que va realizando el UAV. En el panel de Ejecución, se presenta el desplazamiento del dron en una gráfica 3D (Figura B.7). Una vez terminada la simulación, se habilita el botón “Resultados” para su visualización en dos pestañas.



**Figura B.7.** Pantalla principal – Resultados de la ejecución

En la primera pestaña, Errores, se presenta una gráfica donde se visualizan los errores resultantes de cada uno de los ejes, así como los índices de desempeño IAE y TVU obtenidos durante la simulación al presionar el botón “Índices” (Figura B.8).



**Figura B.8.** Pantalla principal – Resultados de los errores

En la segunda pestaña de Acción de Control, se muestra una gráfica que varía según la selección de los botones presentados en la parte superior de la sección mostrada. Si se

selecciona el controlador basado en Lyapunov, se aumenta un botón de “Velocidad” debido a que tiene una acción de control adicional a la del control tipo PID (Figura B.9).

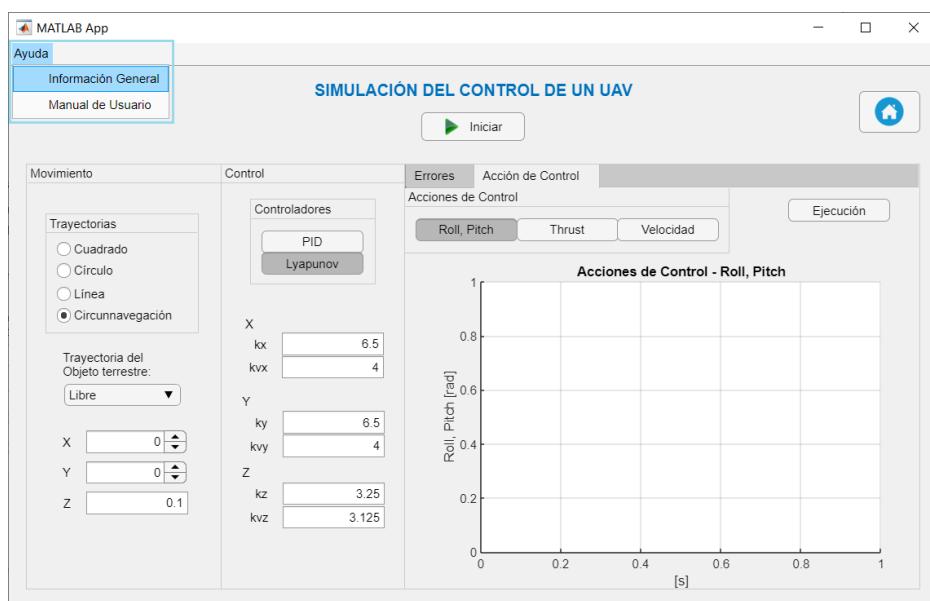


**Figura B.9.** Pantalla principal – Resultados de las acciones de control

En ambos casos, se tiene la opción de volver al panel de Ejecución al presionar el botón “Ejecución” que se encuentra en la parte superior derecha de cada pestaña.

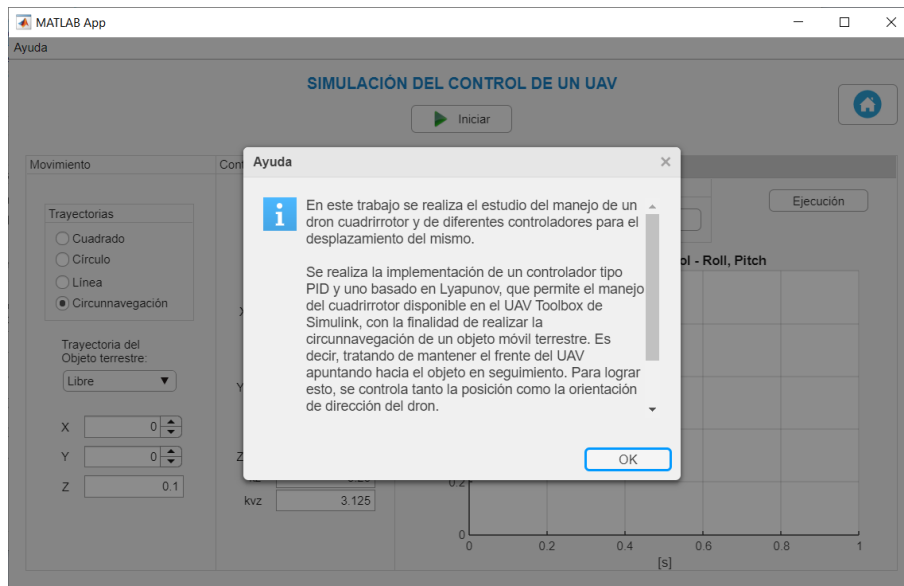
## B.4. AYUDA

En caso de ser necesario, en la Pantalla principal se presenta un botón de ayuda que despliega un submenú con dos botones adicionales, como se puede ver en la Figura B.10.



**Figura B.10.** Pantalla principal – Botón de Ayuda

Al seleccionar el botón “Información General”, se despliega una ventana emergente que resalta el contenido del trabajo de titulación presentada en la Figura B.11.



**Figura B.11.** Ventana de Información General

Al seleccionar el botón “Manual de Usuario”, se desplegará el manual con la información necesaria para hacer uso de la interfaz gráfica presentada en el proyecto de titulación (Figura B.12).



**Figura B.12.** Ventana de Manual de Usuario

## **ORDEN DE EMPASTADO**