

MODELADO Y CONTROL INTELIGENTE DEL CUATRIROTOR QUANSER QBALL-X4

Juan Alfonso Campo Lobato
Marco Cebrián Muiño
Fernando de la Mata de la Iglesia

PROYECTO DE SISTEMAS INFORMÁTICOS, FACULTAD DE INFORMÁTICA
UNIVERSIDAD COMPLUTENSE DE MADRID



Departamento de Arquitectura de Computadores y Automática (DACYA)

Curso 2011/2012

Directora: Matilde Santos Peñas
Director Colaborador: José Antonio Martín Hernández

Se autoriza a la Universidad Complutense a difundir y utilizar con fines académicos, no comerciales y mencionando expresamente a sus autores, tanto la propia memoria, como el código, los contenidos audiovisuales incluso si incluyen imágenes de los autores, la documentación y/o el prototipo desaralabeado

Juan Alfonso Campo
Lobato

Marco Cebrián Muiño

Fernando de la Mata de la
Iglesia

*Dedicamos este proyecto a la UCM,
a la Facultad de Informática y
a todos nuestros profesores
por convertirnos en lo que somos.*

Agradecimientos

Índice general

Agradecimientos.....	VII
Resumen.....	XIII
Summary.....	XIII
1.Descripción del proyecto.....	1
1.1 Motivación.....	1
1.2 Objetivos.....	1
1.3 Estructura de la memoria.....	2
2.Estado del arte.....	5
2.1 Vehículo.....	5
2.2 Controles.....	6
2.2.1 PID.....	7
2.2.2 Controles Inteligentes.....	8
3.Modelo.....	13
3.1 Estudio de las características del sistema.....	13
3.2 Modelado analítico.....	14
3.3 Identificación.....	15
3.4 Modelo Quanser QBall-X4 de J. M. de la Cruz 2011.....	16
3.5 Modelo de D. Sánchez 2009.....	17
3.6 Modificaciones y mejoras.....	19
3.6.1 Suelo real.....	19
3.6.2 Rozamiento con el suelo.....	21
4.Control del Quanser JMC-2011.....	25
4.1 Implementación.....	25
4.2 Resultados.....	28
4.2.1 Seguimiento de trayectorias dinámicas simples.....	28
4.2.2 Seguimiento de trayectorias compuestas.....	28
5.Control de DS-2009.....	31
5.1 PID.....	31
5.1.1 Implementación.....	32
5.1.2 Componentes del calculador inteligente.....	35
5.1.3 Análisis de resultados.....	35
5.2 Fuzzy.....	38
5.2.1 Control de la altura.....	38
5.2.2 Control del guiñada.....	41
5.2.3 Control del cabeceo.....	44
5.2.4 Control del alabeo.....	44
5.2.5 Análisis de resultados.....	45
5.3 Red neuronal.....	51
5.3.1 Neurocontrol de la altura.....	51
5.3.2 Implementación y entrenamiento de la red.....	51
5.3.3 Estudio de los resultados obtenidos.....	53
6.Plataforma de simulación.....	55
6.1 Interfaz 2D.....	55
6.2 Interfaz 3D.....	56
6.2.1 Modos.....	58
6.2.2 Ajustes.....	60
7.Conclusiones y trabajos futuros.....	63

Conclusiones.....	63
Trabajos futuros.....	64
8.Bibliografía.....	65
Anexo I: Especificaciones Técnicas del Qball-X4 UAV.....	67
Anexo II: Ecuaciones cinemáticas y dinámicas del cuatrirotor.....	67

Resumen

En este proyecto se ha desarrollado el modelado, diseño y aplicación de controladores inteligentes en un cuatrirotor, en concreto, el modelo Quanser Qball-X4. Este vehículo autónomo aéreo (UAV) es altamente inestable, por lo que para que su comportamiento sea eficiente debe estar bien controlado, una tarea que es compleja. En este proyecto se propone, en primer lugar, partiendo de modelos disponibles del vehículo aéreo, refinarlos para poder simularlo y estudiar su comportamiento. Tras esta fase, en la que se generan modelos mejorados, se plantea el diseño de controladores, tanto convencionales como inteligentes o una hibridación de ambos, con el objetivo de que despegue y aterrice en una plataforma, y que recorra una trayectoria. Además se dota de una visión tridimensional sencilla para observar los resultados.

Palabras clave: *cuatrirotor, modelado, control inteligente automática, quanser, UAV, PID, red neuronal, difuso*

Summary

System modeling, design and implementation of intelligent controlabeoers in a cuatrirotor, specifically, the model Quanser Qball-X4 . This autonomous air vehicle (UAV) is highly unstable, so in order to get an efficient behavior a good controlabeoer is needed, and that is a complex task. This project proposes the study and simulation of air vehicle models. After this phase, in which improved models are generated, the design of controlabeoers is considered, both conventional and intelligent or a hybrid form of both, with the goal to archieve a takeoff and landing from and on a platform, and a travel path. It also gives a simple three-dimensional view to observe results.

Key words: *cuatrirotor, modeling, intelligent control, quanser, UAV, PID, neural network, fuzzy*

1. Descripción del proyecto

1.1 Motivación

El proyecto consiste en el diseño y aplicación de controladores convencionales e inteligentes al modelo del cuatrirotor Quanser QBall-X4, para luego implementarlo en tiempo real sobre el prototipo del que se dispone en el Dpto. de Arquitectura de Computadores y Automática, en el grupo de Ingeniería de Sistemas y Automática. También se contempla la posibilidad de aplicar técnicas de la inteligencia artificial para la calibración y optimización del modelo si fuera necesario.

El control tiene como objetivo permitir que el vehículo despegue y aterrizaje en una plataforma que puede estar en movimiento y con perturbaciones, típicamente un barco, del cual se dispone de un modelo físico a escala y de un escenario real de operaciones con sus características intrínsecas de inestabilidad.

El cuatrirotor una de sus posibles aplicaciones en un área marítima podría ser desde localizar supervivientes hasta aumentar la visión espacial del exterior del barco. Sus ventajas son que al no necesitar tripulación, los costes en combustible y sistema de vuelo se reduciría de manera significativa, pueden actuar en ambientes de riesgo biológico y son reutilizables.

Por todo lo anterior, es un área muy interesante para la investigación y aplicación del control inteligente en un vehículo aéreo no tripulado.

1.2 Objetivos

- ✓ Estudio y mejora del modelo suministrado del cuatrirotor incorporando efectos dinámicos respecto al suelo, etc. para hacerlo más realista.
- ✓ Simulación y análisis de su comportamiento dinámico. Estudio de las limitaciones físicas en el seguimiento de trayectorias.
- ✓ Diseño y simulación de controladores convencionales e inteligentes, basados en lógica fuzzy para el modelo suministrado y del Quanser.

- ✓ Interfaz visual en 2D y 3D de las trayectorias.
- ✓ Mejora de la sintonía de los controladores.
- ✓ Implementación en el prototipo de los controladores para experimentación real.

1.3 Estructura de la memoria

La memoria consta de ocho capítulos dedicados a cada aspecto del proyecto. En los dos primeros se comenta las motivaciones y objetivos del proyecto, y la información de lo que hay hecho hasta ahora respecto al vehículo, modelo y controles por otros autores.

En los tres siguientes se detalla los modelos estudiados y su validación con sus respectivos diseños de controladores para cada uno. Es la parte más importante del proyecto y por eso se dedica capítulos enteros para entrar en profundidad.

En el punto 6 de Plataforma de simulación se describe el diseño de la interfaz visual 2D y 3D. También por su relevancia tiene su apartado propio.

Finalmente, las conclusiones acerca de los resultados obtenidos en este trabajo y las futuras ampliaciones para poder continuar la labor detallada en la introducción.

En la bibliografía expone todas las referencias de documentación usadas en estilo Harvard.

Como complemento se incluye unos anexos elaborados por entidades u otros autores para mejorar la comprensión de la totalidad del trabajo.

2. Estado del arte

2.1 Vehículo

El Quanser QBall-X4 es un moderno cuatrirotor, es decir, un vehículo aéreo tipo helicóptero de cuatro motores. Es un prototipo de UAV (*Unmanned autonomous vehicle*) que puede ser utilizado para gran variedad de aplicaciones. En su página oficial está disponible toda la información técnica del modelo¹. En la Figura 2.1 se muestra una fotografía del mismo.

El cuatrirotor tiene una cabina central y cuatro motores que alimentan cuatro hélices. Está protegido por una jaula de fibra de carbón, lo que permite dotarlo de un entorno de seguridad cuando se utiliza en un laboratorio, dentro de un edificio, encontrarse obstáculos u otros vehículos con lo que puede colisionar. Para más información consultar el Anexo I.

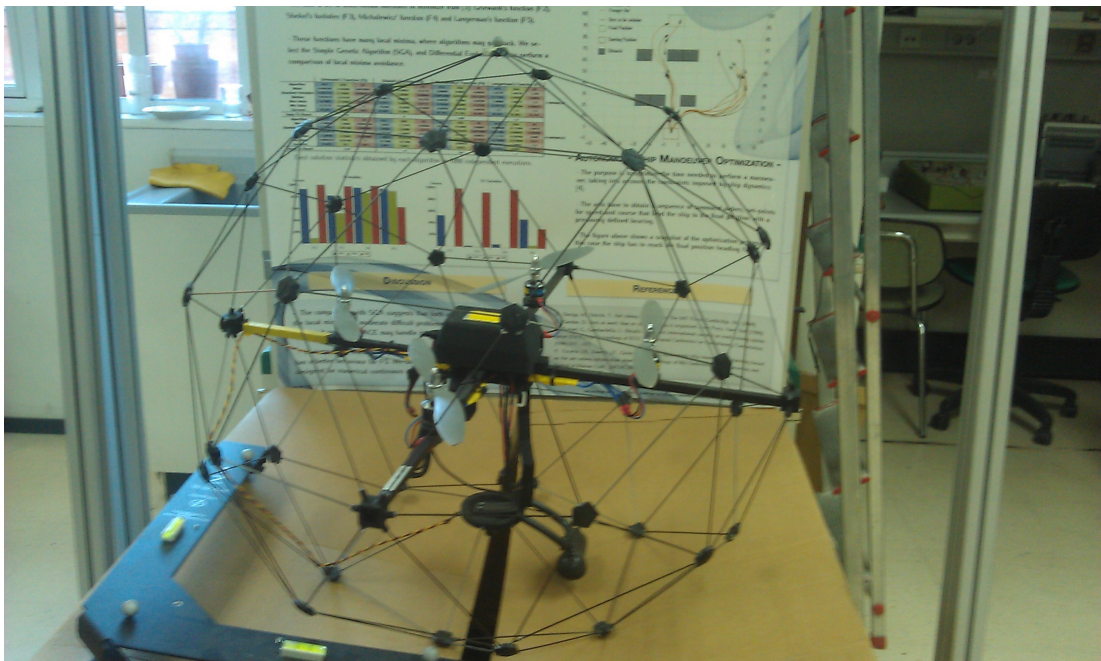


Figura 2.1: Quanser QBall-X4 disponible en la Facultad de Físicas de la UCM

¹ http://quanser.com/english/html/UVS_Lab/fs_Qball_X4.htm

Está también dotado de sensores y drivers para los motores, a los que se accede desde una placa de adquisición de datos (DAQ) con las correspondientes entradas/salidas que está incorporada al helicóptero. El módulo HiQ está embebido en el sistema, y permite medidas inerciales (IMU) con gran resolución.

Tradicionalmente los UAV han sido aviones que se pilotaban de forma remota, aunque actualmente esta tendencia está cambiando y cada vez más se está utilizando el control autónomo de los mismos. Hay que mencionar que realmente las aeronaves que son pilotadas remotamente están mal llamadas UAV, siendo su nombre, si se emplea la terminología correcta, Aeronaves R/C (radio controladas).

A día de hoy, los UAV militares realizan tanto misiones de ataque como de reconocimiento, y se han contabilizado numerosos casos exitosos de aplicaciones con estos tipos de aeronaves.

Los UAV también se utilizan en pequeño, pero creciente número de aplicaciones civiles, como prevención y lucha contra incendios, reconocimiento y toma de muestras en ambientes de alta toxicidad o radiactividad y ayuda en el rescate de náufragos.

Los UAV pueden ser clasificados por multitud de criterios, aunque normalmente se hacen por el tipo de misión, o por su tamaño y alcance.

2.2 Controles

En cuanto al control inteligente en UAV como se puede consultar en las bases de datos bibliográficas de Web of Science y de IEEE Xplore, no arroja más de 300 resultados². Esto implica que es un campo muy innovador frente a cualquier otra

² Consultado el 14/12/2011 con la búsqueda "intelligent control uav "

aplicación del modelado y control. Destacar que hasta ahora en la Web of Science no hay ningún artículo de control aplicado a un cuatrirotor.

En los capítulos siguientes se detalla dos modelos previos de referencia que son el modelo Quanser QBall-X4 de Jesús Manuel de la Cruz 2011 y el modelo de David Sánchez 2009.

2.2.1 PID

El controlador PID es el mecanismo de control más común. Basado en realimentación, calcula el error entre el valor medido y el valor que se quiere obtener. El algoritmo de control está fundamentado en tres parámetros: el proporcional, el integral y el derivativo. En la Figura 2.2 se muestra el diseño de dicho controlador (Åström & Hägglund, 2009.). Es el primer control y el básico que se usa como referencia para el proyecto.

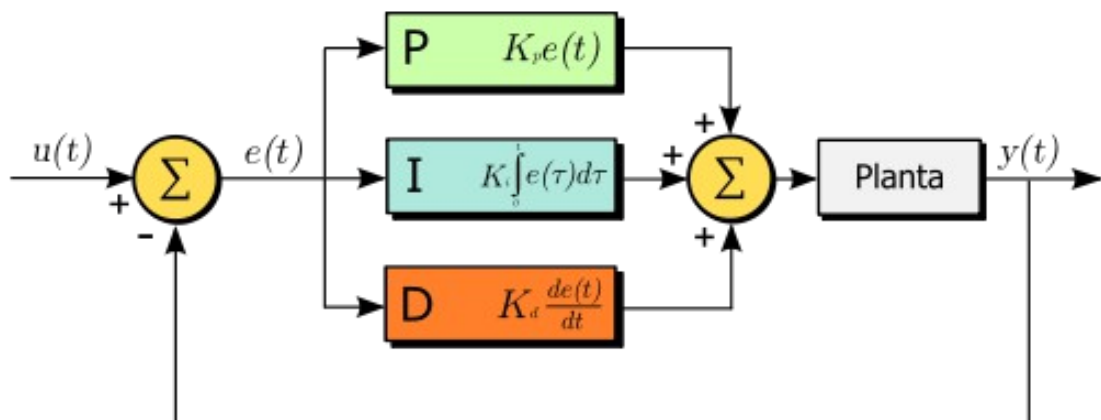


Figura 2.2: Esquema de un control PID

2.2.2 Controles Inteligentes

El control inteligente permite generalizar el concepto de control. Se caracteriza por permitir adaptación, aprendizaje, planificación bajo una elevada incertidumbre y procesamiento de una gran cantidad de datos (Cotero Ochoa).

El proyecto se enfoca, a pesar de la diversidad de metodologías, en el control difuso.

2.2.2.1 Control difuso

La lógica difusa es una técnica de la inteligencia artificial relacionada con la forma de razonamiento humana. A diferencia de la lógica clásica, en la que la pertenencia a un conjunto sólo puede tomar dos valores, verdadero (1) o falso (0), en la difusa una variable puede tener un valor de verdad gradual comprendido entre cero y uno. En la Figura 2.3 se observa la diferencia entre conjunto nítido y un conjunto difuso.

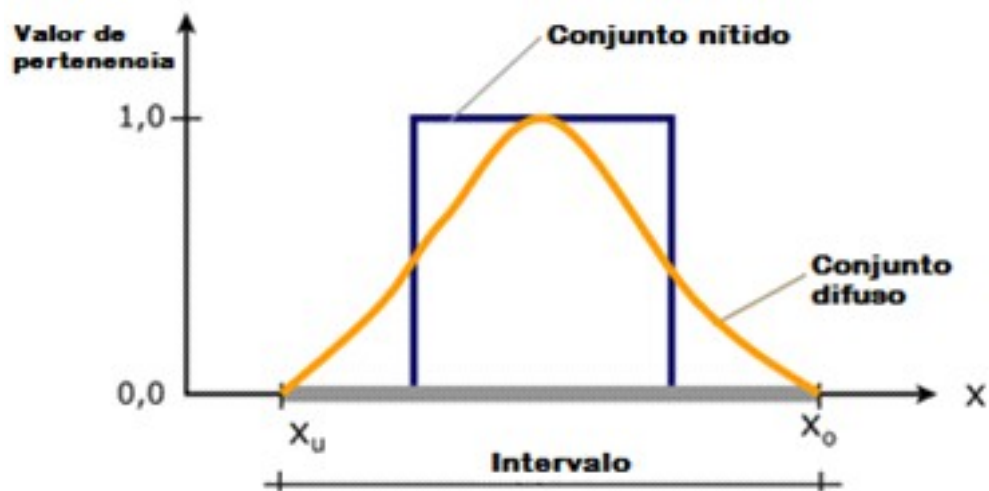


Figura 2.3: Diferencia de un controlador difuso y uno nítido

Esta lógica trabaja con conjuntos difusos. Un conjunto difuso es un conjunto en el que la pertenencia de sus elementos se define acorde a una función de pertenencia, que indicará en qué grado un valor se corresponde con uno o varios conjuntos difusos. La información de entrada se traduce en términos difusos en el proceso de fuzzyficación. En la Figura 2.4 se muestra un ejemplo de esta transformación.

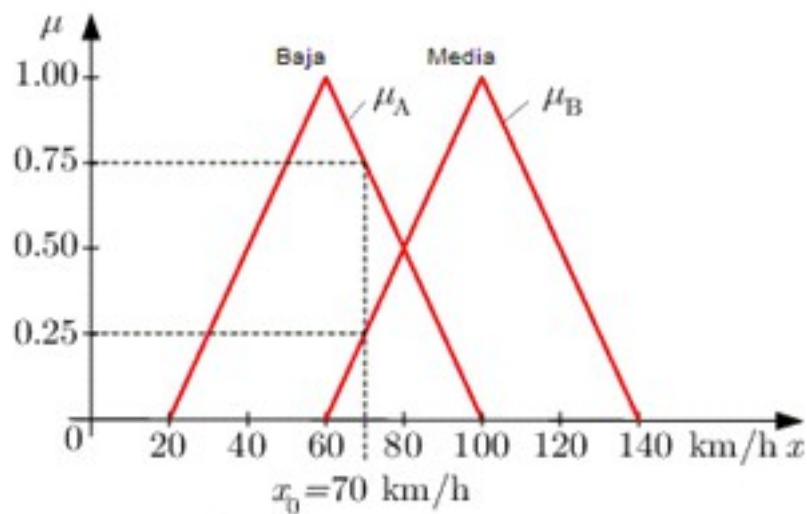


Figura 2.4: Ejemplo de fuzzyficación del valor $x_0 = 70$ Km/h en las funciones de pertenencia baja (μ_A) y media (μ_B)

En la imagen anterior se puede observar cómo la velocidad 70 Km/h pertenece un 75% al conjunto difuso baja y un 25% al de media.

El primer paso para la obtención de un valor de salida es la utilización de un sistema de reglas con encadenamiento hacia delante de un único nivel. Estas reglas son de la forma

Si variable condición entonces acción

Por ejemplo:

Si la velocidad es muy alta entonces aplica mucho el freno

La activación de estas reglas producirán determinadas salidas en uno o varios conjuntos difusos que serán interpretadas en el proceso de defuzzificación y traducidas a un único valor numérico que será el utilizado finalmente. Una explicación gráfica se enseña en la Figura 2.5 (Mukaidono, 2004).

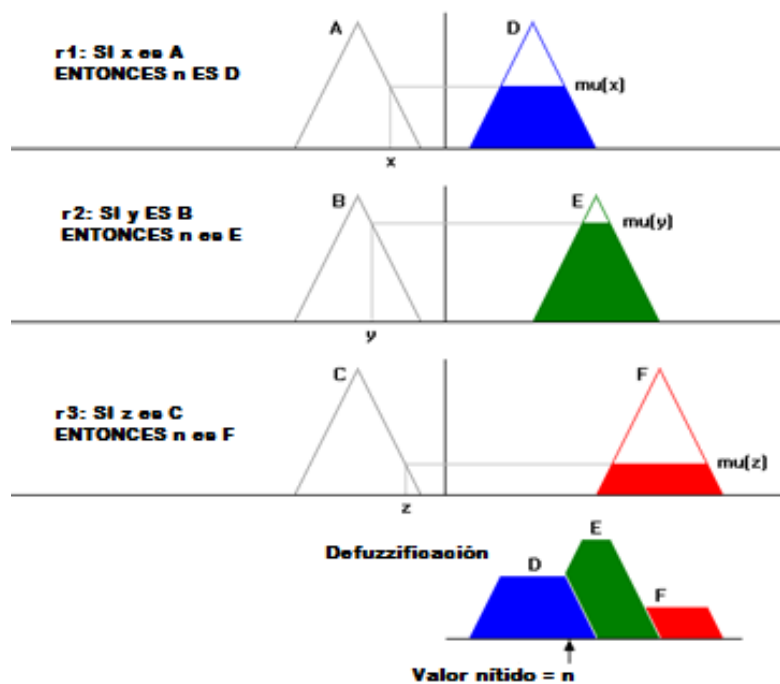


Figura 2.5: Defuzzificación mediante el método centro de gravedad

2.2.2.2 Redes neuronales

El paradigma de redes neuronales está inspirado en gran medida en el comportamiento biológico de estas células: frente a un conjunto de estímulos de entrada produce una salida determinada. Además, también como en la realidad,

estas neuronas actuarán en conjunto mediante la formación de grandes redes. Dentro de estas redes existen grupos de neuronas que estarán en el mismo nivel, lo que se denomina capa. De esta forma, tendremos en el primer nivel la capa de entrada, a continuación n capas ocultas y por último la capa de salida. Un ejemplo de esto lo podemos apreciar en la Figura 2.6.

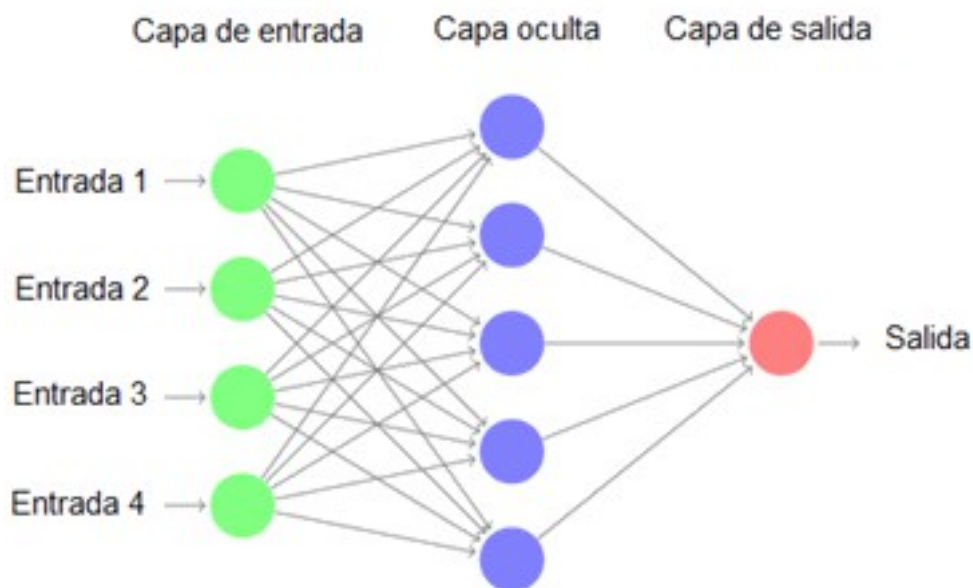


Figura 2.6: Red neuronal con 4 entradas y 1 capa oculta

La salida que produce una neurona viene determinada por tres funciones: una función de propagación, una de activación y una de transferencia

Los valores que genera una red neuronal se producen en base a unos datos de entrada para los que se conoce su salida, con los que se ha entrenado a la red previamente (Kosko, 1992).

3. Modelo

3.1 Estudio de las características del sistema

Analizando el objeto para su representación abstracta de sus propiedades se concluye delimitar respecto a su entorno solo su estado (posición y velocidad). Logicamente las magnitudes físicas de gravedad, factor de rozamiento, etc serán constantes del sistema. Mientras que el peso o tamaño forman parte de los parámetros del diseño (en este caso del Quanser Qball-X4).

El control, que será el que de mejor respuesta de los diseñados y simulados, tiene que ajustarse a unos valores muy concretos para que se estabilice en el tiempo menor posible y mejorar los resultados actuales en el campo de estudio. Al tratarse de un sistema complejo de doce ecuaciones es difícil determinar los valores paramétricos de los controles y, por tanto, es una de las partes más importantes del estudio.

Para la simulación se usa el software disponible Matlab-Simulink que ofrece las herramientas necesarias para pruebas. Se cuenta ya con un modelo genérico de cuatrirotor que una vez refinado bastará con aplicarle las características físicas del Quanser Qball-X4. Para facilitar la visualización se usa una representación del objeto a tiempo real y así interpretar mejor los datos obtenidos.

Dispone además de un software de control en tiempo real, QuadRC, que permite desaralabeoar y probar controladores sobre el hardware actual a través de un interface con Matlab-Simulink. El código desaralabeoado en Simulink para implementar controladores es automáticamente traducido y ejecutado en los controladores embebidos en el vehículo.

Durante el vuelo, mientras el controlador se está ejecutando, el usuario puede sintonizar parámetros del controlador en tiempo real y observar las medidas de los sensores desde una estación (PC). Las comunicaciones son vía Wi-Fi.

Por todo lo comentado anteriormente, la señal de entrada es el estado del modelo. Éste se compone de doce variables de entrada que son el vector posición x, y, z y la velocidad angular p, q, r respecto a los ejes de coordenadas; el vector de velocidad lineal u, v, w y los ángulos de Euler *cabeceo*, *alabeo* y *guiñada* (*cabeceo*, *alabeo*, *guiñada* en inglés) respecto a los ejes del cuatrirotor como se puede ver en la Figura 3.1. Con esta información, el modelo produce una respuesta a través del impulso de cada rotor que cambiará su estado aproximándose al deseado.

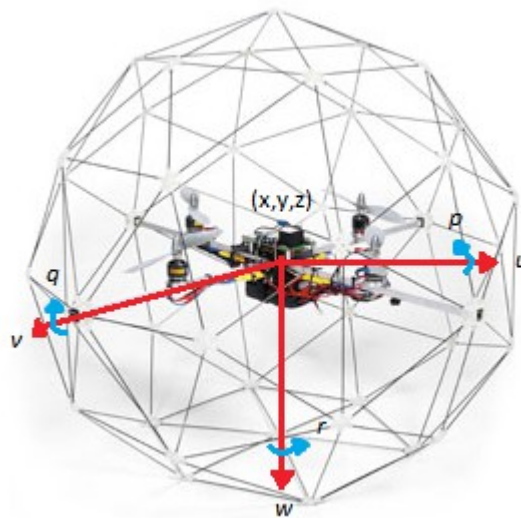


Figura 3.1: Variables del cuatrirotor Quanser QBall-X4

Se contempla las variables externas de perturbación como por ejemplo el viento, que afectan al estado. Para corregirlo el modelo varía su rumbo para alcanzar el esperado como en el sistema real.

3.2 Modelado analítico

Conociendo las leyes físicas para la cinemática y dinámica, se obtiene las ecuaciones necesarias para determinar el estado del sistema. Un sólido rígido en un sistema físico viene determinado por el número de grados de libertad para definir su estado.

En un espacio tridimensional se compone de seis: tres coordenadas de posición y tres de velocidades. Con estas ecuaciones obtenemos un modelo no lineal.

Respecto al cómo se produce movimiento en una dirección y sentido determinados, los cuatro rotores (de ahí su nombre) se encargan de ello. La disposición es en forma de cruz y funcionan por pares. Un par gira en un sentido y el otro par en el opuesto para que cambien el par de motor y el empuje.

Para definir el movimiento se obtienen doce ecuaciones diferenciales para cada variable. En el Anexo II se encuentra toda la información al respecto.

3.3 Identificación

El cuatrirotor para desplazarse de un punto del espacio a otro tiene unos movimientos determinados.

Para elevarse basta con modificar el sentido del giro de un par obteniéndose una fuerza resultante como la suma de los dos pares en el sentido negativo del eje OZ. En el caso contrario de descender, los dos pares giraran en el otro sentido produciendo la misma fuerza pero en el sentido positivo de OZ.

Para girar respecto al eje de guiñada se obtiene incrementando el par en el motor delantero y trasero a la vez que se reduce en los laterales.

Una rotación respecto al eje de cabeceo le permite avanzar o retroceder, reduciendo la velocidad el motor delantero y aumentando el trasero o viceversa.

Cualquier trayectoria se podrá definir como un conjunto de tramos formados por el subconjunto de movimientos descritos anteriormente. A su vez, éstos se componen del conjunto de órdenes a los cuatro motores de potencia y sentido de giro.

3.4 Modelo Quanser QBall-X4 de J. M. de la Cruz 2011

El modelo utilizado para el cuatrirotor Quanser QBall-X4 fue originalmente suministrado por la casa y posteriormente modificado por el catedrático Jesús Manuel de la Cruz. Esta última convirtió el modelo lineal original en un modelo no-lineal, permitiendo simular operaciones con seis grados de libertad. En la Figura 3.2 se muestra el diagrama del modelo.

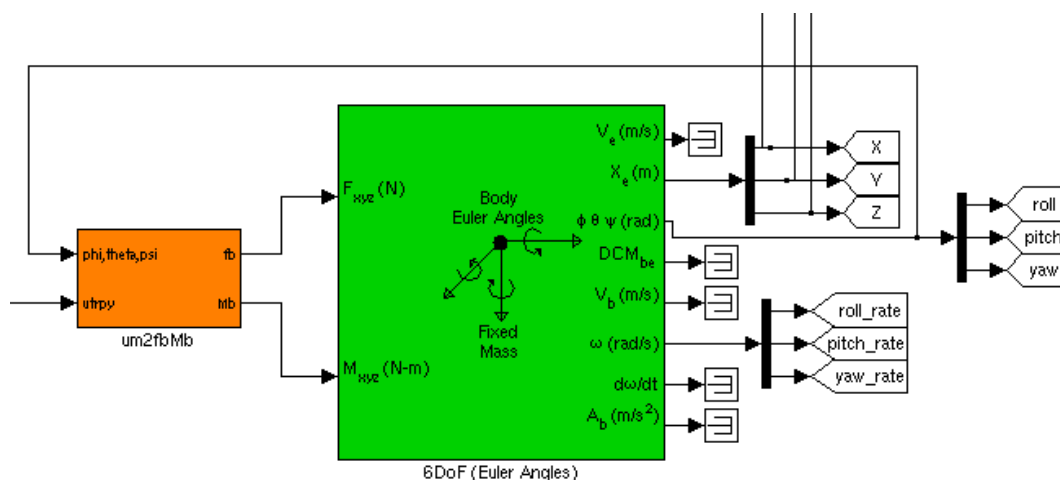
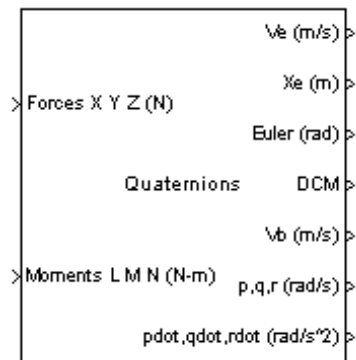


Figura 3.2: Modelo JMC-2011 empleado en Simulink

El bloque Simulink utilizado para representar las ecuaciones del movimiento con seis grados de libertad es propiedad de MatLab, y está enmarcado dentro del set Aeroespacial. Posteriormente se ajustaron los parámetros para el cuatrirotor original.

Aerospace Blockset

6DoF (Quaternion)



Entradas (2) : Fuerzas (X,Y,Z) y momentos.

Salidas (7) : Velocidades, posiciones, ángulos de Euler, matriz de transformación de coordenadas, aceleraciones y velocidades angulares.

3.5 Modelo de D. Sánchez 2009

La implementación de las ecuaciones de un sólido rígido con seis grados de libertad en Matlab-Simulink está compuesto por tres módulos, QuadS2, MotorDynamics y Limitations/adaptations como se muestra en la Figura 3.3.

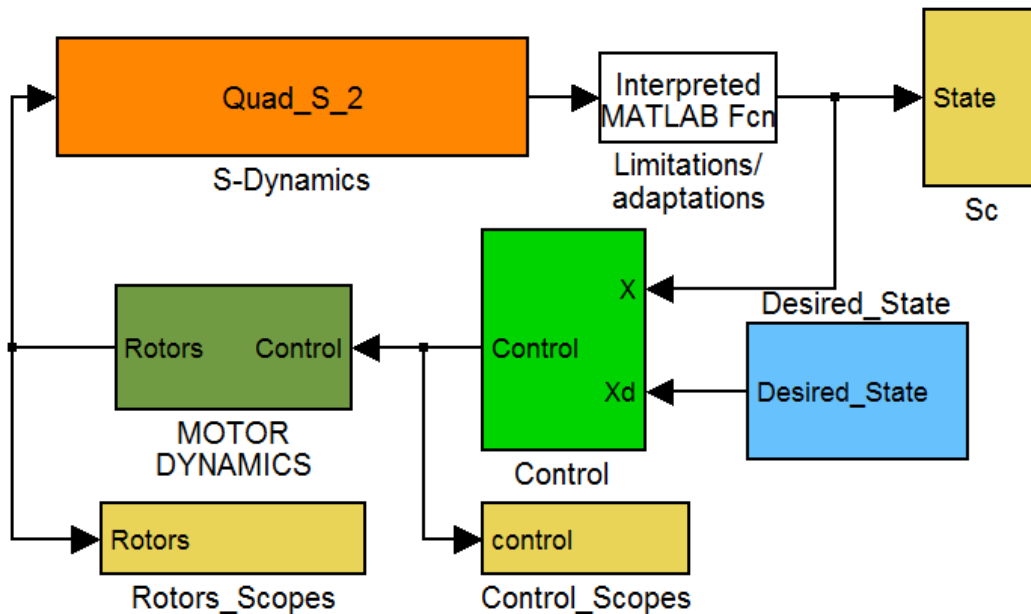


Figura 3.3: Modelo DS-2009 empleado en Simulink

QuadS2: Implementa mediante funciones MATLAB la dinámica del movimiento de un sólido rígido con seis grados de libertad.

Limitations/adaptations: Implementa mediante una función MATLAB las limitaciones físicas del sistema, que la altura no tenga valores negativos y establece las equivalencias de los ángulos de Euler. El método utilizado originalmente para impedir alturas negativas, enmascara los resultados del modelo y hace imposible simular aterrizajes.

Motor Dynamics: En el bloque MOTOR DYNAMICS se calculan las señales de impulso para cada motor en función de la señal de control proporcionada y los parámetros establecidos para un cuatrimotor en concreto. En la Figura 3.4 se muestra la composición del bloque en Simulink. Estos parámetros se pueden modificar a través del archivo *parameters.m*. El retardo de la modulación de la señal

de la emisora se refleja con el bloque de Trasmision-Delay preconfigurado a un valor adecuado.

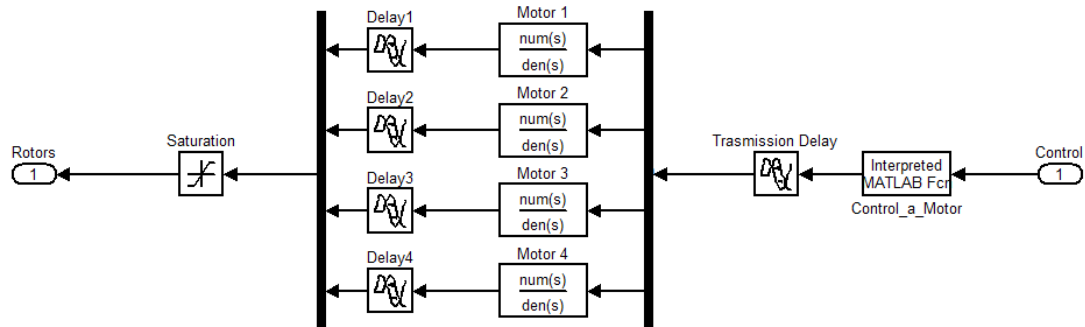


Figura 3.4: Detalle del bloque MOTOR DYNAMICS

3.6 Modificaciones y mejoras

Con objeto de mejorar el modelo de David Sánchez 2009 se han añadido los elementos que permiten simular despegues y aterrizajes reales, incluyendo las posibles deceleraciones del móvil con la superficie de contacto (suelo, portaviones, etc.)

3.6.1 Suelo real

El primer problema que presentaba el modelo original era la ausencia de una altura a la que situar la referencia del suelo (línea de tierra) sobre la que situar a la aeronave. Originalmente no se permitían alturas negativas en los resultados de las gráficas pero el modelo si operaba con ellas, induciendo a error al operador del mismo. Por ejemplo, si el móvil partía de la posición (0,0,0) comenzaba inmediatamente su descenso en el tiempo de retardo que los motores tardaban en reaccionar a las instrucciones del controlador. Esto no se reflejaba en la gráficas, lo único apreciable era un tiempo excesivamente alto para que la aeronave abandonara la posición de cero metros de altura.

La solución elegida ha sido utilizar unas ecuaciones diferentes cuando el móvil se encuentra en el suelo (altura cero o aproximadamente cero) en las cuales se tiene en cuenta el empuje aplicado a los motores para producir una aceleración vertical pero se establecen por obligación a cero el alabeo y el cabeceo, como ocurriría realmente respecto al suelo con el cuatrorotor apoyado en el mismo, y se anula toda variación de altura negativa (Figura 3.5).

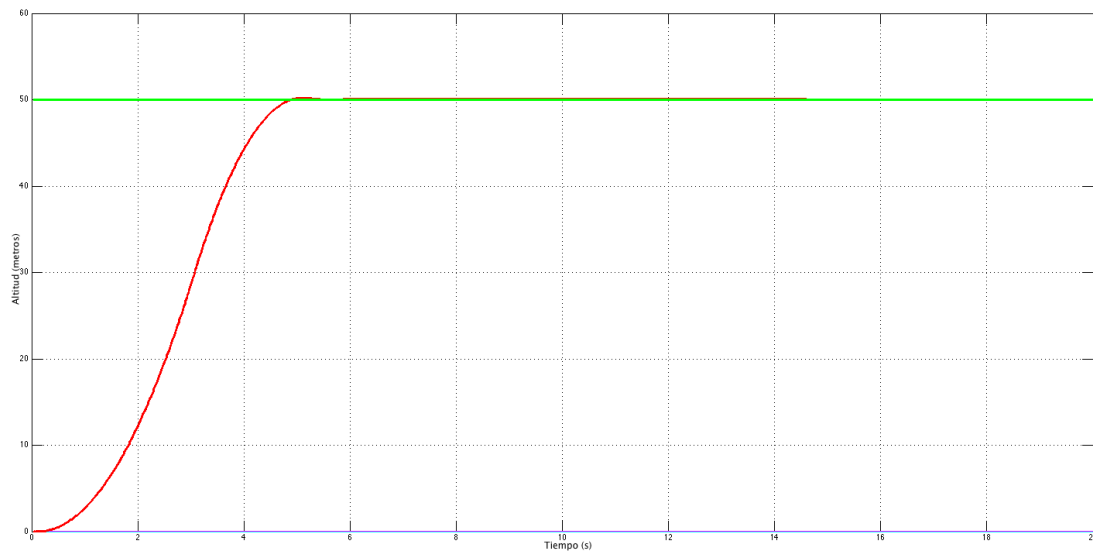


Figura 3.5: Despegue desde el suelo

Además, también se desprecian las micro oscilaciones ocurridas en los ejes X e Y que en existencia de rozamiento estático con el suelo no deberían existir.

Esto permite simular despegues y aterrizajes desde el suelo así como inferir de las gráficas las velocidades a las que se producen los mismos de forma real (Figura 3.6).

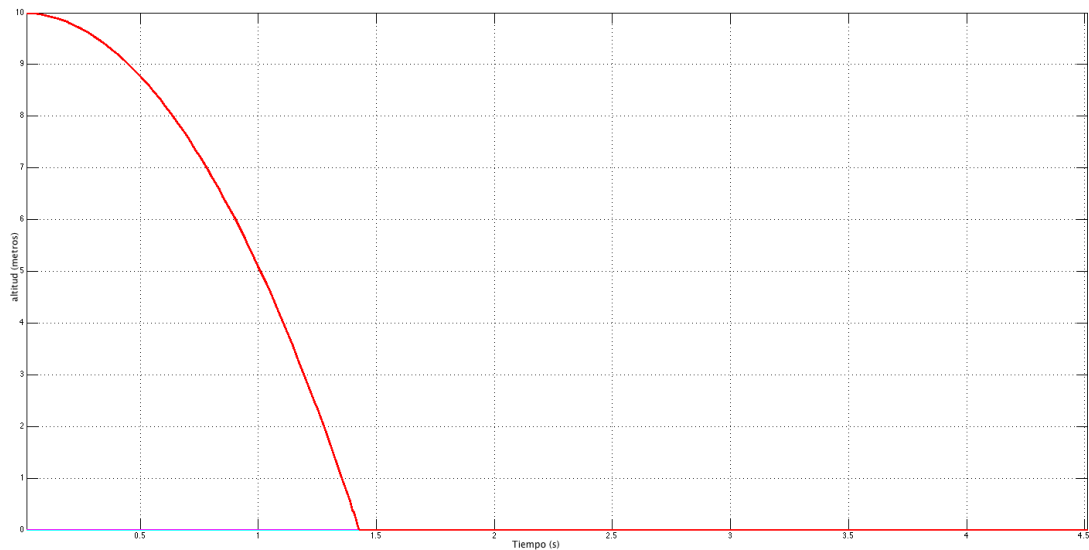


Figura 3.6: Caída real contra el suelo

3.6.2 Rozamiento con el suelo

La otra mejora llevada a cabo consiste en añadir a la simulación el rozamiento básico con el suelo que permita simular condiciones de aterrizaje, de modo que si el cuatrorotor llega al suelo con velocidad en los ejes X e Y no se anulen inmediatamente sino que sufra la deceleración correspondiente al tipo de superficie por donde está resbalando.

Así, podemos definir la fuerza de rozamiento F_r de un cuerpo en desplazamiento horizontal de la forma en la que aparecen las siguientes ecuaciones:

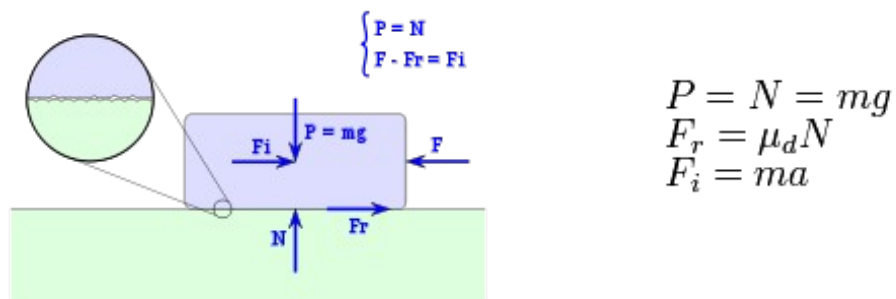


Figura 3.7: Representación gráfica de las fuerzas sometidas en el objeto

Implementado en las ecuaciones de nuestro modelo de la siguiente forma:

$$\text{angulo} = \text{atan}\left(\frac{|V|}{|U|}\right) \quad \text{dotx}(U) = -\text{sign}(U) \cdot M_{\text{Total}} \cdot (\text{Mud} \cdot g \cdot \cos(\text{angulo}))$$

$$\text{dotx}(V) = -\text{sign}(V) \cdot M_{\text{Total}} \cdot (\text{Mud} \cdot g \cdot \sin(\text{angulo}))$$

Esto nos permite simular de forma efectiva la deceleración que sufre el móvil en los ejes X e Y al alcanzar el suelo en función del coeficiente de rozamiento del mismo (según los materiales involucrados de los patines del cuatrirotor y la superficie).

A continuación se muestran los resultados de dejar caer el cuatrirotor con un ángulo inicial de 0,2 y 0,3 radianes en cabeceo y alabeo respectivamente desde una altura de 10 metros para una superficie con un coeficiente de rozamiento dado (Figura 3.8) (Figura 3.9).

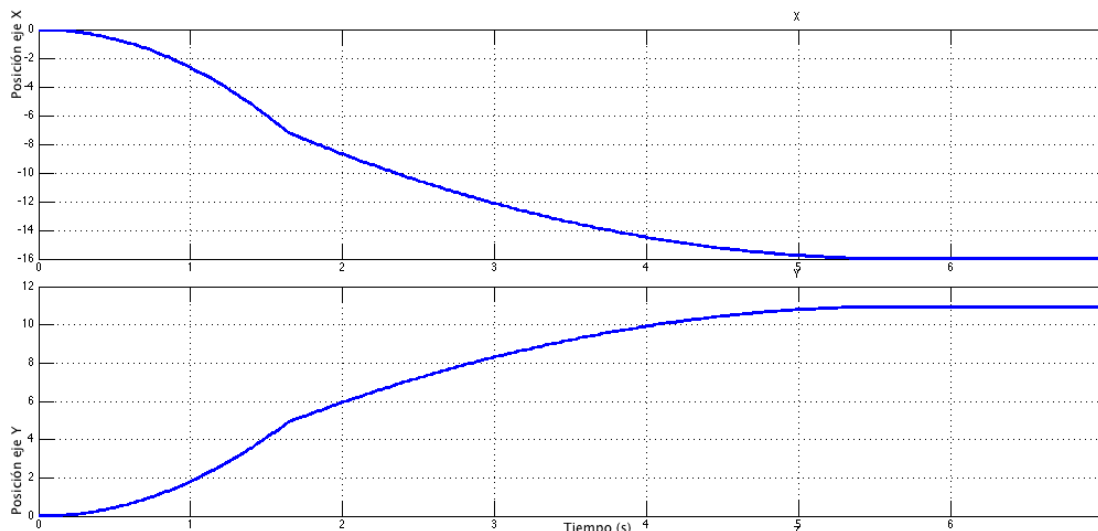


Figura 3.8: Posición del móvil respecto a los ejes X e Y partiendo de (0,0,10)

Para realizar la validación de forma efectiva sería necesario disponer del vehículo real en funcionamiento. Es de suponer que las ínfimas oscilaciones que ocurren en simulación en estado estacionario aumentarían en magnitud y frecuencia pero dado que los parámetros se han ajustado conforme al vehículo real las expectativas de funcionamiento general son muy positivas.

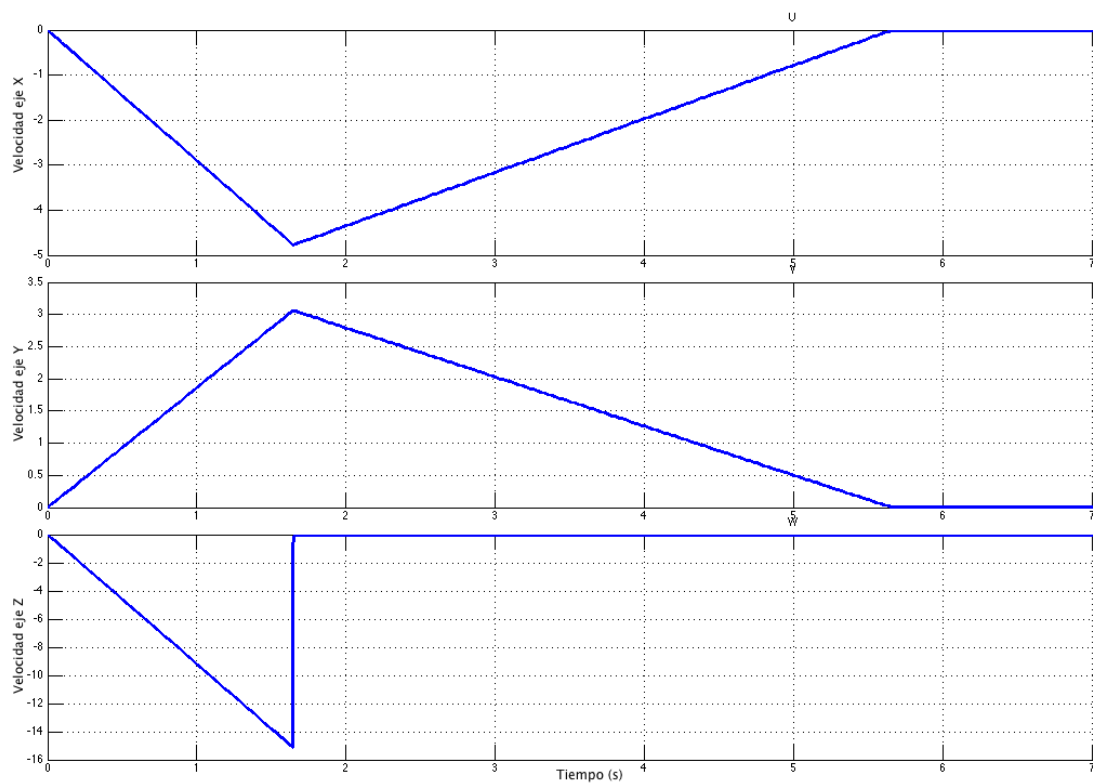


Figura 3.9: Velocidad relativa en los ejes X,Y,Z para el ejemplo anterior

4. Control del Quanser JMC-2011

4.1 Implementación

La implementación del sistema de control sobre el modelo de Quanser modificado se puede definir como un sistema de PID's en cascada más un PID inteligente. (Ver Figura 4.1).

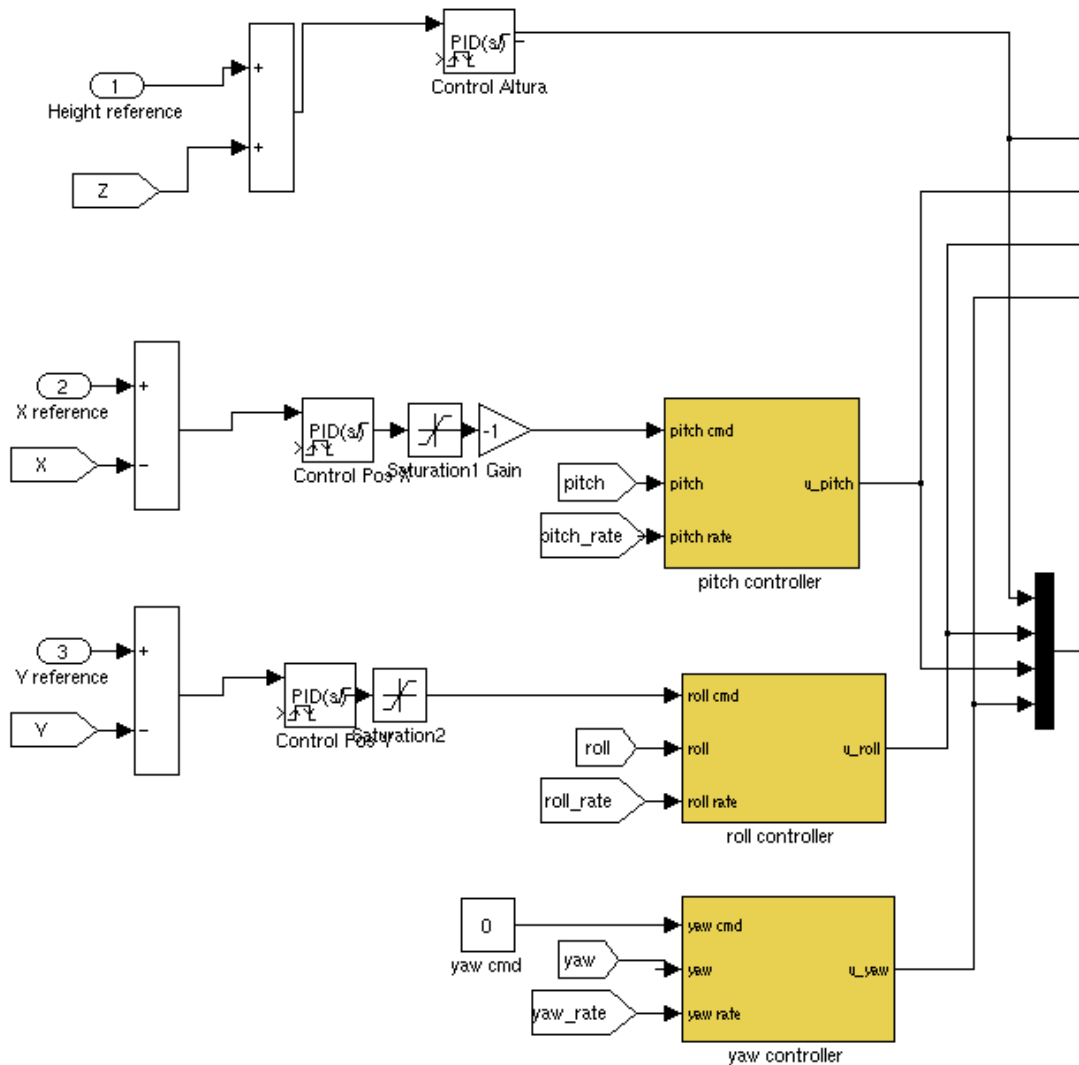


Figura 4.1: Diagrama de control general

Este último, trabaja asociado con un calculador de parámetros que le proporciona los ajustes en tiempo real, similar al mecanismo de ganancia programada pero continuo y aplicable a los 3 parámetros de forma diferenciada. Este mecanismo será

explicado con detalle en el punto 5 del presente documento ya que fue originalmente diseñado para otra plataforma.

Control de altitud

El control de altitud implementado permite un rango de operación en “tramo único” de 400 metros. Esto significa, que utilizado de forma correcta, podrá ejecutar tramos con un diferencial máximo de 400 metros en altura sin error apreciable, ni de sobreelongación ni de estabilidad a largo plazo. (Figura 4.2).

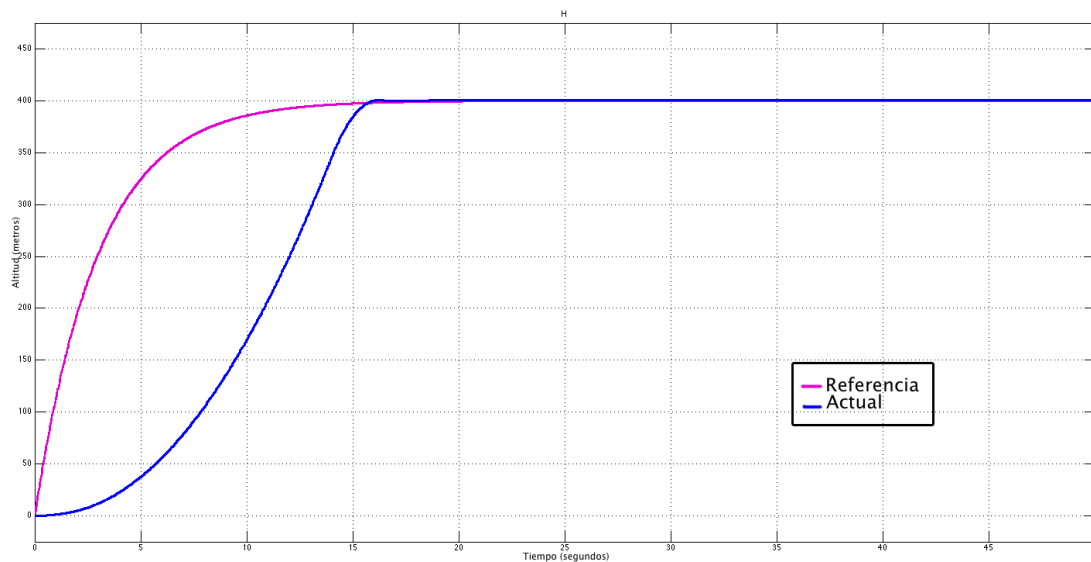


Figura 4.2: Alcance de la altura deseada

Control de posición

En este caso, se ha optado por una versión un poco más agresiva que permite realizar maniobras en pequeños espacios de forma rápida y precisa, por lo que su rango de operación en tramo único se reduce a 125 metros en X e Y. (Ver Figura 4.3 y Figura 4.4)

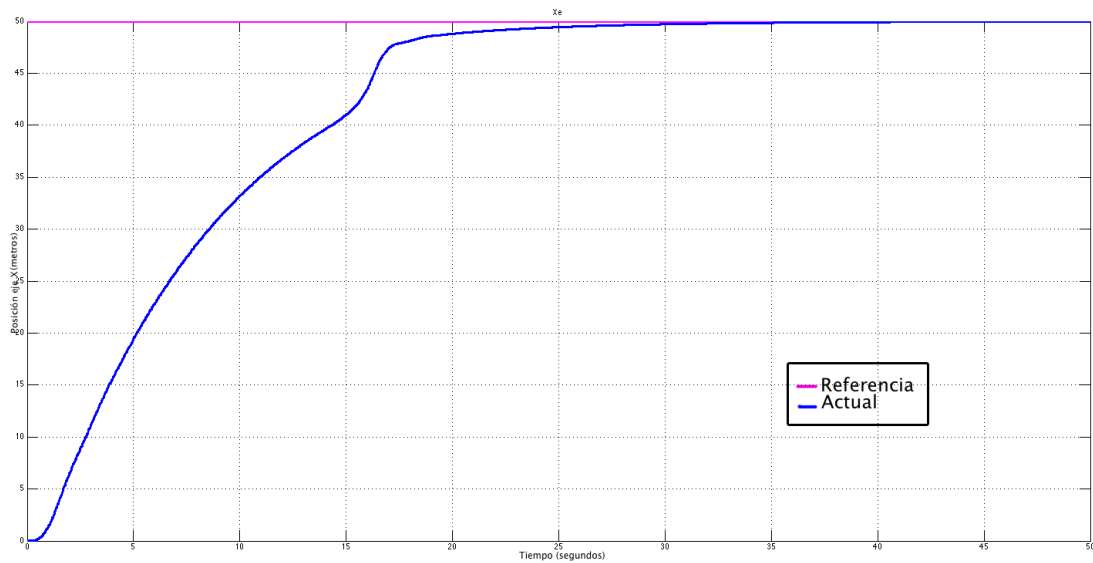


Figura 4.3: Alcance de la posición en el eje X

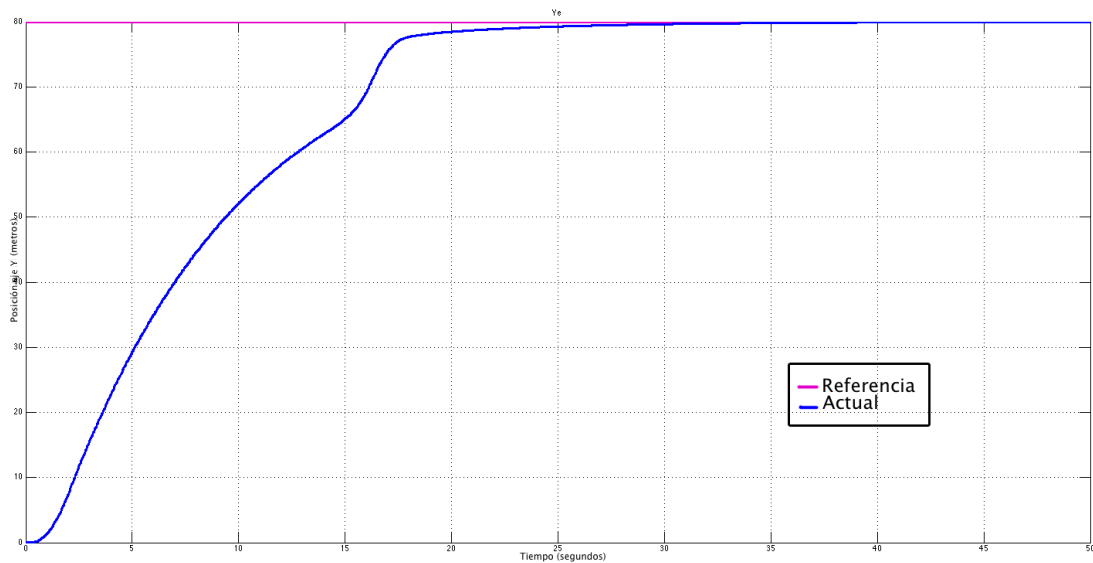


Figura 4.4: Alcance de la posición en el eje Y

Filosofía de operación

El hecho de especificar el tamaño máximo de tramo en ausencia de error, no limita el rango de operación del vehículo en modo alguno. Para tramos mayores, sería necesario un planificador de trayectorias que dividiera el espacio a recorrer en una lista de tramos ordenada como es estándar en todo tipo de proyectos de guiado.

4.2 Resultados

4.2.1 Seguimiento de trayectorias dinámicas simples

El seguimiento de trayectorias dinámicas simples, muestra la capacidad del sistema de control de adaptarse en tiempo real a una señal variable y permite medir la precisión, la estabilidad, y la velocidad de movimiento disponible.

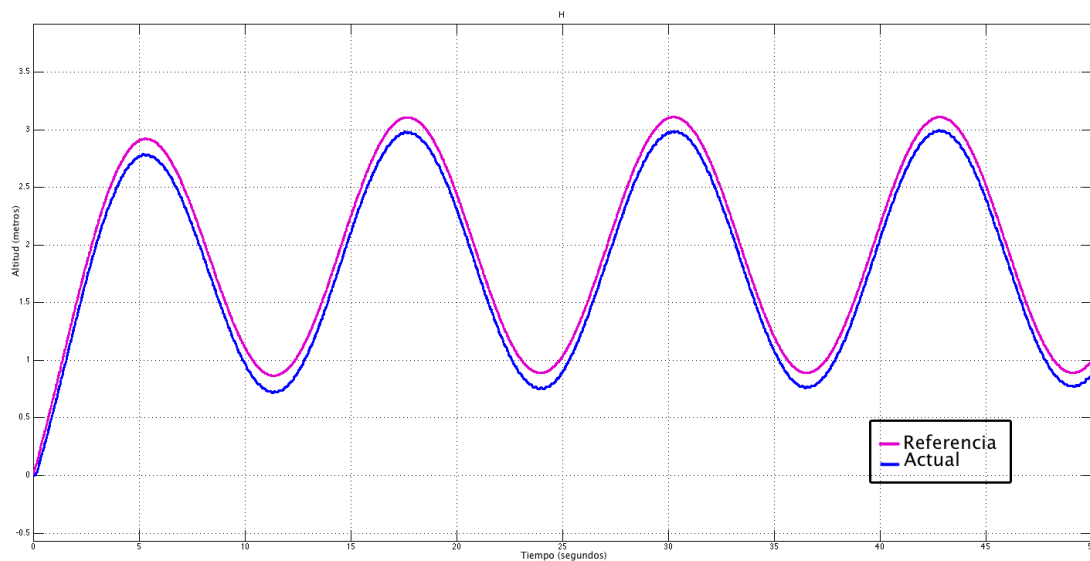


Figura 4.5: Trayectoria dinámica simple

La trayectoria de prueba de la Figura 4.5 es una trayectoria sinusoidal de frecuencia 0,5Hz y amplitud 4. El sistema se comporta de forma muy correcta pese a la relativamente alta frecuencia utilizada. Se puede decir entonces que posee gran capacidad de maniobra y seguimiento de trayectorias en altura.

4.2.2 Seguimiento de trayectorias compuestas

El seguimiento de trayectorias dinámicas compuestas implica realizar movimientos coordinados en los tres ejes a lo largo de una trayectoria que va cambiando con el tiempo. Para esta prueba se ha seleccionado una maniobra real conocida como circuito de tráfico aéreo y que deben realizar todas las aeronaves volando en las

proximidades de un aeródromo. Este procedimiento se puede ver en detalle en la Figura 4.6. El resultado obtenido por el cuatrirotor (Figura 4.7) ejecutando la citada maniobra de vuelo básica es excelente.

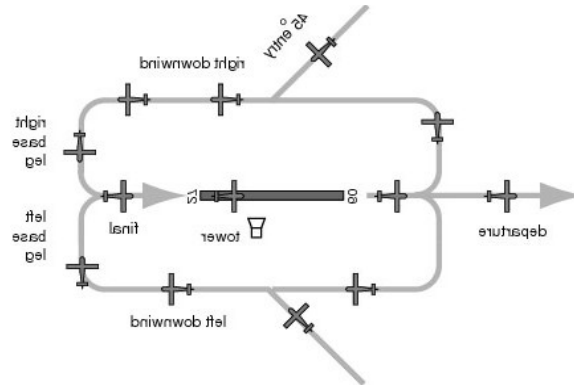


Figura 4.6: Maniobra del circuito de tráfico aéreo

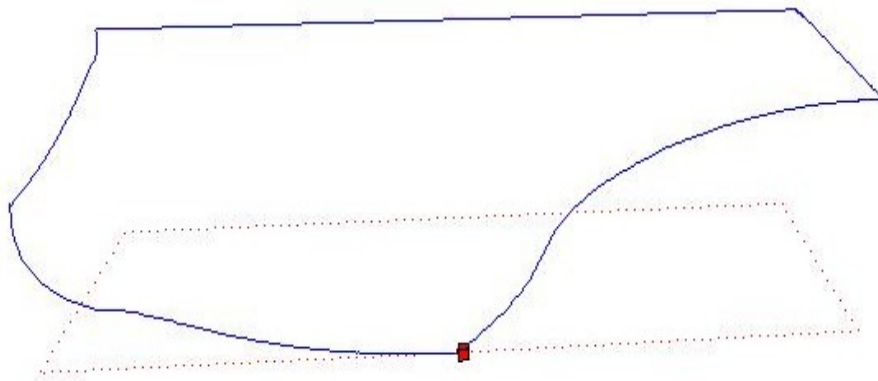


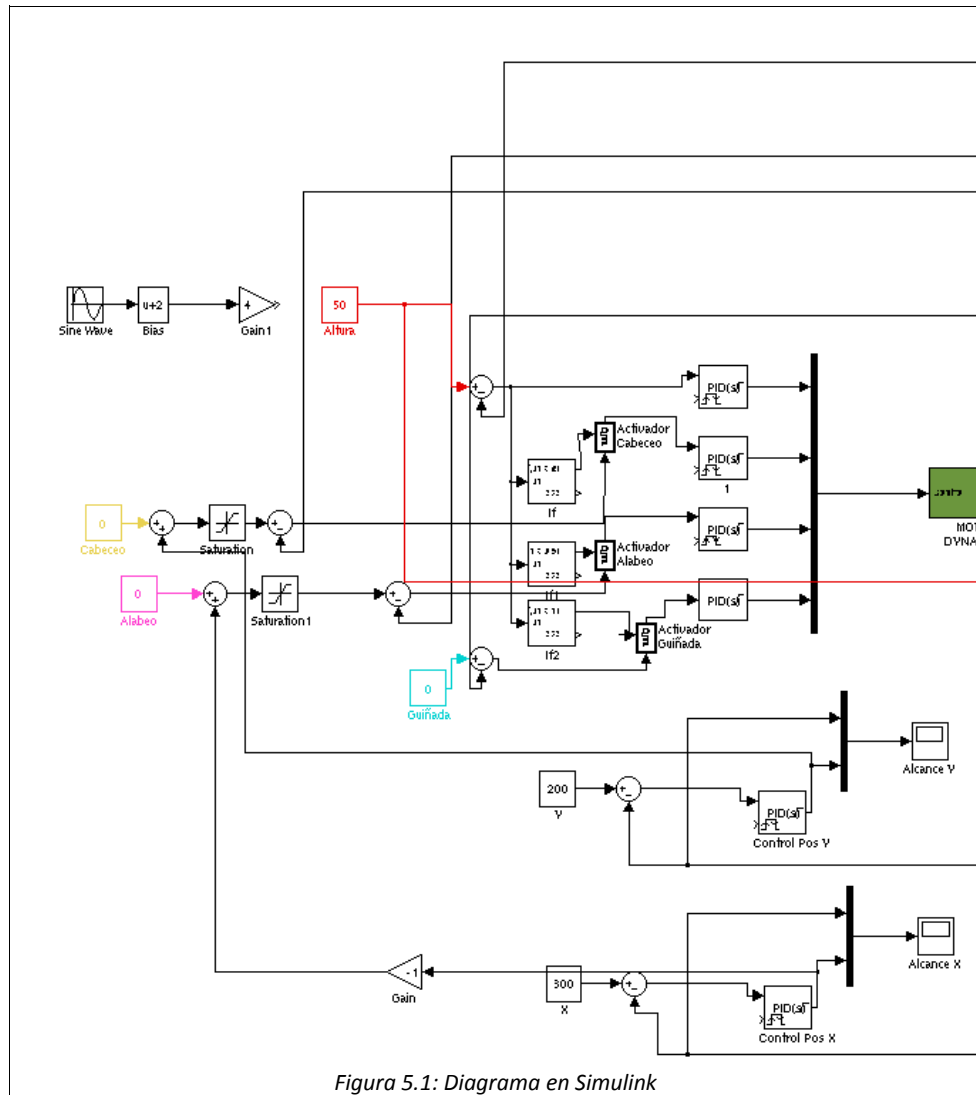
Figura 4.7: Resultado de la simulación de la maniobra

A la vista de los resultados, podemos afirmar que el sistema de control para el modelo Quanser modificado se muestra muy efectivo y muy preciso dentro de los rangos de operación especificados.

5. Control de DS-2009

5.1 PID

A continuación en la Figura 5.1 se muestra el esquema general del control mediante PID en cascada con ganancia programada inteligente utilizado para conseguir un rendimiento óptimo en este cuatrirotor.



Como se puede ver, existen tres grandes bloques: El primero, los PID de los parámetros de vuelo (Altura,cabeceo,alabeo,guiñada), otro para la posición (X,Y) y un último bloque de activadores central que activa o desactiva los PID según corresponda.

5.1.1 Implementación

La implementación del sistema de control sobre el modelo D.Sanchez-2009 mejorado se puede definir como un sistema de PID en cascada relacionados mediante activadores condicionales más un PID inteligente en el control de altura. Esto permite establecer una serie de prioridades sobre las señales de control, y dotar al vehículo de una maniobrabilidad y velocidad de respuesta sin precedentes.

Sistema de prioridades

Las prioridades sobre las señales de control se establecen del siguiente modo: El parámetro más prioritario es la altitud, no existiendo control sobre el resto de los parámetros hasta que la señal de altitud se considera viva (en aviación, un parámetro se considera vivo o *alive* cuando este comienza a aproximarse al valor de la referencia deseada) a continuación, se activaran por estricto orden, cabeceo, guiñada y alabeo según corresponda.

La activación de las citadas señales de control sólo depende de la altitud, así el ajuste de cabeceo, guiñada y alabeo se realizara de forma prácticamente simultanea siempre que la altitud se encuentre dentro de la tolerancia especificada respecto a la referencia deseada.

Como se puede ver en la Figura 5.2, la línea amarilla que representa la señal de cabeceo es la primera que comienza a operar seguida por la línea de guiñada y alabeo en poco menos de un segundo.

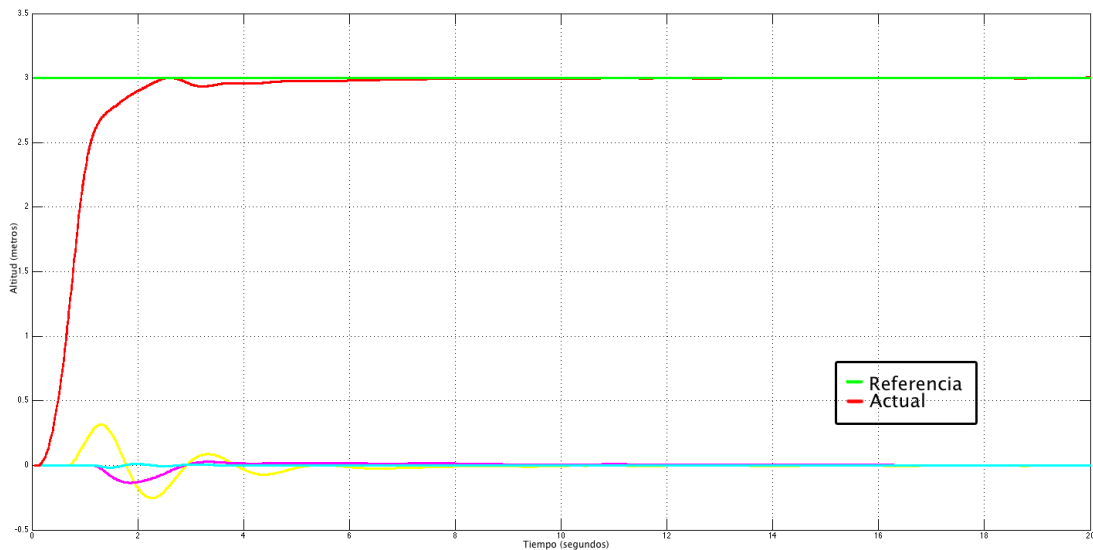


Figura 5.2: Vuelo de control a posición definida

Control de altitud

El control de altitud implementado permite un rango de operación en “tramo único” de 400 metros. Esto significa, que utilizado de forma correcta, podrá ejecutar tramos con un diferencial máximo de 400 metros en altura sin error apreciable, ni de sobreelongación ni de estabilidad a largo plazo.

Control de posición

En este caso, se ha optado por una versión un poco más agresiva que permite realizar maniobras en pequeños espacios de forma rápida y precisa, por lo que su rango de operación en tramo único se reduce a 300 metros en X e Y.

Implementación del PID Inteligente

Como se ha comentado anteriormente, el control de la altura necesita de unos parámetros dinámicos que deben ser calculados en función del tamaño del tramo a

ejecutar, en la Figura 5.3 que se muestra la filosofía de operación de dicho calculador. El calculador en su versión final sólo necesita como variable de entrada el tamaño del tramo, quedando la entrada de velocidad para posteriores implementaciones más complejas.

PID Adaptativo

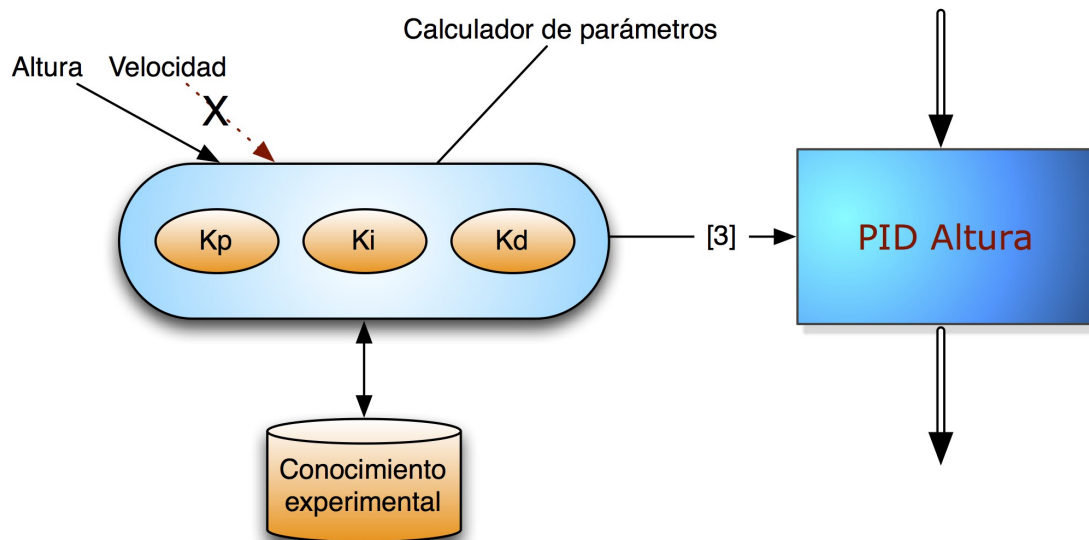


Figura 5.3: Esquema PID adaptativo

El cálculo de los parámetros se realiza por interpolación directa sobre los resultados obtenidos mediante prueba y error en las diferentes pruebas experimentales. Estos resultados se encuentran almacenados en el sistema y se detallan a continuación. Los tres parámetros se tratan de forma completamente independiente.

El cálculo final se realiza según la variable definida como TamañoDeTramo o lo que es lo mismo la diferencia de alturas entre la posición inicial y la final.

$$\text{TamañoDeTramo}(Z) = (Z_{\text{final}} - Z_{\text{inicial}}) + 1$$

5.1.2 Componentes del calculador inteligente

Componente P

La componente P disminuye a mayor tamaño de tramo pero no lo hace de forma uniforme(Figura 5.4).

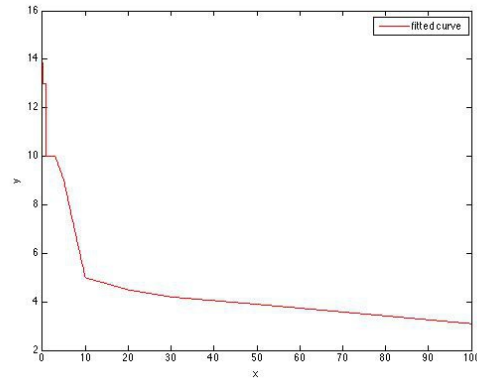


Figura 5.4: Componente P

Componente I

La componente I se asemeja a una función exponencial decreciente en función del tamaño del tramo(Figura 5.5).

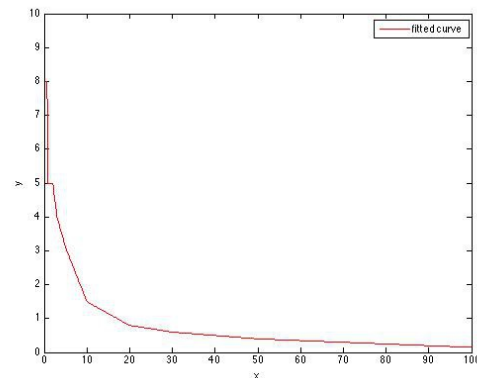


Figura 5.5: Componente I

Componente D

La componente D se ajusta a ninguna función conocida y debe ser codificada manualmente para obtener unos buenos resultados(Figura 5.6).

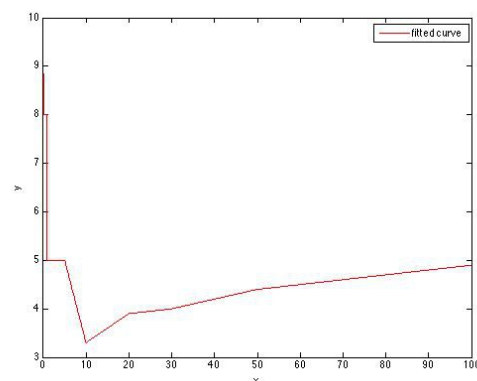


Figura 5.6: Componente D

5.1.3 Análisis de resultados

5.1.3.1 Seguimiento de trayectorias dinámicas simples

Se muestra a continuación en la Figura 5.7 el seguimiento de una trayectoria dinámica basada en la función seno en altitud con una amplitud de 8 metros y una frecuencia de 0,5 Hz.

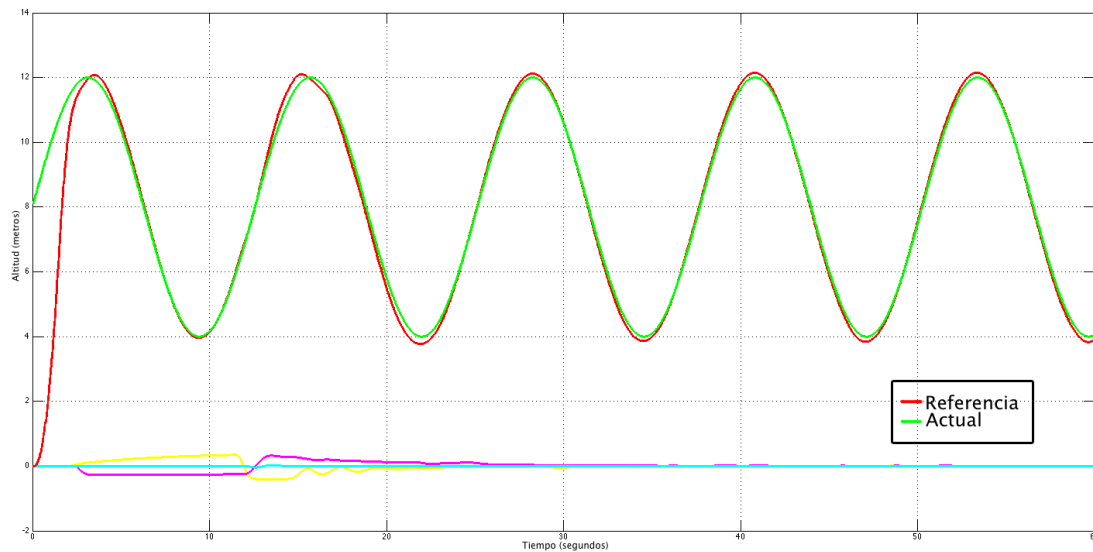


Figura 5.7: Seguimiento de trayectoria sinusoidal con amplitud 8 metros

A la vez que se realizaba el seguimiento de la trayectoria sinusoidal en altura, se le pedía que ejecutara un desplazamiento programado en X e Y a la posición (300,200)

Como se puede ver en el resultado 2D de la simulación en la Figura 5.8, la precisión alcanzada durante todo el ejercicio es extremadamente buena. Podemos decir que se trata de un control fiable.

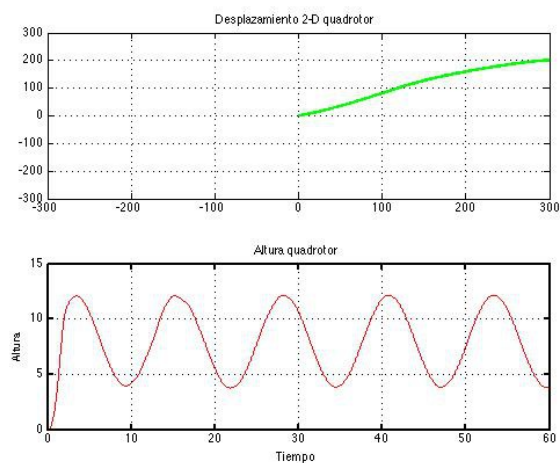


Figura 5.8: Vista en planta de la trayectoria

5.1.3.2 Vuelo real hacia un objetivo manteniendo parámetros de altitud

Se puede ver el resultado de la Figura 5.9 de ejecutar un vuelo real desde la posición (0,0,0) hasta la posición (300,200,50). Es decir, manteniendo 50 metros de altitud en todo momento, desplazarse a una posición dada aproximadamente a 360 metros de distancia.

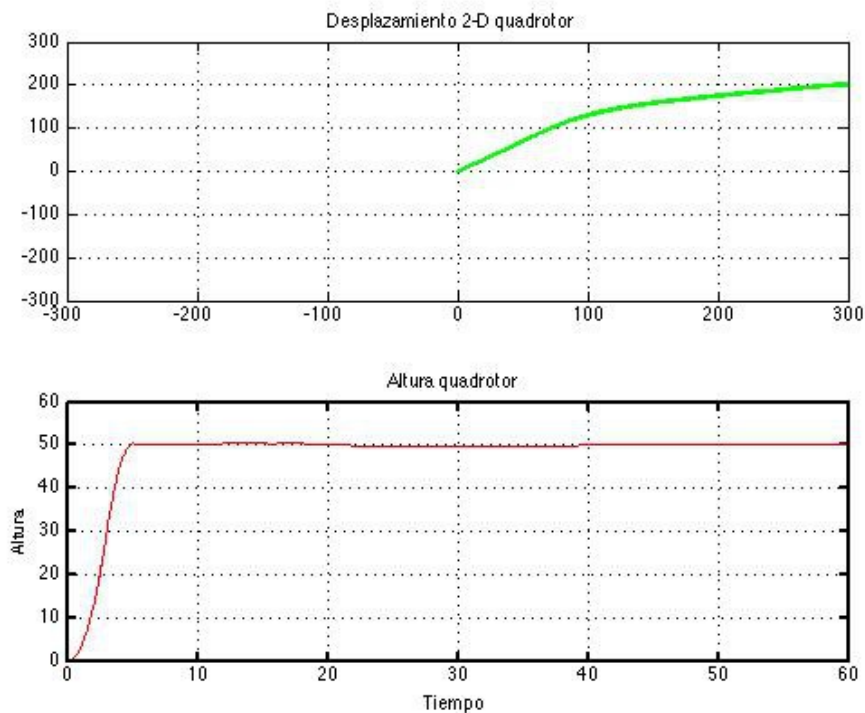


Figura 5.9: Resultados del vuelo real

La calidad en la captura de las posiciones es excelente, no existiendo sobreelongación o error alguno y operando de forma acorde con la aviación según el patrón lento-rápido-lento de un desplazamiento estandar.

5.2 Fuzzy

El control difuso utiliza como entradas el error, la derivada del error y la integral del error de la variable que se trate de controlar; o en otras palabras, es un control fuzzy-PID.

En una primera versión se usó las ideas obtenidas en el modelo de la caja negra para usar la potencia que ofrece la lógica difusa ideal sin funcionar correctamente. Después se utilizó un control basado en un PID por su buen resultado conocido, con entradas del error y la derivada del error de la variable que se trata de controlar.

Para todos los controladores se ha utilizado como sistema de inferencia Mamdani y como método de defuzzyficación bisector.

Comentar que debido a la complejidad del modelo el control fuzzy-PID se ha implementado sólo para las variables de altura y guiñada. Tanto el cabeceo como el alabeo se controlan mediante reguladores PID que son exactamente los mismos que los utilizados en el capítulo en el que se explica este tipo de control para el cuatrorotor. Esto se comentará con mayor grado de detalle en el apartado de conclusiones.

Para todos los controladores se ha utilizado como sistema de inferencia Mamdani y como método de defuzzyficación bisector.

5.2.1 Control de la altura

Para controlar la altura se le suministra a los motores la potencia necesaria para subir, mantenerse o bajar, que será por tanto la salida que proporcione este control.

Como entradas utilizará el error y la derivada del error, con ganancias $P=2$ y $D=5$, obtenidas mediante prueba y error, siendo la parte integral igual a cero ($I=0$).

El universo de discurso para la entrada error de altura estará formada por los conjuntos difusos `error_negativo`, `error_cero` y `error_positivo` (Figura 5.10).

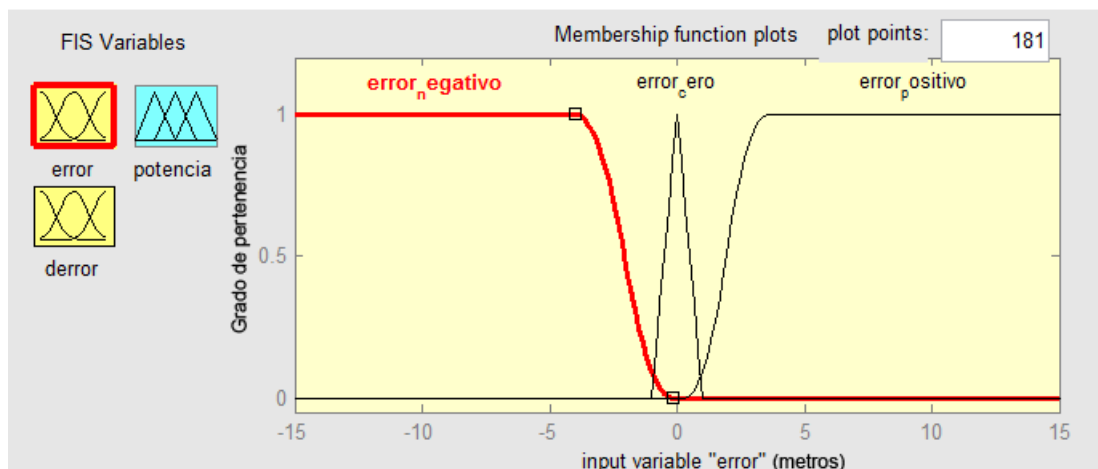


Figura 5.10 Conjuntos difusos para la variable de entrada error

La entrada derivada del error en altura estará formada por los conjuntos difusos `derivada_negativa`, `derivada_cero` y `derivada_positiva` (Figura 5.11).

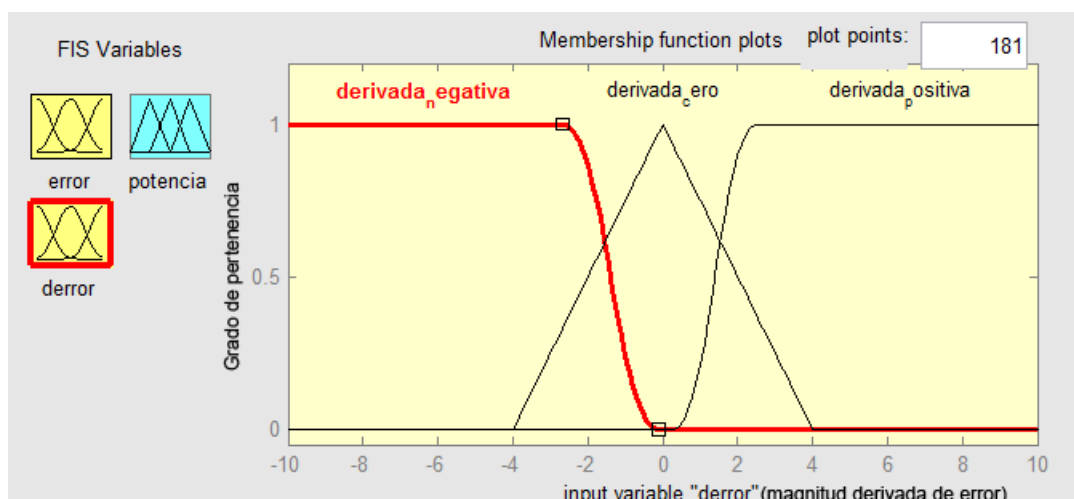


Figura 5.11 Conjuntos difusos para la variable de entrada derivada del error

La salida utilizará los conjuntos borrosos subir, mantener y bajar(Figura 5.12).

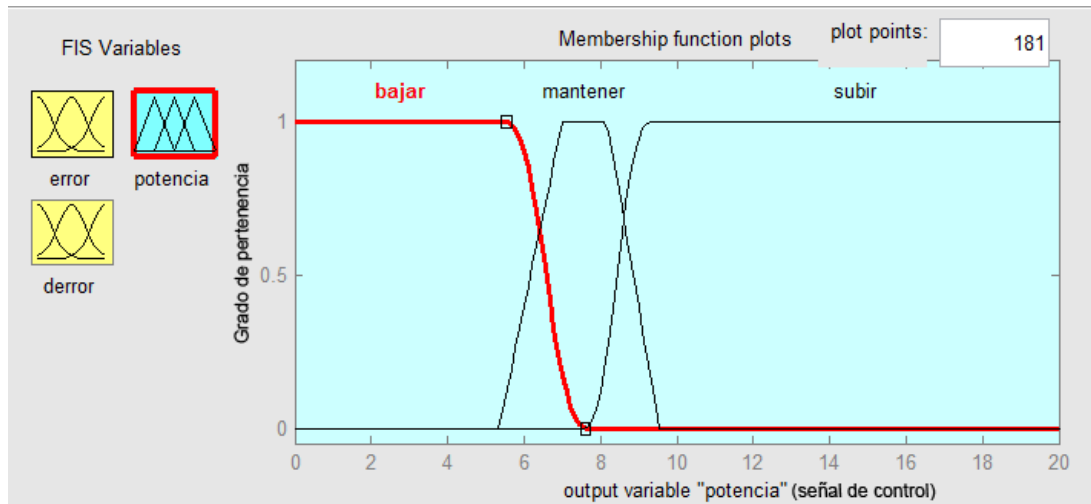


Figura 5.12 Conjuntos borrosos para salida del control

El conjunto de reglas utilizado ha sido el siguiente:

Error Z/Error DZ	Error negativo	Error Cero	Error Positivo
Error Negativo	<i>bajar</i>	<i>bajar</i>	<i>mantener</i>
Error Cero	<i>bajar</i>	<i>mantener</i>	<i>subir</i>
Error Positivo	<i>mantener</i>	<i>subir</i>	<i>subir</i>

5.2.2 Control del guiñada

El control del guiñada se realizará utilizando como entradas el error de la variable, su derivada y su integral, con ganancias $P=20$, $D=30$ e $I=20$ obtenidas mediante prueba y error.

Para la entrada error se encuentran los conjuntos borrosos SHorario (girar en sentido horario), mantenerse, y SAntihorario (girar en sentido antihorario) (Figura 5.13).

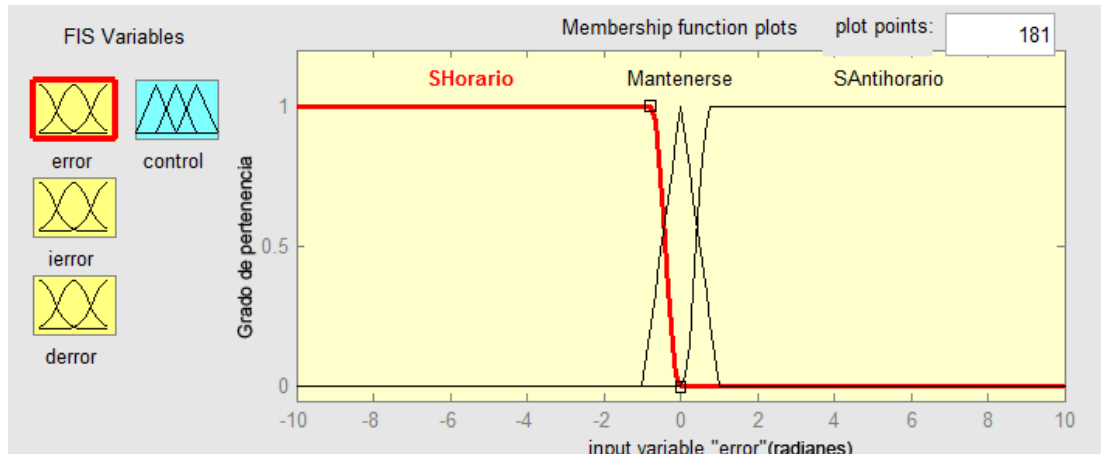


Figura 5.13 Conjuntos difusos para la entrada error de la variable guiñada

Para la integral del error se utilizan los conjuntos difusos Negativa, Igual y Positiva al que pertenecerá el resultado en función de su signo (Figura 5.14).

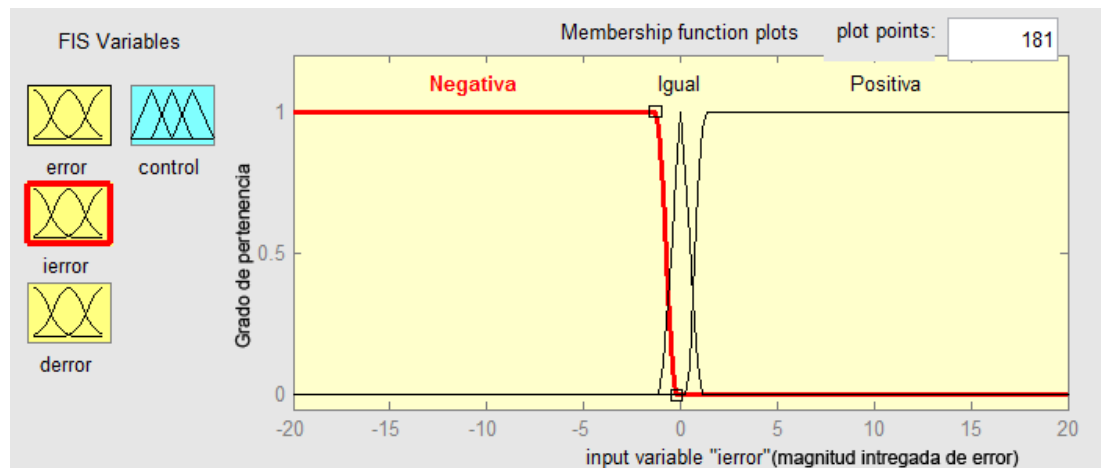


Figura 5.14 Conjuntos difusos para la entrada derivada del error de la variable guiñada

Por último, para la entrada derivada del error se utilizan los mismos conjuntos que en la integral sólo que con intervalos de pertenencia diferentes (Figura 5.15).

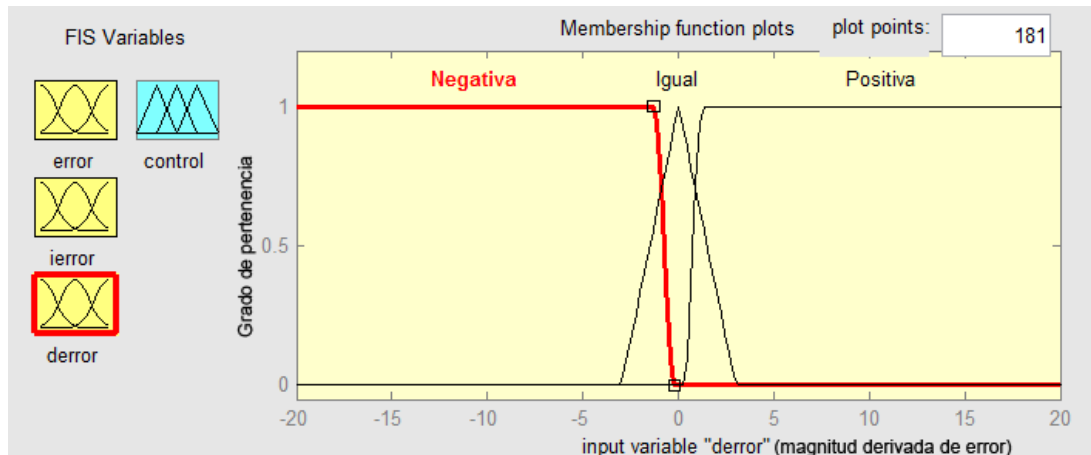


Figura 5.15 Conjuntos difusos para la entrada derivada del error de la variable guiñada.

La salida presenta los conjuntos borrosos im (girar mucho hacia la izquierda), i (girar a la izquierda), m (mantener), d (girar a la derecha) y dm (girar mucho a la derecha) (Figura 5.16).

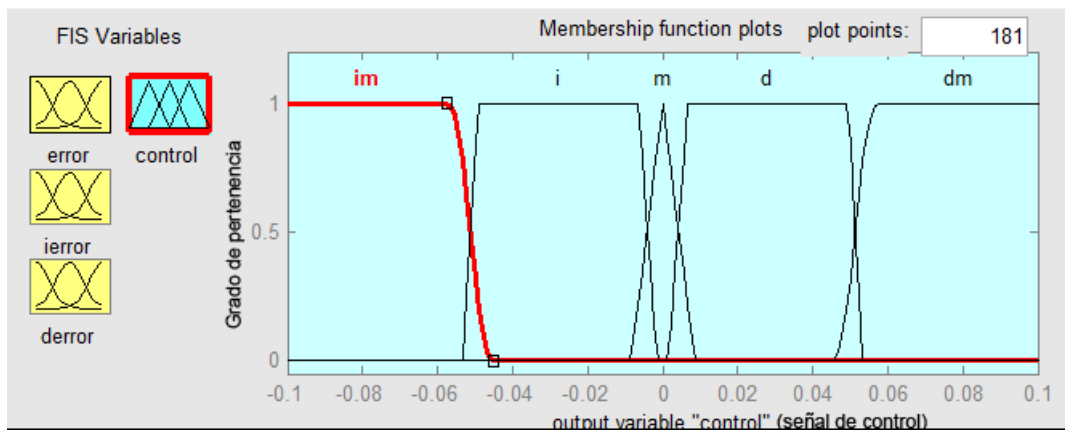


Figura 5.16 Conjuntos borrosos para la salida del control difuso guiñada

El conjunto de reglas utilizadas es el siguiente:

DErrorY-IErrorY/ErrorY	Sentido horario	Mantenerse	Sentido Antihorario
Negativa-Negativa	<i>dm</i>	<i>d</i>	<i>i</i>
Negativa-Igual	<i>dm</i>	<i>d</i>	<i>i</i>
Negativa-Positiva	<i>dm</i>	<i>d</i>	<i>i</i>
Igual-Negativa	<i>d</i>	<i>d</i>	<i>i</i>
Igual-Igual	<i>d</i>	<i>m</i>	<i>i</i>
Igual-Positiva	<i>d</i>	<i>i</i>	<i>i</i>
Positiva-Negativa	<i>d</i>	<i>i</i>	<i>im</i>
Positiva-Igual	<i>d</i>	<i>i</i>	<i>im</i>
Positiva-Positiva	<i>d</i>	<i>i</i>	<i>im</i>

5.2.3 Control del cabeceo

Se ha aplicado control PID convencional del cabeceo (ver 5.1 PID).

5.2.4 Control del alabeo

Se ha aplicado control PID clásico del alabeo (ver 5.1 PID).

5.2.5 Análisis de resultados

En este apartado se testeará el control de las dos variables para las que se ha realizado la implementación, aumentando la complejidad progresivamente, empezando primero con órdenes independientes referentes únicamente a cada una de las variables en cuestión y posteriormente con órdenes en las que intervienen varias, en las que además puede existir acoplamiento.

5.2.5.1 Resultados individuales

5.2.5.1.1 Variable altura

En este apartado los ángulos de cabeceo, alabeo y guiñada tienen valor 0.

(1) Prueba de ascenso de 5 metros a 15 metros (Figura 5.17).

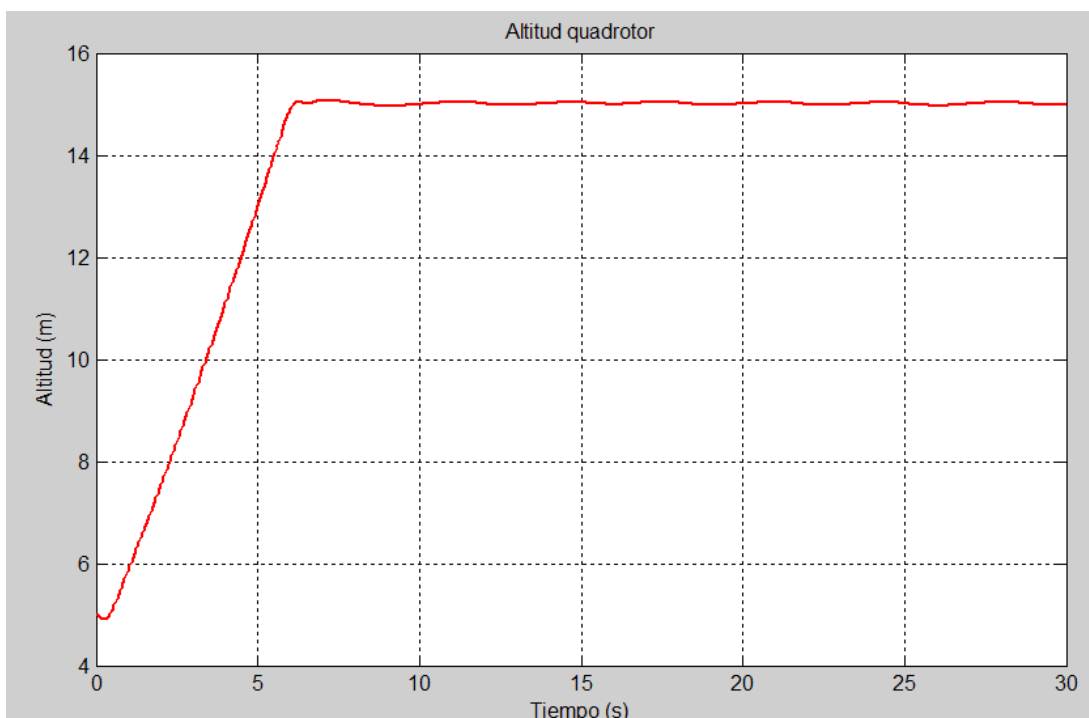


Figura 5.17 Simulación de ascenso de 5 m a 15 m

Se aprecia en la gráfica que se produce una subida limpia y rápida hasta aproximadamente el valor deseado que se alcanza en torno a los 6 segundos, manteniéndose estable a partir de ese momento el resto de la simulación.

(2) Prueba de descenso de 10 metros a 6 metros (Figura 5.18).

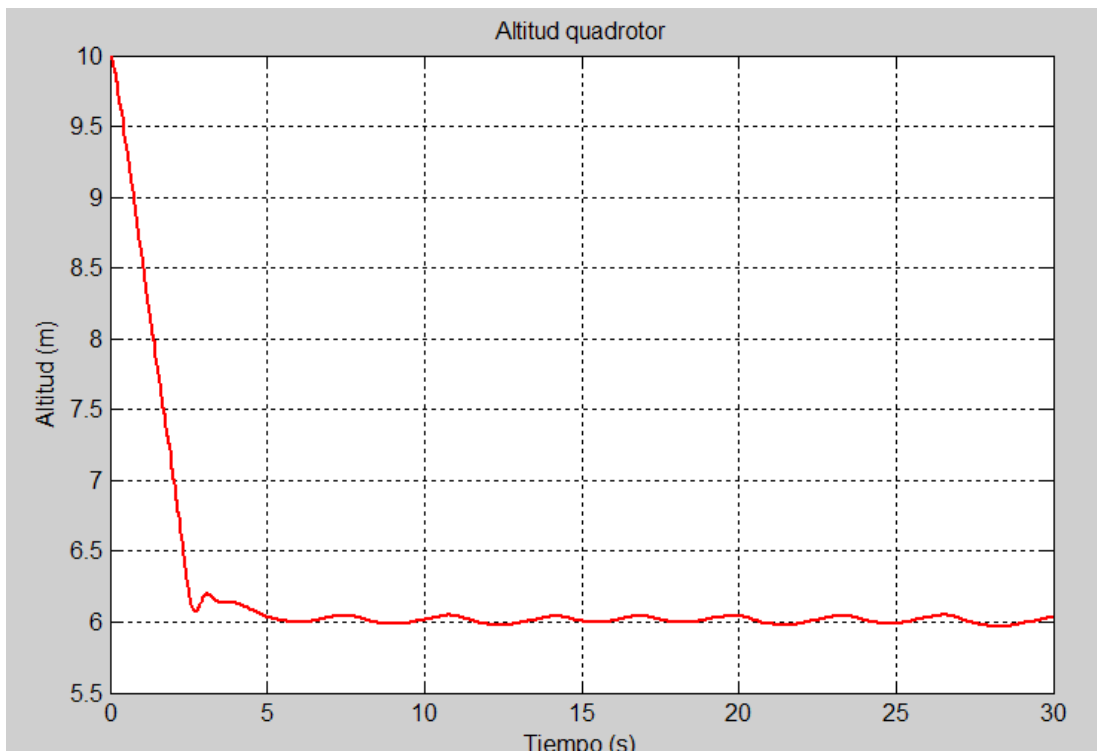


Figura 5.18 Simulación de descenso de 10 m a 6 m

Como se puede observar se produce un descenso rápido, quedándose un poco por encima del valor deseado poco antes de alcanzarlo casi por completo. Se aprecia que el resto de la simulación se mantiene estable con pequeñas oscilaciones (inferiores a 4 cm) en la altura objetivo.

5.2.5.1.2 Variable guiñada

En este apartado los ángulos de las variables alabeo y cabeceo serán iguales a 0 y la altura se mantendrá constante a 10 metros.

(1) Giro en sentido horario de 90° (Figura 5.19)

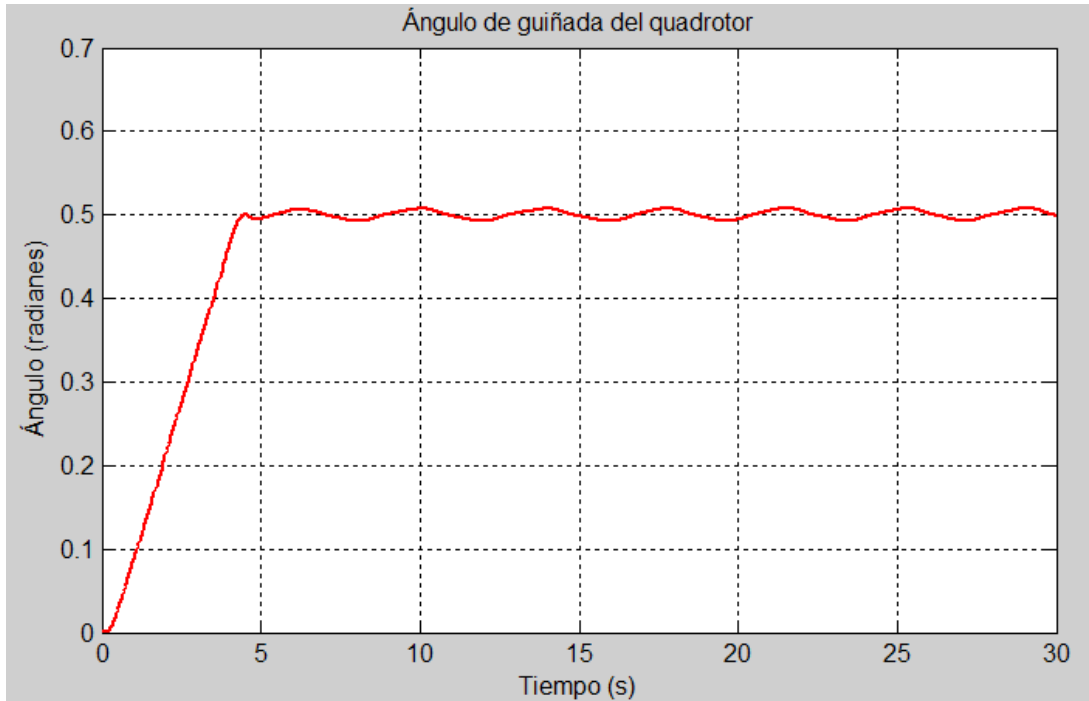


Figura 5.19 Simulación de variación de guiñada de 0 a 0.5 radianes

Se observa el cuatrorotor responde de forma rápida, alcanzando en 5 segundos el ángulo deseado, que en la gráfica viene expresado en radianes (0.5 rad).

(2) Giro en sentido antihorario de 90° (Figura 5.20)

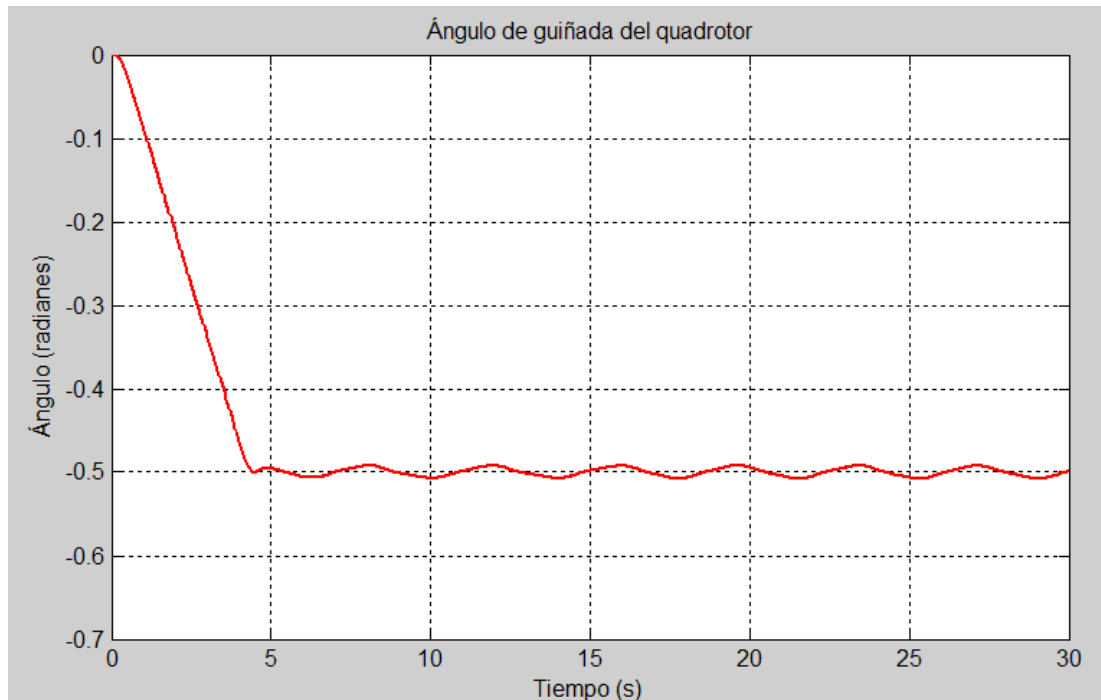


Figura 5.20 Simulación de variación de guiñada de 0 a -0.5 radianes

Las conclusiones que se pueden extraer de la observación de esta gráfica son análogas a la anterior.

5.2.5.2 Resultados combinados

(2) Ascenso de 2 a 15 metros con un ángulo de cabeceo de 0.1 radianes (18°), ángulo de alabeo de 0.083 radianes (15°) y ángulo de guiñada 0.25 radianes (45°)

Se presentan en primer lugar los resultados de cada variable para corroborar que se alcanzan los valores deseados (Figura 5.21, Figura 5.22, Figura 5.23 y Figura 5.24).

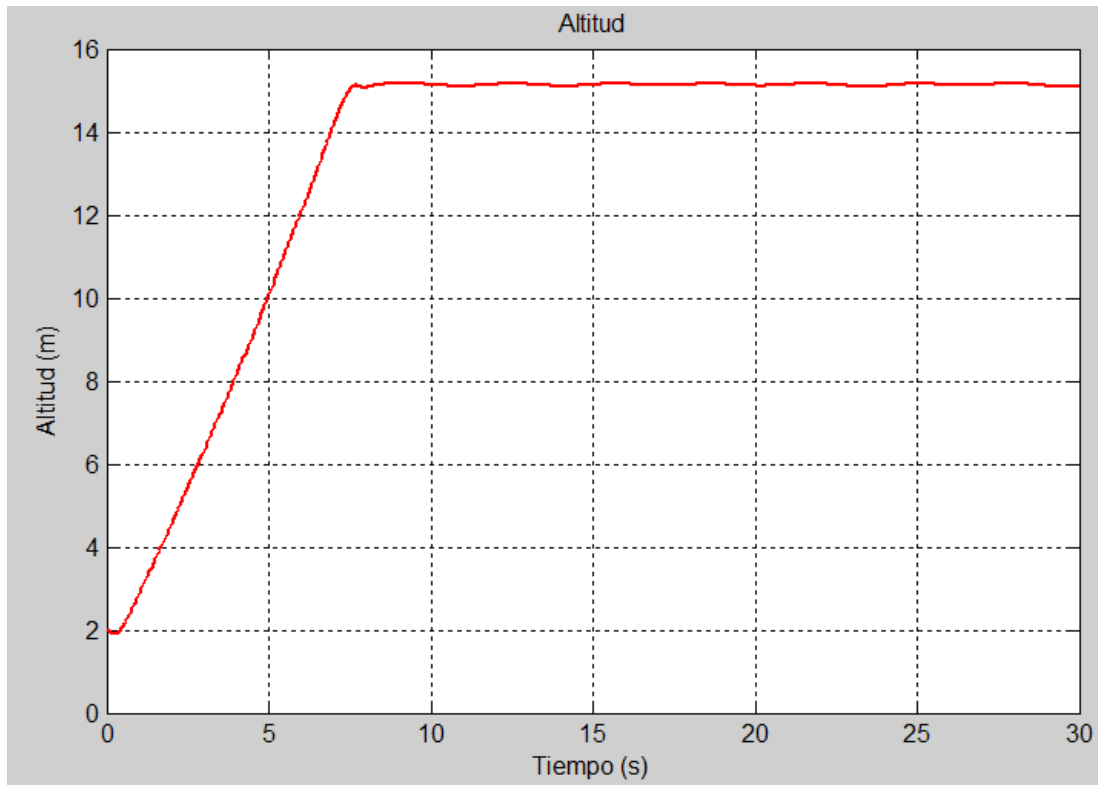


Figura 5.21 Simulación de ascenso de 2 a 15 metros

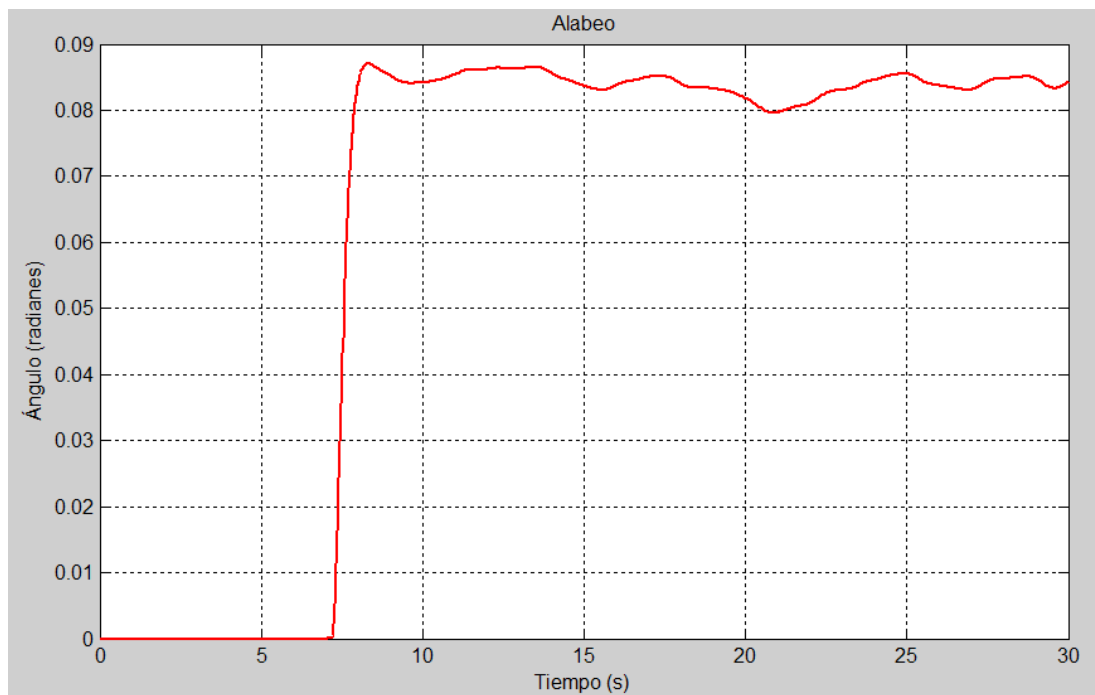


Figura 5.22 Simulación cambio de ángulo de cabeceo de 0 a 0.1 radianes

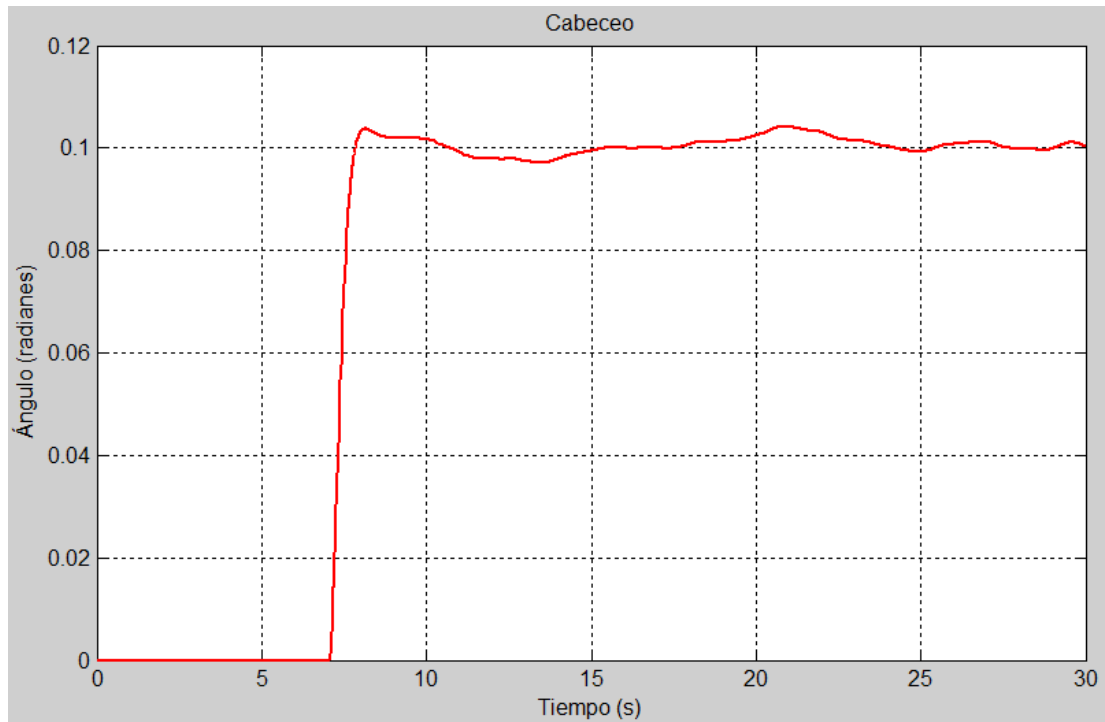


Figura 5.23 Simulación cambio de ángulo de cabeceo de 0 a 0.083 radianes

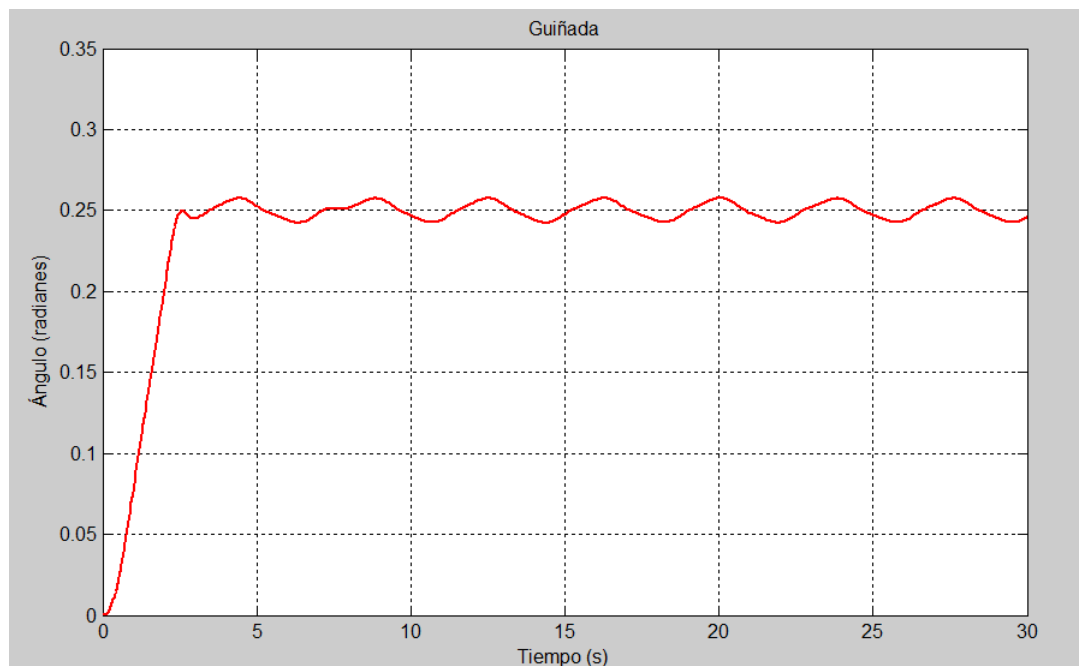


Figura 5.24: Simulación variación de guiñada de 0 a 0.25 radianes

Para apreciar realmente que han intervenido varias variables, a continuación se muestra la representación en planta del cuatrirotor (Figura 5.25)

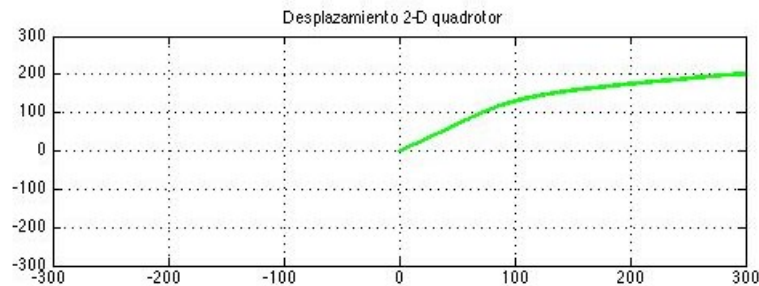


Figura 5.25 Desplazamiento del quadrotor en el eje X-Y

Ésta última gráfica es la que clarifica realmente lo que está pasando al cambiar los ángulos de cabeceo y alabeo desde sus valores iniciales a otros diferentes positivos: efectivamente el cuatrirotor al inclinarse hacia abajo y hacia la derecha esta provocando un desplazamiento en el primer cuadrante del eje X-Y.

5.3 Red neuronal

5.3.1 Neurocontrol de la altura

La herramienta con la que se ha realizado la implementación de las redes neuronales ha sido la toolbox `nntool` de matlab. Para realizar el control neuronal de la altura se han utilizado como datos de entrenamiento el par entrada-salida y `error_en_altura-potencia_aplicada_en_motores`, esto es, para que la red deduzca lo que tiene que hacer, la entrenaremos suministrándole el voltaje que hay que aplicar a los motores para una determinada diferencia de altura. Estos valores del voltaje se han obtenido del PID adaptativo .

5.3.2 Implementación y entrenamiento de la red

Mediante `nntool` se ha creado una red que dispone de una entrada con una neurona, el error, que es la diferencia entre la altura deseada y la actual del cuatrirotor, doce capas ocultas con doce neuronas cada una y una salida. La función

de transferencia utilizada en todas las capas ha sido la tangente hiperbólica sigmoïdal. De una forma gráfrica, esta red se puede apreciar en la Figura 5.26.

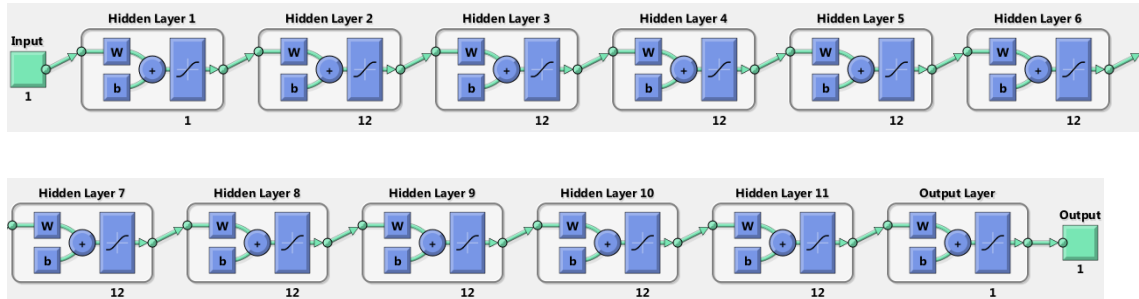


Figura 5.26: Red neuronal utilizada para el control de la altura

Para el entrenamiento se han utilizado un total de 35607 datos de entrada-salida, en un intervalo de +6 metros (esto es, sólo valido para ascensos) tomados de 0,25 en 0,25 metros (Figura 5.27).

Los parametros de entrenamiento de la red han sido los siguientes:

- Numero de iteraciones: 15000
- Numero de validaciones: 30
- Error mınimo para validacion: 0.001

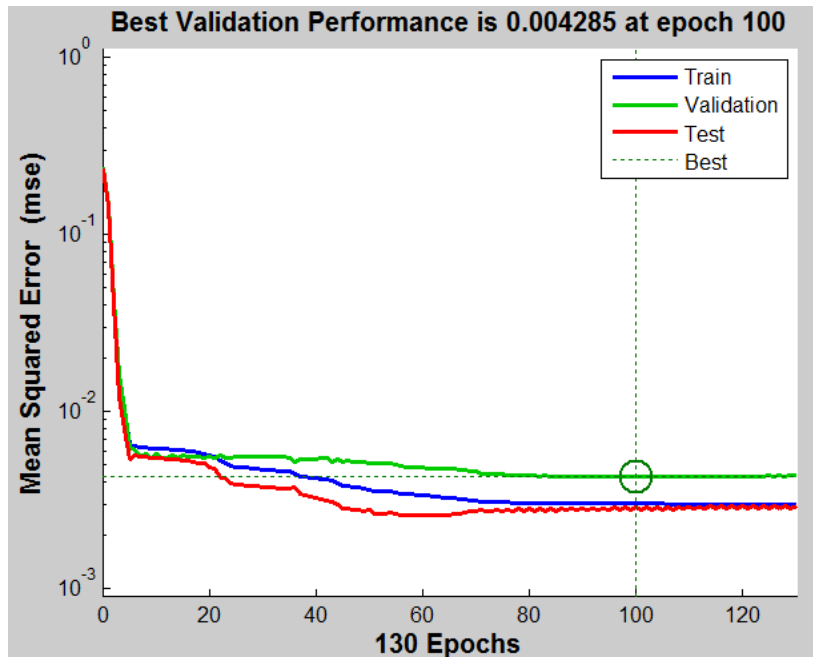


Figura 5.27: Evolución del aprendizaje de la red

Posteriormente se ha generado el correspondiente bloque para simulink utilizando esta red y se ha integrado en el modelo PID adaptativo sustituyendo al regulador que controla la altura.

5.3.3 Estudio de los resultados obtenidos

Se probará el funcionamiento de la red para un ascenso desde 5 a 8 metros (Figura 5.28). Como se puede apreciar la red no es capaz de realizar su cometido y el cuatrirotor, desciende de forma brusca y colisiona contra el suelo.

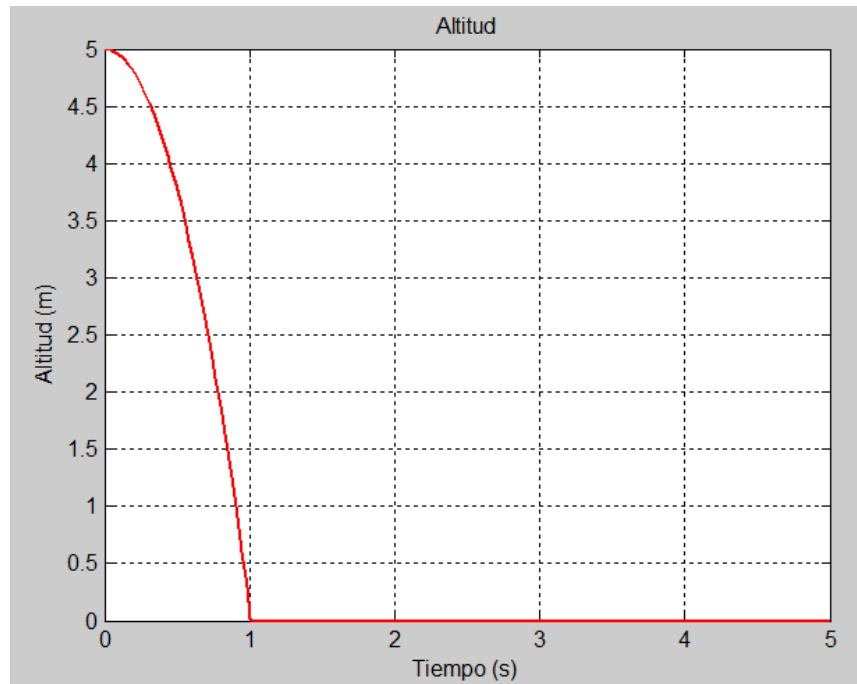


Figura 5.28: Intento fallido de ascenso del quadrotor de 5 a 8 metros

Creemos que esto se debe a que la red neuronal no es capaz de reaccionar con la misma sensibilidad y rapidez que el PID (problema similar al comentado en el fuzzy). ya que a pesar de incrementar el conjunto de datos de entrenamiento y variar los parámetros del mismo, los resultados obtenidos son los iguales.

6. Plataforma de simulación

6.1 Interfaz 2D

Una vez finalizada la simulación, desaparece la interfaz 3D en tiempo real y se muestran las gráficas y la información relevante en 2D. La primera gráfica representa la trayectoria en planta recorrida por el móvil en X e Y, o lo que es lo mismo, dibuja X respecto a Y. En la segunda se representa la altura del móvil en el tiempo (Figura 6.1).

Con esta forma de representar los resultados de una manera sencilla y práctica se obtiene la trayectoria del cuatrirotor en los tres ejes.

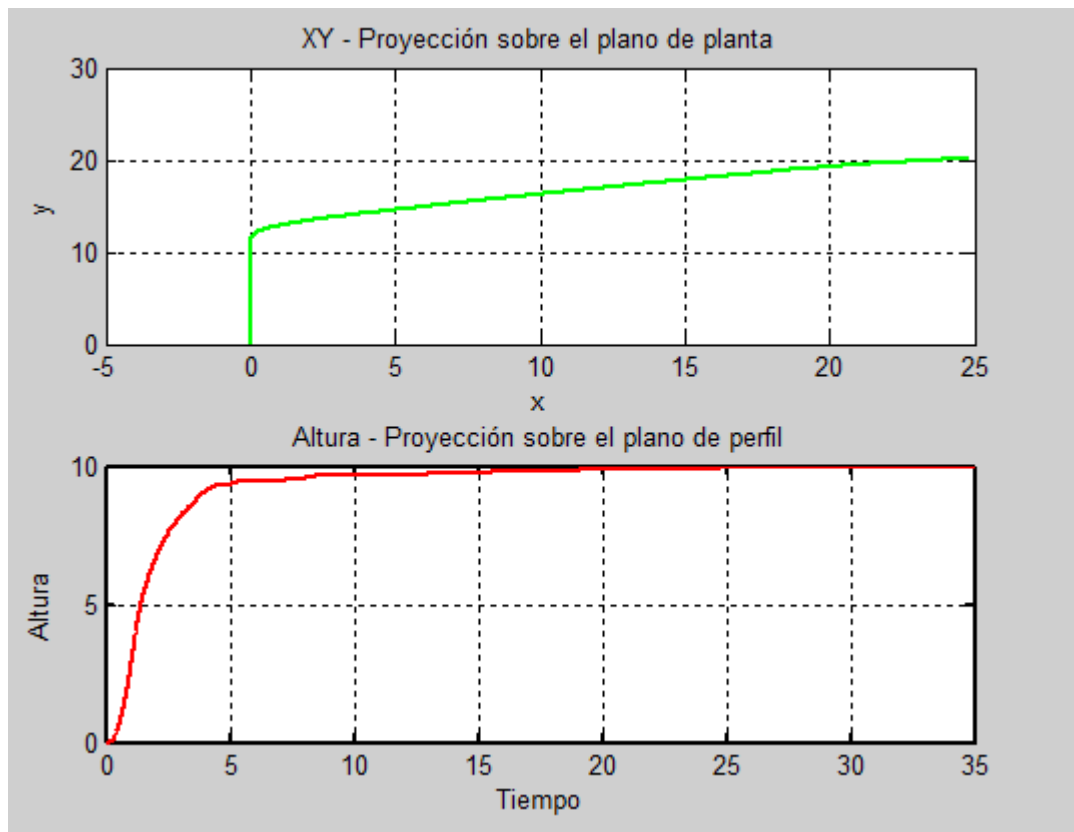


Figura 6.1: Interfaz 2D de la trayectoria en planta y en perfil

6.2 Interfaz 3D

De las diferentes opciones para la visualización en tres dimensiones se usa la proporcionada por Matlab-Simulink usada ya previamente en otros trabajos (Corke, 2011). Durante la simulación del cuatrirotor, se muestra una ventana donde se puede observar en 3D el resultado de la ejecución en tiempo real. En la parte superior se muestran los valores de la posición (x , y , z) en cada instante (Figura 6.2). Esta vista, resulta muy útil para poder observar con precisión qué maniobras está realizando el vehículo. Para representar el sentido de giro de los rotores se ha utilizado un código de color, el color verde indica un sentido de giro y el rojo el sentido opuesto.

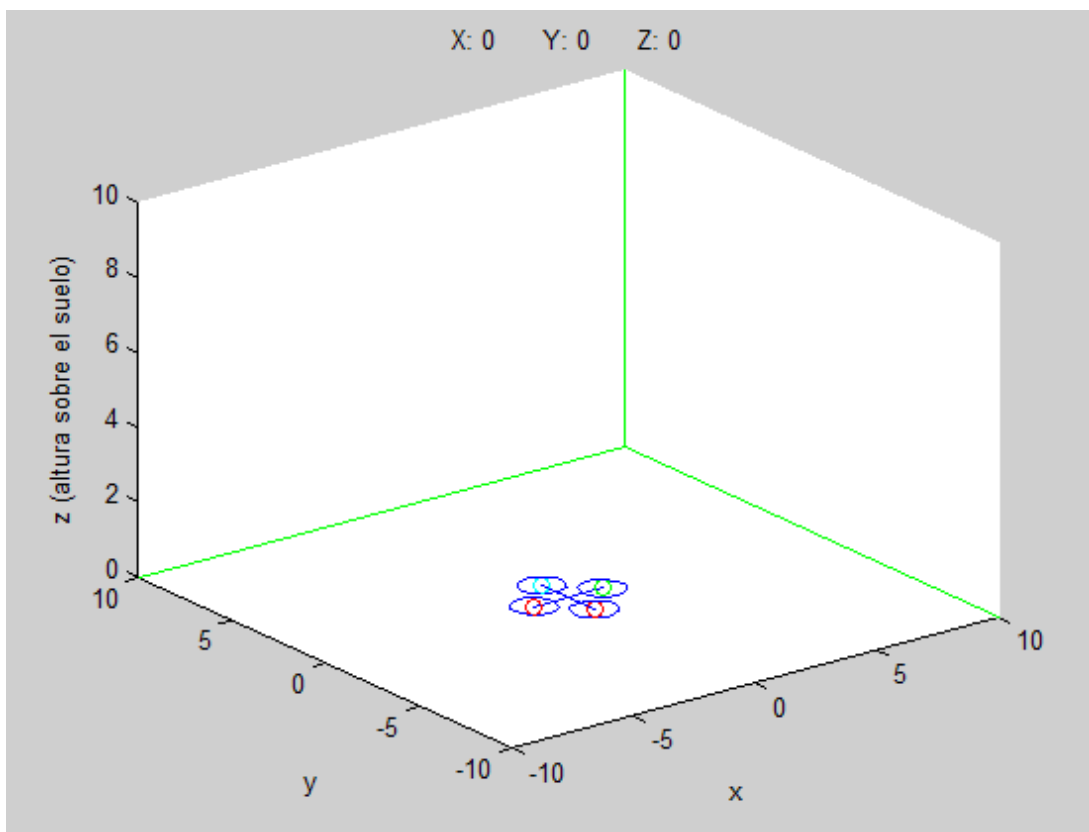


Figura 6.2: Interfaz 3D

En la parte central se presentan los ejes de coordenadas absolutos (en azul) y los relativos superpuestos (en verde) que acompaña al movimiento del vehículo si el

modo seguimiento está activado. El eje Z tiene centrada la altura con el valor mínimo y en los ejes X e Y el valor central indica el punto en el que se encuentra (siempre que esté activado el seguimiento). Esto es una ayuda extra para distinguirlo de la visión en perspectiva que puede conducir a errores de interpretación.

Como se ha expuesto anteriormente, existen dos sistemas de coordenadas. El azul, que corresponde con el sistema de coordenadas absoluto inicial, indicando desde donde comienza la simulación y el verde, que es relativo al movimiento de la aeronave y sirve para representar el efecto de perspectiva (Figura 6.3).

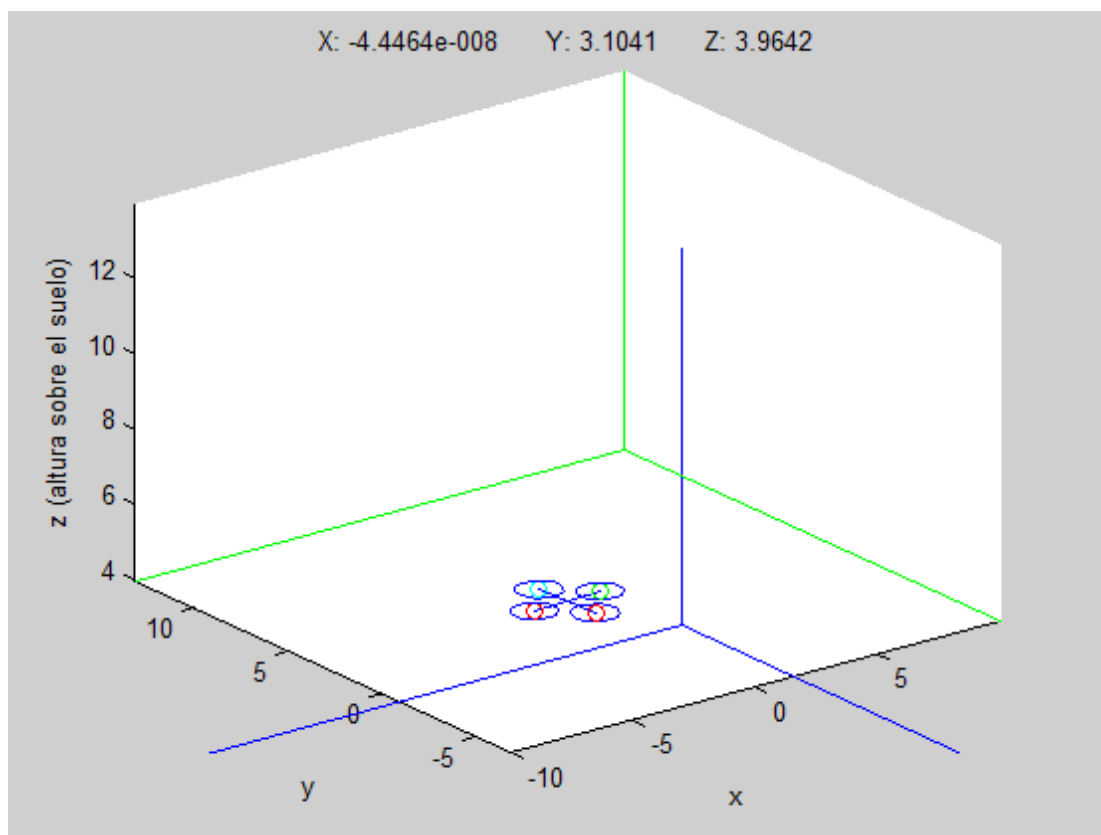


Figura 6.3: Los dos sistemas de coordenadas de la interfaz 3D

6.2.1 Modos

Es posible configurar la plataforma de simulación en los siguientes dos modos de operación: uno estático, donde la cámara no se desplaza, y otro dinámico, donde la cámara se desplaza siguiendo al móvil.

Si está activada la primera opción (estático), unas cruces rojas indicarán en los ejes de coordenadas el valor actual (aparte de la etiqueta de datos en la parte superior) y una cruz negra denota la proyección del centro del objeto en el suelo. Los límites de los ejes se amplían según sea necesario como también ocurre con el zoom según las condiciones dadas(Figura 6.4).

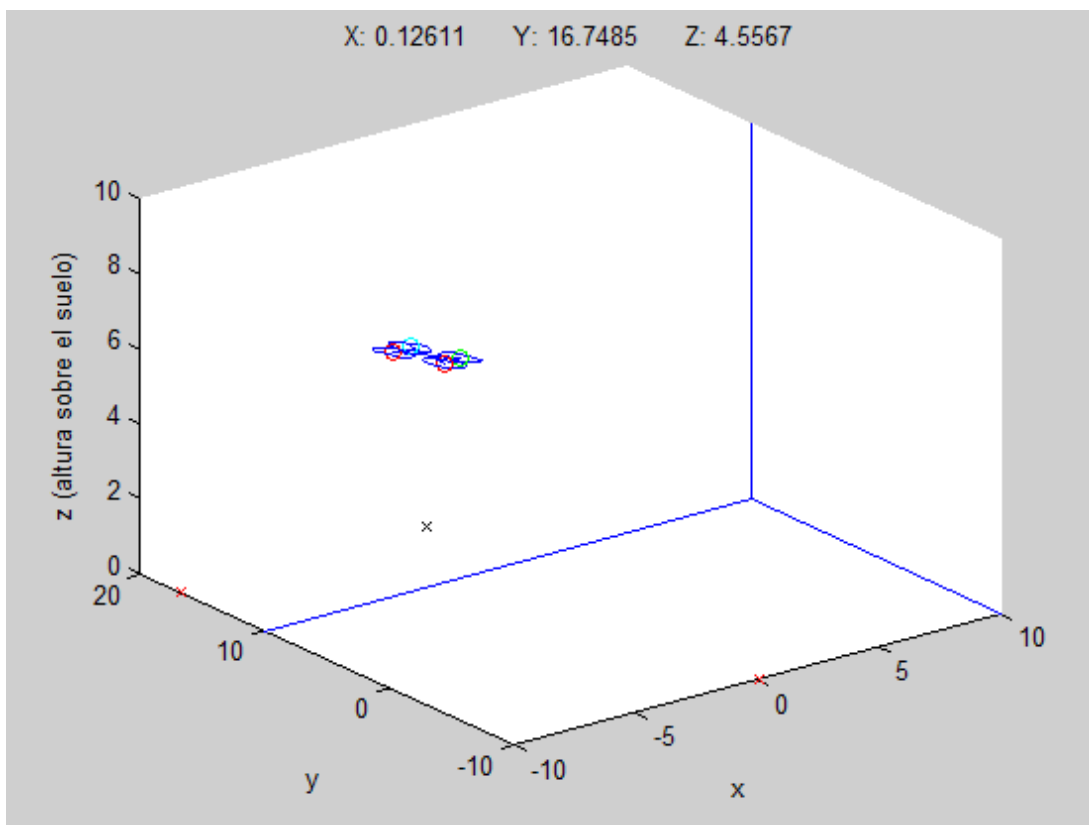


Figura 6.4: Modo estático

En el modo dinámico los ejes de referencia se modifican en función de la trayectoria dejando el objeto en el centro de la representación (Figura 6.5).

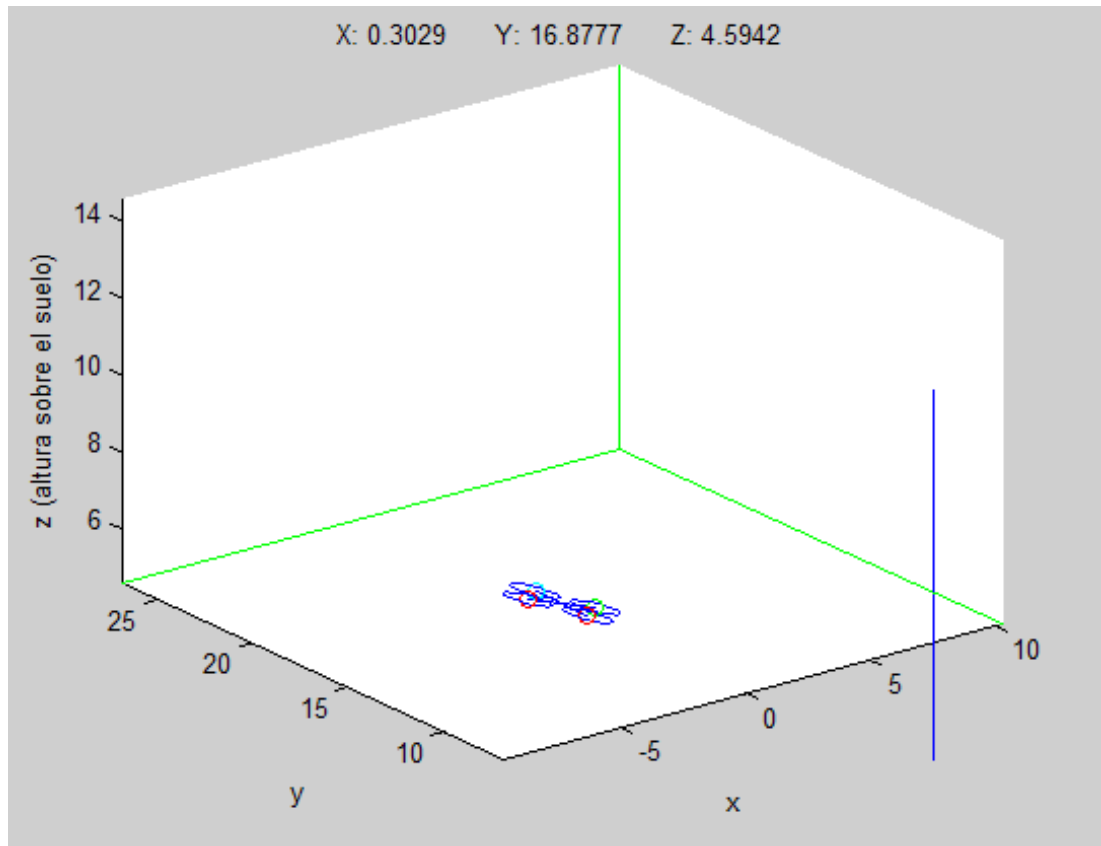


Figura 6.5: Modo dinámico

Para cambiar de modo, en el bloque Simulink Vista 3D basta con cambiar la segunda variable de la función (las entradas en las propiedades del objeto). Uno (activado), para modo dinámico y cero (desactivado), para modo estático.

6.2.2 Ajustes

El escalado de la figura se puede ajustar también poniendo un valor máximo para los ejes X e Y y otro para el Z. En la Figura 6.6 se muestra como quedaría poniendo de límites 20 para las tres componentes.

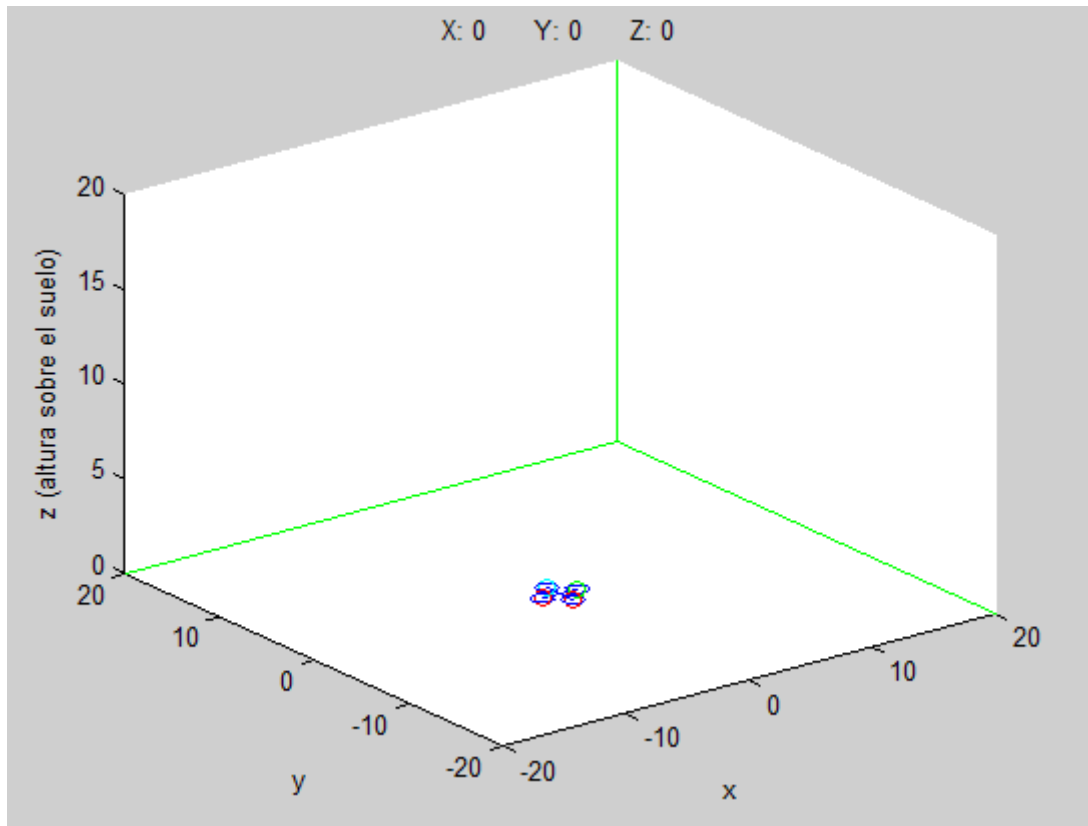


Figura 6.6: Vista con escalado al 20

Para cambiar de escalado en el bloque Simulink Vista 3D basta con cambiar la primera variable de la función (las entradas en las propiedades del objeto). Es un array de dos valores (máximo x-y, máximo z) o de uno (máximo x-y-z).

7. Conclusiones y trabajos futuros

Conclusiones

- Se ha mejorado el comportamiento del modelo mediante la inclusión de la fuerza de rozamiento y de una referencia de suelo real.
- Se ha conseguido el seguimiento dinámico de trayectorias con una excelente precisión en los dos modelos utilizados.
- Se han diseñado y probado en simulación controladores PID, controladores difusos y basados en redes neuronales. Destacar que con los primeros se han obtenido unos excelentes resultados.
- Se han implementado desde cero una vista en 2D y en 3D que permiten observar con alto nivel de detalle el comportamiento del cuatrirotor hasta alcanzar los parámetros deseados. La vista 3D además permite distinguir el sentido de rotación de los motores y los ángulos de inclinación de la aeronave.
- Se han sintonizado todos los tipos de controles en los dos modelos para ampliar lo máximo posible el rango de operación en rango único.
- A la vista de los resultados obtenidos hemos determinado que el control fuzzy no es el idóneo en este caso concreto por la gran complejidad que entraña su implementación con la que además no se obtienen resultados tan buenos como con la técnica PID.
- Dados los malos resultados obtenidos con redes neuronales no se recomienda el uso de este tipo de control para otros proyectos de carácter similar.
- El uso de PID en cascada unidos a calculadores de parámetros que tengan como entrada los propios parámetros del vuelo (velocidad y altitud) ha demostrado ser la técnica con mejores resultados y por ello se recomienda para futuros proyectos o ampliaciones de este mismo.

Trabajos futuros

- Probar y refinar el modelo en base a las plantas reales (cutrirotor de Quanser y cuatrirotor de la UCM) para concluir el modelado en su máxima expresión y asegurar la veracidad de los sistemas de control citados en el presente documento.
- Dotar al control de posiciones en ambos modelos de un mecanismo de cálculo de parámetros igual que el utilizado para el control de altitud de forma que el rango de tramo único se amplíe notablemente.

8. Bibliografía

- ÅSTRÖM, K.J., HÄGGLUND, T. 2009. Control PID avanzado. Pearson Educación S.A.
- CASTILLO, P., GARCÍA, P., LOZANO, R. & ALBERTOS, P. 2007. Modelado y estabilización de un helicóptero con cuatro rotores. Revista Iberoamericana de Automática e Informática Industrial.
- CORKE, P. 2011. Robotics, Vision and Control Fundamental Algorithms in MATLAB®. Springer Berlin / Heidelberg.
- COTERO OCHOA, J. B. Control clásico y control inteligente [Online]. Available: <http://www.mde.iteso.mx/titulacion/estudio%20de%20caso/ejemplos/marco%20teorico%20en%20progreso.pdf>.
- SÁNCHEZ BENÍTEZ, D., SANTOS PEÑAS, M. Apuntes de la asignatura Inteligencia Artificial Aplicada al Control: “Tutorial de iniciación a nntool: Implementación de la función XOR mediante una red neuronal”. Universidad Complutense de Madrid. 2011.
- OGATA, K. *Ingeniería de control moderna*. Dormido Canto,S., Dormido Canto, R. (trad.); Dormido Bencomo, S. (ed. lit.). Madrid: Pearson Educación. 2010. 894 p.
- MUKAIDONO, M. *Fuzzy logic for beginners*. Singapur: World Scientific. 2004. 105 p.
- KOSKO, B. *Neural networks and fuzzy systems: a dynamical systems approach to machine intelligence*. Nueva Jersey: Prentice-Hall Internacional. 1992. 449 p.

Anexo I: Especificaciones Técnicas del Qball-X4 UAV

Característica	Descripción
Diametro	0,7m
Baterías	2 baterías LiPo , 2500mAh, 3 celdas
Autonomía de duración del vuelo	15 minutos de tiempo de vuelo por cada carga
Motores	4 motores de 740KW con hélices de 10 pulgadas
Peso con las baterías	1410g
Capacidad de carga máxima	400g

Anexo II: Ecuaciones cinemáticas y dinámicas del cuatrirotor

Éste se compone de doce variables de entrada que son el vector posición x, y, z y la velocidad angular p, q, r respecto a los ejes de coordenadas; el vector de velocidad lineal u, v, w y los ángulos de Euler *cabeceo*, *alabeo* y *guiñada* (*cabeceo*, *alabeo*, *guiñada* en inglés) respecto a los ejes del cuatrirotor como se puede ver en la

El vector posición $\mathbf{x} = (x, y, z)$ está definido respecto al sistema inercial, mientras que el vector de velocidad $\mathbf{v}_i = (u, v, w)$ lo está respecto al sistema unido al cuadrorotor. Para pasar de uno a otro tendremos que usar la matriz de Euler para giros sobre los tres ejes. Suponiendo que los giros se producen (en sentido horario) primero respecto a

ANEXO

k_v , luego respecto a j_v y finalmente respecto a i_v , la matriz de rotación tendrá la conocida forma:

$$R_{v\bar{i}}(\varphi, \theta, \psi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Entonces:

$$R_{i/v} = R_{v\bar{i}}^{-1} = R_{v\bar{i}}^T = \begin{bmatrix} c(\theta)c(\psi) & s(\phi)s(\theta)c(\psi) - c(\phi)s(\psi) & c(\phi)s(\theta)c(\psi) + s(\phi)s(\psi) \\ c(\theta)s(\psi) & s(\phi)s(\theta)s(\psi) + c(\phi)c(\psi) & c(\phi)s(\theta)s(\psi) - s(\phi)c(\psi) \\ -s(\theta) & s(\phi)c(\theta) & c(\phi)c(\theta) \end{bmatrix}$$

Tendremos que:

$$\frac{d}{dt} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = R_{i/v}(\varphi, \theta, \psi) \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (\text{A})^1$$

Para relacionar (p, q, r) con $d/dt(\varphi, \theta, \psi)$, consideraremos las rotaciones dadas por SR_v paso a paso.

$$\begin{aligned} \begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \\ &+ \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & \sin \phi \\ 0 & -\sin \phi & \cos \phi \end{bmatrix} \begin{bmatrix} \cos \theta & 0 & -\sin \theta \\ 0 & 1 & 0 \\ \sin \theta & 0 & \cos \theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = \\ &= \begin{bmatrix} 1 & 0 & -s(\theta) \\ 0 & c(\phi) & s(\phi)c(\theta) \\ 0 & -s(\phi) & c(\phi)c(\theta) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{aligned}$$

E invirtiendo:[Slotine][Stevens]

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s(\phi)t(\theta) & c(\phi)t(\theta) \\ 0 & c(\phi) & -s(\phi) \\ 0 & s(\phi)s\theta c(\theta) & c(\phi)s\theta c(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (\text{B})$$

La fuerza de rozamiento que sufre el vehículo será:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} \frac{(c(\phi)s(\theta)c(\psi)+s(\phi)s(\psi))\sum_{i=1}^4 F_i - 1/2 C_x A_c \rho U|U|}{M_T} \\ \frac{(c(\phi)s(\theta)s(\psi)-s(\phi)c(\psi))\sum_{i=1}^4 F_i - 1/2 C_y A_c \rho V|V|}{M_T} \\ \frac{c(\phi)c(\theta)\sum_{i=1}^4 F_i}{M_T} - g - \frac{1/2 C_z A_c \rho W|W|}{M_T} \end{bmatrix} \quad (\text{C})$$

Las ecuaciones de Euler que describen como las componentes del vector velocidad angular evolucionan en el tiempo, en respuesta a los torques aplicados [Fried]:

$$\begin{aligned} d(\mathbf{J}\boldsymbol{\omega})/dt + \boldsymbol{\omega} \times (\mathbf{J}\boldsymbol{\omega}) &= (\tau_\phi, \tau_\theta, \tau_\psi) \\ \mathbf{J} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} + \begin{bmatrix} i & j & k \\ p & q & r \\ J_x p & J_y q & J_z r \end{bmatrix} &= \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \\ \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \mathbf{J}^{-1} \left\{ \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} - \begin{bmatrix} i & j & k \\ p & q & r \\ J_x p & J_y q & J_z r \end{bmatrix} \right\} &= \begin{bmatrix} \tau_\phi/J_x \\ \tau_\theta/J_y \\ \tau_\psi/J_z \end{bmatrix} + \begin{bmatrix} \frac{(J_x - J_z)}{J_x} qr \\ \frac{(J_z - J_x)}{J_y} pr \\ \frac{(J_z - J_y)}{J_z} pq \end{bmatrix} \quad (\text{D}) \end{aligned}$$

Siendo $(\tau_\phi, \tau_\theta, \tau_\psi)$ y (J_x, J_y, J_z) los anteriormente expuestos.

Las ecuaciones (A), (B), (C), y (D) forman el sistema de ecuaciones que describen al cuatrorotor.