



CARTA DE AUTORIZACIÓN

CÓDIGO

AP-BIB-FO-06

VERSIÓN

1

VIGENCIA

2014

PÁGINA

1 de 1

Neiva, 20 de septiembre del 2018

Señores

CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD SURCOLOMBIANA

Ciudad

El (Los) suscrito(s):

Fernando Campos Archila, con C.C. No. 1075297858,

Valentina Pinzón Saavedra, con C.C. No. 1075287028,

autor(es) de la tesis y/o trabajo de grado titulado Sistema de control de altura del quadrotor Ar. Drone 2.0 mediante lógica difusa en un ambiente controlado presentado y aprobado en el año 2018 como requisito para optar al título de

Ingeniero electrónico

Autorizo (amos) al CENTRO DE INFORMACIÓN Y DOCUMENTACIÓN de la Universidad Surcolombiana para que con fines académicos, muestre al país y el exterior la producción intelectual de la Universidad Surcolombiana, a través de la visibilidad de su contenido de la siguiente manera:

- Los usuarios puedan consultar el contenido de este trabajo de grado en los sitios web que administra la Universidad, en bases de datos, repositorio digital, catálogos y en otros sitios web, redes y sistemas de información nacionales e internacionales “open access” y en las redes de información con las cuales tenga convenio la Institución.
- Permita la consulta, la reproducción y préstamo a los usuarios interesados en el contenido de este trabajo, para todos los usos que tengan finalidad académica, ya sea en formato Cd-Rom o digital desde internet, intranet, etc., y en general para cualquier formato conocido o por conocer, dentro de los términos establecidos en la Ley 23 de 1982, Ley 44 de 1993, Decisión Andina 351 de 1993, Decreto 460 de 1995 y demás normas generales sobre la materia.
- Continúo conservando los correspondientes derechos sin modificación o restricción alguna; puesto que de acuerdo con la legislación colombiana aplicable, el presente es un acuerdo jurídico que en ningún caso conlleva la enajenación del derecho de autor y sus conexos.

De conformidad con lo establecido en el artículo 30 de la Ley 23 de 1982 y el artículo 11 de la Decisión Andina 351 de 1993, “Los derechos morales sobre el trabajo son propiedad de los autores”, los cuales son irrenunciables, imprescriptibles, inembargables e inalienables.

EL AUTOR/ESTUDIANTE: Fernando Campos Archila

EL AUTOR/ESTUDIANTE: Valentina Pinzón Saavedra

Firma: 

Firma: 

Vigilada Mineducación



**TÍTULO COMPLETO DEL TRABAJO:** Sistema de control de altura del quadrotor AR. Drone 2.0 mediante lógica difusa en un ambiente controlado

**AUTOR O AUTORES:**

Primero y Segundo Apellido	Primero y Segundo Nombre
Campos Archila	Fernando
Pinzón Saavedra	Valentina

**DIRECTOR Y CODIRECTOR TESIS:**

Primero y Segundo Apellido	Primero y Segundo Nombre
Robayo Betancourt	Faiber Ignacio

**ASESOR (ES):**

Primero y Segundo Apellido	Primero y Segundo Nombre
----------------------------	--------------------------

**PARA OPTAR AL TÍTULO DE:** Ingeniero electrónico

**FACULTAD:** Ingeniería

**PROGRAMA O POSGRADO:** Electrónica

**CIUDAD:** Neiva

**AÑO DE PRESENTACIÓN:** 2018

**NÚMERO DE PÁGINAS:** 96

**TIPO DE ILUSTRACIONES (Marcar con una X):**

Diagramas  Fotografías  Grabaciones en discos \_\_\_ Ilustraciones en general  Grabados \_\_\_  
Láminas \_\_\_ Litografías \_\_\_ Mapas \_\_\_ Música impresa \_\_\_ Planos \_\_\_ Retratos \_\_\_ Sin ilustraciones \_\_\_  
Tablas o Cuadros \_\_\_

Vigilada mieducación

La versión vigente y controlada de este documento, solo podrá ser consultada a través del sitio web Institucional [www.usco.edu.co](http://www.usco.edu.co), link Sistema Gestión de Calidad. La copia o impresión diferente a la publicada, será considerada como documento no controlado y su uso indebido no es de responsabilidad de la Universidad Surcolombiana.



**SOFTWARE** requerido y/o especializado para la lectura del documento: Adobe reader o cualquier lector pdf.

**MATERIAL ANEXO:**

**PREMIO O DISTINCIÓN** (En caso de ser LAUREADAS o Meritoria):

**PALABRAS CLAVES EN ESPAÑOL E INGLÉS:**

<u>Español</u>	<u>Inglés</u>	<u>Español</u>	<u>Inglés</u>
1. Control	Control	6. Cabeceo	Pitch
2. Difuso	Fuzzy	7. Alabeo	Roll
3. dron	drone	8. Guiñado	Yaw
4. AR. Drone 2.0	Ar. Drone 2.0	9. Altura	Altitude
5. Quadrotor	Quadrotor	10. Matlab	Matlab

**RESUMEN DEL CONTENIDO:** (Máximo 250 palabras)

El sistema de control de altura del quadrotor AR. Drone 2.0 aplicando lógica difusa en un ambiente previamente establecido dispone de un sistema de altura controlado tanto en simulación como en la plataforma real. Se desarrollan tres controladores mediante la lógica difusa cuyos parámetros son obtenidos de los sensores de tal manera que permita controlar altura y ángulos de orientación (Pitch, Roll y Yaw) siempre y cuando se consideren ciertos niveles de carga de batería para que el sistema no se vuelva inestable.

Para la visualización e interacción con el dron, se diseña e implementa una interfaz de Matlab que permite la comunicación entre el usuario y todas las funciones del sistema de tal manera que se puede elegir el modo de ejecución, seguir los parámetros de referencia de manera autónoma, almacenar dato para un posterior análisis, visualizar los desplazamientos para observar la eficiencia, entre otras



**ABSTRACT:** (Máximo 250 palabras)

The altitude control system of quadrotor AR. Drone 2,0 applying fuzzy logic in an previously established environment has a height system controlled both in simulation and in the real platform. Three controllers are developed by fuzzy logic whose parameters are obtained from the sensors in such a way that it allows to control height and orientation angles (Pitch, Roll and Yaw) as long as certain levels of battery charge are considered so that the system is not become unstable.

For the visualization and interaction with drone, a Matlab interface is designed and implemented that allows communication between the user and all system functions in such a way that the mode execution can be chosen, following the reference parameters autonomously, store data for a later analysis, visualize the displacements to observe the e-ciency, among others.

#### APROBACION DE LA TESIS

Nombre Presidente Jurado:

Firma:

Nombre Jurado: José de Jesús Salgado Patrón

Firma:

Nombre Jurado: Diego Fernando Sendoya

Firma:

**SISTEMA DE CONTROL DE ALTURA DEL QUADROTOR AR.  
DRONE 2.0 MEDIANTE LÓGICA DIFUSA EN UN AMBIENTE  
CONTROLADO**

**FERNANDO CAMPOS ARCHILA  
VALENTINA PINZÓN SAAVEDRA**

**UNIVERSIDAD SURCOLOMBIANA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
NEIVA  
2018**

**SISTEMA DE CONTROL DE ALTURA DEL QUADROTOR AR.  
DRONE 2.0 MEDIANTE LÓGICA DIFUSA EN UN AMBIENTE  
CONTROLADO**

**FERNANDO CAMPOS ARCHILA  
VALENTINA PINZÓN SAAVEDRA**

**Trabajo De Grado Para Optar Al Título De Ingeniero Electrónico**

**Director  
Faiber Ignacio Robayo Betancourt  
Mag. en Ingeniería de Control Industrial**

**UNIVERSIDAD SURCOLOMBIANA  
FACULTAD DE INGENIERÍA  
PROGRAMA DE INGENIERÍA ELECTRÓNICA  
NEIVA  
2018**

Nota de aceptación:

---

---

---

---

---

---

---

---

Firma del presidente del Jurado

---

Firma del Jurado

---

Firma del Jurado

Neiva, Septiembre 19 de 2018

## DEDICATORIA

*Principalmente a mis padres los cuales han sido la base y el impulso para avanzar hasta ahora. A mis hermanos por la colaboración que me han brindado. A mi compañera de proyecto de grado por el trabajo realizado. A mis instructores por la formación recibida durante la carrera, en especial al ingeniero Faiber Robayo por su instrucción, su guía durante el proyecto. Por último a mis amigos y compañeros que de una u otra forma me brindaron su ayuda, ánimo y confianza.*

***Fernando Campos Archila***

*A Dios y a la Santísima Virgen María, por estar conmigo en cada paso que doy, por fortalecer mi corazón e iluminar mi mente, por colmarme de paciencia y sabiduría. A mi padre Moisés Pinzón por los ejemplos de perseverancia, constancia y entrega, a mi madre Nohora Saavedra por su amor y enseñanza constante. A mi hermano Juan Jesús para que vea en mí un ejemplo a seguir. A mi tía Mónica Pinzón que ha estado en cada una de mis etapas y a su esposo Francisco Rosero por sus consejos y apoyo. A mi tía Enid Pinzón y a su esposo Emil Gutiérrez por su generosidad y confianza. A mis abuelos Alonso Pinzón y Gladys Caicedo por quererme, apoyarme y velar por mí siempre, esto también se lo debo a ustedes. A Daniel Murcia y a su familia por su cariño y apoyo. A mis amigos Karito, Lau y Nico por su apoyo incondicional. A mi compañero de tesis Fernando por la paciencia y por el buen equipo que formamos. A los maestros, ingenieros y trabajadores de ingeniería por su acompañamiento en este gran camino.*

***Valentina Pinzón Saavedra***



## AGRADECIMIENTOS

*Recorrer en esta carrera no ha sido sencillo, por ello agradezco a mis padres Fernando Campos y Clementina Archila por su apoyo incondicional y consejos en cada etapa. A mi tía Nidia Elda Archila por estar pendiente y atenta de mí. A mis hermanos los cuales me han apoyada de diversas maneras para avanzar esta carrera. A los ingenieros de los cuales pude recibir la formación necesaria para llegar hasta acá en especial el Ingeniero Faiber Robayo por el tiempo la confianza depositada en nosotros. A mi compañera de proyecto Valentina Pinzón por el trabajo realizado en conjunto, la paciencia, pero sobre todo por la confianza que tuvo en mí. A mis amigos y compañeros que me brindaron su apoyo en múltiples ocasiones muchas veces sin requerirlo. Muchas Gracias por confiar en mí y ayudarme a cumplir este gran logro de mi vida.*

***Fernando Campos Archila***

*Primero a Dios por acompañarme y guiarme siempre. A mis padres, mi hermano, mis tías, mis abuelos por apoyarme en todo momento, por lo que me han inculcado, por creer en mí, por ser parte fundamental en mi vida y representar la unidad familiar. Al ingeniero Faiber Robayo por creer en nosotros. A la Universidad Surcolombiana por el apoyo y facilidades otorgadas. A mi compañero de tesis por su compañía y por haber tenido la paciencia necesaria. A mi novio. A mis amigos. A mis compañeros y futuros colegas. A todas y cada una de las personas con quienes he compartido a lo largo de mi carrera, hicieron de esto un gran sueño cumplido.*

***Valentina Pinzón Saavedra***

# TABLA DE CONTENIDOS

	Pág
<b>1. INTRODUCCIÓN</b>	<b>15</b>
<b>2. OBJETIVOS</b>	<b>16</b>
2.1. OBJETIVO GENERAL . . . . .	16
2.2. OBJETIVOS ESPECÍFICOS . . . . .	16
<b>3. MARCO TEÓRICO</b>	<b>17</b>
3.1. ANTECEDENTES . . . . .	17
3.2. VEHICULOS AEREOS NO TRIPULADOS(UAV) . . . . .	19
3.3. QUADROTOR AR. DRONE 2.0 . . . . .	20
MODELAMIENTO . . . . .	22
PLACA DE NAVEGACIÓN . . . . .	24
SISTEMA DE COMUNICACIÓN . . . . .	26
3.4. MATLAB . . . . .	27
3.5. ÁNGULOS DE EULER . . . . .	29
3.6. LÓGICA DIFUSA . . . . .	30
Funciones de membresía difusa . . . . .	31
Controlador lógico difuso . . . . .	31
<b>4. DISEÑO Y DESARROLLO DEL PROYECTO</b>	<b>35</b>
4.1. AR. DRONE SIMULINK DEVELOPMENT-KIT V1.1 . . . . .	35
AR. Drone Wi-Fi-Block . . . . .	36
AR. Drone Simulation Block . . . . .	37
Position Estimation . . . . .	37

Visualization Of Drone States . . . . .	38
Guidance Logic . . . . .	38
Baseline Controller . . . . .	39
4.2. CONTROL FUZZY . . . . .	40
Diseñador de lógica difusa . . . . .	40
Editor de funciones de membresía . . . . .	41
Editor de reglas . . . . .	41
Visor de reglas . . . . .	42
Visor de superficie . . . . .	43
4.3. INTERFAZ GRÁFICA . . . . .	49
Vista General . . . . .	49
Vista Parámetros . . . . .	51
Vista Simulación . . . . .	53
Vista Real . . . . .	55
4.4. IMPLEMENTACIÓN . . . . .	57
Simulación del Sistema . . . . .	57
Implementación del Sistema . . . . .	58
<b>5. RESULTADOS</b>	<b>62</b>
5.1. SISTEMA LÓGICO DIFUSO “FIS9” . . . . .	62
5.2. SISTEMA LÓGICO DIFUSO “FIS11GA” . . . . .	70
5.3. SISTEMA LÓGICO DIFUSO “FIS11TR” . . . . .	77
<b>6. CONCLUSIONES</b>	<b>86</b>
<b>7. RECOMENDACIONES</b>	<b>89</b>
<b>BIBLIOGRAFÍA</b>	<b>90</b>



## LISTA DE FIGURAS

	<b>Pág</b>
Figura 1. Tipos de aeronaves . . . . .	20
Figura 2. Dron carcasa interiores . . . . .	21
Figura 3. Dron carcasa exteriores . . . . .	21
Figura 4. Parrot AR. Drone 2.0 . . . . .	22
Figura 5. Modelamiento cinemático del helicóptero de cuatro rotores . . . . .	22
Figura 6. Placa de navegación del AR. Drone 2.0 . . . . .	24
Figura 7. Entorno MATLAB . . . . .	28
Figura 8. Entorno MATLAB SIMULINK . . . . .	29
Figura 9. Ángulos de Euler (a)Yaw (b)Pitch (c)Roll . . . . .	30
Figura 10. Diagrama proceso de inferencia difusa . . . . .	31
Figura 11. Proceso de fusificación . . . . .	32
Figura 12. Diagrama de bloques del hardware . . . . .	35
Figura 13. AR. Drone Wi-Fi Block . . . . .	36
Figura 14. AR. Drone Simulation Block . . . . .	37
Figura 15. Position estimation . . . . .	37
Figura 16. Visualization of Drone states . . . . .	38
Figura 17. Guidance Logic . . . . .	39
Figura 18. Baseline Controller . . . . .	39
Figura 19. Diseñador de lógica difusa . . . . .	40
Figura 20. Editor de funciones de membresía . . . . .	41
Figura 21. Editor de reglas . . . . .	42
Figura 22. Visor de reglas . . . . .	43
Figura 23. Visor de superficie . . . . .	44

Figura 24.	Nueve funciones Gaussianas para las variables de Entrada . . . . .	45
Figura 25.	Nueve funciones Triangulares para la variable de Salida . . . . .	45
Figura 26.	Once funciones Gaussianas para las variables de Entrada . . . . .	46
Figura 27.	Once funciones Triangulares para la variable de Salida . . . . .	46
Figura 28.	Once funciones Triangulares para las variables de Entrada . . . . .	47
Figura 29.	<b>Pitch</b> en el AR. Drone 2.0 . . . . .	47
Figura 30.	Variables de Entrada y Salida para el control difuso del Pitch . . . . .	48
Figura 31.	<b>Roll</b> en el AR. Drone 2.0 . . . . .	48
Figura 32.	<b>Yaw</b> en el AR. Drone 2.0 . . . . .	49
Figura 33.	Intrucciones Generales . . . . .	50
Figura 34.	Vista General . . . . .	50
Figura 35.	Instrucciones De Parámetros . . . . .	51
Figura 36.	Vista De Parámetros . . . . .	52
Figura 37.	Instrucciones De Simulación . . . . .	53
Figura 38.	Vista De Simulación . . . . .	54
Figura 39.	Gráfica Simulada De Ruta en tres dimensiones . . . . .	55
Figura 40.	Instrucciones Real . . . . .	55
Figura 41.	Vista Real . . . . .	56
Figura 42.	Implementación Simulada . . . . .	58
Figura 43.	Dron posicionado para implementación . . . . .	59
Figura 44.	Dron junto a su interfaz para implementación real . . . . .	60
Figura 45.	Dron en ejecución de altura . . . . .	61
Figura 46.	Respuesta simulada con 9 funciones . . . . .	63
Figura 47.	Respuesta real con 9 funciones . . . . .	63
Figura 48.	Respuesta ante perturbación 9 funciones . . . . .	64

Figura 49.	Respuesta simulada con multiples puntos 9 funciones . . . . .	65
Figura 50.	Respuesta real con multiples puntos 9 funciones . . . . .	65
Figura 51.	Respuesta ante perturbación con multiples puntos 9 funciones . . . . .	66
Figura 52.	Respuesta simulada de altura 9 funciones . . . . .	67
Figura 53.	Respuesta real de altura 9 funciones . . . . .	67
Figura 54.	Respuesta simulada vista 2D 9 funciones . . . . .	68
Figura 55.	Respuesta real vista 2D 9 funciones . . . . .	68
Figura 56.	Respuesta simulada vista superior 9 funciones . . . . .	69
Figura 57.	Respuesta real vista superior 9 funciones . . . . .	69
Figura 58.	Respuesta simulada vista 3D 9 funciones . . . . .	70
Figura 59.	Respuesta real vista 3D 9 funciones . . . . .	70
Figura 60.	Respuesta simulada con 11 funciones gaussianas . . . . .	71
Figura 61.	Respuesta real con 11 funciones gaussianas . . . . .	71
Figura 62.	Respuesta simulada múltiples puntos 11 funciones gaussianas . . . . .	72
Figura 63.	Respuesta real multiples puntos 11 funciones gaussianas . . . . .	73
Figura 64.	Respuesta simulada de altura 11 funciones gaussianas . . . . .	74
Figura 65.	Respuesta real de altura 11 funciones gaussianas . . . . .	74
Figura 66.	Respuesta simulada vista 2D 11 funciones gaussianas . . . . .	75
Figura 67.	Respuesta real vista 2D 11 funciones gaussianas . . . . .	75
Figura 68.	Respuesta simulada vista superior 11 funciones gaussianas . . . . .	76
Figura 69.	Respuesta real vista superior 11 funciones gaussianas . . . . .	76
Figura 70.	Respuesta simulada vista 3D 11 funciones gaussianas . . . . .	77
Figura 71.	Respuesta real vista 3D 11 funciones gaussianas . . . . .	77
Figura 72.	Respuesta simulada con 11 funciones triangulares . . . . .	78
Figura 73.	Respuesta real con 11 funciones triangulares . . . . .	79

Figura 74.	Respuesta simulada multiples puntos 11 funciones triangulares . . . . .	80
Figura 75.	Respuesta real multiples puntos 11 funciones triangulares . . . . .	80
Figura 76.	Respuesta simulada de altura 11 funciones triangulares . . . . .	81
Figura 77.	Respuesta real de altura 11 funciones triangulares . . . . .	81
Figura 78.	Respuesta simulada vista 2D 11 funciones triangulares . . . . .	82
Figura 79.	Respuesta real vista 2D 11 funciones triangulares . . . . .	82
Figura 80.	Respuesta simulada vista superior 11 funciones triangulares . . . . .	83
Figura 81.	Respuesta real vista superior 11 funciones triangulares . . . . .	83
Figura 82.	Respuesta simulada vista 3D 11 funciones triangulares . . . . .	84
Figura 83.	Respuesta real vista 3D 11 funciones triangulares . . . . .	84
Figura 84.	Diagrama interno del bloque AR. Drone Wi-Fi Block . . . . .	92
Figura 85.	Diagrama interno del bloque Position Estimation . . . . .	92
Figura 86.	Diagrama interno del bloque Visualization of block of Drone states . . . . .	93
Figura 87.	Diagrama interno del bloque Guidance logic . . . . .	93
Figura 88.	Diagrama interno del bloque Baseline controller . . . . .	94



## LISTA DE ANEXOS

	<b>Pág</b>
Anexo A. Diagrama interno del bloque AR. Drone Wi-Fi Block . . . . .	92
Anexo B. Diagrama interno del bloque Position Estimation . . . . .	92
Anexo C. Diagrama interno del bloque Visualization of block of Drone states . . . . .	93
Anexo D. Diagrama interno del bloque Guidance logic . . . . .	93
Anexo E. Diagrama interno del bloque Baseline controller . . . . .	94

## GLOSARIO

**ARM Cortex A8:** Es un núcleo de procesador de 32 bits.

**b/g/n:** Tipos de Wi-Fi, basado cada uno de ellos en un estándar IEEE 802.11.

**DHCP:** (Dynamic Host Configuration Protocol)El protocolo de configuración dinámica de host.

**Difusa:** Se basa en lo relativo de lo observado como posición diferencial.

**DSP:** (digital signal processor)Un procesador digital de señales.

**FPS:** (Frames per second) fotogramas por segundo.

**GB:** (Gigabyte)Es una unidad de almacenamiento de información.

**GHz:** (Gigahertz)Es una unidad de frecuencia.

**GPU:** (Graphics Processing Unit)Unidad de procesamiento gráfico.

**HD:** (High Definition)Alta definición.

**IP:** (Internet Protocol)Es un número que identifica, de manera lógica y jerárquica, a una Interfaz en red.

**IPv4:** (Internet Protocol version 4 )El Protocolo de Internet versión 4.

**LSB:** (Least significant bit)Es la posición del bit en un número entero binario.

**mAh:** Miliamperio-hora.

**MEMS:** (Microelectromechanical Systems)Sistemas Microelectromecánicos.

**MHz:** (Megahertz)Es una unidad de medida de la frecuencia.

**QVGA:** (Quarter Video Graphics Array)Es la resolución gráfica de 320x240 píxeles.

**rpm:** (Revoluciones por minuto)Es una unidad de frecuencia.

**TCP:** (Transmission Control Protocol)Protocolo de control de transmisión.

**UAS:** (Unmanned Aircraft System)Sistema aéreo no tripulado.

**UAV:** (Unmanned Aerial Vehicle)Vehículo aéreo no tripulado.

**UDP:** (User Datagram Protocol)El protocolo de datagramas de usuario.

**Wi-Fi:** Es una tecnología que permite la interconexión inalámbrica de dispositivos electrónicos.

## **RESUMEN**

El sistema de control de altura del quadrotor AR. Drone 2.0 aplicando lógica difusa en un ambiente previamente establecido dispone de un sistema de altura controlado tanto en simulación como en la plataforma real. Se desarrollan tres controladores mediante la lógica difusa cuyos parámetros son obtenidos de los sensores de tal manera que permita controlar altura y ángulos de orientación (Pitch, Roll y Yaw) siempre y cuando se consideren ciertos niveles de carga de batería para que el sistema no se vuelva inestable.

Para la visualización e interacción con el dron, se diseña e implementa una interfaz de Matlab que permite la comunicación entre el usuario y todas las funciones del sistema de tal manera que se puede elegir el modo de ejecución, seguir los parámetros de referencia de manera autónoma, almacenar datos para un posterior análisis, visualizar los desplazamientos para observar la eficiencia, entre otras.

### **PALABRAS CLAVES**

Ambiente controlado, control de altura, dron, lógica difusa, Pitch, Roll, Yaw.

## **ABSTRACT**

The altitude control system of quadrotor AR. Drone 2,0 applying fuzzy logic in an previously established environment has a height system controlled both in simulation and in the real platform. Three controllers are developed by fuzzy logic whose parameters are obtained from the sensors in such a way that it allows to control height and orientation angles (Pitch, Roll and Yaw) as long as certain levels of battery charge are considered so that the system is not become unstable.

For the visualization and interaction with drone, a Matlab interface is designed and implemented that allows communication between the user and all system functions in such a way that the mode execution can be chosen, following the reference parameters autonomously, store data for a later analysis, visualize the displacements to observe the efficiency, among others.

### **KEYWORDS**

Altitude control, controlled environment, drone, fuzzy logic, Pitch, Roll, Yaw.

# 1. INTRODUCCIÓN

A nivel académico e investigativo, se ha avanzado de forma exponencial en la importancia de la implementación de nuevas tecnologías que apoyen al ser humano en diferentes tareas que presenten un alto riesgo como lo son el uso de UAV, debido a su versatilidad. La investigación constante en la mecánica, funcionalidad y aplicaciones de los UAV ha hecho que sean cada vez más accesibles permitiendo conocer su clasificación que varía dependiendo de sus componentes. Una de sus clasificaciones depende de las hélices que el UAV tenga, en caso de tener cuatro hélices este es un quadrotor uno de los más conocidos en el mercado es el AR. Drone 2.0.

El avance tecnológico constante ha logrado que el AR. Drone 2.0, considerado un sistema altamente complejo, se pueda usar a través de técnicas de control inteligente que independientemente de su complejidad no requiera de su modelo matemático como lo hace la lógica difusa, mediante la cual es posible automatizar una de las principales funciones y la base de múltiples aplicaciones del AR. Drone 2.0 que consiste principalmente en el nivel de altura y el trazado de ruta brindándole autonomía y control siendo estas funciones utilizadas en diversas áreas para desempeñar tareas variadas.

Para el trabajo con procesos lógicos difusos existe en particular un software llamado MATLAB que es bastante empleado en ingeniería permitiendo que su cotidianidad y familiarización con estos procesos haga más sencillo asimilar la comunicación y el control que realiza el AR. Drone 2.0.

Diseñar e implementar un sistema de control de altura aplicando lógica difusa para el quadrotor AR. Drone 2.0 en un ambiente controlado permite seguir diversos puntos en un ambiente previamente establecido, no solo de altura sino en ubicación permitiendo hacer seguimiento de trayectorias. La instrumentación es parte importante al momento de ejecutar el control por lo tanto es indispensable hacer uso de los sensores para medir las diferentes variables relacionadas al vuelo. El uso del software MATLAB hace posible la realización de una interfaz gráfica que sea intuitiva y amigable de tal manera que permite la comunicación entre el aeromodelo y el ordenador donde se puede ingresar y modificar los niveles de altura del quadrotor, ingresar un seguimiento de trayectoria, registrar y presentar los niveles de altura ingresados y recorridos por el quadrotor disponiendo de la información relevante para futuras investigaciones o aplicaciones.

## **2. OBJETIVOS**

### **2.1. OBJETIVO GENERAL**

Diseñar e implementar un sistema de control de altura aplicando lógica difusa para el quadrotor AR. Drone 2.0 en un ambiente controlado.

### **2.2. OBJETIVOS ESPECÍFICOS**

- Diseñar el sistema de control de altura que permita seguir diversos puntos en un ambiente previamente establecido.
- Incorporar al sistema de control una interfaz que permita la comunicación entre el aeromodelo y el ordenador al igual que los niveles de altura del quadrotor.
- Registrar y presentar los niveles de altura ingresados y recorridos por el quadrotor, permitiendo que el usuario disponga de la información relevante para futuras y diversas investigaciones o aplicaciones.

## 3. MARCO TEÓRICO

### 3.1. ANTECEDENTES

Los vehículos aéreos no tripulados (UAV) surgieron ya hace un tiempo, sin embargo en la actualidad se ha incrementado exponencialmente su uso debido a su fácil acceso, los más populares son los drones, ya que son una herramienta que lleva a cabo múltiples aplicaciones en las cuales facilita el trabajo del hombre como la distribución de señal gratuita de internet por medio de drones como lo idea Facebook<sup>1</sup>, los trazados de mapas o modelos de elevación de terreno<sup>2</sup>, transporte y envío de mercancía<sup>3</sup>, búsqueda y rescate, agricultura entre otras.

Para realizar todas estas aplicaciones en un principio se tiene que controlar el vuelo del dron, para lograr este fin se diseñan e implementan múltiples sistemas por variados métodos de control:

Fendy Santoso, Matthew A. Garratt y su grupo de investigación en la Universidad de Nueva Gales del Sur Canberra, Canberra Australia propusieron diseñar un sistema de control no lineal híbrido para la dinámica de un quadrotor, usan un modelo que aprovecha las ventajas del control predictivo del modelo no lineal con las ventajas proporcionadas por un compensador de lógica difusa, obtienen una gran eficiencia en comparación a un sistema de control clásico PD, además de una gran robustez frente a las perturbaciones<sup>4</sup>.

Sandipan Saha y Sabiha Wadoo plantean controlar óptimamente de forma lineal un AR. Drone 2.0, usando un regulador lineal cuadrático (LQR) y un sistema de control lineal cuadrático Gaussiano (LQG) logrando una gran estabilidad en el dron con un tiempo de estado estacionario de 2.5 a 3 segundos, además aplican un filtro Kalman teniendo en cuenta la sensorica que posee el dron logrando así un gran desempeño lo cual es observable en las gráficas simuladas en Matlab<sup>5</sup>.

Z. Nelson Gatica, P. Carlos Muñoz y A. Patricio Sellado en el Depto. de Ing. Eléctrica Universidad de La Frontera Temuco, Chile idean un control difuso PID real para el UAV AR. Drone 2.0 que le permitiera volar bajo perturbaciones en espacios conocidos, inicialmente para un vuelo en interiores se logra un controlador que permite al vehículo volar en el mismo sitio, es decir un vuelo estacionario, presentan un controlador PID difuso, logrando óptimos resultados y comparando estos con los obtenidos mediante un controlador clásico PID donde logran disminuir el error promedio de 11 a 8 cm<sup>6</sup>.

---

<sup>1</sup>Rodríguez, *Google con globos y Facebook con drones ofreceran internet gratis para todos.*

<sup>2</sup>Varios, *Impacto y uso de drones para generar cartografía.*

<sup>3</sup>Penalva, *Así es el dron repartidor de Amazon: todavía poco más que humo que promete entregar paquetes en media hora.*

<sup>4</sup>Santoso y col., *Robust Hybrid Nonlinear Control Systems for the Dynamics of a Quadcopter Drone.*

<sup>5</sup>Saha y Wadoo, *Linear optimal control of a parrot AR.Drone 2.0.*

<sup>6</sup>Gatica, Muñoz y Sellado, *Real fuzzy PID control of the UAV AR.Drone 2.0 for hovering under disturbances in known environments.*

Arturo Alonso Casanova De Los Santos en la Universidad Tecnológica De La Mixteca, Huajuapán De León, Oaxaca, México propone un control difuso del quadrotor AR. Drone 2.0® para el seguimiento autónomo de trayectorias, para esto hace uso de la lógica difusa por medio de la cual diseña e implementa 4 controladores, uno para cada Angulo de Euler (Pitch, Roll y Yaw) y un último para la altura, presenta un panel de control en Matlab desde donde se pueden ejecutar las funciones del vuelo hacia el dron, en general tuvo una respuesta buena aunque sensible a perturbaciones, presentando múltiples oscilaciones en los resultados que obtuvo, pese a ello el desempeño del controlador fue de calidad<sup>7</sup>.

Álvaro José Herrera Távara de la Universidad Ricardo Palma, Lima, Perú presenta el proyecto de diseño e implementación del sistema de vuelo autónomo de un vehículo aéreo no tripulado para el reconocimiento en zonas hostiles azotadas por el narcoterrorismo en la Amazonía peruana, en su proyecto comienza por el diseño y la elaboración del dron con el cual va a trabajar, una vez lo tiene listo diseña e implementa un sistema de control clásico PID para darle autonomía suficiente al vehículo para que ejecute algunas funciones de vuelo, los resultados de vuelo fueron en un principio un tanto bruscos pero logra solucionarlos por medio de programación adicional<sup>8</sup>.

Bernardino Vargas Mollinedo en la Universidad Nacional del Altiplano, Puno, Perú plantea el diseño de un control multivariable para un dron, obtiene el modelo matemático con el cual diseña el sistema de control, posteriormente realiza pruebas de manera simulada para obtener una respuesta optima, utiliza un controlador PD no lineal de polos iguales, adicionalmente aplica el criterio de Ziegler-Nichols para mejorar la respuesta de estabilidad obtenida, simuladamente el comportamiento del dron es bueno<sup>9</sup>.

Fabio Nelson Castañeda García, Ramiro Andrés Henao y Fabio Alejandro Valencia en la Universidad Católica de Manizales, Manizales, Colombia establecen el diseño e implementación del sistema de control de vuelo de un UAV, para el control de vuelo se utiliza una brújula y horizonte artificial usando Arduino-Processing -MPU6050, este software permitió que el dron realizara un vuelo estable<sup>10</sup>.

Brian Mauricio Lara Sosa, Elkin Yilmar Fagua Perez y su equipo de trabajo exponen el diseño e implementación de un sistema de control de vuelo para un vehículo aéreo no tripulado tipo cuadricóptero, realizan el modelamiento del dron por medio de las ecuaciones Euler Lagrange , para la estrategia de control hacen uso del modo deslizante (Sliding Mode Controller, SMC) el cual es sumamente robusto ya que obliga al sistema a anular las dinámicas no modeladas adicionalmente no tiene en cuenta incertidumbres paramétricas, presenta una alta insensibilidad a determinadas perturbaciones al sistema, los resultados obtenidos son buenos, sin embargo presentan desplazamientos no deseados en X y Y dado que no se están controlando los desplazamientos a través de esos ejes<sup>11</sup>.

<sup>7</sup>Santos, *Control difuso del quadrotor AR.Drone 2.0. para el seguimiento autónomo de trayectorias.*

<sup>8</sup>Herrera Tavera, *Diseño e implementación del sistema de vuelo autónomo de un vehículo aéreo no tripulado para el reconocimiento en zonas hostiles azotadas por el narcoterrorismo en la Amazonía peruana.*

<sup>9</sup>Vargas Mollinedo, *Diseño de un sistema de control multivariable para un dron.*

<sup>10</sup>Castañeda García, Henao y Valencia, *Diseño e implementación del sistema de control de vuelo de un UAV.*

<sup>11</sup>Sosa y col., *Diseño e implementación de un sistema de control de vuelo para un vehículo aéreo no tripulado tipo cuadricóptero.*



Andrés Felipe Lasso Aldana y Alexander Murcia Castro en la Universidad Surcolombiana, Neiva, Huila, Colombia plantean el diseño e implementación de un sistema de guía inercial para el control de vuelo autónomo en un UAV, para lograr esto diseñan el sistema de guía inercial haciendo uso de los diferentes instrumentos de medida que posee el UAV para que por medio de un algoritmo DCM pueda calcular los datos referentes a los ángulos de Euler para su posterior corrección, los resultados que obtuvieron fueron óptimos manifestando así el buen desempeño del sistema de guía inercial. Aunque este proyecto tiene el precedente en la región Surcolombiana, es de anotar que esta propuesta solamente realizó telemetría, pero no control y por lo tanto se propone aquí un valor agregado<sup>12</sup>.

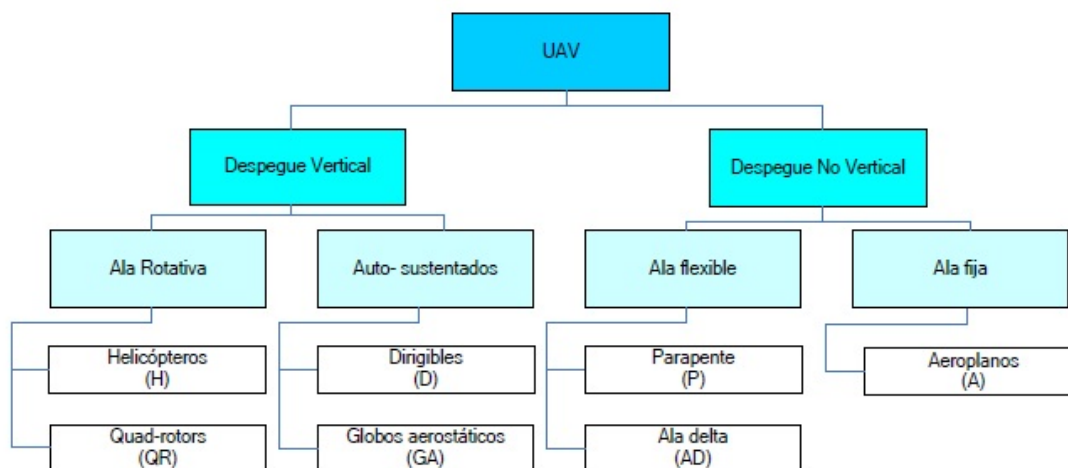
### 3.2. VEHICULOS AEREOS NO TRIPULADOS(UAV)

Se entiende por un Aeronave No Tripulada (UAV: Unmanned Aerial Vehicle o también UAS: Unmanned Aircraft System o UAVS: Unmanned Aircraft Vehicle System) a aquella que es capaz de realizar una misión sin necesidad de tener una tripulación embarcada. Debe entenderse que ésta condición no excluye la existencia de piloto, controlador de la misión u otros operadores, que pueden realizar su trabajo desde tierra. La extensión del concepto de vehículo a sistema, refleja que el UAVS precisa, no solo de la aeronave adecuadamente instrumentada, si no también de una estación en tierra, que complementa la instrumentación y capacidades embarcadas. Es conveniente considerar que esta definición podría incluir algunos casos que quedan fuera del concepto subyacente de UAV. Así los globos aerostáticos, utilizados por ejemplo con fines meteorológicos, responden a la definición y sin embargo no son considerados como UAV dada su no controlabilidad. Lo mismo cabría decir de los misiles autopilotados o con control remoto. A la hora de establecer una clasificación de los UAV es posible atender a diferentes criterios. Tal vez el más simple sea el que se basa en el tipo de aeronave del UAV. De acuerdo a este pueden distinguirse a aquellas de despegue vertical de las que no lo son, estando dentro de las primeras las de ala rotativa o hélice (helicópteros y quad-rotors entre otros), los de ala flexible (parapentes, ala delta) y los auto-sustentados (dirigibles y globos). Dentro de los de despegue no vertical, se encuentran los de ala fija (aeroplanos). La Figura 1, muestra los diferentes tipos de aeronaves utilizados en los UAV<sup>13</sup>.

<sup>12</sup>A.F. Lasso, *Diseño e implementación de un sistema de guía inercial para el control de vuelo autónomo en un UAV*.

<sup>13</sup>A y col., *Vehículos aéreos no tripulados para uso civil. Tecnología y aplicaciones*.

**Figura 1:** Tipos de aeronaves



Fuente: <https://www.aviacioncivil.com.ve/>

aplicaciones-civiles-de-los-vehiculos-aereos-tripulados-vant-unmanned-aerial-vehicles-uav/

## Aplicaciones

En las aplicaciones civiles, la portabilidad, la capacidad de despegue y aterrizaje vertical, la maniobrabilidad y la capacidad de vuelo estacionario de los helicópteros y el muy bajo costo de los multi-rotors, han hecho que estas sean el tipo de aeronaves más populares”, generalmente limitadas a las categorías ‘micro’. En muchos de los casos la distancia entre la estación base de control y el vehículo no tripulado, está entre las decenas y las centenas de metros, y en algunos casos pocos kilómetros. Es frecuente también que las tareas a realizar requieran volar a baja velocidad, o incluso en vuelo estacionario<sup>13</sup>.

### 3.3. QUADROTOR AR. DRONE 2.0

Es un cuadricóptero de la casa fabricante Parrot posee una batería de 1100 mAh que le da una autonomía de 12 minutos de vuelo, posee una aplicación móvil presente en el AppStore y Googleplay que permite pilotarlo, el enlace entre el dron y el dispositivo que lo controle se hace por medio de una red Wi-Fi que crea el propio dron, dicha red tiene un alcance de 50 metros, adicionalmente el dron cuenta con 2 cámaras, la cámara frontal es la principal, permite grabar videos en HD con una resolución de 720p a 30 FPS, para el procesamiento de datos posee un procesador ARM Cortex A8 de 32 bits a 1 GHz con DSP de videoTMS320DMC64x a 800 MHz Linux 2.6.32 RAM DDR2 de 1 GB a 200 MHz, también posee un puerto USB 2.0 de alta velocidad para extensiones y grabación Wi-Fi b/g/n, posee los sensores necesarios para el control y la estabilización necesaria como un giroscopio de 3 ejes 2.000°/s, acelerómetro de 3 ejes +/-50 mg, magnetómetro de 3 ejes, precisión 6°, sensor de presión +/- 10 Pa (80 cm a nivel del mar), sensor de ultrasonidos para medición de altitud respecto al suelo, cámara QVGA vertical a 60 fps para medición

de la velocidad respecto al suelo, para el movimiento del vehículo cuenta con 4 motores sin escobillas de tipo “inrunner”: 14,5 vatios y 28 500 rpm<sup>14</sup>.

El dron está diseñado de tal manera que posee dos carcasas especiales para su respectiva adaptación, en la Figura 2 se muestra el dron con carcasa de interiores y en la Figura 3 se muestra el dron con carcasa de exteriores. En la Figura 4 se muestra el dron con sus respectivas carcasas y accesorios.

**Figura 2:** Dron carcasa interiores



Fuente: <https://www.parrot.com/es/drones/parrot-ardrone-20-elite-edition#parrot-ardrone-20-elite-edition-details>

**Figura 3:** Dron carcasa exteriores



Fuente: <https://www.parrot.com/es/drones/parrot-ardrone-20-elite-edition#parrot-ardrone-20-elite-edition-details>

---

<sup>14</sup>Parrot, *Drone 2.0*.

**Figura 4:** Parrot AR. Drone 2.0

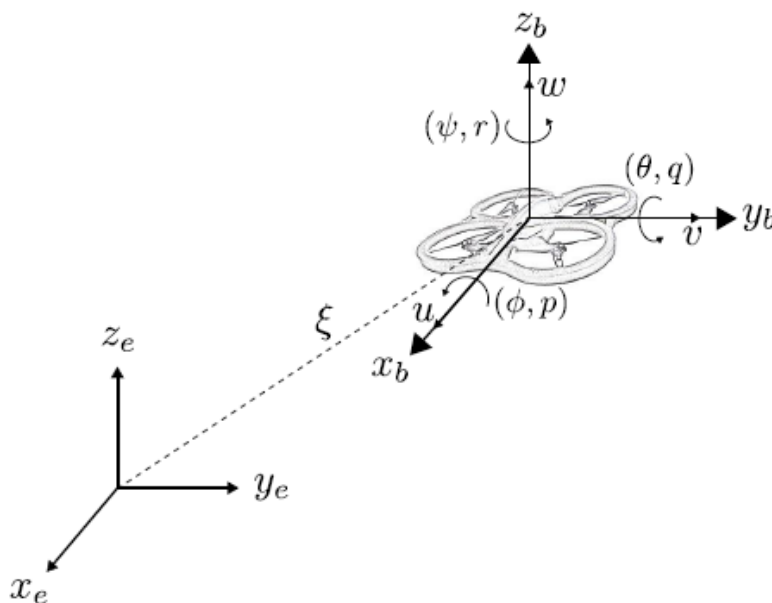


Fuente: <https://www.parrot.com/es/drones/parrot-ardrone-20-elite-edition#parrot-ardrone-20-elite-edition-details>

## MODELAMIENTO

La cinemática de un cuerpo rígido que se mueve en el espacio muestra la relación entre las velocidades de traslación y rotación entre el marco del cuerpo y el marco inercial como se muestra en la Figura 5.

**Figura 5:** Modelamiento cinemático del helicóptero de cuatro rotores



Fuente: <http://amca.mx/memorias/amca2014/media/files/0265.pdf>

La posición y rotación del helicóptero de cuatro rotores con respecto al marco inercial están dados por  $\xi = [x, y, z]^T$  y  $\eta = [\phi, \theta, \psi]^T$ , respectivamente. Las velocidades lineales del helicóptero dadas en el marco del cuerpo están dadas por  $V = [u, v, w]^T$  y  $\omega = [p, q, r]^T$  representan las velocidades angulares. La relación entre las velocidades en el marco inercial y las velocidades lineales en el marco del cuerpo está dada por

$$\dot{\xi} = R_b^e V \quad (1)$$

donde

$$R_b^e = \begin{bmatrix} c\theta c\psi & s\phi s\theta c\psi - c\phi s\psi & c\phi s\theta c\psi + s\phi s\psi \\ c\theta s\psi & s\phi s\theta s\psi + c\phi c\psi & c\phi s\theta s\psi - s\phi c\psi \\ -s\theta & s\phi c\theta & c\phi c\theta \end{bmatrix} \quad (2)$$

con  $c\phi = \cos \phi$ ,  $s\phi = \sin \phi$ ,  $c\theta = \cos \theta$ ,  $s\theta = \sin \theta$ ,  $c\psi = \cos \psi$  y  $s\psi = \sin \psi$ . Las velocidades de los ángulos de orientación en el marco inercial y las velocidades angulares en el marco del cuerpo están relacionadas por

$$\Omega = W_\eta \dot{\eta} = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \sin \phi \cos \theta \\ 0 & -\sin \phi & \cos \phi \cos \theta \end{bmatrix} \dot{\eta} \quad (3)$$

Invirtiendo (3) se obtiene

$$\dot{\eta} = W_\eta^{-1} \Omega = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & -\sin \phi \sec \theta & \cos \phi \sec \theta \end{bmatrix} \Omega \quad (4)$$

La dinámica representa la evolución de un sistema sujeto a fuerzas y pares externos. Existen varios enfoques para obtener el modelo dinámico de un helicóptero de cuatro rotores entre los que se encuentran el enfoque de Newton-Euler, Euler-Lagrange y el de los Cuaterniones. Utilizando el enfoque de Euler-Lagrange y basados en García et al. (2013), el modelo dinámico de un helicóptero de cuatro rotores está dado por

$$m\ddot{\xi} = R_b^e \begin{bmatrix} 0 \\ 0 \\ F \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} \quad (5)$$

$$\ddot{\eta} = \tilde{\tau} \quad (6)$$

Con  $m$  la masa del helicóptero,  $g$  la constante de gravedad,  $F$  el empuje principal y  $\tilde{\tau} = [\tilde{\tau}_\phi, \tilde{\tau}_\theta, \tilde{\tau}_\psi]^T$  torques auxiliares definidos por

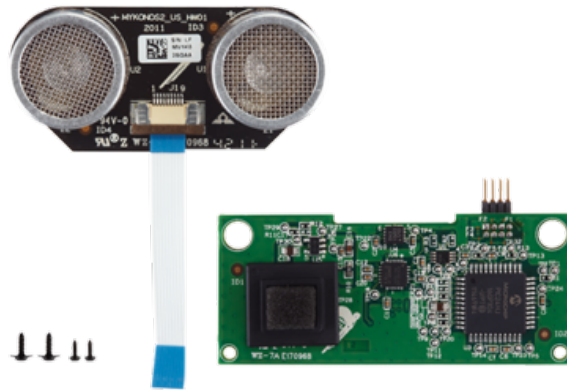
$$\tilde{\tau} = \mathbb{J}^{-1}(\tau \frac{d}{dt} C(\eta, \dot{\eta}) \dot{\eta}) \quad (7)$$

donde  $\mathbb{J}$  es la matriz de inercia,  $C(\eta_i, \dot{\eta}_i)$  es el término de Coriolis y  $\tau$  son los pares generados por los rotores del helicóptero.<sup>15</sup>

## PLACA DE NAVEGACIÓN

La placa de navegación está equipada con distintos sensores para garantizar la estabilidad del dron como se observa en la Figura 6.

**Figura 6:** Placa de navegación del AR. Drone 2.0



Fuente: <https://www.parrot.com/es/piezas-repuesto/drones/placa-de-navegacion-para-parrot-ardrone-20#placa-de-navegacion-para-parrot-ardrone-20-details>

## Giroscopio

Sensor de velocidad angular de eje X, Y y Z de salida digital con escala completa programable por el usuario. Tiene una característica muy interesante que consiste en poder cambiar el rango de medida de la velocidad angular por medio de la escritura de registros. Los rangos disponibles son  $\pm 250^\circ/\text{s}$ ,  $\pm 500^\circ/\text{s}$  y  $\pm 2000^\circ/\text{s}$ .<sup>16</sup>

El típicamente utilizado es el giroscopio MEMS de estructura vibrante, que detecta la variación de la velocidad de rotación sobre un eje. Es llamado de estructura vibrante ya

<sup>15</sup>Santiaguillo-Salinas y Aranda-Bricaire, *Seguimiento de Trayectorias para un Helicóptero de 4 Rotores AR. Drone 2.0 Utilizando ROS*.

<sup>16</sup>Sheet, *MPU-6000 and MPU-6050 Register Map and Description*.

que al rotar el giroscopio sobre un eje, por efecto Coriolis, se produce una vibración que es medida por unos platos capacitivos. La señal generada por los platos es tratada para que sea proporcional a la velocidad angular.<sup>17</sup>

### **Magnetómetro de tres ejes**

Los magnetómetros sirven para medir la fuerza de los campos magnéticos y su dirección. Existen magnetómetros escalares y vectoriales, los magnetómetros escalares miden la fuerza del campo magnético y los vectoriales calculan la fuerza del campo magnético en una dirección. Los magnetómetros vectoriales miden varias componentes del campo magnético de forma electrónica. Con tres magnetómetros (o magnetómetro de tres ejes) situados ortogonalmente se puede averiguar las componentes de un campo magnético en un punto por lo que se podría averiguar la posición y dirección de ese punto en relación a la magnetósfera terrestre.<sup>18</sup>

El magnetómetro MEMS es un sensor magnético de efecto Hall monolítico de silicio de 3 ejes con concentrador magnético. Amplio rango de medición dinámica y alta resolución con menor consumo de corriente. La resolución de datos de salida es de 13 bits ( $0.3\mu\text{T}$  por LSB). El rango de medición a escala completa es de  $\pm 1200\mu\text{T}$ .<sup>19</sup>

### **Acelerómetro digital de tres ejes**

Un acelerómetro es un dispositivo que mide las fuerzas inerciales que se le apliquen, como por ejemplo la gravedad. Esta funcionalidad es muy útil para comprobar la inclinación del cuadricóptero respecto al eje vertical. Al medir la aceleración en los tres ejes de coordenadas (XYZ) y tener una referencia constante como es la gravedad, se puede conocer la dirección en la que se encuentra el suelo y saber la inclinación respecto a éste. El acelerómetro MEMS de triple eje tiene 3 ejes de salida digital con un rango programable de escala completa de  $\pm 2\text{g}$ ,  $\pm 4\text{g}$ ,  $\pm 8\text{g}$  y  $\pm 16\text{g}$ . Detección y señalización de orientación. Detección de tap. Interrupciones programables por el usuario. Interrupción de alto G. Autocomprobación del usuario.<sup>19</sup>

### **Altímetro**

El altímetro es un sensor que contienen los drones, para regular automáticamente la altura del vuelo, lo que permite realizar grabaciones con alturas estables; que no disminuyan ni aumenten durante la grabación; se puede con el control de distancia, hacer avanzar el dron, hacerlo girar, e incluso retrocederlo; sin que este pierda la estabilidad y gracias al altímetro, mantendrá la altura del vuelo que se escoge para la mejor toma fotográfica o filmográfica.<sup>19</sup>

---

<sup>17</sup>Cantero Guerrero, *Diseño y fabricación de un vehículo aéreo no tripulado*.

<sup>18</sup>García García, *Diseño y construcción de magnetómetro triaxial para análisis y experimentación de aislamientos magnéticos*.

<sup>19</sup>Sheet, *MPU-9150 Register Map and Description*.

## Sensor de ultrasonidos

El sensor de distancia ultrasónico proporciona mediciones de distancia precisas y sin contacto desde aproximadamente 2 cm (0,8 pulgadas) hasta 3 metros (3,3 yardas). Es muy fácil de conectar a microcontroladores. El sensor funciona transmitiendo un estallido ultrasónico (muy por encima del rango de audición humana) y proporcionando un pulso de salida que corresponde al tiempo requerido para que el eco de ráfaga regrese al sensor. Al medir el ancho del pulso de eco, la distancia al objetivo puede calcularse fácilmente.<sup>20</sup>

## SISTEMA DE COMUNICACIÓN

El software incrustado en el AR. Drone 2.0 que viene cargado de fábrica en su placa madre, responsable de la lectura de los sensores, automatización de maniobras de vuelo complejas e interacción con dispositivos externos mediante una conexión Wi-Fi. En su configuración por defecto, el firmware del dron actúa como un access point que crea su propia red Wi-Fi que sirve de canal para la comunicación bilateral entre el y un dispositivo externo cliente (host). Los protocolos de este proceso de comunicación son establecidos por el firmware del dron, tal como lo son la estructura de los mensajes intercambiados y su efecto sobre la máquina. Según este esquema existen cuatro tipos de mensajes, cada uno con un protocolo específico:

- A. Envío de instrucciones de navegación y de modificación de la configuración.
- B. Recepción de la configuración actual del dron.
- C. Stream de video.
- D. Datos de navegación (navdata).

En su configuración por defecto el AR. Drone 2.0 crea su propia red para comunicarse con dispositivos externos. En este modo, el dron ejecuta un servidor DHCP con las siguientes características:

- IP Drone: 192.168.1.1
- IP Host: 192.168.1.2 (sólo el dispositivo con esta IP puede controlar al dron)
- Rango DHCP: 192.168.1.2-5
- Gateway: 192.168.1.1
- Wi-Fi
- Encriptación: Ninguna
- ESSID: adrone2\_xxx(por defecto)
- Protocolos:
  - TCP/IP
  - UDP/IP
  - IPv4

Para usuarios avanzados, es posible configurar el dron vía telnet, asignándole una IP estática y creando una conexión con un punto de acceso determinado; haciendo esto, una

---

<sup>20</sup>Parallax, *Ping ultrasonic distance sensor (# 28015)*.



aplicación en el cliente podría conectarse con múltiples drones a la vez. No obstante lo anterior, tal opción no es relevante para efectos de este trabajo, por lo cual a partir de este punto se considerará que la IP del AR. Drone será siempre fija, de valor 192.168.1.1.<sup>21</sup>

### 3.4. MATLAB

MATLAB es un lenguaje de alto desempeño diseñado para realizar cálculos técnicos. Integra el cálculo, la visualización y la programación en un ambiente fácil de utilizar donde los problemas y las soluciones se expresan en una notación matemática. Es un sistema interactivo cuyo elemento básico de datos es el arreglo que no requiere de dimensionamiento previo. Esto permite resolver muchos problemas computacionales, específicamente aquellos que involucren vectores y matrices, en un tiempo mucho menor al requerido para escribir un programa en un lenguaje escalar no interactivo tal como C o Fortran. Se utiliza ampliamente en:

- Cálculos numéricos.
- Desarrollo de algoritmos.
- Modelado, simulación y prueba de prototipos.
- Análisis de datos, exploración y visualización.
- Graficación de datos con fines científicos o de ingeniería.
- Desarrollo de aplicaciones que requieran de una interfaz gráfica de usuario (GUI, Graphical User Interface).

En el ámbito académico y de investigación, es la herramienta estándar para los cursos introductorios y avanzados de matemáticas, ingeniería e investigación. En la industria MATLAB es la herramienta usada para el análisis, investigación y desarrollo de nuevos productos tecnológicos.

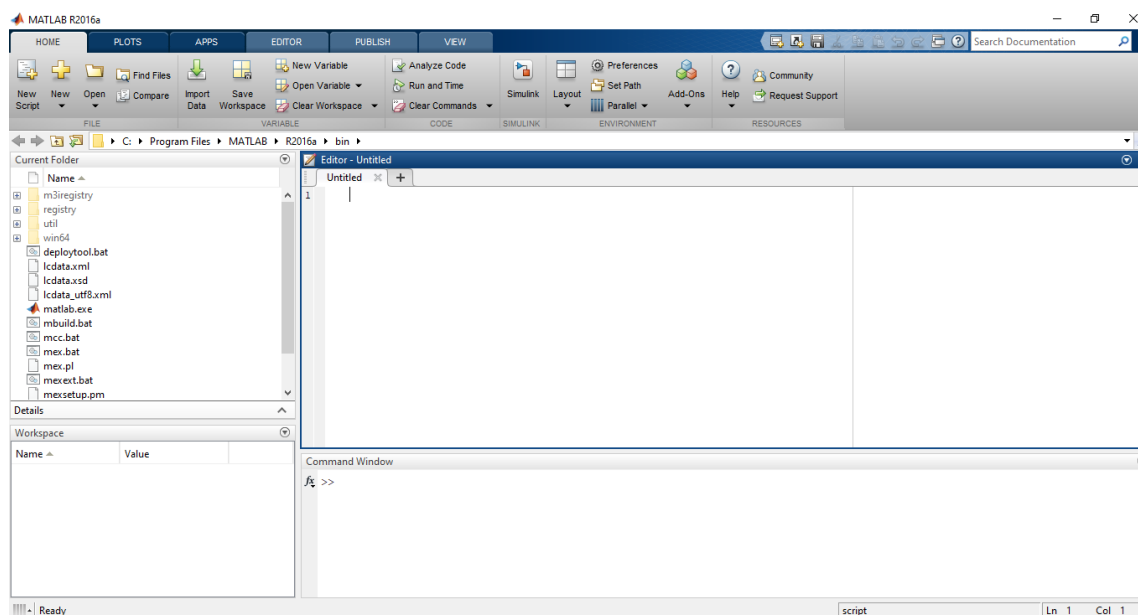
La ventaja principal de MATLAB es el uso de familias de comandos de áreas específicas llamadas toolboxes. Lo más importante para los usuarios de MATLAB es que los toolboxes le permiten aprender y aplicar la teoría. Los toolboxes son grupos de comandos de MATLAB (archivos M) que extienden el ambiente de MATLAB para resolver problemas de áreas específicas de la ciencia e ingeniería. Por ejemplo, existen toolboxes para las áreas de Procesamiento Digital de Señales, Sistemas de Control, Redes Neuronales, Lógica Difusa, Wavelets, etc.<sup>22</sup>. El entorno de MATLAB se muestra en la Figura 7.

---

<sup>21</sup>Hernández, *PLATAFORMA DE COMUNICACIÓN ENTRE LIVE ROBOT PROGRAMMING Y EL ROBOT AR.DRONE 2.0*.

<sup>22</sup>MathWorks, *the Mathworks*.

**Figura 7:** Entorno MATLAB



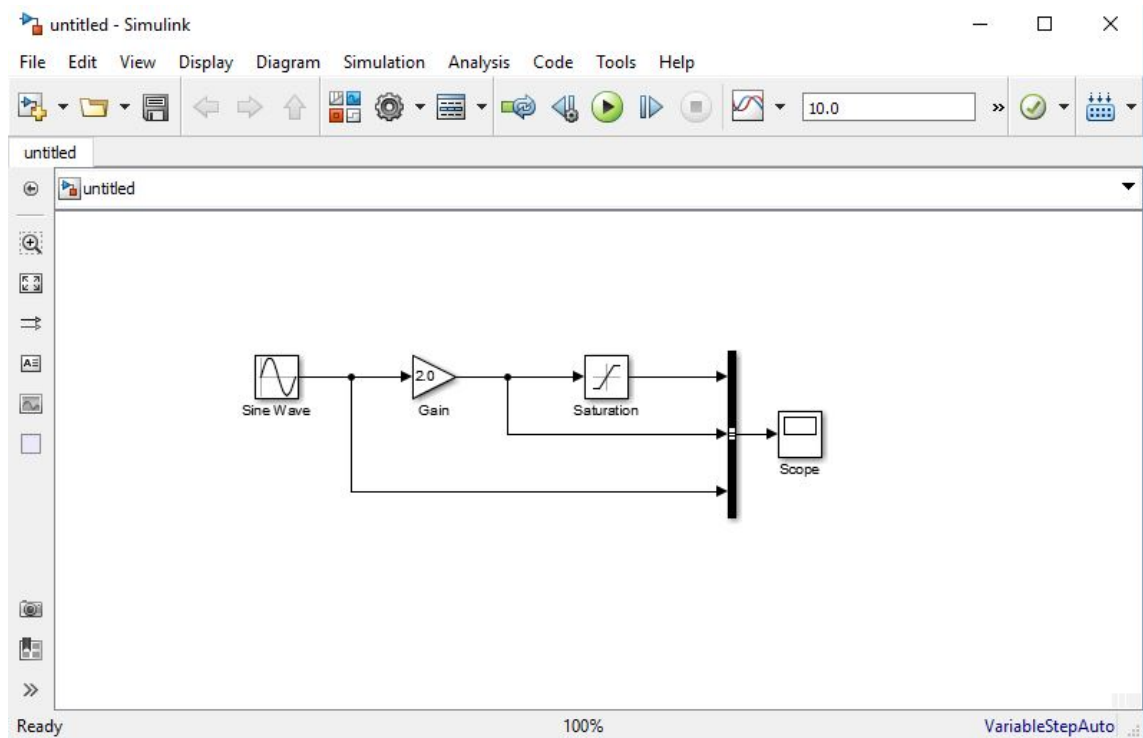
Fuente: <https://postparaprogramadores.com/matlab-para-ingenieros/>

## SIMULINK

Es un entorno de diagrama de bloques como se muestra en la Figura 8 para la simulación multidominio y diseño basado en modelos. Es compatible con el diseño a nivel de sistema, la simulación, la generación automática de código y la prueba y verificación continua de sistemas integrados. Simulink proporciona un editor gráfico, bibliotecas de bloques personalizables y solucionadores para modelar y simular sistemas dinámicos. Se integra con MATLAB, lo que le permite incorporar algoritmos de MATLAB en modelos de simulación y los resultados de exportación a MATLAB para su posterior análisis<sup>23</sup>.

<sup>23</sup>Simulink, *Simulation and Model-Based Design*.

**Figura 8:** Entorno MATLAB SIMULINK



Fuente: <https://www.indiamart.com/proddetail/matlab-simulink-training-4186164148.html>

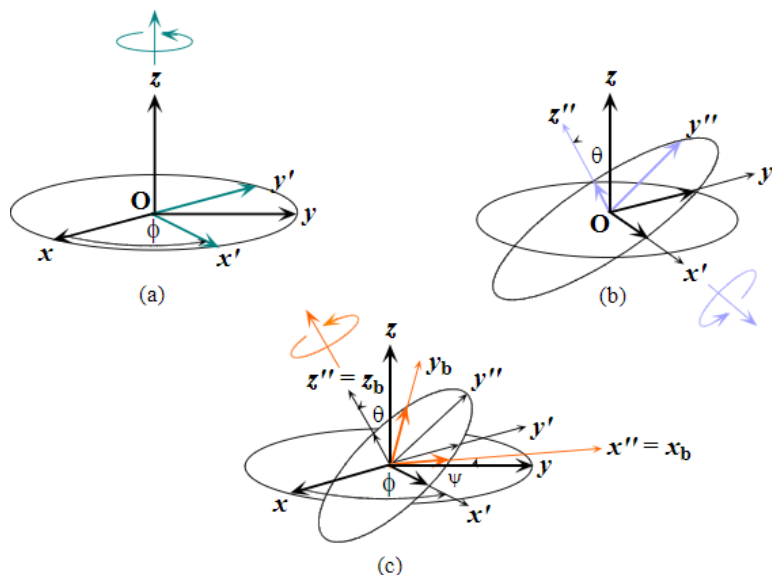
### 3.5. ÁNGULOS DE EULER

Los tres ángulos de Euler,  $\psi$ ,  $\theta$  y  $\phi$  en los resultados de Análisis de movimiento son la secuencia de los ángulos de rotación de los ejes Z, X y Z medidos con respecto al sólido del sistema de coordenadas de rotación.

- $\psi$  El primer ángulo de rotación en la secuencia de rotación del ángulo de Euler.  $\psi$  mide la rotación con respecto al eje Z del sistema de coordenadas de rotación (Yaw, guiñada). Como se observa en la Figura 9 (a).
- $\theta$  El segundo ángulo de rotación en la secuencia de rotación del ángulo de Euler.  $\theta$  mide la rotación con respecto al eje X del sistema de coordenadas de rotación después de que se haya aplicado la rotación  $\psi$  (Pitch, cabeceo). Como se observa en la Figura 9 (b).
- $\phi$  El tercer ángulo de rotación en la secuencia de rotación del ángulo de Euler.  $\phi$  mide la rotación con respecto al eje Z del sistema de coordenadas de rotación después de que se hayan aplicado las rotaciones  $\psi$  y  $\theta$  (Roll, alabeo). Como se observa en la Figura 9 (c).<sup>24</sup>

<sup>24</sup>SolidWorks, *Solidworks corporation*.

**Figura 9:** Ángulos de Euler (a)Yaw (b)Pitch (c)Roll



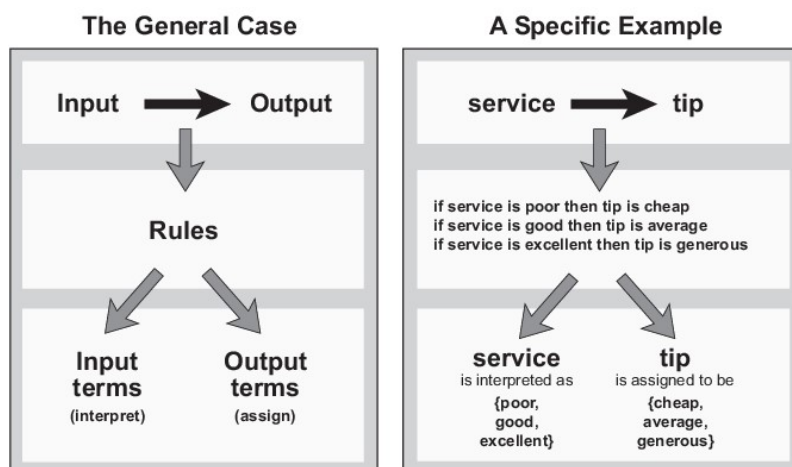
Fuente:

<http://www.udesantiagovirtual.cl/moodle2/mod/book/view.php?id=24916&chapterid=287#fg3.3>

### 3.6. LÓGICA DIFUSA

El objetivo de la lógica difusa es asignar un espacio de entrada a un espacio de salida y el mecanismo principal para hacerlo es una lista de sentencias if-then llamadas reglas. Todas las reglas se evalúan en paralelo y el orden de las reglas no es importante. Las reglas en sí son útiles porque se refieren a variables y los adjetivos que describen esas variables. Antes de que pueda construir un sistema que interpreta reglas debe definir todos los términos que planea usar y los adjetivos que los describen. Para decir que el agua está caliente se debe definir el rango en el que se espera que varíe la temperatura del agua así como también lo que se quiere decir con la palabra “caliente”. El siguiente diagrama en la Figura 10 proporciona una hoja de ruta para el proceso de inferencia difusa. Muestra la descripción general de un sistema difuso a la izquierda y un sistema difuso específico a la derecha.

**Figura 10:** Diagrama proceso de inferencia difusa



Fuente: <https://la.mathworks.com/help/fuzzy/foundations-of-fuzzy-logic.html>

Para resumir el concepto de inferencia difusa que se muestra en esta figura la inferencia difusa es un método que interpreta los valores en el vector de entrada y en función de un conjunto de reglas asigna valores al vector de salida.

### Funciones de membresía difusa

El grado de membresía en los conjuntos difusos, puede ser representado por una función continua, la cual es denominada como función de membresía. En otras palabras si  $F$  es un conjunto difuso, entonces la función de membresía  $\mu_F(x)$  mide el grado con el cual el valor  $x$  pertenece al conjunto  $F$ , de la cual puede tomar diferentes formas, dependiendo de la aplicación en particular. Algunas de las funciones más conocidas son: función singleton, cuyo valor de membresía es igual a 1 para un solo punto y el resto es cero, función trapezoidal, triangular, sigmodial, gaussiana, entre otras.

### Controlador lógico difuso

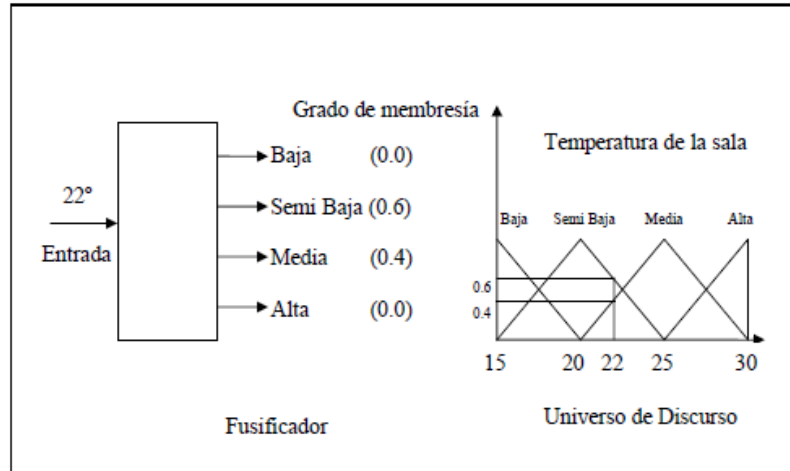
Un controlador lógico difuso (CLD), emplea principios de lógica difusa y permite convertir estrategias de control lingüístico, basado en conocimiento experto, en una estrategia de control automático. El método propuesto por C.C. Lee, el cual está compuesto por las siguientes partes:

- **Fusificación**

Es el proceso de asignar valores de membresía o pertenencia a un valor numérico de entrada para cada una de las etiquetas difusas que forman la variable lingüística “Temperatura de la sala” puede tomar los valores “baja”, “semi-baja”, “media” y “alta”; para este caso la entrada al fusificador es un valor de temperatura preciso (crisp), y la salida estará formada

por los valores de verdad de cada una de las etiquetas “baja”, “semi-baja”, “media” y “alta”, como se muestra en la Figura 11.

**Figura 11:** Proceso de fusificación



Fuente: <https://cdigital.uv.mx/bitstream/123456789/1736/1/Tesis-Jimenez-Madrigal-Rene.pdf>

- **Base de conocimiento**

Se obtiene la experiencia desarrollada por un operador y de conocimientos sobre Ingeniería de control, y depende del proceso a controlar y los requerimientos de diseño. Está compuesta por dos partes, una base de datos y una base de reglas de control difuso (utiliza variable lingüísticas). Las bases de reglas también se pueden obtener utilizando métodos de optimización.

**Base de datos**

Incluye la discretización y normalización de los universos de entrada y salida, la definición de subconjuntos (partición de los universos y funciones de pertenencia) y satisfacer la propiedad de completitud.

a) Discretización

Debido a la necesidad de discretizar, ya que los datos se procesan en forma digital, este proceso genera niveles cuantizados, cada uno de los cuales presenta un elemento gráfico en un universo de discurso.

b) Normalización

La normalización del universo discreto puede ser lineal o no.

c) Partición de los universos

Se refiere al número de etiquetas que toma una variable lingüística; por ejemplo la variable “temperatura de la sala”, puede tomar los valores “muy baja”, “baja”, “media” y “alta”. El número de términos está determinado por las características del sistema a controlar y la calidad del control.

#### d) Funciones de pertenencia

Las funciones de pertenencia o membresía, representan gráficamente la relación que mantienen los elementos de un subconjunto difuso, dentro de un universo de discurso, con el grado de pertenencia al conjunto en cuestión. La representación de estas figuras puede tomar diferentes formas, siendo las más utilizadas las de forma triangular y trapezoidal. La utilización de alguna de ellas puede ser de manera arbitraria, dependiendo de la aplicación en particular.

#### e) Completitud

Esta propiedad indica que el algoritmo debe ser capaz de inferir una acción correcta para cada estado del proceso.

### **Base de reglas**

La estrategia de control, derivada de la experiencia se expresa mediante el uso de algoritmos difusos. Las reglas de control que forman el algoritmo difuso pueden definirse usando los siguientes criterios:

#### a) Selección de las variables

Las variables de entrada se seleccionan basándose en la experiencia y en conocimientos de Ingeniería y el cambio de error (derivada del error).

#### b) Origen y obtención de las reglas de control

En un controlador lógico difuso, su operación dinámica está caracterizada por un conjunto de reglas compuestas por variables lingüísticas basadas en el conocimiento de un experto, estas usualmente son de la forma:

*IF* (un conjunto de condiciones son satisfechas) *Then* (un conjunto de consecuentes que pueden inferir).

Donde los antecedentes y los consecuentes de las reglas *IF-Then* son asociados con conceptos difusos (términos lingüísticos), formando lo que se conoce como declaraciones condicional difusa en donde el antecedente es una condición sobre la base del estado de las variables del proceso y el consecuente es una acción de control para el sistema a controlar (proceso).

#### c) Tipos de reglas de control

C.C. Lee, menciona que existen dos tipos de reglas usadas en el diseño de un controlador lógico difuso:

### **Reglas de evaluación de estado**

Este tipo de reglas es el que más se usa, y en el caso de los sistemas MISO, del inglés entradas múltiples salida única, toman la siguiente forma:

### **Reglas de evaluación de objeto**

Este tipo de reglas involucra la evaluación del estado actual así como el resultado de la acción de control. Se aplica en esquemas de control difuso de tipo predictivo.

#### **• Lógica de decisiones**

Como ya se ha mencionado un controlador lógico difuso puede emular a un hábil experto operador humano, mediante funciones de implicación difusa, y mecanismos de inferencia principalmente. Aunque no son los únicos mecanismos que existen. Inferencia difusa Después que las variables de entrada han sido convertidas a valores variables lingüísticas, el paso de inferencia difusa identifica las reglas que se aplican a cada situación, y mediante un método llamado MAX/MIN, determina los valores de la variable lingüística de salida, en el caso que se tenga solo una salida. Este método está compuesto de dos pasos:

Agregación, que relaciona los antecedentes de las reglas.

Composición, el cual procesa los consecuentes de las mismas.

#### **• Defusificación**

Básicamente este proceso, es un mapeo de un espacio de acciones de control difuso definido sobre un Universo de discurso de salida, en un espacio de acciones de control no-difuso (valores precisos), siendo muy importante esta Interfaz, ya que la mayoría de las aplicaciones prácticas de control requieren variables numéricas. De las estrategias más usadas se pueden mencionar las siguientes:

- *Método del criterio máximo*: Da como resultado el punto en el cual la distribución de posibilidades de la acción de control alcanza el valor máximo.

- *Método del promedio máximo*: Esta estrategia genera una acción de control la cual es el promedio de todas las acciones de control cuyos grados de pertenencia alcanzan el valor máximo.

- *Método del centro de áreas*: Es una de las estrategias más utilizadas, se calcula el centro de gravedad de la distribución de las salidas de control difuso para obtener la señal de control, la cual se aplica al sistema a controlar.<sup>25</sup>

---

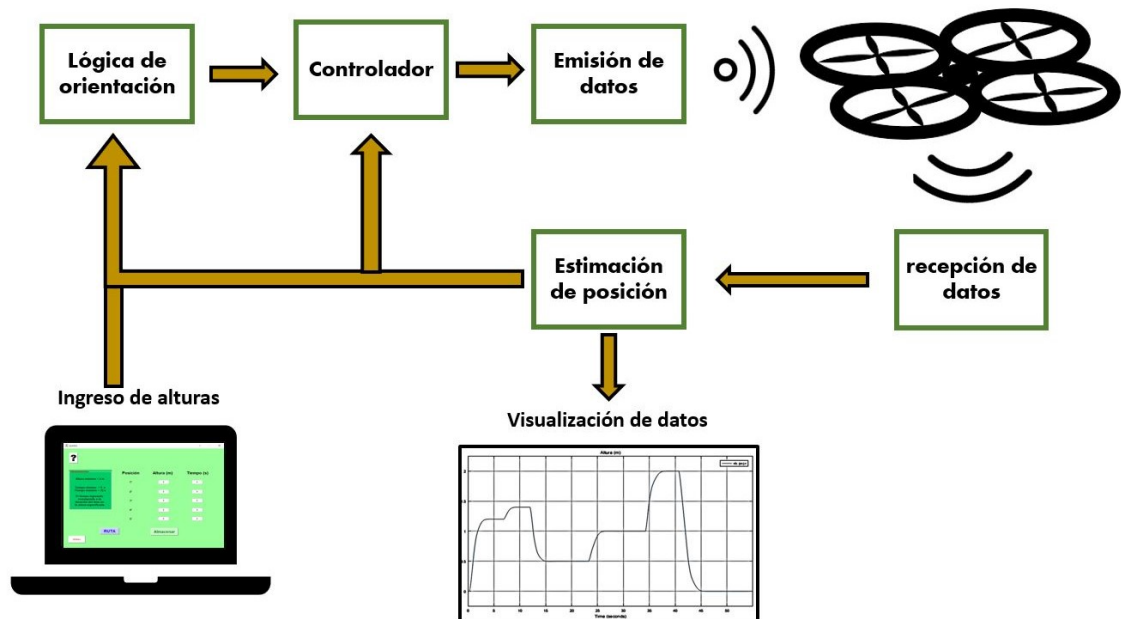
<sup>25</sup>Jiménez Madrigal, *Diseño de un controlador lógico difuso aplicado al control de posición de un servomotor de CD usando un algoritmo genético*.



## 4. DISEÑO Y DESARROLLO DEL PROYECTO

En la Figura 12 se muestra el diagrama de bloques de como está constituido el hardware del proyecto.

Figura 12: Diagrama de bloques del hardware



### 4.1. AR. DRONE SIMULINK DEVELOPMENT-KIT V1.1

Para realizar el sistema de control y su correspondiente interfaz gráfica se decide trabajar con el software Matlab, ya que posee un paquete de desarrollo específico para trabajar con el AR. Drone 2.0, este kit cuenta con bloques y ejemplos para simular el control o implementarlo en el dron directamente en tiempo real, los bloques de simulación cuenta con un modelo aproximado basado en identificaciones previas del vehículo, a su vez la parte de control real permite enviar y leer los datos de las variables necesarias para el vuelo del dron en tiempo real, todo esto permite al usuario realizar control de posición, velocidad, seguimiento de trayectorias de manera simulada como ejecutarlas de forma real por medio del control Wi-Fi, este kit fue desarrollado en el año 2013 para los proyectos de práctica de investigación de MathWorks.

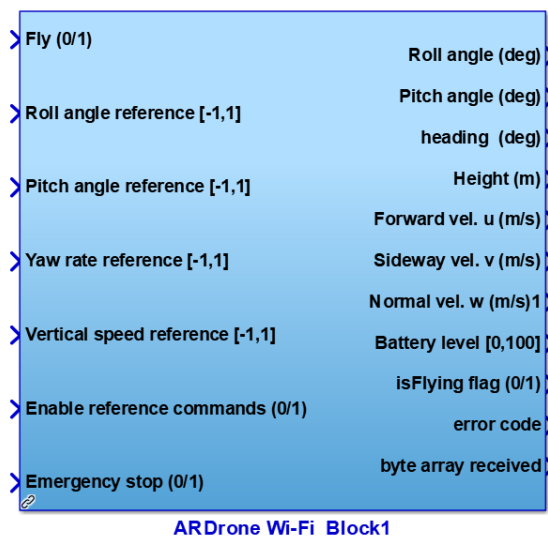
Se profundiza en este kit de desarrollo donde se puede observar que cuenta con ocho funciones de Matlab y seis modelos en Simulink los cuales contienen múltiples bloques que permiten realizar funciones específicas necesarias para este proyecto solo se hace uso de dos modelos de Simulink el `setupWPTrackingSim.m` que trabaja con base a una simulación y el `setupWPTracking.m` para interactuar de manera directa y en tiempo real con el dron,

estos modelos son muy similares entre si ya que los dos están compuestos por cinco grandes subsistemas de bloques que desempeñan una función específica y necesaria los modelos comparten cuatro subsistemas relativamente iguales pero difieren en el último subsistema debido a que en el modelo simulado este último subsistema contendrá diversos bloques que en conjunto permite simular el modelo del dron en el archivo real este subsistema de simulación se ve sustituido por un subsistema de comunicación Wi-Fi con el dron, a continuación se presentan los bloques de manera más explícita:

### AR. Drone Wi-Fi-Block

En este subsistema se pueden encontrar los bloques encargados de la emisión y la recepción de datos entre el ordenador y el dron para establecer dicha comunicación el vehículo cuenta con su propia red Wi-Fi abierta y permite ingresar a las funciones del dron a través de puertos establecidos, el puerto para la lectura de datos es el 5554 y el de emisión es el 5556, todo esto se realiza con base en el protocolo de comunicación UDP este protocolo pertenece al nivel de transporte y se basa en el intercambio de datagramas se caracteriza por permitir el envío de paquetes datos a través de la red sin que se haya establecido previamente una conexión ya que el datagrama tiene suficiente información de direccionamiento en su cabecera tampoco tiene confirmación de llegada lo que no permite la retransmisión de datos favoreciendo así las transmisiones que necesitan realizarse en tiempo real con el menor número de retardos. Para la recepción de datos se utiliza el protocolo UDP 15B2h y para el envío el UDP 15B4h la información que se envía y recibe son las velocidades del dron, los ángulos de Euler, la altura y otros estados como lo son el despegue o aterrizaje y el nivel de batería. La Figura 13 muestra el bloque Wi-Fi con sus respectivas entradas y salidas, en el anexo A se muestra el contenido del bloque y su programación interna.

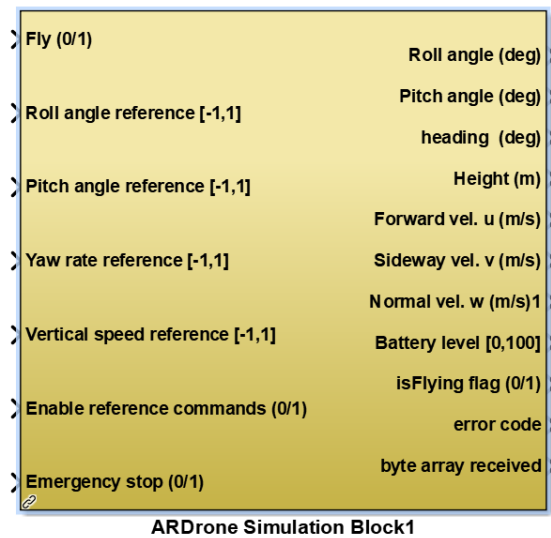
**Figura 13:** AR. Drone Wi-Fi Block



## AR. Drone Simulation Block

Este subsistema cuenta con un modelo matemático obtenido por varias identificaciones que se realizan con Matlab el modelo se expresa por el método de espacio de estado de las cuatro variables principales, los tres ángulos de Euler (Pitch, Roll y Yaw) y la altura, este modelo simula el comportamiento que tendría el dron. La Figura 14 muestra el bloque de simulación con sus respectivas entradas y salidas.

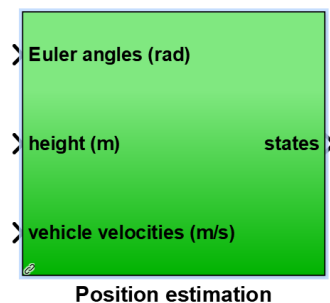
**Figura 14:** AR. Drone Simulation Block



## Position Estimation

Este subsistema permite conocer y estimar la posición del dron por medio de las velocidades del vehículo, los ángulos de Euler y la altura a través del proceso de integración se puede determinar la posición del dron, su orientación y el respectivo desplazamiento en cualquier dirección de los tres ejes al igual que su rotación. La Figura 15 muestra el bloque de estimación de la posición con sus respectivas entradas y salidas, en el anexo B se muestra el contenido del bloque y su programación interna.

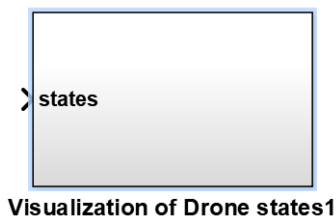
**Figura 15:** Position estimation



## Visualization Of Drone States

Este subsistema se encuentra inmediatamente después del subsistema de estimación de posición y permite visualizar por medio del bloque scope los desplazamientos en altura, X, Y y el Yaw o cabeceo, por medio de ellos se puede visualizar en tiempo real la posición actual del dron, al igual que el espacio recorrido por el mismo. La Figura 16 muestra el bloque de visualización de los estados del dron con sus respectivas entradas y salidas, en el anexo C se muestra el contenido del bloque.

**Figura 16:** Visualization of Drone states



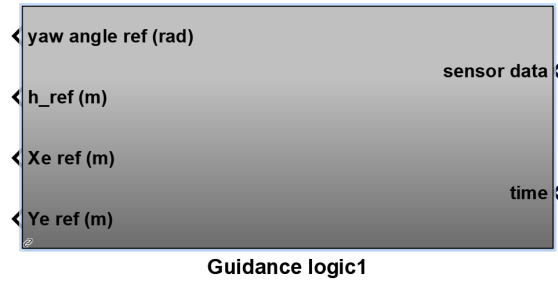
## Guidance Logic

En este subsistema se agrega la posición de altura, posición en X, en Y, la rotación de cabeceo en radianes y el tiempo que permanecerá el dron en determinado punto, este subsistema es el que permite trazar trayectorias, cuando se trabaja con estas se tiene presente el inicio, el final del recorrido, la posición actual y el tiempo que debe permanecer en cada punto, esto determina los datos que serán utilizados como referencia en la etapa de control, para corregir los datos reales provenientes del dron, en este seguimiento el sistema va avanzando punto a punto sin conocer más allá del siguiente punto inmediato, es decir que si va a recorrer los puntos en orden sin saltarlos para determinar los puntos deseados y su respectivo orden el subsistema hace uso de un bloque que ejecutará la función `getWaypoints` en Matlab, esta función trae las instrucciones necesarias para que el usuario lo pueda modificar según la ruta que desea establecer como aparece a continuación:

```
Function [ waypoint ] = getWaypoints ()
%Number of waypoints. Edit this value as desired.
nPoints = nP;
waypointsListARDrone = zeros(5, nPoints);
% Edit the following entries for k = 1, 2, ..., nPoints
% waypointsListARDrone(:, k) = [Xe(m) , Ye(m), h (m), Yaw(deg) , waiting-time(sec)
waypointsListARDrone(:, 1) = [ 1 ; 1 ; 2 ; 0 ; 12 ] ;
waypointsListARDrone(:, 2) = [ -2 ; 1 ; 1 ; 0 ; 5 ] ;
waypointsListARDrone(:, 3) = [ 4 ; 2 ; 0.5 ; 25 ; 10 ] ;
```

Los datos de X, Y y Altura se deben ingresar en metros, el Yaw o ángulo de cabeceo se debe ingresar en grados por último el tiempo que debe permanecer el dron en cada punto se debe ingresar en segundos. La Figura 17 muestra el bloque lógica de guía con sus respectivas entradas y salidas, en el anexo D se muestra el contenido del bloque y su programación interna.

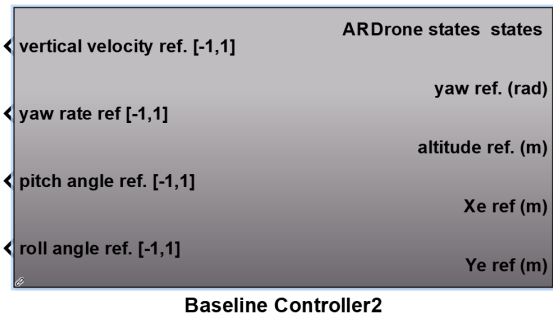
**Figura 17:** Guidance Logic



### Baseline Controller

En este subsistema se encuentra la etapa de control la cual se encarga de corregir el error proveniente de los datos del dron con respecto a los datos de referencia provenientes del subsistema de guía en él se controla las cuatro variables que me determinan el desplazamiento y posición del dron es decir el Pitch, el Roll, el Yaw y la Altura, el kit de desarrollo sólo cuenta con un control de tipo proporcional para cada variable con una ganancia específica para cada una de ellas. La Figura 18 muestra el bloque controlador de línea de base con sus respectivas entradas y salidas. En el anexo E se muestra el contenido del bloque y su programación interna.

**Figura 18:** Baseline Controller



La forma de operar del modelo real del kit de desarrollo para el AR. Drone 2.0 se puede resumir en:

- A. Ingresar la trayectoria deseada en *getWaypoints.m*
- B. Procesar la trayectoria en el subsistema **Guidance logic**
- C. Controlar la trayectoria en el subsistema **Baseline Controller**
- D. Enviar los datos corregidos al dron desde el subsistema **AR. Drone Simulation Block**
- E. Recibir los datos del dron desde el subsistema **AR. Drone Simulation Block**
- F. Estimar la posición del dron en el subsistema **Position estimation**
- G. Visualizar la posición del dron en el subsistema **Visualization of Drone states**
- H. Enviar los datos recibidos del dron al subsistema **Baseline Controller** para corregirlos
- I. Repetir a partir del paso 3 hasta que se terminen los puntos.

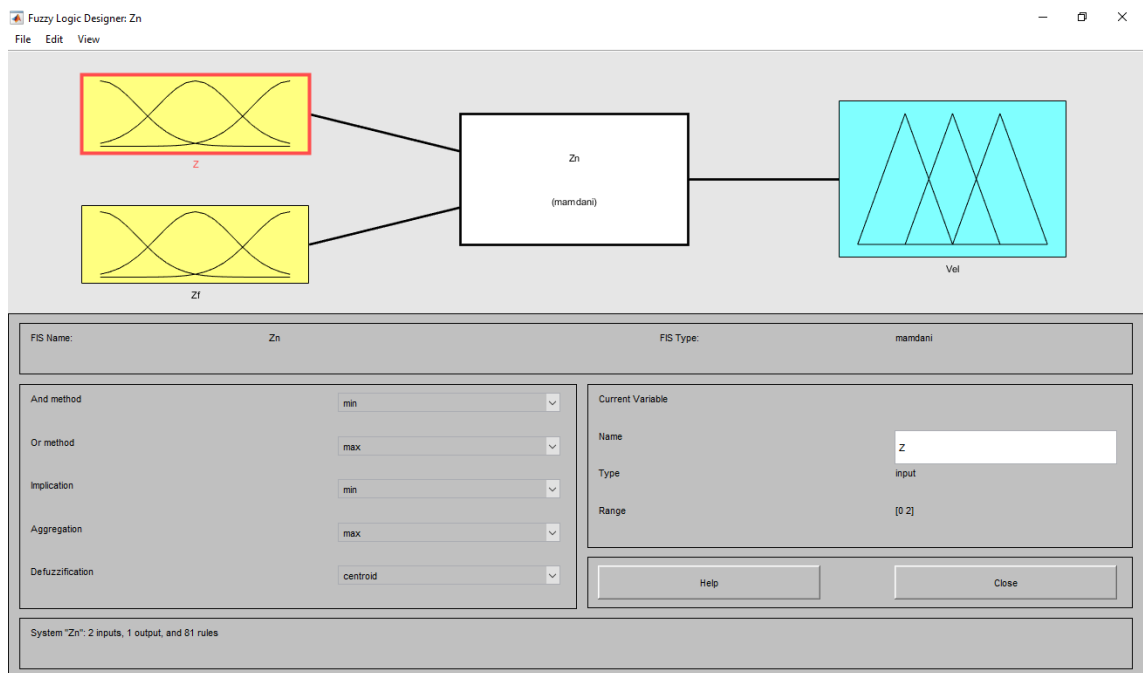
## 4.2. CONTROL FUZZY

Para la etapa de control se utiliza lógica difusa debido a las ventajas útiles que presenta para el proyecto como su facilidad de implementación en este caso particular se hace uso del Fuzzy logic toolbox de Matlab/simulink además la lógica difusa tiene un gran desempeño frente a sistemas no lineales o de difícil modelación ya que no depende de ecuaciones matemáticas complejas y extensas y lo que es más importante no se necesita conocer el modelo matemático que rige el funcionamiento del sistema, al ser el dron un vehículo aéreo presenta un modelamiento complejo y extenso de su sistema por ello para no detenerse en el modelamiento se trabaja con lógica difusa. Como se mencionó en el párrafo anterior se hace uso del Fuzzy logic toolbox de Matlab/simulink esta herramienta permite construir y analizar los resultados de un sistema de inferencia difusa (FIS) por medio de sus diversos componentes:

### Diseñador de lógica difusa

En esta parte como se muestra en la Figura 19 se define la cantidad de variables de entrada que en este caso son dos y salida que siempre es una estas no están limitadas directamente pero si son demasiadas se verá comprometido el desempeño del equipo en el cual se ejecuten. Aquí se pueden editar los nombres de las variables y el método de defuzzificación.

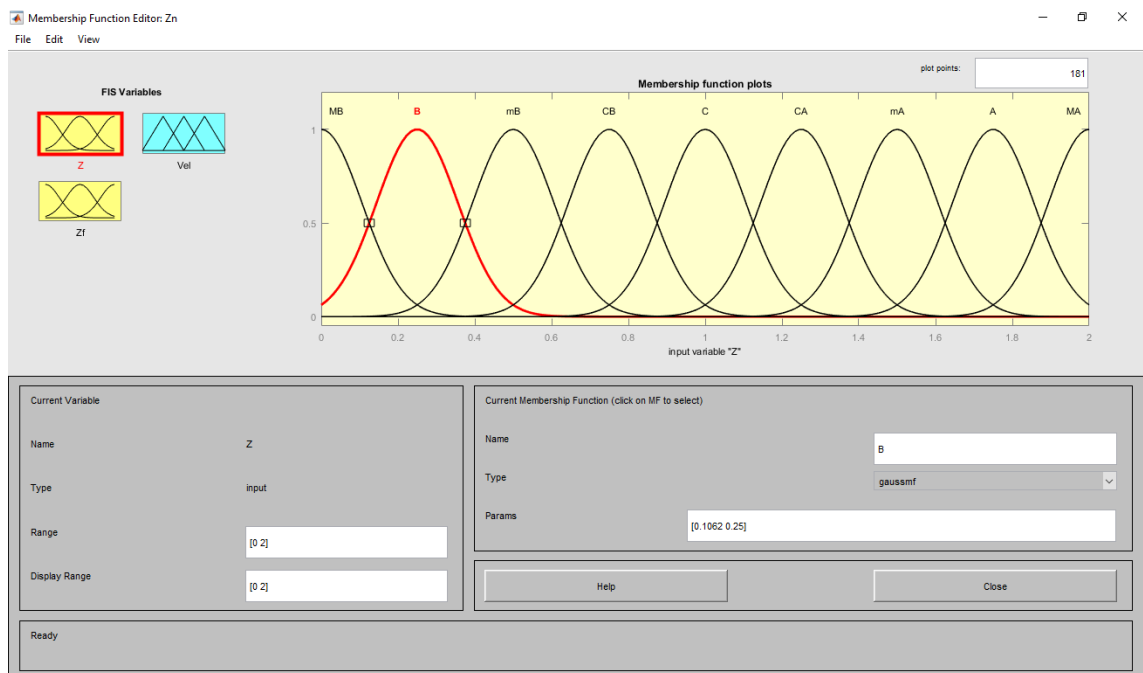
**Figura 19:** Diseñador de lógica difusa



## Editor de funciones de membresía

En este editor como se muestra en la Figura 20 se puede modificar las funciones de membresía de las variables de entrada y la de salida para cada variable se puede elegir la cantidad de funciones de membresía que en este caso son nueve el tipo de curva que tiene cada función de membresía que en este caso se usan gaussianas y triangulares, los rangos máximos y mínimos de cada variable como el de cada función de membresía y el nombre de cada función.

**Figura 20:** Editor de funciones de membresía



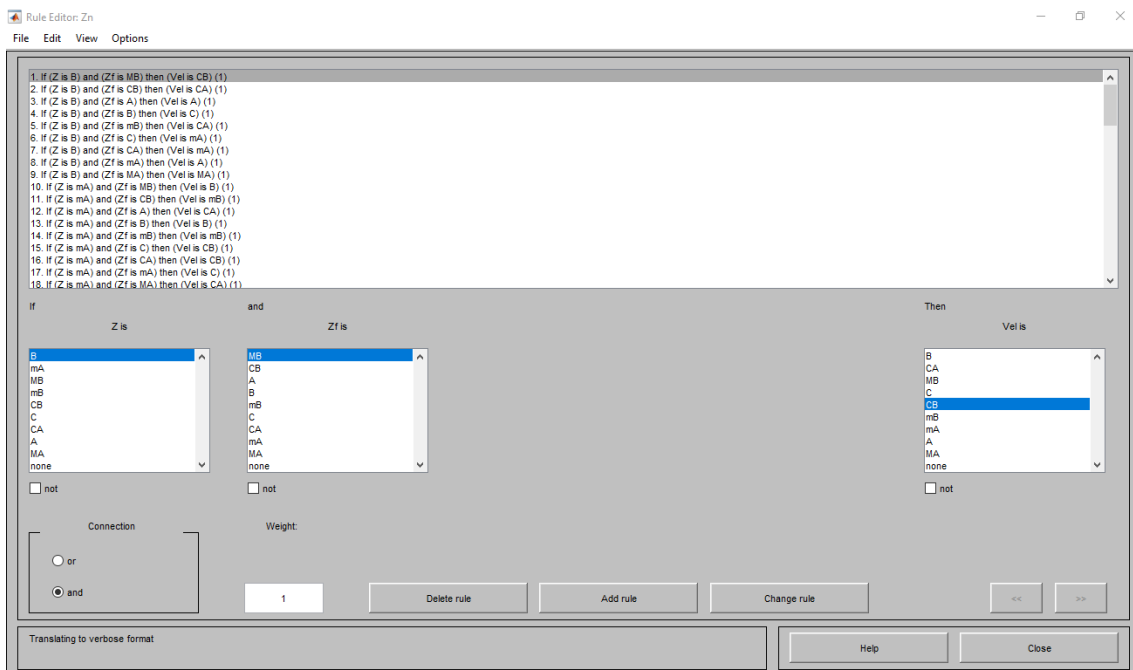
## Editor de reglas

Como se muestra en la Figura 21 el editor de reglas permite establecer las reglas que definen el comportamiento del sistema, para esto se selecciona una función de membresía de una de las variables de entrada y se relaciona con otra función de membresía de otra variable de entrada por medio de una sentencia lógica de disyunción ( $\vee$ , Or, o) o de conjunción ( $\wedge$ , And, y) luego de relacionarlas se elige entre ellas una función de membresía perteneciente a la variable de salida que se quiera, las funciones en general pueden ser de forma afirmativa o negativa obteniendo las respectivas reglas.

$$1. (Z = B) \& (Zf = MB) \Rightarrow (Vel = CB)(1)$$

Además se puede borrar, cambiar y adicionar las reglas según se considere pertinente por el diseñador.

Figura 21: Editor de reglas



## Visor de reglas

Este visor como se muestra en la Figura 22 permite visualizar las reglas de forma activa y cómo estas de manera singular afectan el resultado final del sistema.



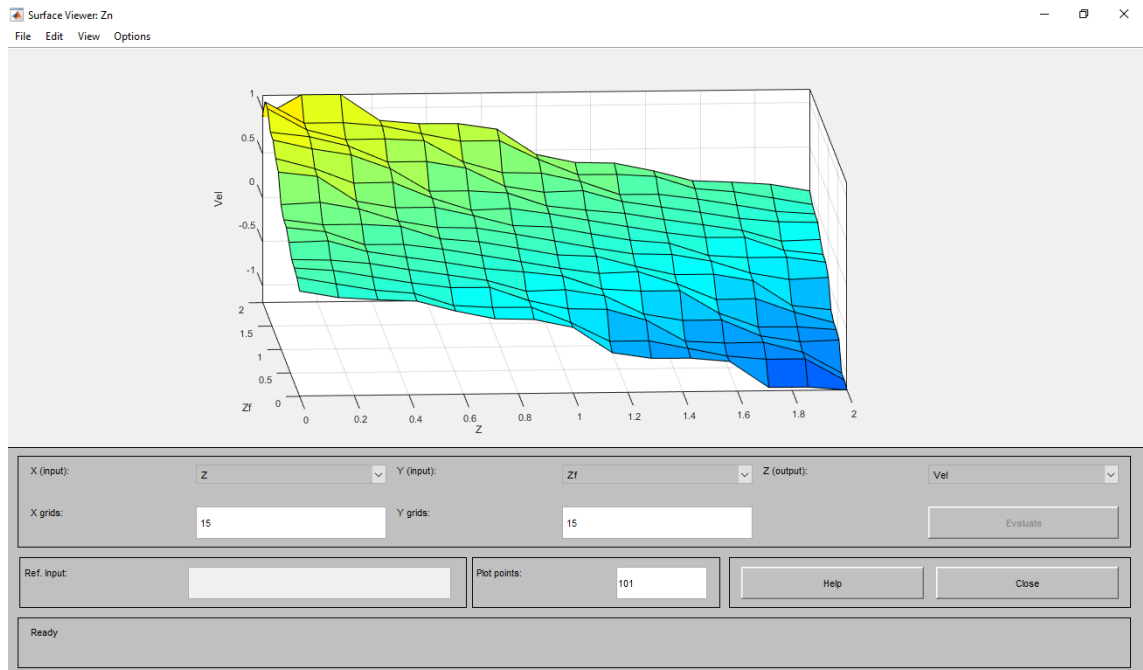
**Figura 22:** Visor de reglas



### Visor de superficie

Este visor como se muestra en la Figura 23 permite visualizar la relación entre una de las salidas y las entradas en un mapa superficial que en sus ejes tiene las variables de entrada y salida.

**Figura 23:** Visor de superficie

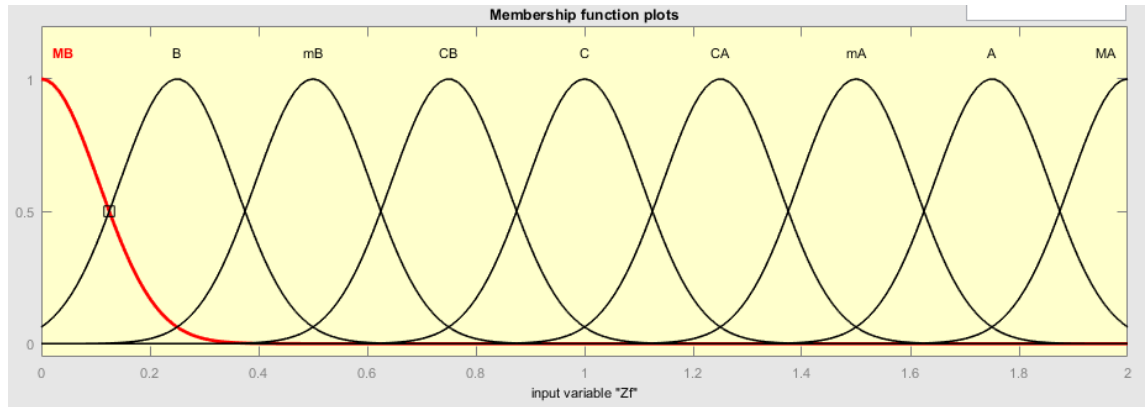


La herramienta de MATLAB de lógica difusa permite elegir la clase de sistema difuso que se va a trabajar entre sistema Sugeno y sistema Mamdani en este caso se elige el último ya que tiene mayor similitud con el análisis humano y una mayor aceptación sin embargo cabe mencionar que el sistema Sugeno es computacionalmente más eficiente.

Para este proyecto se realizan cuatro controladores difusos uno para cada ángulo de Euler presente y otro adicional para la altura todo esto por medio de tres sistemas de inferencia difusa diferentes. En general todos los sistemas comparten dos entradas, una de referencia y la otra es la señal proveniente del dron. En la salida se muestra una variable la cual es enviada al vehículo además realizan el proceso de defusificación por el método del centroide el cual determina la salida del sistema difuso hallando el centro del área bajo la curva de la función de membresía que sea pertinente en el momento se hace énfasis que el sistema de control de altura está diseñado para una elevación máxima de dos metros esta altura se escoge como forma de muestra para el proyecto pero el dron utilizado puede alcanzar alturas de 50 metros idealmente, a continuación se ahonda en el sistema de control diseñado y aplicado:

Como se muestra en la Figura 24 para la altura se utilizan nueve funciones de membresía tanto en las entradas como en la salida las funciones de la entrada son del tipo gaussiana (gaussmf) con un rango de [0 2] expresando que las entradas son alturas donde su máximo valor será de dos metros las funciones están esparcidas a lo largo del rango de manera uniforme.

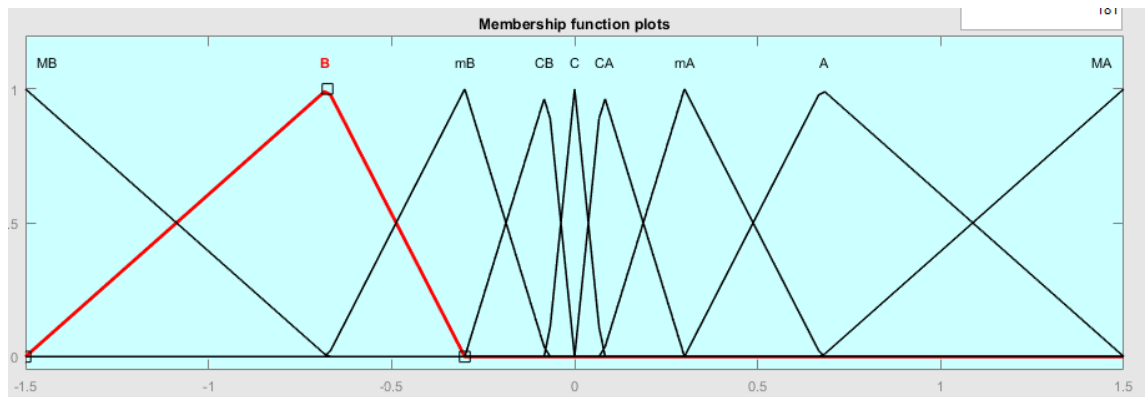
**Figura 24:** Nueve funciones Gaussianas para las variables de Entrada



Como se muestra en la Figura 24 la nomenclatura hace referencia a los estados de elevación que se toman o se divide el rango a trabajar: Muy Bajo, Bajo, medianamente Bajo, Centro Bajo, Centro, Centro Alto, medianamente Alto, Alto y Muy Alto, respectivamente.

En cuanto a la salida como se muestra en la Figura 25 el rango manejado es de  $[-1.5 \ 1.5]$  que es la velocidad de referencia en metros por segundo (m/s) para la salida se emplean señales triangulares las cuales en su área inferior son amplias en los extremos pero en cuanto se van acercando al centro van estrechándose ya que se necesitan cambios más bruscos cuanto más alejado se está del punto de referencia y más suaves en cuanto se va aproximando a la altura de referencia.

**Figura 25:** Nueve funciones Triangulares para la variable de Salida

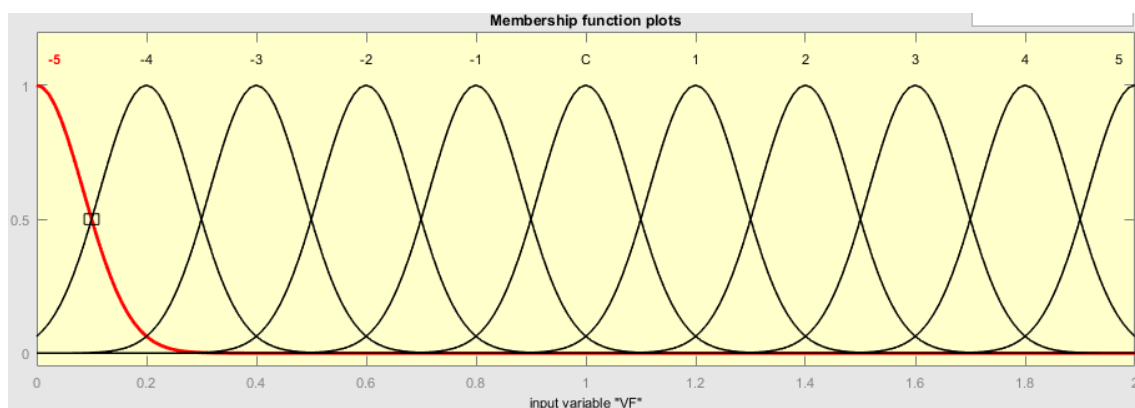


Para que el control funcione se emplean 81 reglas que relacionan todas las funciones de membresía entre sí.

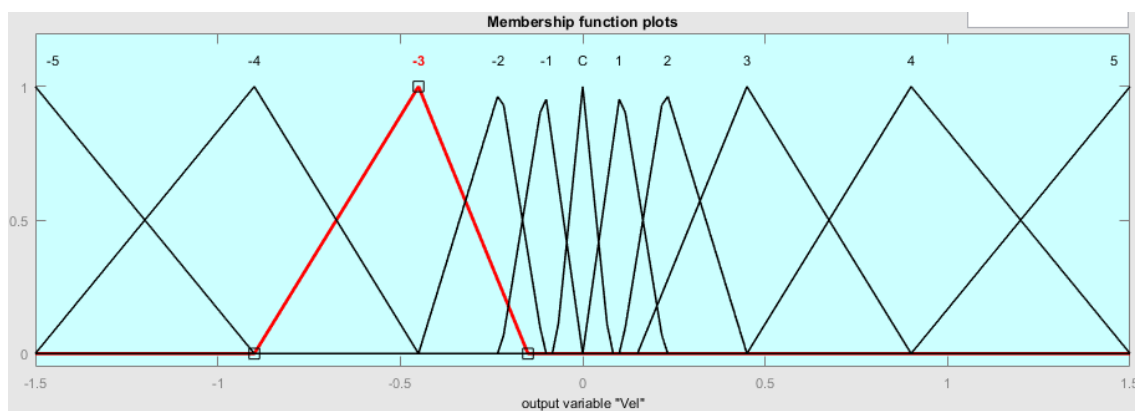
También se experimenta con otros sistemas de control similares que se diferencian en la cantidad de funciones de membresía empleadas, el tipo de función y el número de

reglas se utilizan 11 funciones de membresía en las entradas como se muestra en la Figura 26 y la salida como se muestra en la Figura 27 aplicando el mismo raciocinio empleado anteriormente para la ubicación de las funciones de membresía para no hacer de la nomenclatura de estas funciones algo tedioso se opta por darles valores numéricos que van del -5 pasando por el centro hasta el 5, el número de reglas utilizadas es de 121.

**Figura 26:** Once funciones Gaussianas para las variables de Entrada

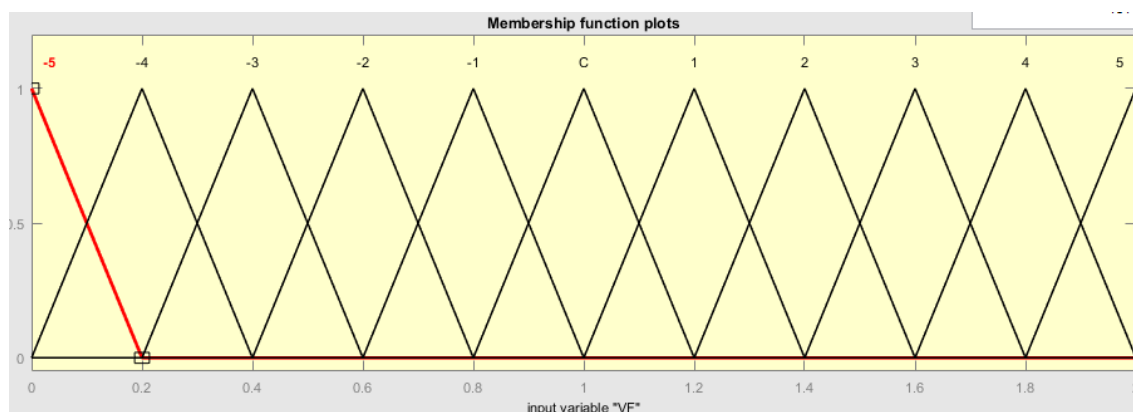


**Figura 27:** Once funciones Triangulares para la variable de Salida



El otro sistema de control que se trabaja es similar al anterior pero se usan señales triangulares en las entradas como se muestra en la Figura 28 esparcidas uniformemente a lo largo del rango de trabajo.

**Figura 28:** Once funciones Triangulares para las variables de Entrada



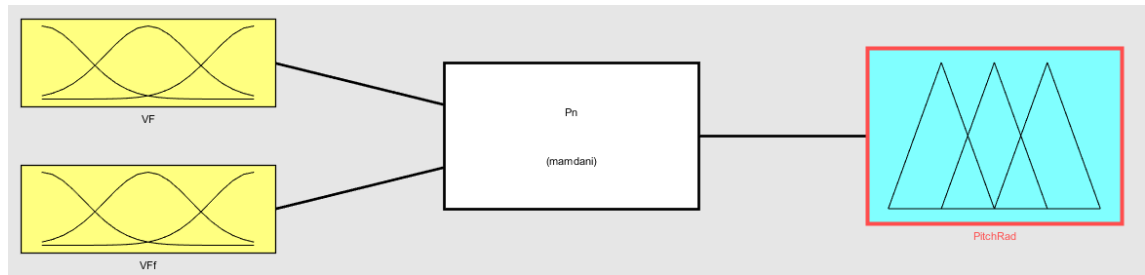
Al controlar los ángulos de dirección se usan las mismas metodologías usadas anteriormente pero variando los rangos de entrada y salida como se expresa a continuación:

**Para el Pitch**(como se muestra en la Figura 29) las entradas son las velocidades en metros por segundo en dirección delantera o trasera lo que se representa en su avance a lo largo del eje X, los rangos trabajados son de  $[-2 \ 2]$  (m/s) para la entrada de la señal proveniente del dron y de  $[-3 \ 3]$  (m/s) para la señal de referencia, la salida del controlador es el ángulo de referencia del pitch que es enviada al dron, este toma valores de  $[-1.5 \ 1.5]$  radianes como se muestra en la Figura 30.

**Figura 29:** Pitch en el AR. Drone 2.0



**Figura 30:** Variables de Entrada y Salida para el control difuso del Pitch



**Para el Roll** (como se muestra en la Figura 31) las entradas son las velocidades en metros por segundo en dirección lateral positiva o negativa lo que se representa en su avance a lo largo del eje Y los rangos trabajados son de  $[-1.5 \ 1.5]$  (m/s) para la entrada de la señal proveniente del dron y de  $[-3 \ 3]$  (m/s) para la señal de referencia, la salida del controlador es el ángulo de referencia del roll que es enviada al dron este toma valores de  $[-1.5 \ 1.5]$  radianes al igual que el pitch como se muestra en la Figura 30.

**Figura 31:** Roll en el AR. Drone 2.0



**Para el Yaw** (como se muestra en la Figura 32) las entradas son los ángulos en radianes del cabeceo (Yaw) positivo o negativo lo que se representa en su rotación frontal o desplazamiento angular en  $\phi$  desde la perspectiva de un plano esférico los rangos trabajados son de  $[-\pi \ \pi]$  radianes para las dos entradas (señal proveniente del dron y la señal de referencia), la salida del controlador es la velocidad angular de referencia del ángulo Yaw que es enviada al dron este toma valores de  $[-6 \ 6]$  radianes por segundo como se muestra en la Figura 30.

**Figura 32:** Yaw en el AR. Drone 2.0



Por último cabe recalcar que la cantidad, el tipo y la distribución de las funciones al igual que las reglas usadas se hallan después de un análisis heurístico según el criterio de los diseñadores y el conocimiento sobre el modelo a trabajar.

### 4.3. INTERFAZ GRÁFICA

Para la interfaz gráfica se hace uso de la herramienta guide para su creación por medio de MATLAB GUI la cual facilita el diseño ya que es posible interactuar directamente con los elementos que compondrán la interfaz o también se puede modificar por medio de código para trabajar en tiempo se elige realizar en MATLAB ya que todo el proceso de control esta hecho en este, migrar los datos de un programa a otro podría presentar problemas o retardos no deseados.<sup>26</sup>

Anteriormente se ha mencionado que se trabaja de forma simulada y real por ello se realizan primero 2 entornos de trabajo uno para cada forma, después se realiza uno adicional donde el usuario puede ingresar de forma libre las alturas y el tiempo con las que desea trabajar en el dron para finalizar se realiza una última vista la cual es la que se ve al comenzar la aplicación y es la que permite al usuario navegar entre las 3 primeras vistas anteriormente mencionadas, a continuación se muestran y explican el funcionamiento de cada una de ellas:

#### Vista General

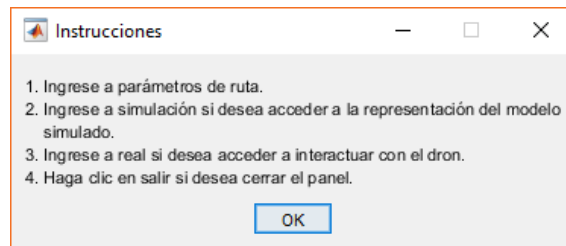
Cada vez que se ingresa a una vista se abre de manera automática una ventana de aviso como se muestra en la Figura 33 la cual contiene las instrucciones para el buen uso de la

---

<sup>26</sup>Guide, *The mathworks*.

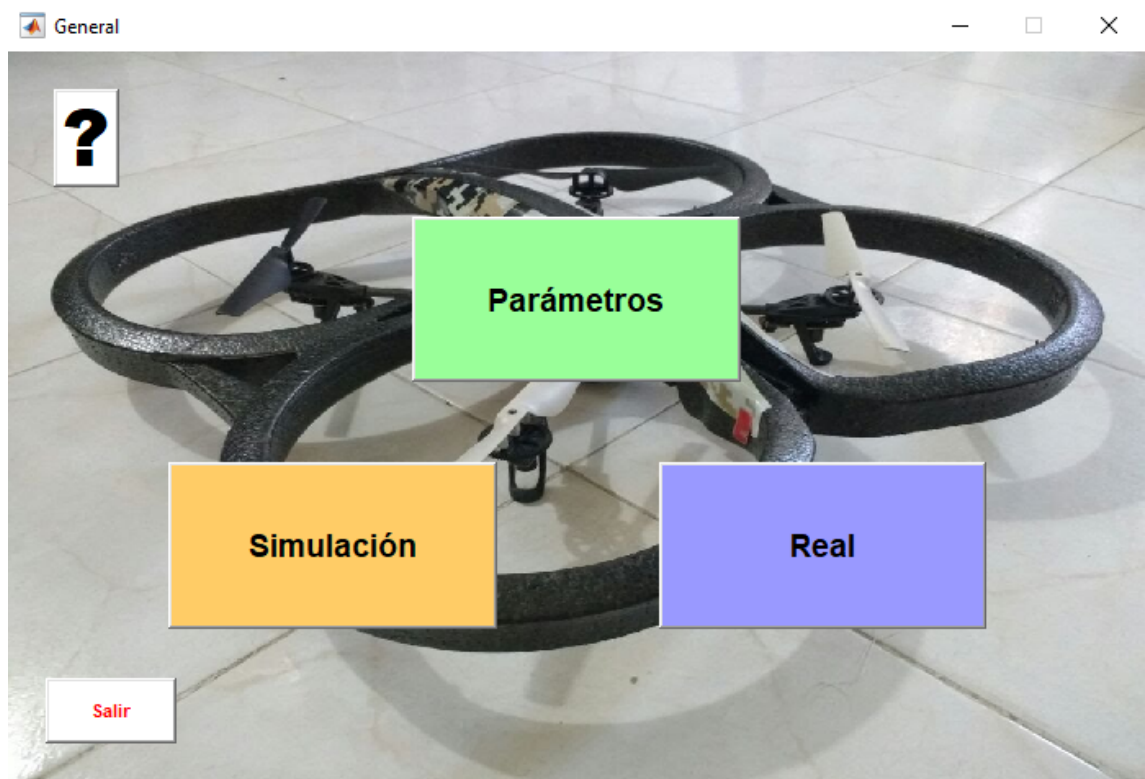
interfaz en la cual se encuentre, no se puede avanzar en la aplicación de no ser que se dé clic en el botón “OK” o se cierre la ventana esto hace que el usuario lea las instrucciones:

**Figura 33:** Instrucciones Generales



Esta vista como se muestra en la Figura 34 contiene los botones que permiten ingresar a los demás paneles, es decir que tiene la función de enlazar las diferentes vistas diseñadas.

**Figura 34:** Vista General



**Botón de ayuda [?]:** Al clickear este botón se despliega nuevamente la ventana emergencia en la cual están contenidas las instrucciones.

**Botón Parámetros:** Permite ingresar a la ventana de Parámetros en la cual se puede ingresar las alturas y los tiempos de los puntos de referencia.



**Botón Simulación:** Permite ingresar al panel de control que trabaja con el modelo simulado del dron donde se pueden visualizar de manera ideal las alturas ingresadas en la ventana de parámetros.

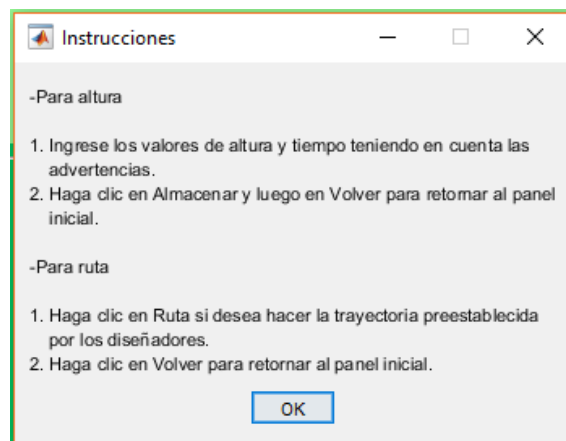
**Bloque Real:** Permite ingresar al panel de control que interactúa con el dron físico en tiempo real desde este se pueden ejecutar las diferentes funciones de control sobre el dron y visualizar las alturas que está cursando al instante.

**Botón Salir:** Permite salir de la aplicación.

### Vista Parámetros

Nuevamente lo primero que se ve es la ventana como se muestra en la Figura 35 que contiene las instrucciones para el buen desempeño de la aplicación.

**Figura 35:** Instrucciones De Parámetros



En esta vista como se muestra en la Figura 36 se puede ingresar hasta un total de 5 alturas de referencia con sus respectivos tiempos de vuelo, adicionalmente se cuenta con un botón de ruta el cual carga los datos de una única ruta de muestra de seguimiento de trayectoria diseñado.

Figura 36: Vista De Parámetros

**Botón de ayuda [?]:** Al clicar este botón se despliega nuevamente la ventana emergencia en la cual están contenidas las instrucciones.

**A. Advertencias:** Se da a conocer información relevante y de gran importancia al usuario para que lo tenga presente al ingresar los demás datos.

**B. Posición:** Se muestra el orden que toman los puntos de referencia ingresados teniendo un máximo de 5 puntos.

**C. Altura(m):** En esta parte se encuentran los espacios editables donde el usuario puede ingresar un máximo de 5 alturas dadas en metros, las alturas que se escriban se deben hacer conforme a las advertencias en este caso si se ingresa un valor que no sea numérico o un valor que sea menor a cero o mayor a 2 la casilla elimina este dato hasta que se ingrese uno válido.

**D. Tiempo:** Nuevamente cuenta con 5 espacios editables donde se ingresa el tiempo en segundos que el dron permanece en la altura correspondiente, al igual que en “Altura” si se ingresa un valor que no es válido (valores que no sean numéricos o un valor que sea menor a 5 o mayor a 35) la casilla elimina este dato hasta que se ingrese uno válido.

**Botón Almacenar:** Almacena los datos ingresados en las casillas de altura y tiempo en forma de matriz numérica que se utiliza como referencia en las etapas de control.

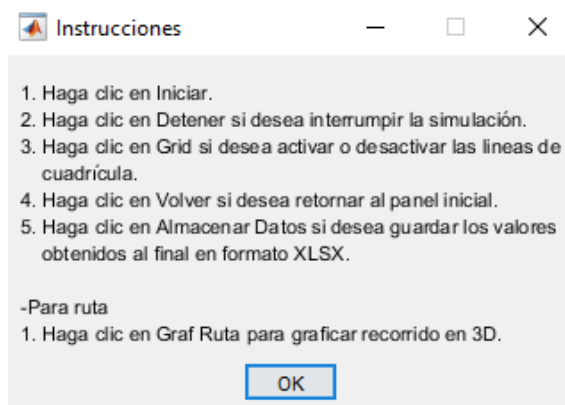
**Botón RUTA:** Si se hace clic en este botón carga automáticamente una ruta predeterminada por los diseñadores a manera de muestra de seguimiento de trayectorias de igual forma habilita la opción de graficar la ruta en un plano tridimensional en los paneles de simulación y real.

**Botón Volver:** Retorna a la pantalla general y cierra la ventana actual.

### Vista Simulación

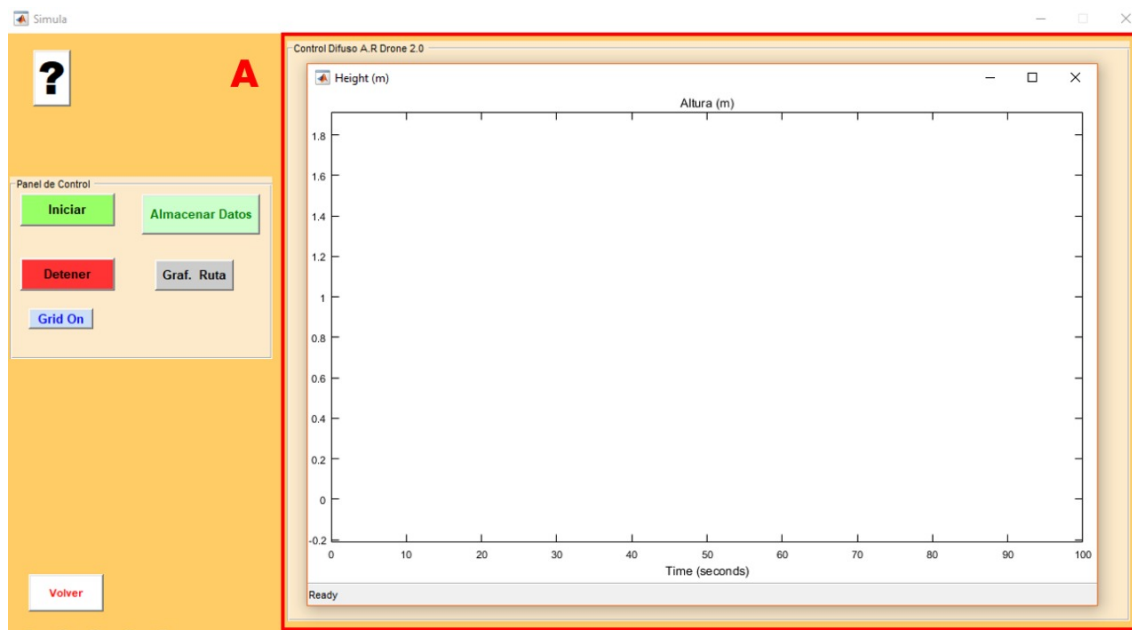
Al igual que en la otras vistas al elegirse esta interfaz lo primero que se despliega es la ventana de instrucciones como se muestra en la Figura 37 para el correcto uso de la aplicación.

**Figura 37:** Instrucciones De Simulación



En la ventana de simulación como se muestra en la Figura 38 están contenido los botones necesarios para interactuar de manera simulada con el modelo del dron al igual cuenta con un espacio en el cual se puede visualizar el desarrollo del comportamiento del dron en una gráfica, a continuación se muestra todo con más detalle:

**Figura 38:** Vista De Simulación



**A. Gráfica De Altura:** En este se grafican las alturas recorridas por el dron en el transcurso del tiempo de manera simulada esta es una gráfica de Simulink la cual es llamada a la pantalla cada vez que se ejecuta alguna función.

**Botón Iniciar:** Permite que se empiece a ejecutar el modelo en simulink comenzando así a simular el vuelo del dron y mostrando su desplazamiento vertical en la gráfica.

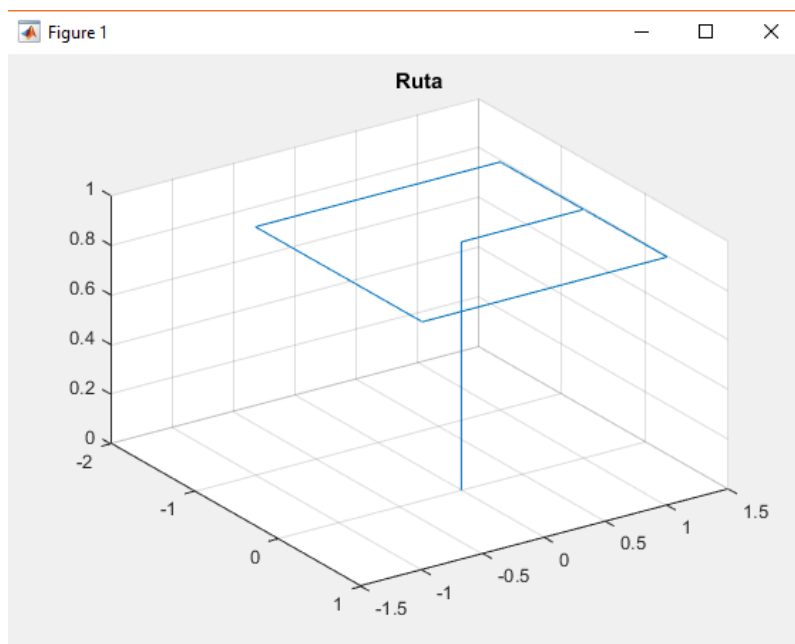
**Botón Detener:** Interrumpe los procesos que se estén llevando a cabo en el modelo simulado.

**Botón Grid On/Grid Off:** Hace visible u oculta la cuadrícula de la gráfica.

**Botón Almacenar Datos:** Registra en un archivo de Excel los datos obtenidos en dos vectores, uno para los valores de tiempo y otro para los de altura (Tener en cuenta que los valores se toman cada 0,065 Segundos que es el tiempo de muestreo trabajado), todo contenido en una hoja que tiene por nombre la fecha y hora del momento en el cual se guardan dichos datos para que sea más fácil su identificación posteriormente, si el archivo de Excel no existe la aplicación crea uno nuevo en donde se almacenan los datos.

**Botón Graf. Ruta:** Este botón solo está habilitado cuando en la ventana de “Parámetros” se elija la función de ruta, si está disponible el botón grafica la trayectoria de forma simulada del dron en una gráfica de tres dimensiones como se muestra en la Figura 39 esta gráfica es fácilmente manipulable por el usuario para que la pueda observar desde cualquier perspectiva.

**Figura 39:** Gráfica Simulada De Ruta en tres dimensiones



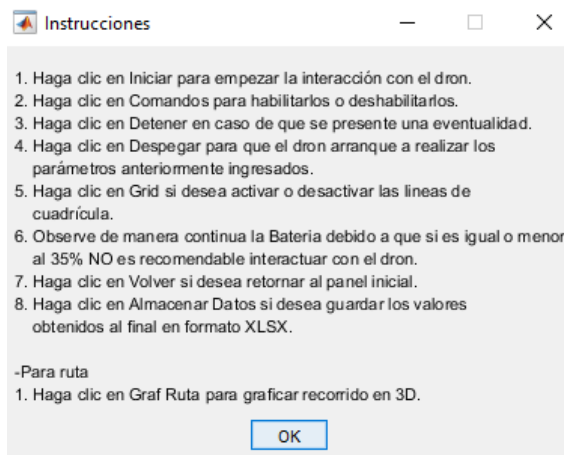
**Botón de ayuda [?]:** Al clicar este botón se despliega nuevamente la ventana emergencia en la cual están contenidas las instrucciones.

**Botón Volver:** : Retorna a la pantalla general no sin antes guardar los cambios hechos al modelo previamente después cierra la ventana actual.

### Vista Real

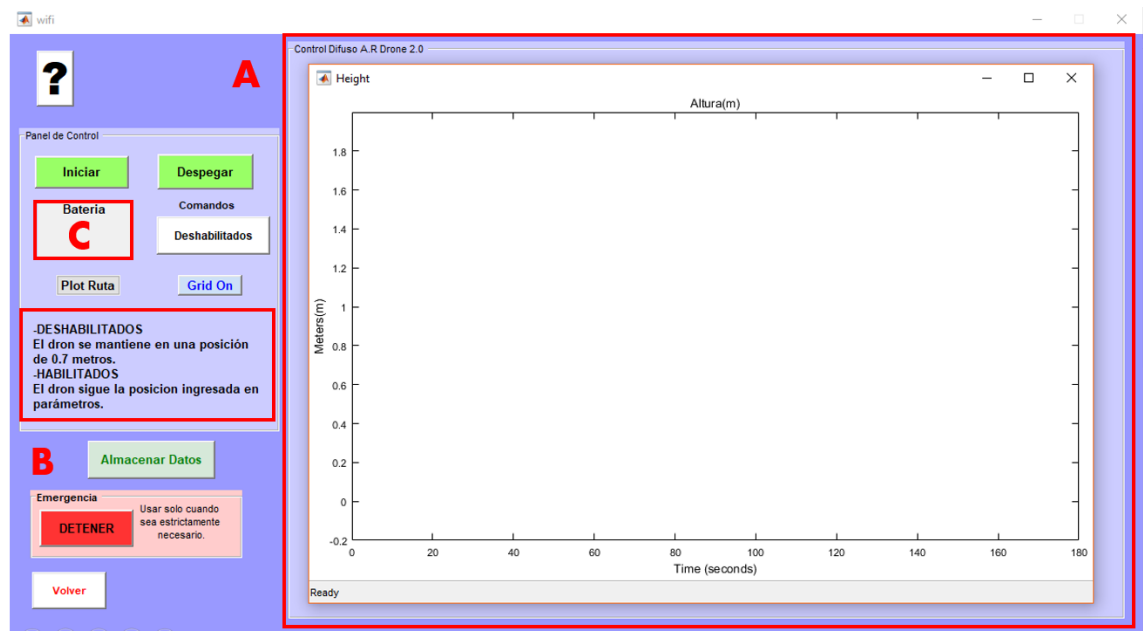
Como se ha venido trabajando en las diferentes vistas que componen esta interfaz gráfica al ingresar a la vista Real lo primero que aparece es la ventana de instrucciones como se muestra en la Figura 40 para que el usuario se guíe.

**Figura 40:** Instrucciones Real



En la ventana “Real” como se muestra en la Figura 41 es donde el usuario puede interactuar de forma directa y en tiempo real con el dron por medio de funciones de despegue, aterrizaje etc. las cuales se explican a continuación:

**Figura 41: Vista Real**



**A. Gráfica de Altura:** En este se grafica la altura que tenga el dron en el transcurso del tiempo de forma real esta es una gráfica de Simulink la cual es llamada a la pantalla cada vez que se ejecuta alguna función.

**B. Explicación del apartado de Comandos:** Se da a conocer al usuario lo que implica el habilitar o deshabilitar los comandos del dron.

**C. Nivel de batería:** En esta parte se puede visualizar circunstancialmente el nivel de batería con el que cuenta el dron en el transcurso del tiempo ya que como lo dicen las instrucciones si el nivel de carga es menor al 35 % el sistema de control del dron se ve comprometido directamente afectando el desempeño de este.

**Botón Iniciar/Detener:** Permite que se empiece a ejecutar el modelo en simulink o se detenga, antes de iniciar es aconsejable cerciorarse que el equipo esté conectado a la red Wi-Fi del dron, una vez se inicie se empezarán a ejecutar todos los procesos de enlace entre el dron y el equipo, por ello es posible que tarde un tiempo razonable en este punto dependiendo de las capacidades computacionales del equipo en el cual se ejecute.

**Botón Despegar/Aterrizar:** Una vez que el modelo haya iniciado por medio de este botón puede ponerse en vuelo el dron para que siga de forma automática los diferentes puntos de referencia ingresados en caso de cualquier contratiempo leve o finalización de los puntos este botón también permite aterrizar el vehículo.

**Comandos Deshabilitados/Habilitados:** Permite habilitar o deshabilitar los comandos que se envían al dron con respecto al seguimiento autónomo de alturas de referencia como se explica en el texto del apartado de explicación de comandos en la interfaz gráfica.

**Botón Plot Ruta:** Este botón solo está habilitado cuando en la ventana de “Parámetros” se elija la función de ruta si está disponible el botón grafica la trayectoria de forma simulada del dron en una gráfica de tres dimensiones esta gráfica es fácilmente manipulable por el usuario para que la pueda observar desde cualquier perspectiva.

**Botón Grid On/Grid Off:** Hace visible u oculta la cuadrícula de la gráfica.

**Botón Almacenar Datos:** Registra en un archivo de Excel los datos obtenidos en dos vectores, uno para los valores de tiempo y otro para los de altura (Tener en cuenta que los valores se toman cada 0,065 Segundos que es el tiempo de muestreo trabajado), todo contenido en una hoja que tiene por nombre la fecha y hora del momento en el cual se guardan dichos datos para que sea más fácil su identificación posteriormente si el archivo de Excel no existe la aplicación crea uno nuevo en donde se almacenan los datos.

**Botón de Emergencia Detener:** Este botón está aislado del resto del conjunto de botones ya que como lo expresa su nombre solo se enfatiza utilizarlo en caso de alguna eventualidad grave, la función de este botón es detener todas las funciones del dron lo que hace que este se precipite a tierra de manera inmediata lo que a grandes alturas o sobre terrenos no adecuados puede comprometer la integridad y funcionamiento del dron.

**Botón de ayuda [?]:** Al clicar este botón se despliega nuevamente la ventana emergencia en la cual están contenidas las instrucciones.

**Botón Volver:** Retorna a la pantalla general no sin antes guardar los cambios hechos al modelo previamente después cierra la ventana actual.

#### 4.4. IMPLEMENTACIÓN

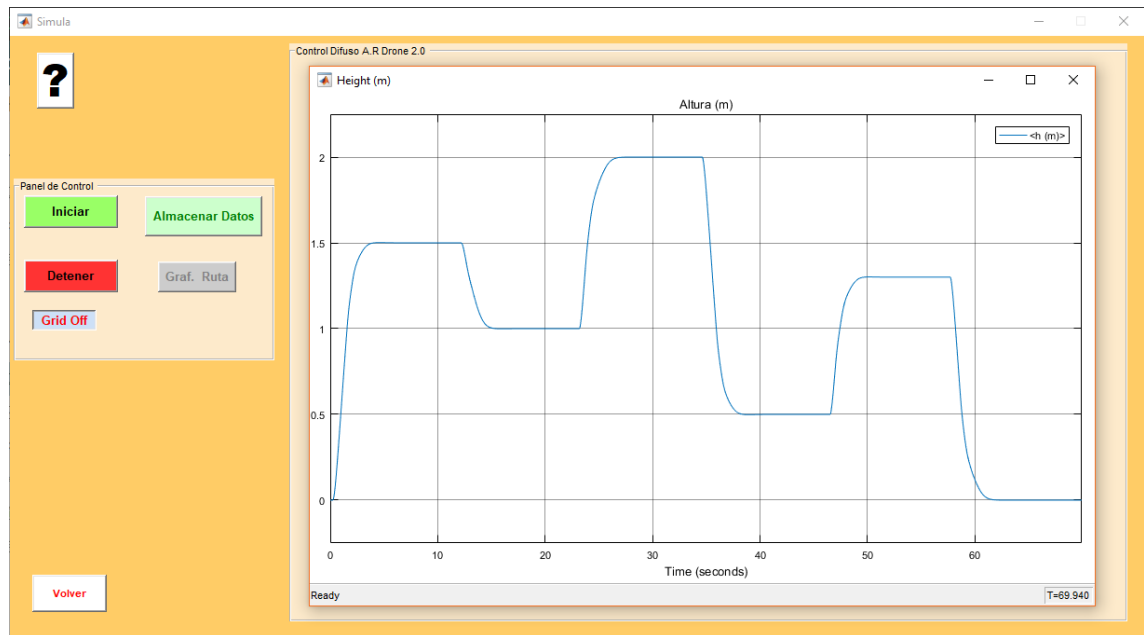
La ejecución del proyecto se lleva a cabo en dos fases, en primera instancia se pone a prueba el sistema de control de manera simulada, una vez se tiene una respuesta óptima se procede a implementar todo de manera real realizando la comunicación y el control directo con el dron en tiempo real.

##### Simulación del Sistema

Se implementa de manera ideal el sistema de control es decir no se tiene en cuenta ningún tipo de perturbación, ni el terreno donde se ejecuta, ni la fricción con el aire, ni los retrasos por las respectivas transmisiones hacia el dron y viceversa. Lo primero es dirigirse a la sección de parámetros e ingresar los diferentes puntos de referencia que se deseen y almacenarlos o bien elegir la función de ruta una vez hecho esto se retorna al panel principal

y se dirige a la opción de simulación la cual empieza a cargar una vez esta la interfaz lista se puede iniciar la simulación.

**Figura 42:** Implementación Simulada



Una vez se ha ejecutado la simulación se puede observar la gráfica del comportamiento de la elevación del dron con respecto al tiempo como se muestra en la Figura 42 y es posible almacenar los datos graficados para hacer uso de ellos en futuras ocasiones para esto se pulsa el botón “Almacenar Datos”, se pueden volver para ingresar nuevos datos o cambiar al modo real cliqueando en el botón “Volver”.

### Implementación del Sistema

Para la implementación real se busca un espacio idóneo en el cual realizar las pruebas ya que se establece desde un inicio que es en un ambiente controlado por ello se decide trabajar en 3 lugares los cuales no fueron afectados en gran medida por las diversas perturbaciones como lo pueden ser las redes Wi-Fi cercanas, objetos en el campo y principalmente el viento además se tiene conocimiento que el dron hace uso de su cámara inferior para detectar cambios de desplazamiento en cualquier dirección por medio del procesamiento de imágenes por ello el suelo del lugar debe contar con figuras o formas que el dron pueda diferenciar en caso de desplazamiento al igual que la luminosidad del lugar que debe ser buena como se muestra en la Figura 43.



**Figura 43:** Dron posicionado para implementación



Ya en el coliseo cerrado de la Universidad Surcolombiana como se muestra en la Figura 44 se inicia la aplicación para la ejecución del control, se comienza por ingresar los niveles de altura y el tiempo que el dron debe permanecer en dicha altura o se puede elegir la función de ruta para que el dron trace la trayectoria preestablecida, se regresa al panel general y se elige el botón “Real”.

**Figura 44:** Dron junto a su interfaz para implementación real



Una vez se carga la ventana se puede iniciar el modelo, cuando este inicie se puede elegir la función despegar para que el dron realice el seguimiento de alturas que el usuario insertó previamente siempre y cuando estén habilitados los comandos, como se muestra en la Figura 45 si no están habilitados el dron sólo se eleva a una altura fija de 0.7 metros, el recorrido que se está ejecutando se puede observar en tiempo real por medio de la grafica de la interfaz, cuando termine los puntos anteriormente ingresados o en caso de alguna eventualidad se puede aterrizar el vehículo así como agregar una cuadrícula a la gráfica para facilitar la visualización de los niveles de altura que se estén trazando.

**Figura 45:** Dron en ejecución de altura



Una vez terminado todo es posible almacenar en un archivo Excel los desplazamientos hechos por el dron al igual que si se elige la opción de “Ruta” se puede visualizar la trayectoria terminada en un plano de tres dimensiones manipulable para que se pueda observar desde todas las perspectivas, la interfaz también cuenta con el botón de emergencia que detiene de manera abrupta el dron para la toma de datos se trabaja con un tiempo de muestreo de 0,065 segundos. En general nunca se obtiene un ambiente 100 por ciento controlado por ello es posible visualizar en los resultados comportamientos o datos que no son muy buenos y manifiestan el error de las perturbaciones principalmente el viento.

## 5. RESULTADOS

Para lograr controlar la elevación del dron en vuelo se diseñan tres sistemas de inferencia lógica difusa, estos también se usan para hacer control de trayectoria al momento de la implementación Real y Simulada se ejecutan los sistemas de tres formas diferentes para cada uno; la primera forma es la más básica consiste en agregar un solo punto de referencia, es decir, un vuelo con una elevación fija, la segunda forma consiste en múltiples puntos de referencia consecutivos y la tercera forma es ingresar puntos de referencia tanto en altura como en espacio para que el dron haga seguimiento de una ruta preestablecida. Algunos resultados obtenidos se muestran a continuación, el desempeño de ciertos resultados no es el esperado debido a las diversas perturbaciones que se explican al momento de presentarse, los resultados están seccionados según el sistema difuso usado para obtenerlos de igual manera cada sección está dividida por cada forma de implementación (punto de referencia fijo, múltiples puntos de referencia y seguimiento de ruta). Dentro de cada forma se muestra primero su respuesta simulada la cual puede ser usada como referencia dado su excelente desempeño seguidamente están los valores obtenidos de manera real.

### 5.1. SISTEMA LÓGICO DIFUSO “FIS9”

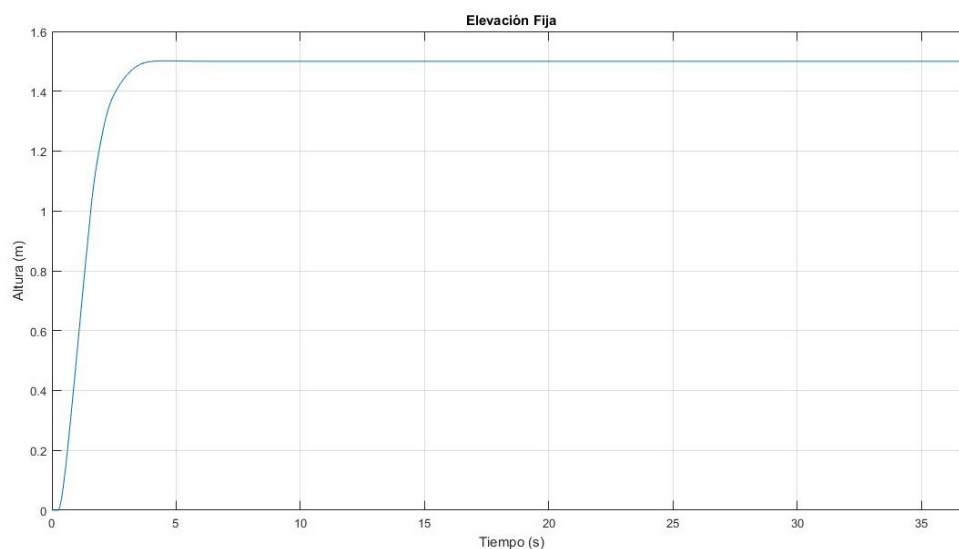
Este sistema de control es el primero que se diseña, consta de:

- 9 Funciones gaussianas para las variables de entrada.
- 9 Funciones triangulares para la variable de salida.
- 81 Reglas.

**Primer tipo de vuelo:** Para la ejecución del vuelo a una elevación fija se utiliza un punto de referencia a 1.5 metros.

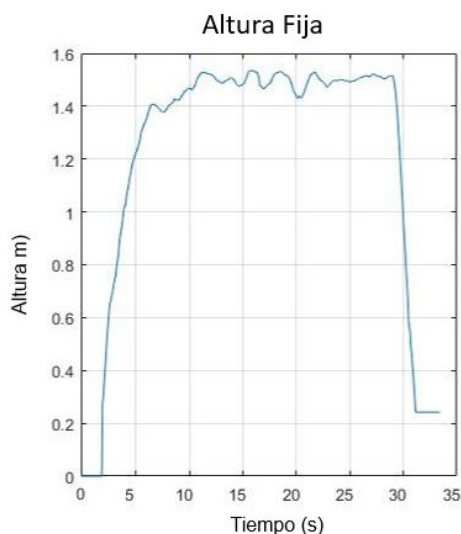
La respuesta simulada con un punto de referencia fijo y 9 funciones gaussianas para las variables de entrada obtenido se muestra en la Figura 46 el tiempo de establecimiento es bastante corto y el error de estado estacionario es nulo, además no presenta sobreimpulso alguno.

**Figura 46:** Respuesta simulada con 9 funciones



En la Figura 47 se muestra la respuesta real de la altura y difiere con la respuesta simulada en la Figura 46 ya que la primera cuenta con un mayor tiempo de establecimiento debido a los retardos de comunicación entre el ordenador y el dron aunque hay presencia de oscilaciones los picos de esta no superan al 10% de la señal total teniendo así un error mínimo y aceptable en lo estipulado este comportamiento se presenta de forma constante en los vuelos.

**Figura 47:** Respuesta real con 9 funciones

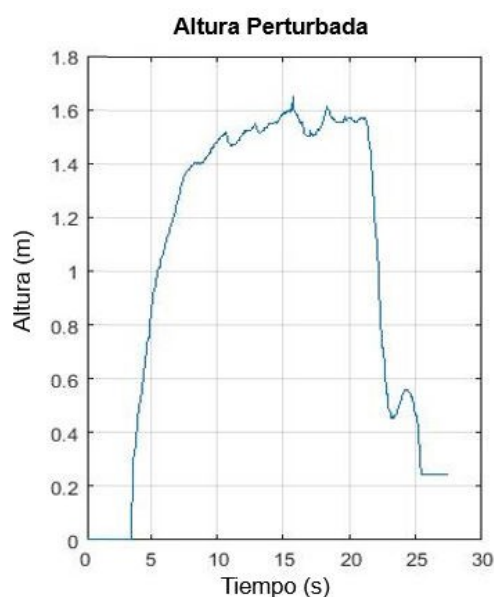


En la Figura 46 se muestra la respuesta real de la altura y difiere en tiempo con la respuesta simulada en la Figura 47 debido a que en la primera necesita enlazar el dron con el ordenador antes de despegar este lapso de tiempo también se gráfica y es el tiempo muerto que se observa al iniciar la gráfica por el contrario en la primera la gráfica inicia

inmediatamente con una presencia casi nula de retardos además el tiempo de vuelo real en un punto difiere del simulado dado a los retardos en la comunicación con el dron y el tiempo que le toma al vehiculo establecerse en la posición indicada.

En la Figura 48 se muestra la respuesta ante una perturbación no controlada con un punto de referencia fijo y 9 funciones gaussianas para las variables de entrada, aunque en la mayoría de las pruebas se tiene perturbaciones mínimas hay ocasiones en las que la fuerza del viento es mayor a lo que puede soportar el dron lo que desencadena una respuesta fuera de lo deseado afectando notablemente su desplazamiento.

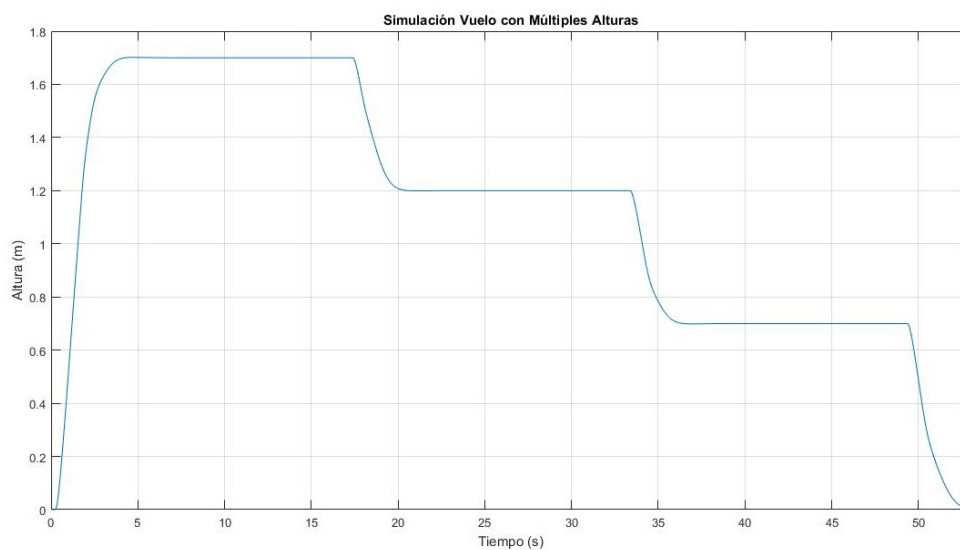
**Figura 48:** Respuesta ante perturbación 9 funciones



**Segundo tipo de vuelo:** Para la ejecución del vuelo en múltiples puntos de referencia de elevación se utilizan tres alturas (1.7 metros, 1.2 metros y 0.7 metros).

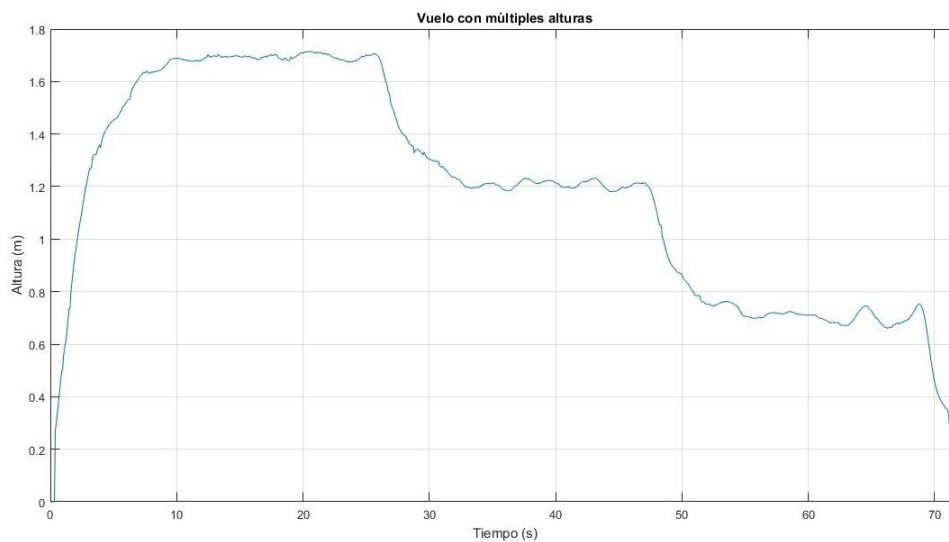
La respuesta simulada con multiples puntos de referencia y 9 funciones gaussianas para las variables de entrada se muestra en la Figura 49 en donde se evidencia el buen desempeño del sistema de control bajo parámetros ideales teniendo por ello un error casi nulo y un tiempo de establecimiento corto.

**Figura 49:** Respuesta simulada con multiples puntos 9 funciones



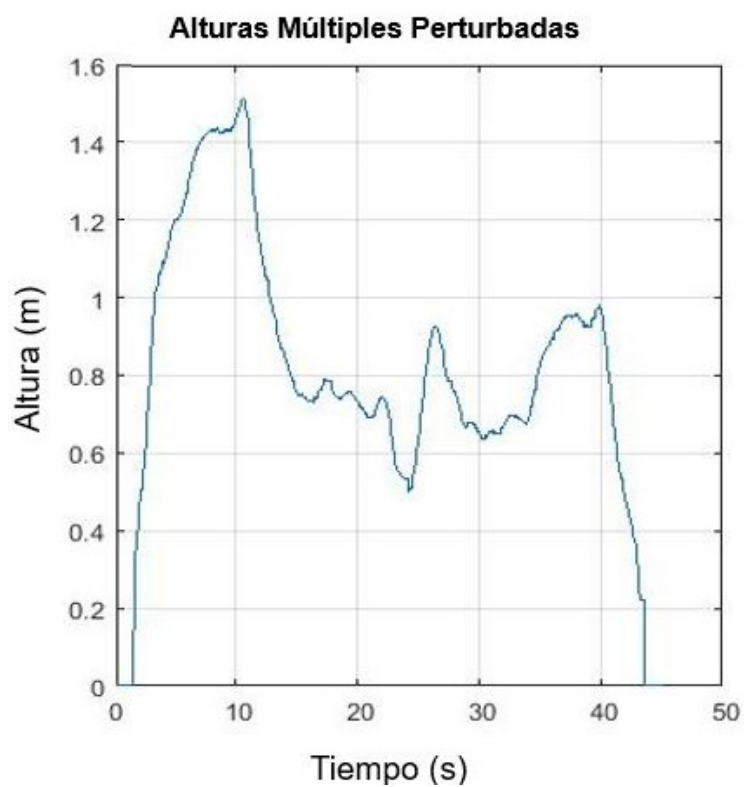
En la Figura 50 se muestra la respuesta real que pese a que los puntos de referencia están a diferentes elevaciones, el desempeño de este sigue siendo óptimo ya que el error de estado estacionario es menor al 10 % y los tiempos de establecimiento son mínimos.

**Figura 50:** Respuesta real con multiples puntos 9 funciones



Al trabajarse con diferentes puntos de referencia el tiempo del dron en vuelo es mayor, por lo tanto en este espacio de tiempo el dron se ve afectado por factores climáticos lo que se evidencia en el resultado del desplazamiento.

**Figura 51:** Respuesta ante perturbación con multiples puntos 9 funciones

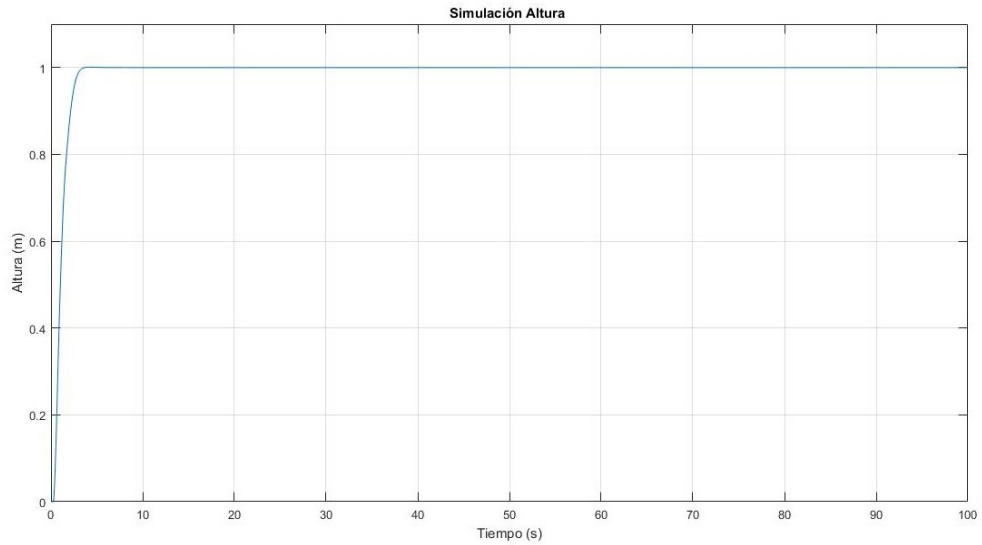


**Tercer tipo de vuelo:** Para la ejecución del seguimiento de ruta se establece que el dron recorra una trayectoria con forma cuadrada de 2 metros por 2 metros a una elevación fija de 1 metro.

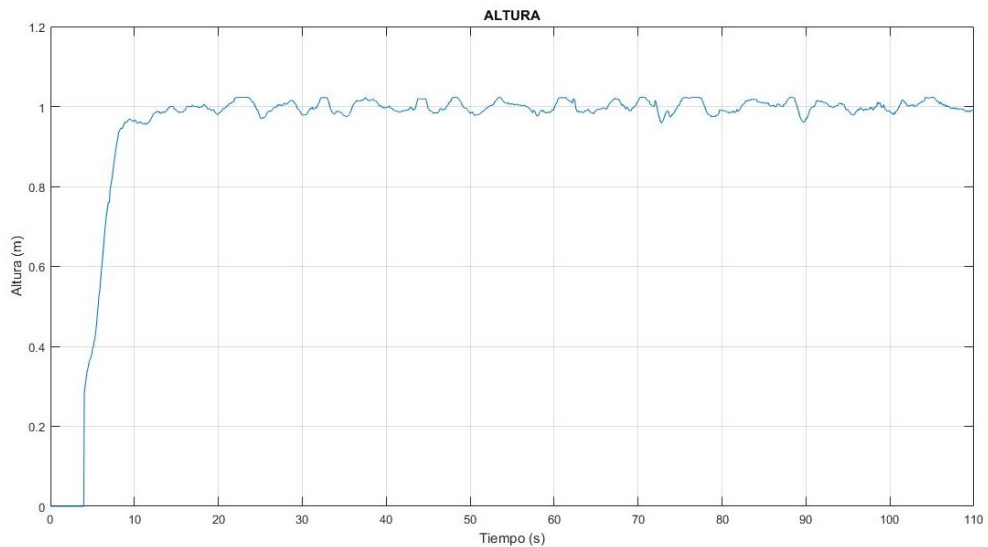
La Figura 52 corresponde a la altura que mantuvo el dron durante la ruta la cual es fija de manera simulada se puede notar que el comportamiento realizado por el dron es el esperado con un error casi nulo, seguido la imagen real que se muestra en la Figura 53 en la cual se visualiza que aunque el comportamiento del dron no es ideal es muy bueno ya que el error es igual o menor al 10 %.



**Figura 52:** Respuesta simulada de altura 9 funciones

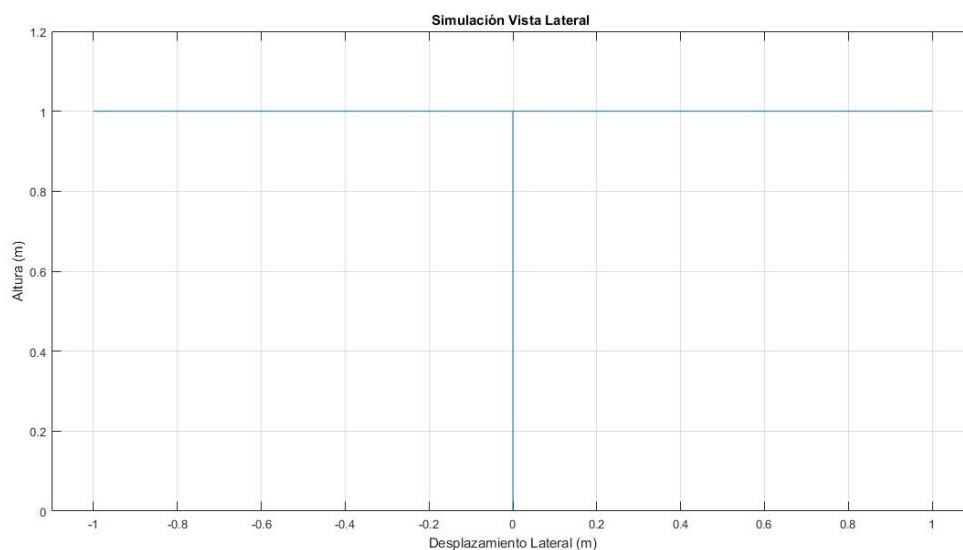


**Figura 53:** Respuesta real de altura 9 funciones

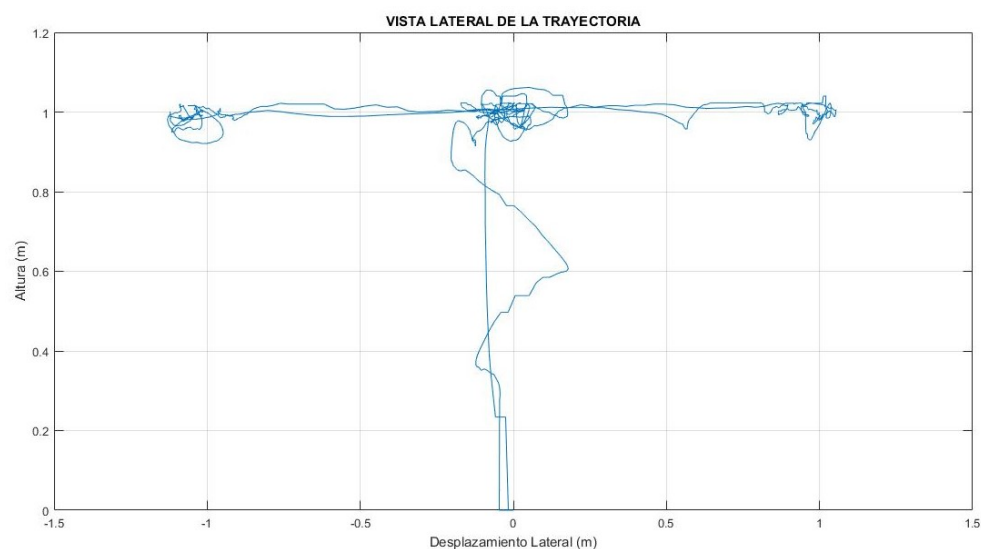


Continuando con la muestra de resultados referentes a la altura se tiene desde otra perspectiva como se muestra en la Figura 54 en la que es posible observar el desplazamiento recorrido por el dron en el espacio hacia adelante o el lado derecho y hacia atrás o el lado izquierdo esto en el plano corresponde a las variables X o Y con respecto a la altura(h) al momento de ejecutar la trayectoria de forma real como se muestra en la Figura 55 se percibe que en la elevación está presente un desplazamiento espacial que en ocasiones es del 50 % de la señal pero que al final de su recorrido retorna a la posición espacial inicial este comportamiento sólo se evidencia en el despegue pese a que este error espacial esta presente mientras se eleva al alcanzar la altura de referencia dicho error ya no está presente. Cuando el dron esta posicionado en la altura de referencia comenzará a seguir la trayectoria establecida en una elevación fija dicha elevación presenta oscilaciones menores al 10 % las cuales no representan ningún obstáculo para que el dron realice el recorrido.

**Figura 54:** Respuesta simulada vista 2D 9 funciones

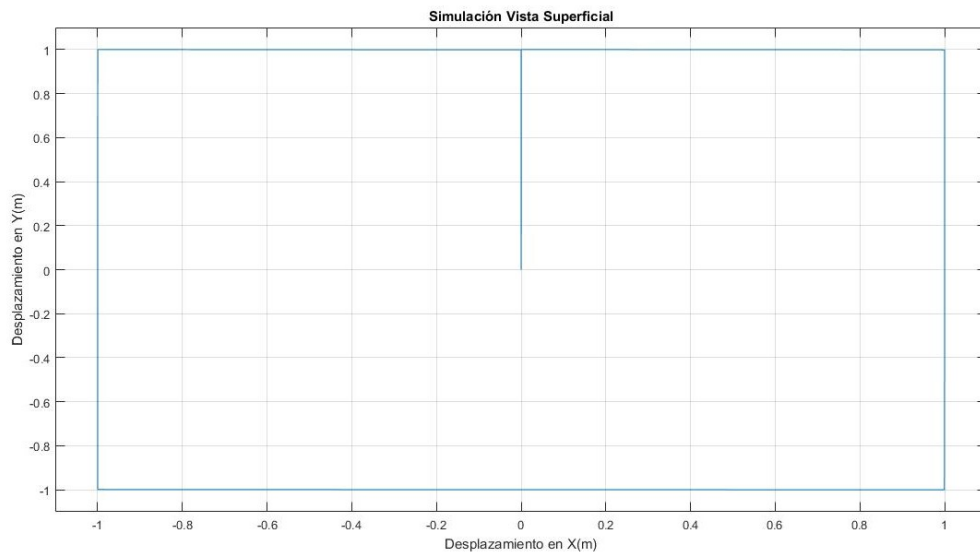


**Figura 55:** Respuesta real vista 2D 9 funciones

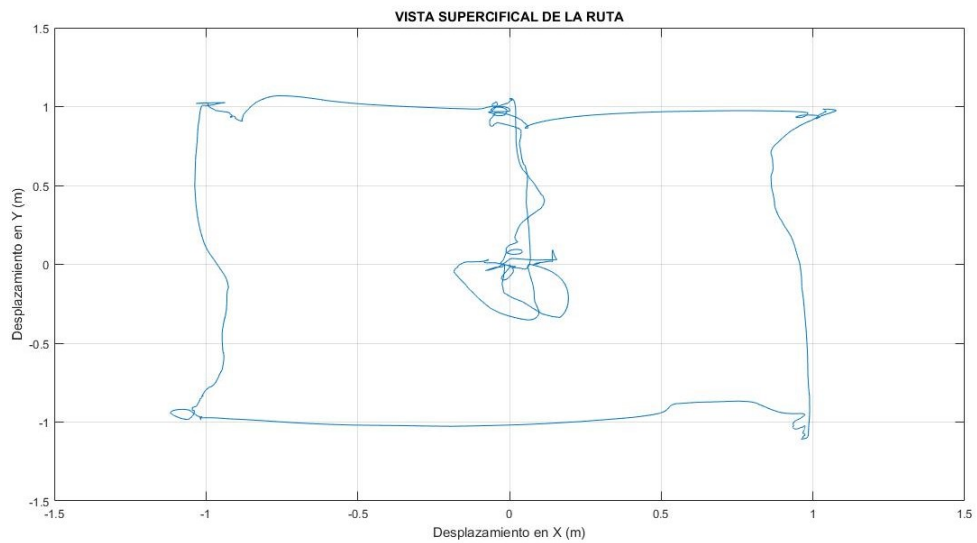


Al visualizar desde una perspectiva superior la trayectoria del dron se puede notar con mayor claridad la ruta recorrida por el dron para efecto de análisis se enfoca en la figura exterior trazada, es decir un cuadrado, de manera simulada se tiene una respuesta ideal como se muestra en la Figura 56 con un trazo perfecto en la implementación real como se muestra en la Figura 57 aunque hay oscilaciones en el desplazamiento de X y Y es posible identificar notoriamente la figura que se quiso trazar en los desplazamientos entre borde y borde pese a que hay desvíos de los puntos de referencia el error nunca supera al 10 % manteniendo así la figura establecida, en los bordes se nota una mayor fluctuación del vuelo del dron ya que en estos puntos el deberá permanecer fijo durante un tiempo desafortunadamente las condiciones no son totalmente controladas lo que con lleva este comportamiento.

**Figura 56:** Respuesta simulada vista superior 9 funciones

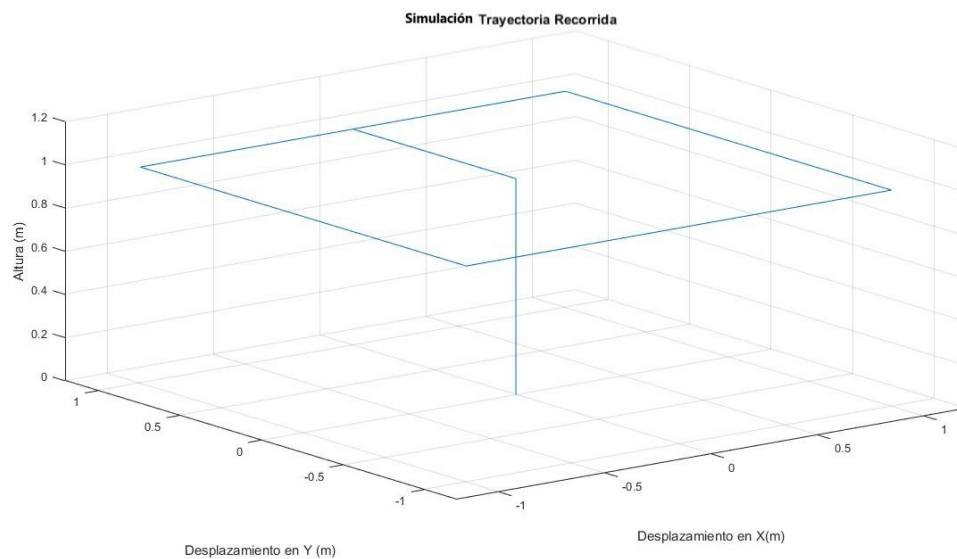


**Figura 57:** Respuesta real vista superior 9 funciones

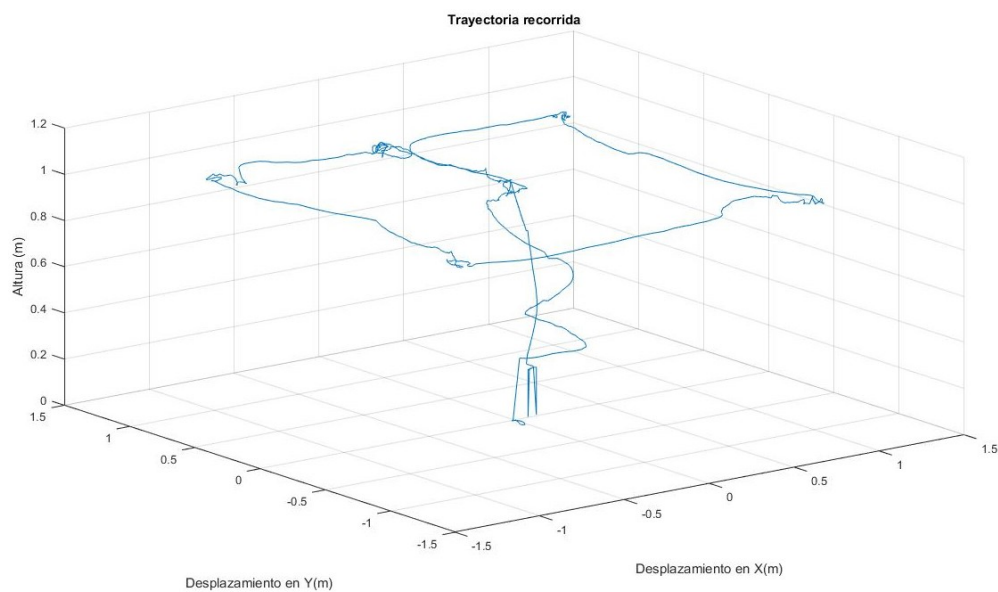


Por último se tiene una visualización general donde se aprecia la elevación y la trayectoria del dron de manera simulada como se muestra en la Figura 58 y en su implementación real como se muestra en la Figura 59.

**Figura 58:** Respuesta simulada vista 3D 9 funciones



**Figura 59:** Respuesta real vista 3D 9 funciones



## 5.2. SISTEMA LÓGICO DIFUSO “FIS11GA”

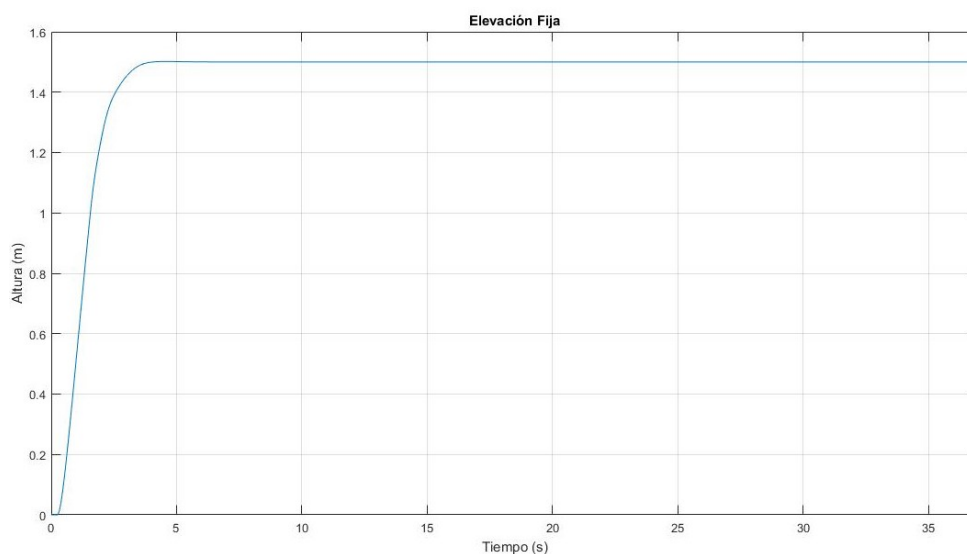
Este sistema de control es el segundo que se diseña, consta de:

- 11 Funciones gaussianas para las variables de entrada.
- 11 Funciones triangulares para la variable de salida.
- 121 Reglas.

**Primer tipo de vuelo:** Para la ejecución del vuelo a una elevación fija se utiliza un punto de referencia a 1.5 metros.

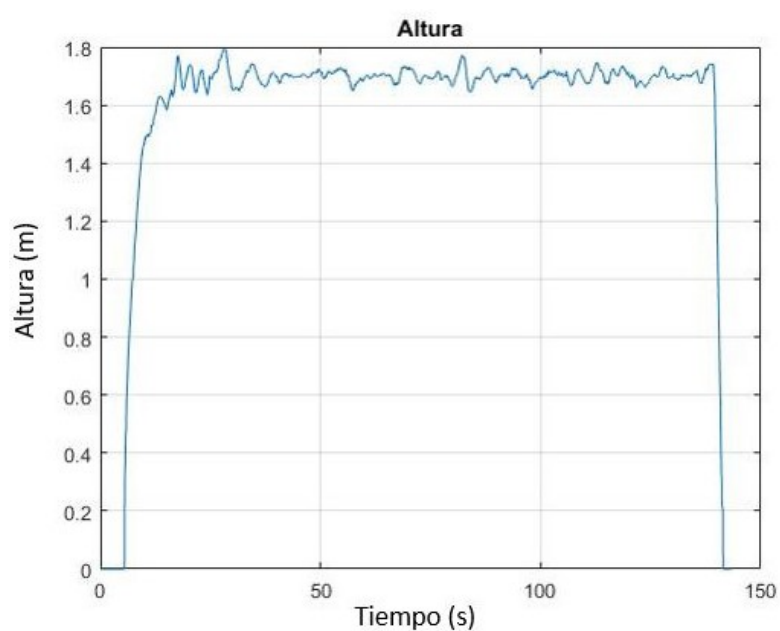
La respuesta simulada que se muestra a continuación en la Figura 60 tiene un tiempo de establecimiento muy pequeño y un error de estado estacionario equivalente a cero al igual que su sobreimpulso.

**Figura 60:** Respuesta simulada con 11 funciones gaussianas



En la Figura 61 se muestra la respuesta de la implementación real que presenta un tiempo de establecimiento bueno a pesar de que tiene oscilaciones estas no superan al 10 % de la señal total teniendo un error pequeño siendo este óptimo y admisible en lo establecido.

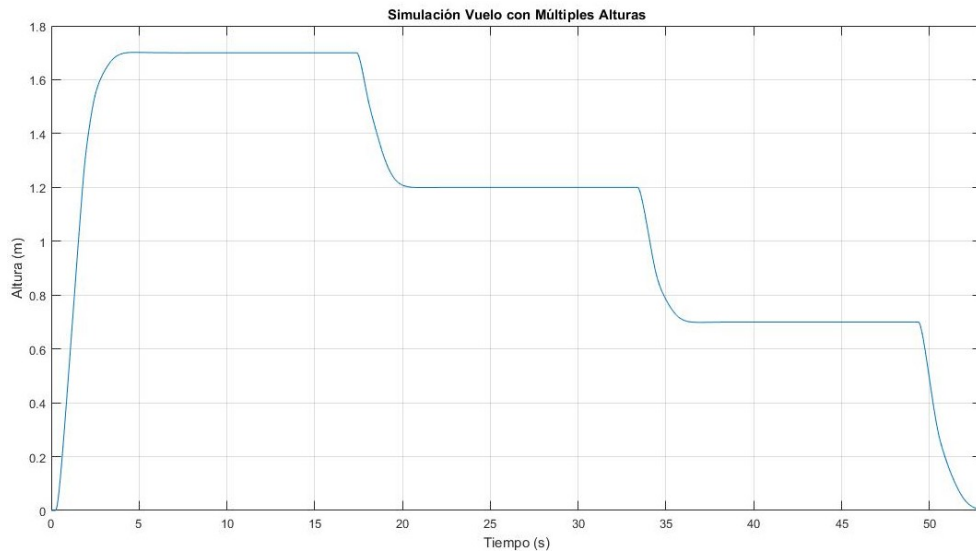
**Figura 61:** Respuesta real con 11 funciones gaussianas



**Segundo tipo de vuelo:** Para la ejecución del vuelo en múltiples puntos de referencia de elevación se utilizan tres alturas (1.7 metros, 1.2 metros y 0.7 metros).

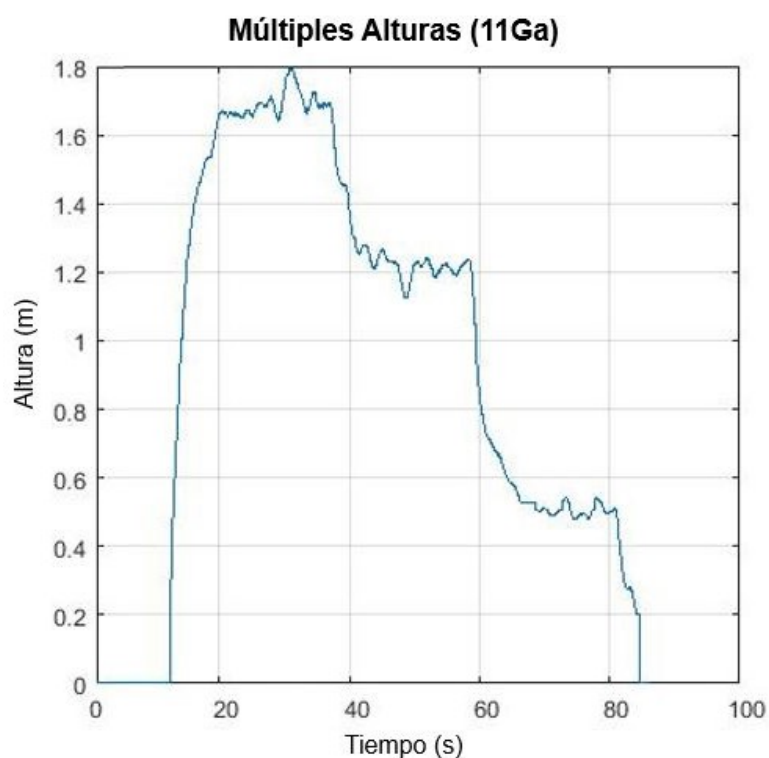
La respuesta simulada como se muestra en la Figura 62 comprueba el buen trabajo del sistema de control bajo parámetros correctos por ello tiene un error igual a cero y un tiempo de establecimiento muy pequeño.

**Figura 62:** Respuesta simulada múltiples puntos 11 funciones gaussianas



La respuesta real que se muestra en la Figura 63 expone que pese a que los puntos de referencia están a diferentes elevaciones el desempeño de este sigue siendo bueno ya que las oscilaciones aunque se presentan en menor medida se hacen notorias durante la mitad del tiempo establecido en cada punto de referencia pero no superan al 10 %.

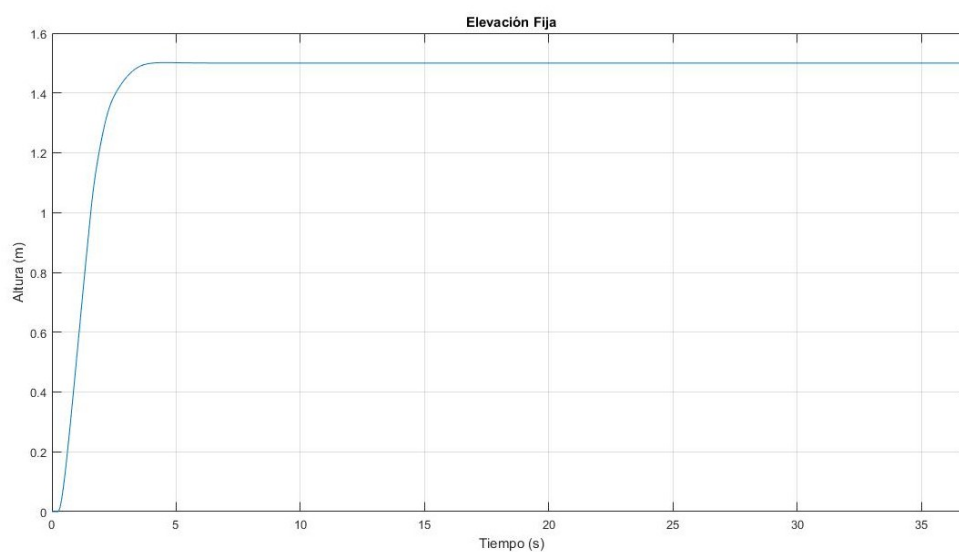
**Figura 63:** Respuesta real multiples puntos 11 funciones gaussianas



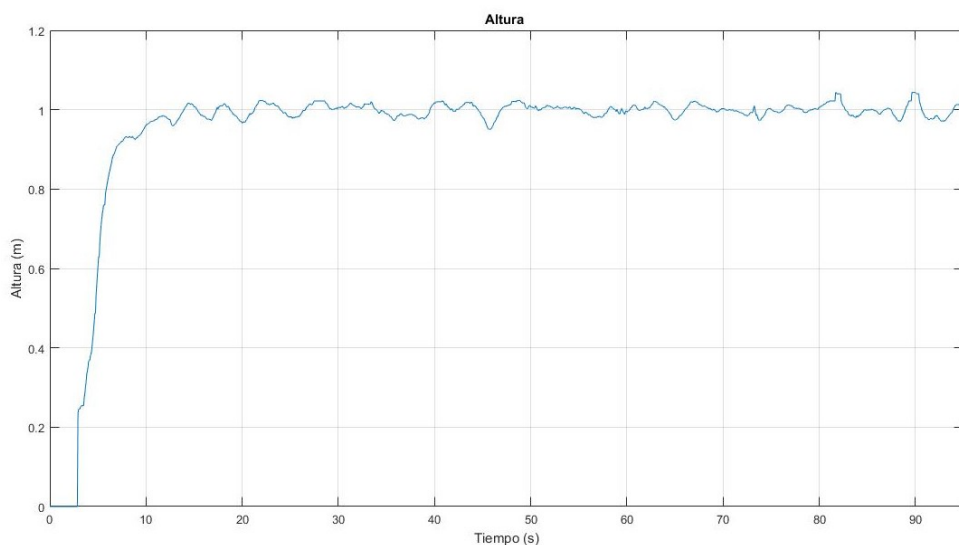
**Tercer tipo de vuelo:** Para la ejecución del seguimiento de ruta se establece que el dron recorra una trayectoria con forma cuadrada de 2 metros por 2 metros a una elevación fija de 1 metro.

La respuesta que se presenta a continuación corresponde a la altura fija que mantuvo el dron durante la ruta en el simulado como se muestra en la Figura 64 se puede observar que el comportamiento realizado por el dron es el esperado con un error igual a cero y en el real como se muestra en la Figura 65 se visualiza que aunque el comportamiento del dron manifiesta oscilaciones su desempeño es muy bueno ya que el error no supera al 10 %.

**Figura 64:** Respuesta simulada de altura 11 funciones gaussianas



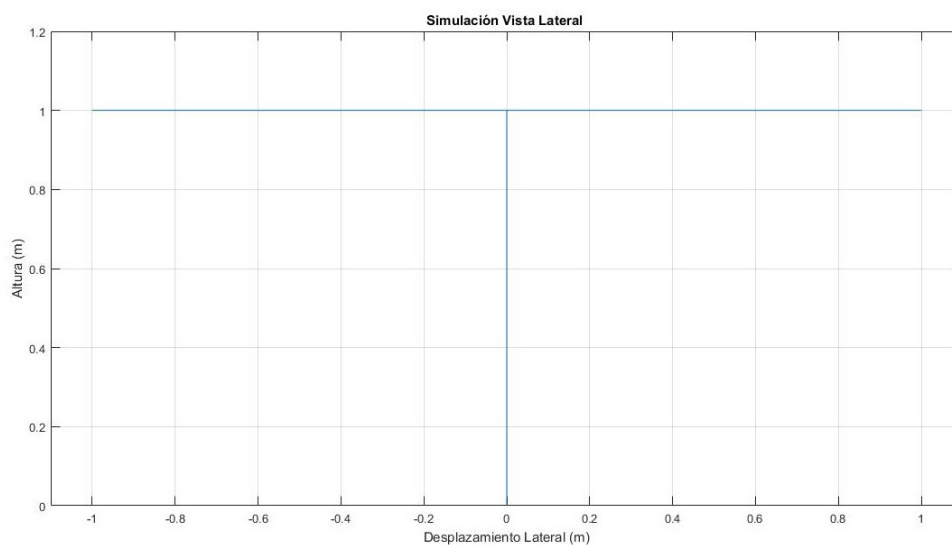
**Figura 65:** Respuesta real de altura 11 funciones gaussianas



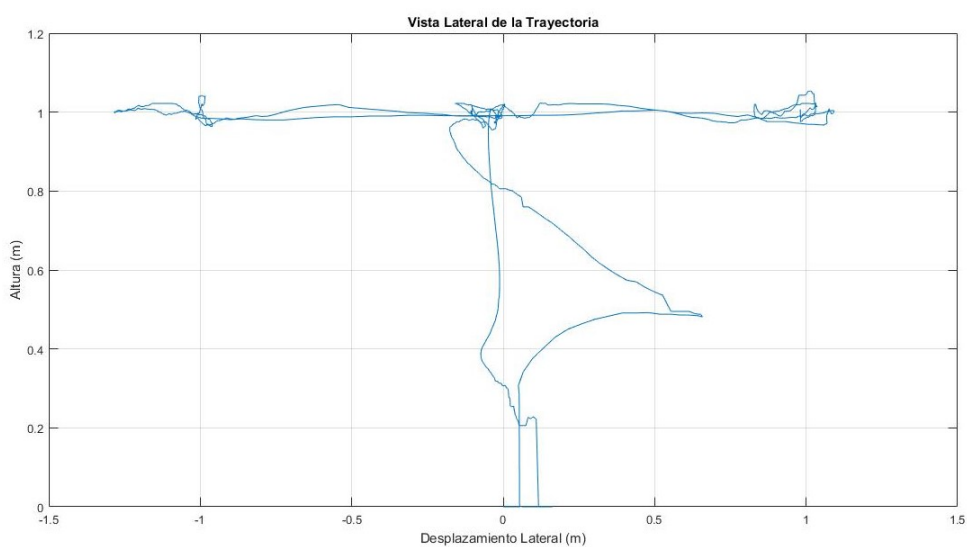
Las respuestas siguientes muestran la altura desde otra perspectiva como se muestra en la Figura 66 en la que se visualiza el recorrido hecho por el dron en el plano correspondiente a las variables X o Y con respecto a su altura(h) fija en el despegue se aprecia un desplazamiento espacial que es igual o menor al 50 % de la señal pero que retorna a su posición espacial inicial al final de su recorrido esto ocurre exclusivamente durante el despegue del dron. Durante el posicionamiento del dron en la altura de referencia inicia su trayectoria como se muestra en la Figura 67 a una elevación fija aunque se observan oscilaciones mínimas estas no representan un inconveniente para el correcto recorrido del dron.



**Figura 66:** Respuesta simulada vista 2D 11 funciones gaussianas

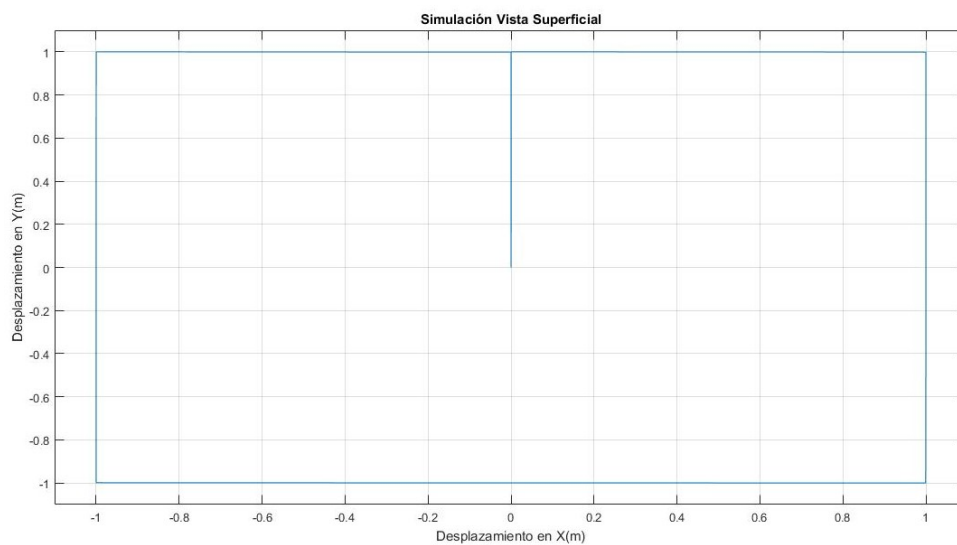


**Figura 67:** Respuesta real vista 2D 11 funciones gaussianas

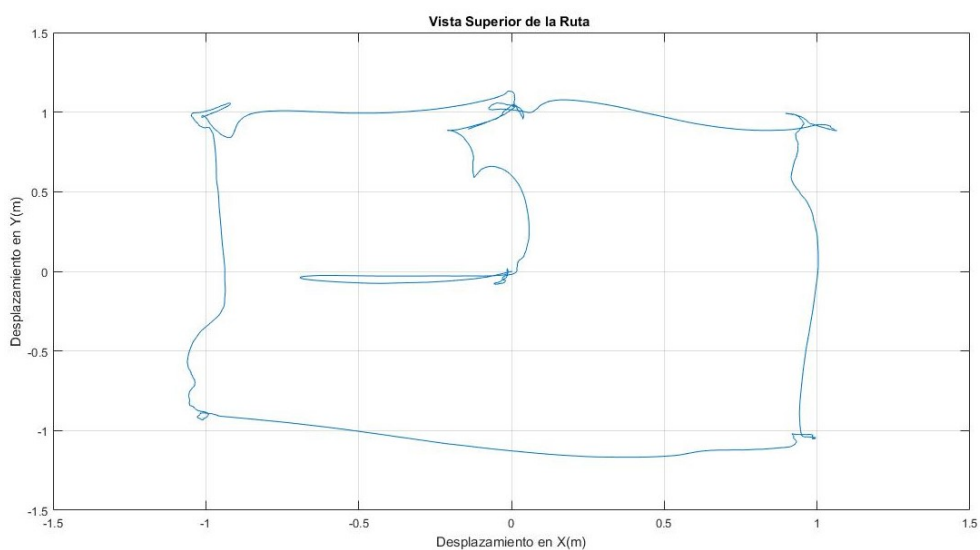


Las siguientes respuestas se observan desde una perspectiva superior en donde es más evidente que la ruta recorrida por el dron es la figura de un cuadrado; la respuesta simulada como se muestra en la Figura 68 es la esperada siguiendo el trazado de la figura en cambio la respuesta real como se muestra en la Figura 69 presenta unas mínimas oscilaciones en el desplazamiento de X y Y debido a los desvíos que se ven claramente están dentro del error aceptable, esto también ocurre debido a que el ambiente no es plenamente controlado.

**Figura 68:** Respuesta simulada vista superior 11 funciones gaussianas

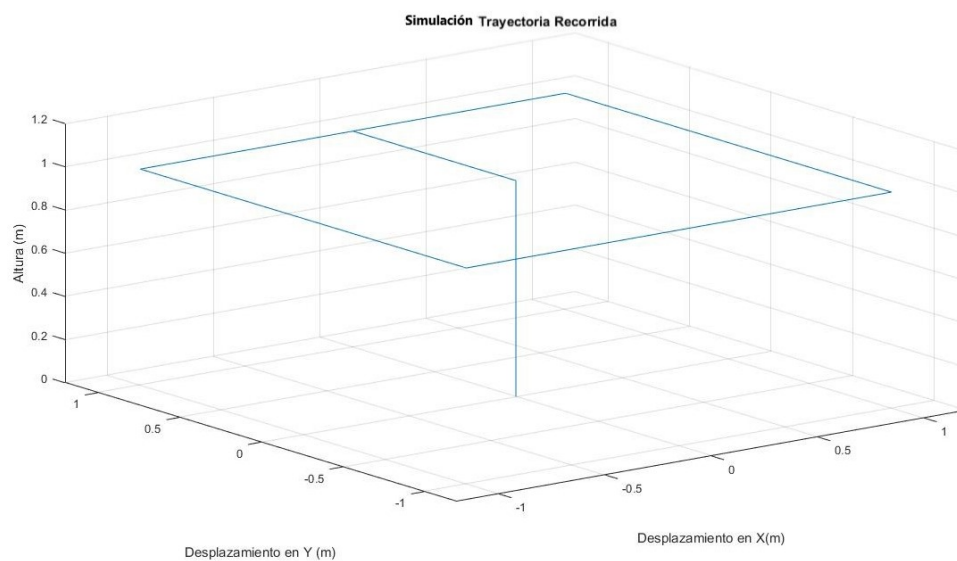


**Figura 69:** Respuesta real vista superior 11 funciones gaussianas

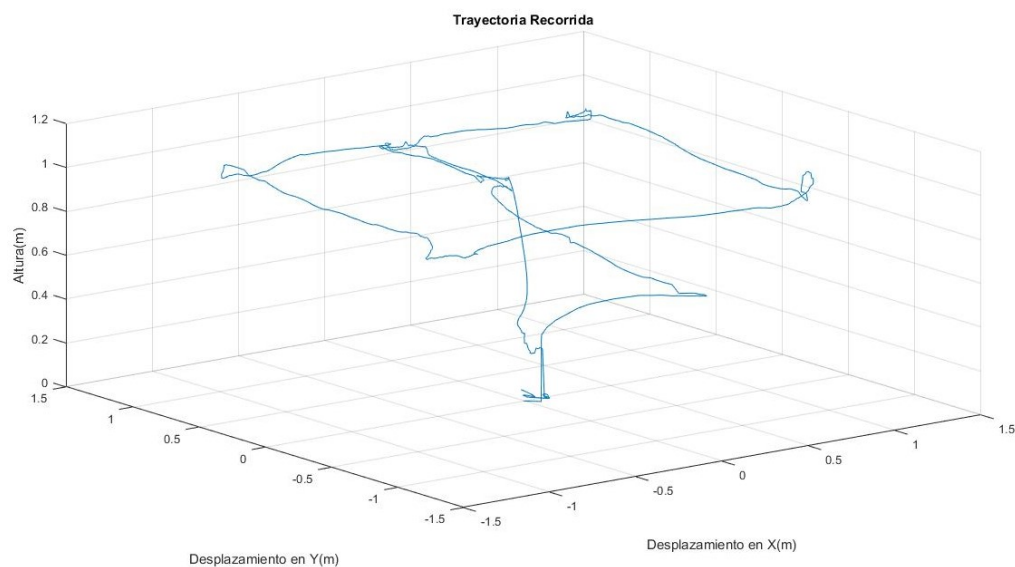


En las últimas respuestas se tiene una visualización general donde se observa la elevación y la trayectoria del dron tanto simulada como se muestra en la Figura 70 como en su implementación real como se muestra en la Figura 71.

**Figura 70:** Respuesta simulada vista 3D 11 funciones gaussianas



**Figura 71:** Respuesta real vista 3D 11 funciones gaussianas



### 5.3. SISTEMA LÓGICO DIFUSO “FIS11TR”

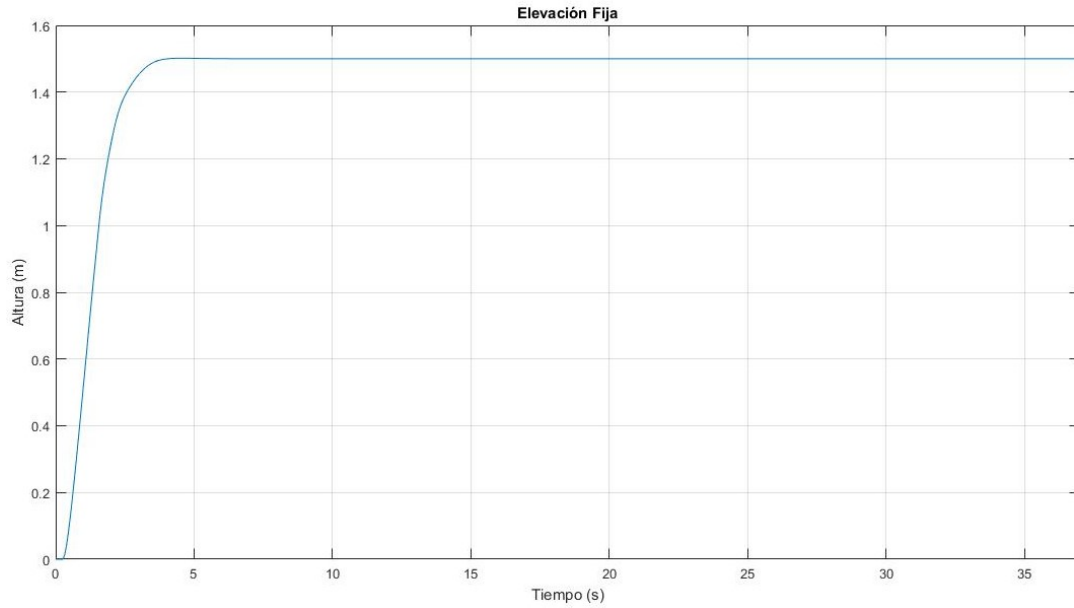
Este sistema de control es el tercero que se diseña, consta de:

- 11 Funciones triangulares para las variables de entrada.
- 11 Funciones triangulares para la variable de salida.
- 121 Reglas.

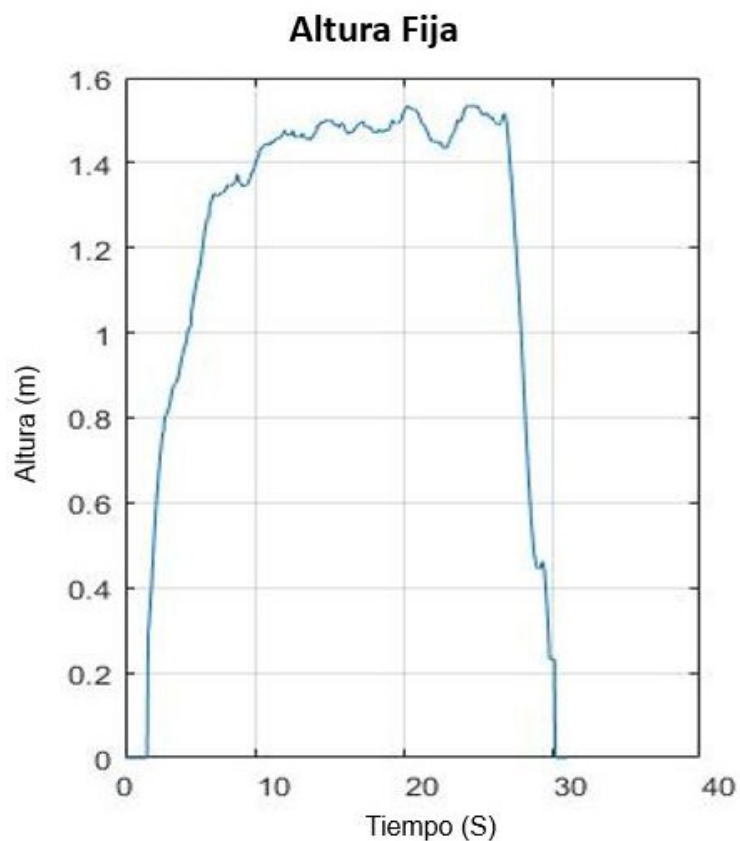
**Primer tipo de vuelo:** Para la ejecución del vuelo a una elevación fija se utiliza un punto de referencia a 1.5 metros.

La respuesta simulada que se evidencia a continuación como se muestra en la Figura 72 tiene un tiempo de establecimiento mínimo y carece de error de estado estacionario de igual forma no hay presencia de sobreimpulso alguno, en la implementación real como se muestra en la Figura 73 el dron alcanza y se mantiene en la altura establecida con un margen de error adecuado, sin embargo presenta gran oscilación en su respuesta y el tiempo que toma en alcanzar la referencia es un poco mayor.

**Figura 72:** Respuesta simulada con 11 funciones triangulares

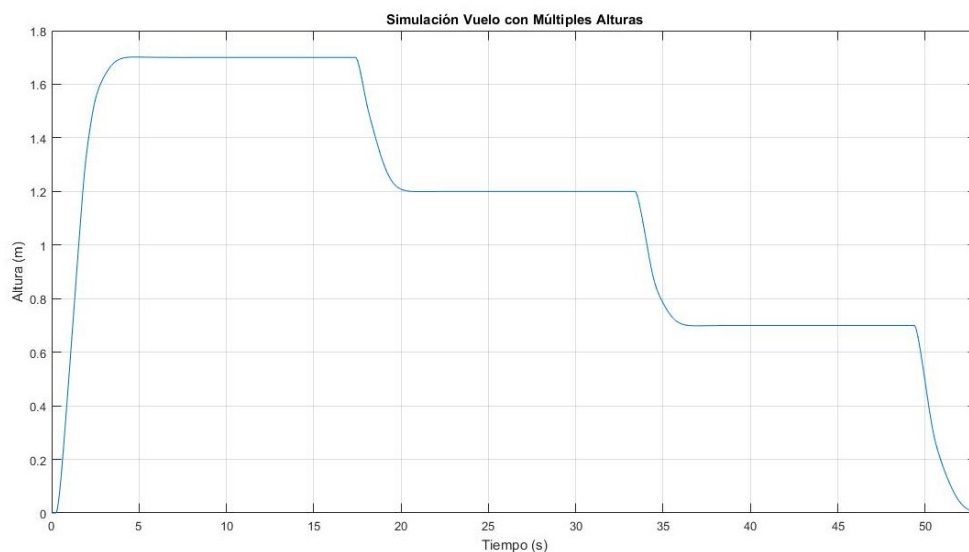


**Figura 73:** Respuesta real con 11 funciones triangulares

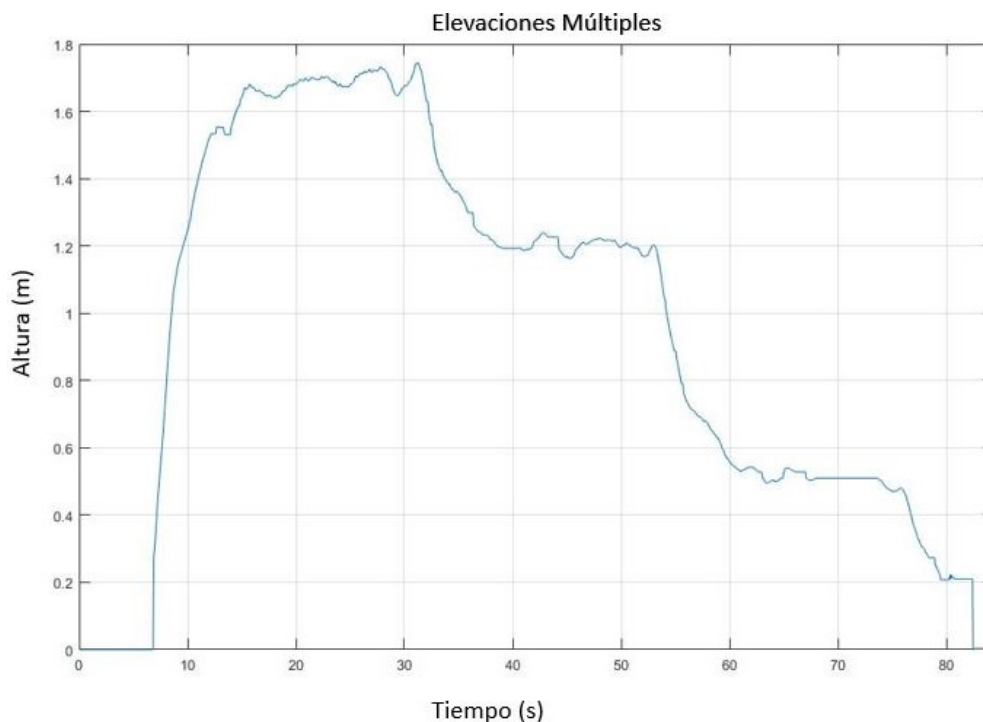


**Segundo tipo de vuelo:** Para la ejecución del vuelo en múltiples puntos de referencia de elevación se utilizan tres alturas (1.7 metros, 1.2 metros y 0.7 metros) de manera simulada como se muestra en la Figura 74 se observa un sobreimpulso que es despreciable en el segundo punto de referencia, de manera general la señal sigue plenamente los puntos de referencia establecidos, en la ejecución del seguimiento de alturas real como se muestra en la Figura 75 se tienen las oscilaciones típicas presentes en el vuelo del vehículo, sin embargo son mínimas al error que se ha venido trabajando.

**Figura 74:** Respuesta simulada multiples puntos 11 funciones triangulares

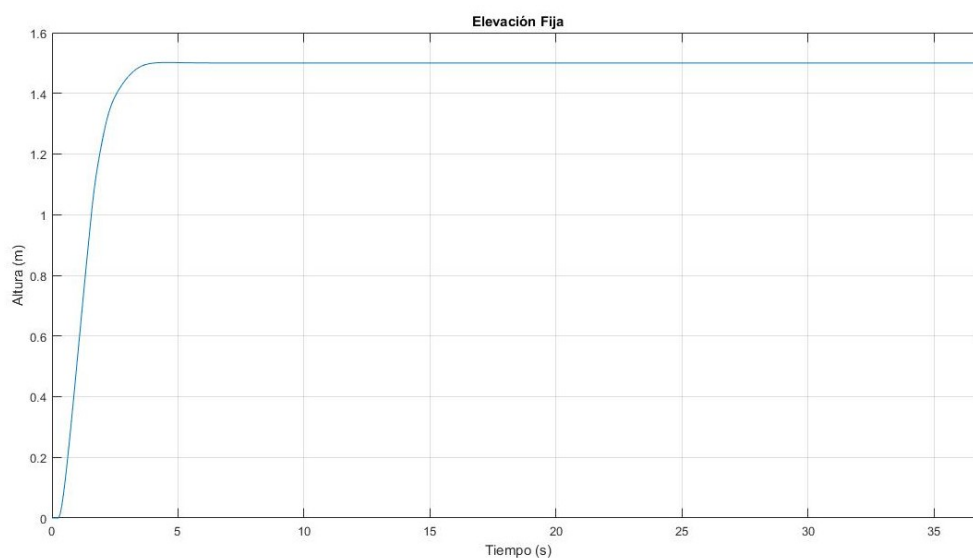


**Figura 75:** Respuesta real multiples puntos 11 funciones triangulares

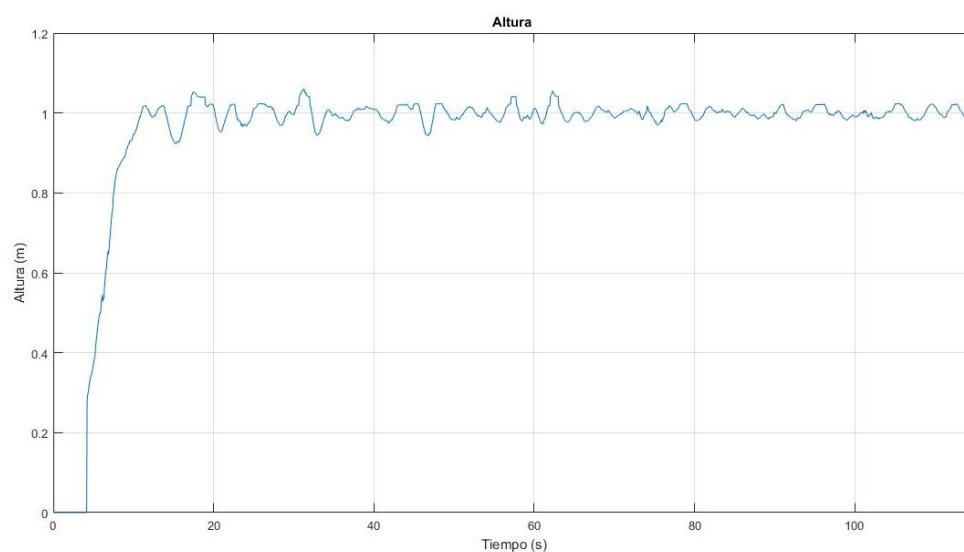


**Tercer tipo de vuelo:** Para la ejecución del seguimiento de ruta se establece que el dron recorra una trayectoria con forma cuadrada de 2 metros por 2 metros a una elevación fija de 1 metro, de forma simulada como se muestra en la Figura 76 la elevación presenta un sobreimpulso mínimo que no es significativo para el desempeño del dron, de manera real como se muestra en la Figura 77 la respuesta aunque es fluctuante las variaciones son sumamente mínimas lo que evidencia la buena acción de control sobre la variable de elevación del dron.

**Figura 76:** Respuesta simulada de altura 11 funciones triangulares

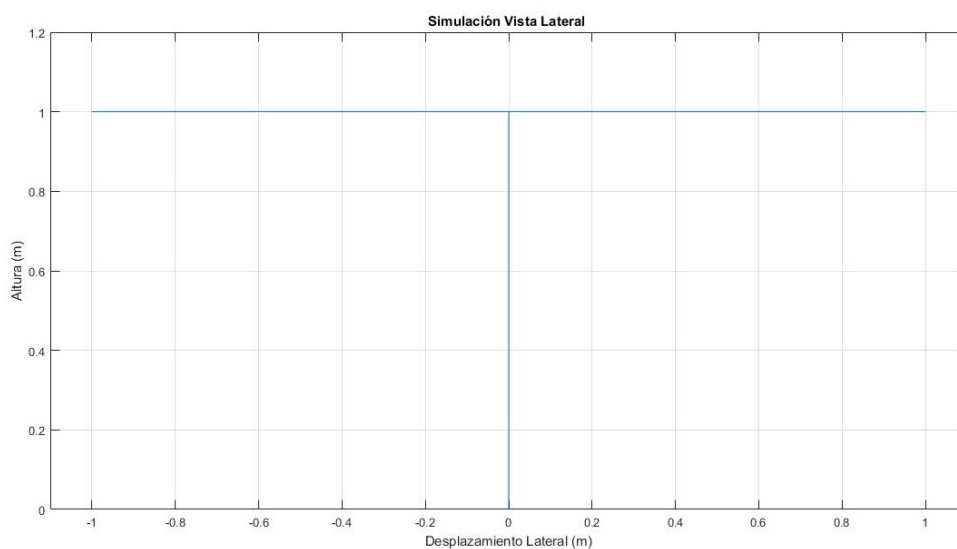


**Figura 77:** Respuesta real de altura 11 funciones triangulares

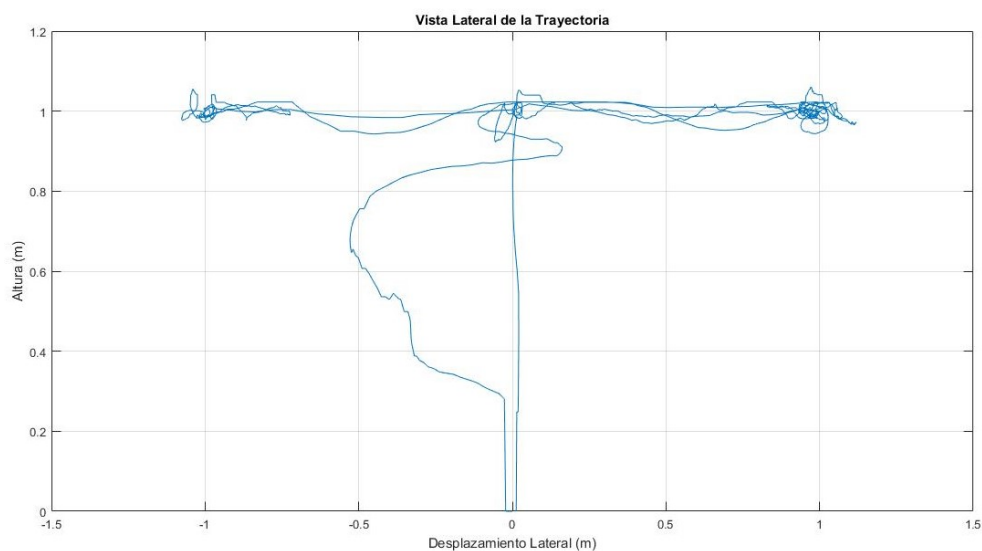


Desde una perspectiva lateral el resultado simulado como se muestra en la Figura 78 presenta un pequeño sobreimpulso en la elevación el cual no es significativo, en la implementación real como se muestra en la Figura 79 nuevamente se observa un comportamiento no controlado en el transcurso de la elevación del dron, sin embargo se posiciona de buena forma en cuanto a altura y espacio horizontal, inicia recorriendo la ruta mantiene la altura con la que se trabaja sin importar los leves vaivenes que tiene, una vez termina la ruta el dron retorna a su punto de origen y aterriza.

**Figura 78:** Respuesta simulada vista 2D 11 funciones triangulares



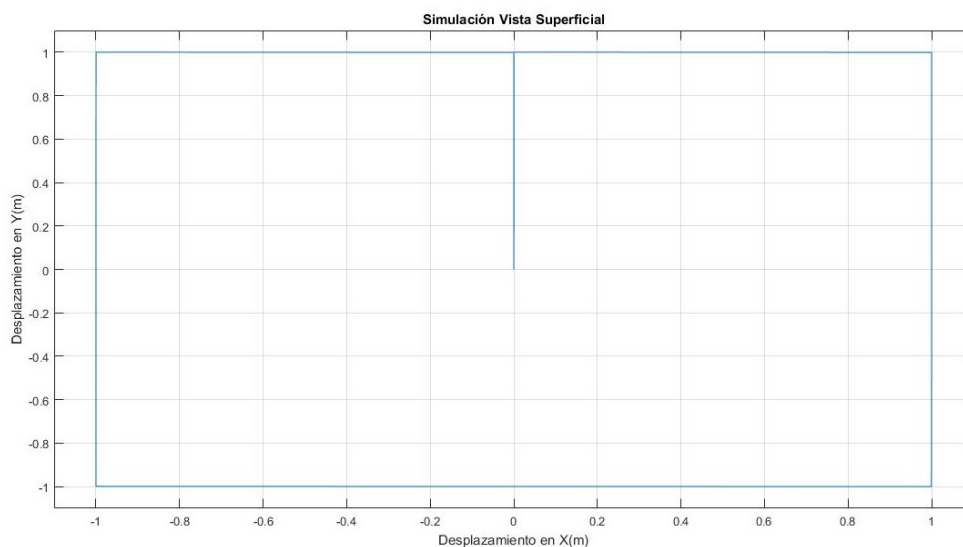
**Figura 79:** Respuesta real vista 2D 11 funciones triangulares



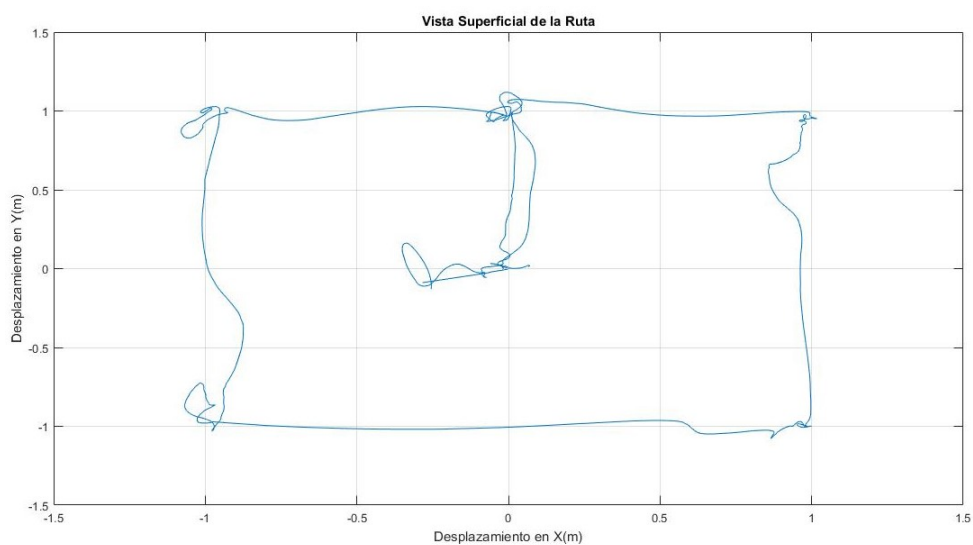
Al trazar la forma de la figura de manera simulada como se muestra en la Figura 80 si se observa con cuidado al final de cada tramo existe un leve bamboleo pero manteniendo la forma de la figura de manera clara, en el desarrollo real como se muestra en la Figura 81 la forma de cuadrado está presente con pérdida de la ruta en algunos tramos pero sin presentar un gran error, en los bordes hay círculos o redundancias de vuelo (sobreimpulso) y se debe al trabajo del sistema de control por mantener al dron fijo en esa posición.



**Figura 80:** Respuesta simulada vista superior 11 funciones triangulares

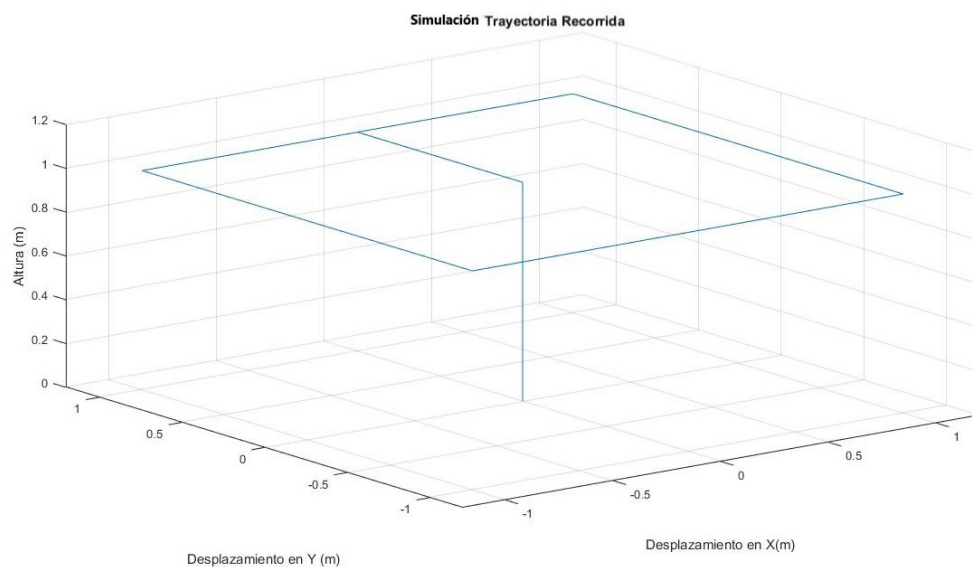


**Figura 81:** Respuesta real vista superior 11 funciones triangulares

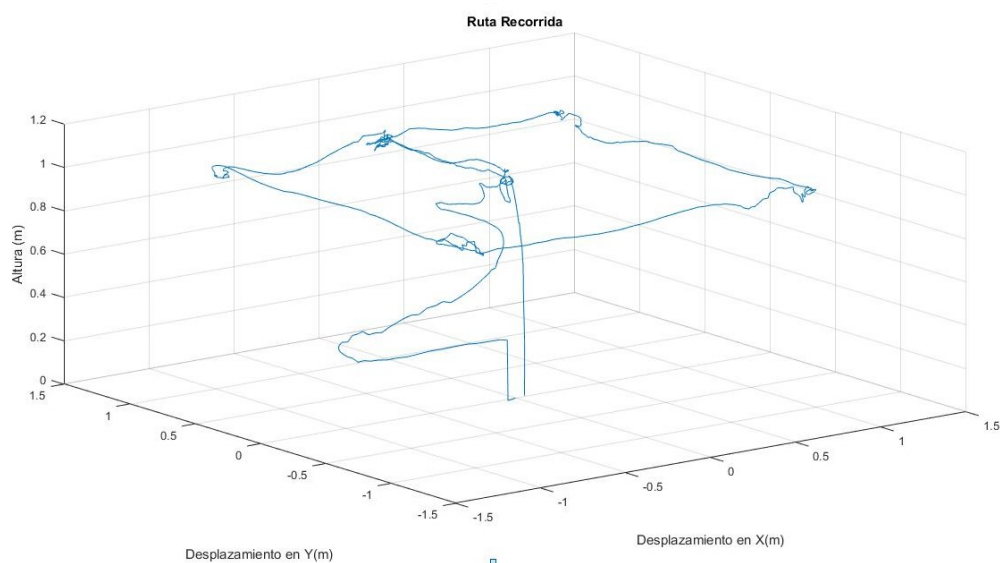


Por último se tiene la vista general de la ruta que siguió el dron desde una perspectiva donde se ve de manera sencilla la altura y la figura formada tanto para el sistema simulado como se muestra en la Figura 82 como probado de manera real como se muestra en la Figura 83.

**Figura 82:** Respuesta simulada vista 3D 11 funciones triangulares



**Figura 83:** Respuesta real vista 3D 11 funciones triangulares



Por último se valida la respuesta del sistema de control frente a otros métodos empleados en proyectos similares, esto permite hacer una confrontación de los parámetros obtenidos dando así una idea del rendimiento del controlador diseñado, sin embargo no es pertinente realizar una comparación total con estos proyectos ya que los parámetros de entrada y el ambiente en el cual fueron desarrollados pueden variar drásticamente entre ellos. Para realizar lo planteado anteriormente se toma como base los resultados del proyecto “Comparación de Técnicas de Control Moderno(Observadores de Estado Vs LQR) Implementadas sobre un AR DRONE 2.0 Parrot” en donde consiguen con un controlador por observador de estados un sobreimpulso de 25.2 %, un tiempo de establecimiento de 2.45s y un error de estado estacionario prácticamente nulo, con el controlador por LQR obtienen un sobreimpulso de 37 %, un tiempo de establecimiento de 1.71s y un error de

estado estacionario que tiende a cero<sup>27</sup>. Con base a los resultados obtenidos en este proyecto se observa que presenta un sobreimpulso menor al 10 %, un tiempo de establecimiento de 5 a 6 segundos aproximadamente y un error de estado estacionario cercano a cero, un común denominador presente en los tres sistemas de control es la oscilación una vez la señal está establecida, no obstante el control difuso al ejecutarse por medio de la toma de decisiones según las reglas y funciones establecidas por el diseñador, permite al sistema una mayor precisión de vuelo produciendo así que la señal presente un sobreimpulso mínimo frente a los otros sistemas de control, pese a esto el tiempo de establecimiento es mucho mayor debido a que el sistema difuso selecciona la salida con base a un análisis de las variables de entrada haciendo que el sistema requiera un poco más de tiempo para llegar al punto de establecimiento, una vez alcanzado el punto de referencia el error de altura es nulo en los tres sistemas de control.

---

<sup>27</sup>Contreras Bernal, Orjuela y Lucia, *Comparación de Técnicas de Control Moderno(Observadores De Estado Vs LQR) Implementadas Sobre un AR Drone 2.0 Parrot.*

## 6. CONCLUSIONES

Se diseñó e implementó de manera exitosa un sistema de control de altura aplicando la lógica difusa para el quadrotor AR. Drone 2.0, se usó lógica difusa ya que no requiere del modelo matemático al cual va a controlar, se diseñó un sistema tipo Mamdani ya que es más aceptado a nivel académico aunque si se quiere que el sistema cuente con una mayor eficiencia computacional se recomienda utilizar un sistema Sugeno. Para emplear los sistemas de inferencia lógica difusa es necesario contar con un amplio conocimiento del sistema a trabajar ya que esto facilita la realización de los parámetros de entrada, los de salida, la cantidad y el tipo de funciones de membresía por variable, el número de reglas a emplear y cómo se relacionan estas últimas entre sí para que la respuesta del sistema sea la esperada.

Los sistemas de inferencia difusa pueden apoyarse en ganancias ubicadas en las entradas y salidas de su sistema, sin embargo una buena determinación de los rangos a emplear pueden sustituir estas ganancias, para las funciones de membresía se usaron de tal forma que estuvieran superpuestas entre sí para disminuir la incertidumbre del sistema, al igual que usar funciones como la triangular o la gaussiana tipo uno o tipo dos también es recomendable ya que los datos están totalmente acotados, sin embargo se considera superior el uso de funciones de membresía tipo gaussianas ya que la sección de mayor relevancia será un conjunto de valores y no un valor único como en las funciones triangulares, en cuanto a la cantidad de funciones de membresía a utilizar si son muy pocas los rangos de valores contenidos deberán ser muy amplios lo que puede desencadenar que el sistema no distinga con claridad entre niveles o desplazamientos de valores cercanos, por otra parte si se usan demasiadas funciones de membresía se soluciona el error anterior ya que cada nivel estará muy bien definido por un rango mínimo pero dado que los sistemas difusos trabajan evaluando por medio de las reglas los parámetros de entradas expresados en rangos de funciones de membresía para así determinar una salida óptima si las funciones son muchas las tomas de decisión necesita más tiempo ya que tendrá que evaluar más casos y el costo computacional será mayor sobre todo si se emplea sistemas Mamdani por eso se aconseja trabajar con el mayor número de funciones de membresía para una mayor precisión pero siempre cuidando que el gasto computacional pueda ser soportado por el equipo donde se está ejecutando el sistema de control.

Para establecer las reglas se busca que las funciones de membresía se relacionen entre ellas abarcando todos los posibles casos teniendo como resultado una respuesta óptima para el sistema por ello entre mayor número de funciones de membresía mayor será el número de reglas que las relacione.

Al momento de controlar el sistema se pretendió solo hacerlo sobre la variable Z ya que es la encargada de la elevación del dron pero al trabajarse se percibe que la elevación se ve fuertemente relacionada con el comportamiento de las otras variables, es decir Pitch, Roll y Yaw ya que la elevación sólo determina la potencia de las hélices del dron por ello se estabilizan los ángulos de Pitch y Roll para que el vehículo mantenga la posición espacial

deseada sin desplazamientos innecesarios y el Yaw para que el dron no rote sobre su mismo eje de manera arbitraria, con los sistemas difusos controlando todas las variables necesarias para el desplazamiento del vehículo se da un paso más allá y se decide no solo hacer control de altura sino además seguimiento autónomo de trayectorias teniendo resultados exitosos.

Matlab no es un software con el que se pueda trabajar en tiempo real fácilmente, por eso se tuvieron ciertas dificultades al momento de presentar los datos reales en la interfaz gráfica pese a ello se logra con éxito una interfaz gráfica que enlaza los modelos de Simulink donde residen los bloques de comunicación y control con una interfaz gráfica sencilla y clara hecha en GUIDE por medio de la cual se pueden ejecutar las funciones básicas para el vuelo del vehículo, la visualización de sus desplazamientos y el almacenamiento de estos datos, el protocolo de comunicación entre el dron y el equipo es de tipo UDP ya que permite el envío de paquetes de datos a través de la red sin que se haya establecido previamente una conexión además brinda una velocidad de transporte más alta. A pesar de ello al ejecutarse toda la aplicación por completo el costo computacional de esta es bastante elevado ya que deberá cargar e iniciar diversos diagramas de bloques que lo componen, iniciar los protocolos de comunicación con el dron tanto para el envío y la recepción de datos lo que genera un gran tiempo de espera al inicio de cada implementación ya sea de forma simulada o real, en esta última el tiempo es mayor ya que su modelo es más complejo estos tiempos adicionales pueden disminuir al trabajar en otro lenguaje de programación o bajo otro tipo de arquitectura.

Se implementó de manera exitosa el sistema de control, la superficie sobre la cual se ubique el dron para la implementación debe ser de nivel uniforme pues el dron determina su altura con base a la medida más inmediata por debajo de él, si hay desniveles u objetos contundentes pueden afectar directamente el desempeño del sistema de control adicionalmente la superficie debe contar con colores, figuras o formas que la identifiquen ya que el dron detecta sus cambios de desplazamiento por medio de la cámara ubicada en la parte inferior, si el suelo es uniforme o las condiciones de luz no son óptimas el dron no percibirá los cambios de posición o al menos no con rapidez lo que hará que el sistema de control no se ejecute de manera eficiente, el dron está hecho de materiales livianos lo que no lo convierte en un sistema muy robusto sobre todo ante el viento según los fabricantes resiste hasta vientos de 15 Km/h pero en la realidad se pierde el control con velocidades muchísimo menores, en algunas ocasiones aunque no se pierda el control si perturba de manera considerable al dron haciendo así que se desvíe de sus puntos de referencia durante el vuelo, las redes Wi-Fi cercanas si son muchas pueden afectar la comunicación con el dron, otro factor muy importante es el nivel de la batería cuando este es inferior al 35 % se ve comprometido todo el sistema por la falta de potencia lo que dificultará en gran medida la controlabilidad del vehículo.

Se logra el registro y la presentación de los niveles de altura recorridos por el quadrotor en tiempo real por medio de una interfaz gráfica realizada en MATLAB la cual al permitir almacenar los datos da la facilidad al usuario para manipularlos o analizarlos posteriormente, de los datos obtenidos de las pruebas hechas se tiene que al utilizarse mayor número de funciones de membresía la respuesta del sistema mejora para este caso particular los datos obtenidos de los vuelos que se implementan en el sistema de control difuso con 11 funciones de membresía son mejor que los obtenidos con el sistema difuso

que solo posee 9 funciones ya que el rango de la variable a tratar se divide en una mayor cantidad de espacios lo que hace que el sistema sea más preciso, sin embargo el sistema difuso con 9 reglas no tuvo un mal desempeño, ahora bien el sistema cuyas 11 funciones de entrada son del tipo gaussiano tuvo un mejor resultado ya que las oscilaciones presentes fueron un poco menores al igual el error en estado estacionario fue un poco menor, de manera general es pertinente mencionar que la actuación de los tres controladores difusos fue el esperado ya que sus respuestas no presentaban un error mayor al 10 % y su tiempo de establecimiento no era muy alto.

Por último este proyecto busca generar las bases para en un futuro realizar el trazado de rutas y posteriormente desarrollar las múltiples aplicaciones que puede llevar a cabo un dron al ser una herramienta versátil, novedosa y amigable con el ambiente generando así un notable impacto en el área que se le de uso. Aunque en un principio represente una gran inversión debido a su costo elevado pero que a mediano plazo puede ser la mejor opción, dándole una o varias aplicaciones adecuadas a dicho vehículo. Al ser un dispositivo tan comercial es accesible a cualquier persona que de acuerdo a las modificaciones necesarias al proyecto puede hacer el mejor uso de este, siguiendo siempre las recomendaciones. En la parte académica se puede investigar más a profundidad sobre métodos más eficientes para el control de sistemas complejos recurriendo a este proyecto como punto de partida para el inicio de nuevos proyectos y su respectiva comparación aumentando así la investigación tanto en la universidad como en la región en este campo tan amplio.

## 7. RECOMENDACIONES

Revisar el nivel de la batería durante la implementación ya que si este baja después de un punto crítico que no necesariamente es la descarga total, comprometerá el control y vuelo del dron lo que provoca pérdida del manejo del vehículo, el no seguimiento de los puntos de referencia o en el peor de los casos un choque contra los elementos circundantes o una caída violenta.

Tener muy presente la capacidad computacional del equipo donde se vaya a realizar la implementación del sistema de control, utilizar siempre la mayor cantidad de funciones de membresía en las variables de entrada y salida, experimentar con un sistema de inferencia lógica Sugeno con el fin de una posible disminución en el tiempo de procesamiento además un equipo con los recursos necesarios disminuirá los tiempos de enlace de los protocolos de comunicación entre el dron y el equipo.

Arriesgarse a llevar esta lógica a lenguajes de programación diferentes donde el rendimiento puede mejorar, se puede añadir nuevas funciones para lograr un control y una autonomía de vuelo mayor.

Trabajar en ambientes especializados o libres en lo posible de corrientes de viento y con una buena iluminación para que el desempeño del sistema del control sea el mayor.

Aventurarse a realizar control de seguimiento de trayectoria directamente sin quedarse solo en el control de altura ya que para estabilizar la altura es necesario controlar todas las variables requeridas para el seguimiento de ruta además el proyecto se torna mas interesante dada a las posibles aplicaciones en las que puede desencadenar.

## BIBLIOGRAFÍA

- [1] A y col. *Vehículos aéreos no tripulados para uso civil. Tecnología y aplicaciones*. Universidad politécnica de Madrid, Madrid, 2016.
- [2] A. Murcia A.F. Lasso. *Diseño e implementación de un sistema de guía inercial para el control de vuelo autónomo en un UAV*. Universidad Surcolombiana, 2012.
- [3] Jorge Cantero Guerrero. *Diseño y fabricación de un vehículo aéreo no tripulado*. Tesis de grado, Universitat Politècnica de Catalunya, 2015.
- [4] Fabio Nelson Castañeda García, Ramiro Andrés Henao y Fabio Alejandro Valencia. *Diseño e implementación del sistema de control de vuelo de un UAV*. Universidad Católica de Manizales, 2016.
- [5] Carlos Andrés Contreras Bernal, Ceferino Orjuela, Nataly Ximena Lucia y col. *Comparación de Técnicas de Control Moderno(Observadores De Estado Vs LQR) Implementadas Sobre un AR Drone 2.0 Parrot*. 2016.
- [6] Sergio García García. *Diseño y construcción de magnetómetro triaxial para análisis y experimentación de aislamientos magnéticos*. 2013.
- [7] Z Nelson Gatica, P Carlos Muñoz y A Patricio Sellado. *Real fuzzy PID control of the UAV AR.Drone 2.0 for hovering under disturbances in known environments*. IEEE, 2017.
- [8] MATLAB User's Guide. *The mathworks*. Inc., Natick, MA, 1998.
- [9] C. Hernández. *PLATAFORMA DE COMUNICACIÓN ENTRE LIVE ROBOT PROGRAMMING Y EL ROBOT AR.DRONE 2.0*. Tesis de grado, Universidad de Chile, Santiago de Chile., 2016.
- [10] Alvaro J Herrera Tavera. *Diseño e implementación del sistema de vuelo autónomo de un vehículo aéreo no tripulado para el reconocimiento en zonas hostiles azotadas por el narcoterrorismo en la Amazonía peruana*. Universidad Ricardo Palma, 2015.
- [11] René Jiménez Madrigal. *Diseño de un controlador lógico difuso aplicado al control de posición de un servomotor de CD usando un algoritmo genético*. 2007.
- [12] MATLAB MathWorks. *the Mathworks*. Inc., Natick, MA, 1992.
- [13] Parallax. *Ping ultrasonic distance sensor (# 28015)*.
- [14] AR Parrot. *Drone 2.0*. Especificaciones Técnicas".[Online] Disponible en: <http://ardrone-2.es/especificaciones-ar-drone-2>, 2012.
- [15] Javier Penalva. *Así es el dron repartidor de Amazon: todavía poco más que humo que promete entregar paquetes en media hora*. [Online] Disponible en: <https://www.xataka.com/drones/>, 2015.
- [16] Emili Rodríguez. *Google con globos y Facebook con drones ofreceran internet gratis para todos*. [Online] Disponible en: <http://www.coaching-tecnologico.com/>, 2015.
- [17] Sandipan Saha y Sabiha Wadoo. *Linear optimal control of a parrot AR.Drone 2.0*. IEEE, 2017.

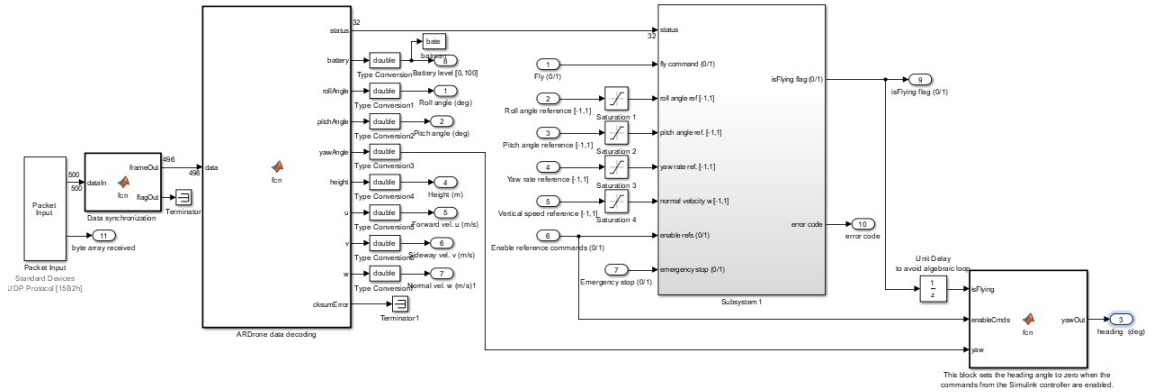


- [18] J Santiaguillo-Salinas y E Aranda-Bricaire. *Seguimiento de Trayectorias para un Helicóptero de 4 Rotores AR. Drone 2.0 Utilizando ROS*. Departamento de Ingeniería Eléctrica, Sección Mecatrónica, 2014.
- [19] Arturo Alonso Casanova De Los Santos. *Control difuso del quadrotor AR.Drone 2.0. para el seguimiento autónomo de trayectorias*. Huajuapán de león, Oaxaca, México., 2015.
- [20] Fendy Santoso y col. *Robust Hybrid Nonlinear Control Systems for the Dynamics of a Quadcopter Drone*. IEEE, 2018.
- [21] Data Sheet. *MPU-6000 and MPU-6050 Register Map and Description*.
- [22] Data Sheet. *MPU-9150 Register Map and Description*.
- [23] Mathworks Simulink. *Simulation and Model-Based Design*. version, 2005.
- [24] Introducing SolidWorks. *Solidworks corporation*. Concord, MA, 2002.
- [25] Brian Mauricio Lara Sosa y col. *Diseño e implementación de un sistema de control de vuelo para un vehículo aéreo no tripulado tipo cuadricóptero*. IEEE, 2017.
- [26] Bernardino Vargas Mollinedo. *Diseño de un sistema de control multivariable para un drone*. Universidad Nacional del Altiplano, 2017.
- [27] Autores Varios. *Impacto y uso de drones para generar cartografía*. [Online] Disponible en: <http://www.milenio.com/opinion/varios-autores/ciencia-tecnologia/>, 2016.

# ANEXOS

## Anexo A. Diagrama interno del bloque AR. Drone Wi-Fi Block

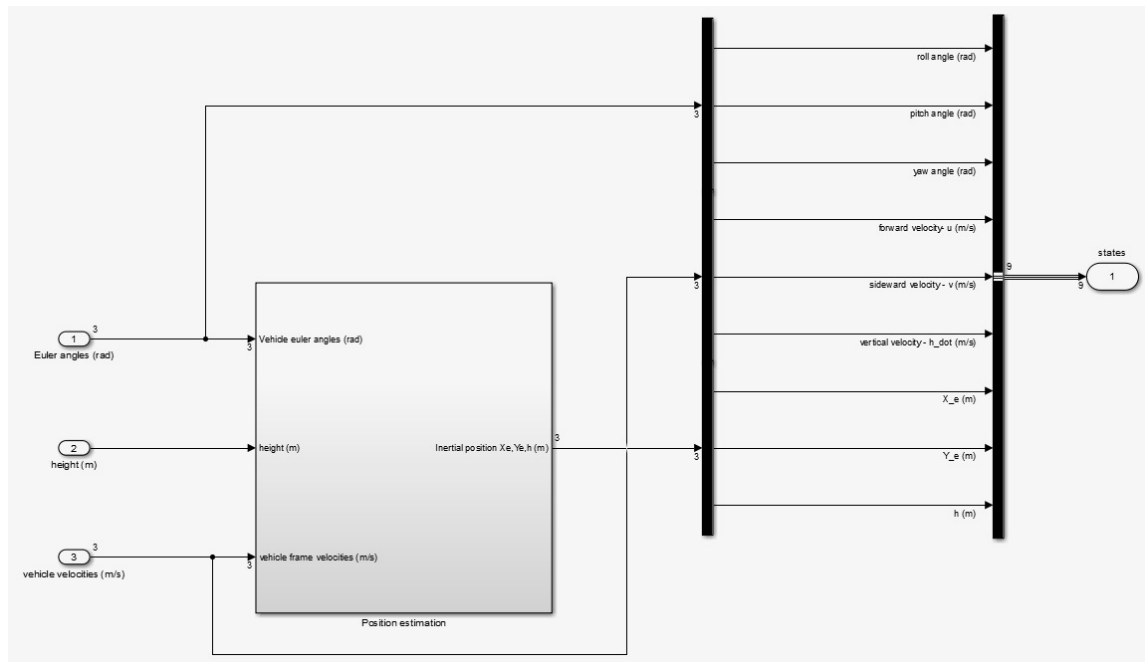
Figura 84: Diagrama interno del bloque AR. Drone Wi-Fi Block



Fuente: Autores

## Anexo B. Diagrama interno del bloque Position Estimation

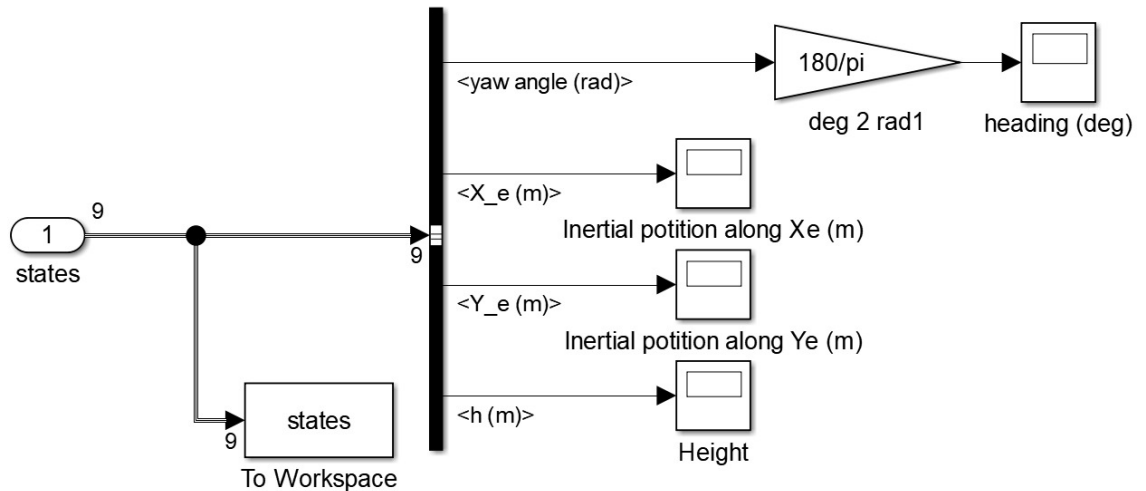
Figura 85: Diagrama interno del bloque Position Estimation



Fuente: Autores

### Anexo C. Diagrama interno del bloque Visualization of block of Drone states

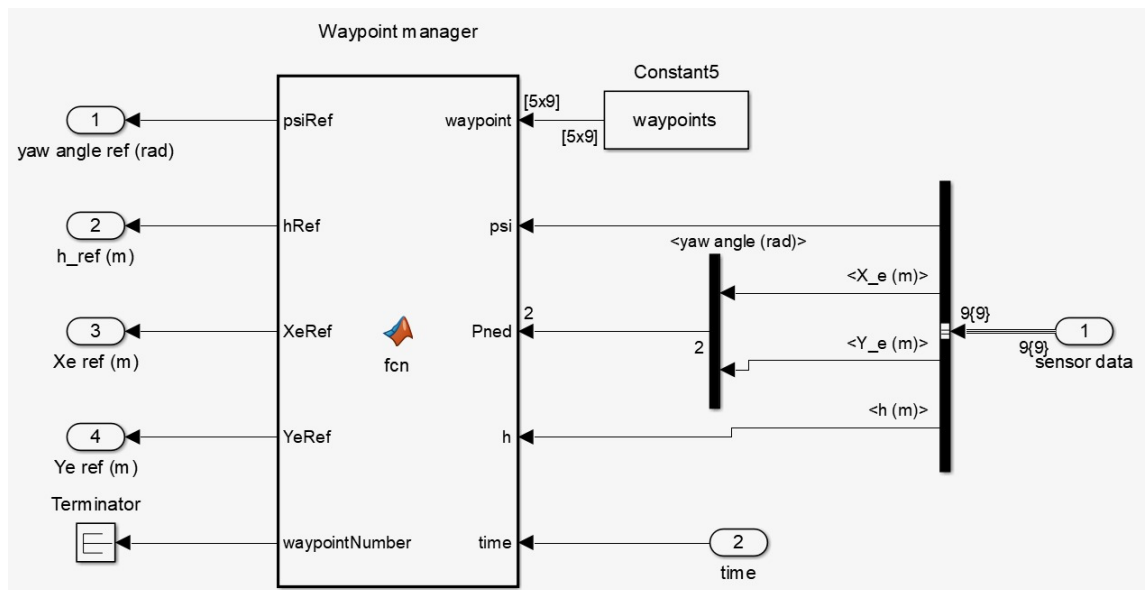
Figura 86: Diagrama interno del bloque Visualization of block of Drone states



Fuente: Autores

### Anexo D. Diagrama interno del bloque Guidance logic

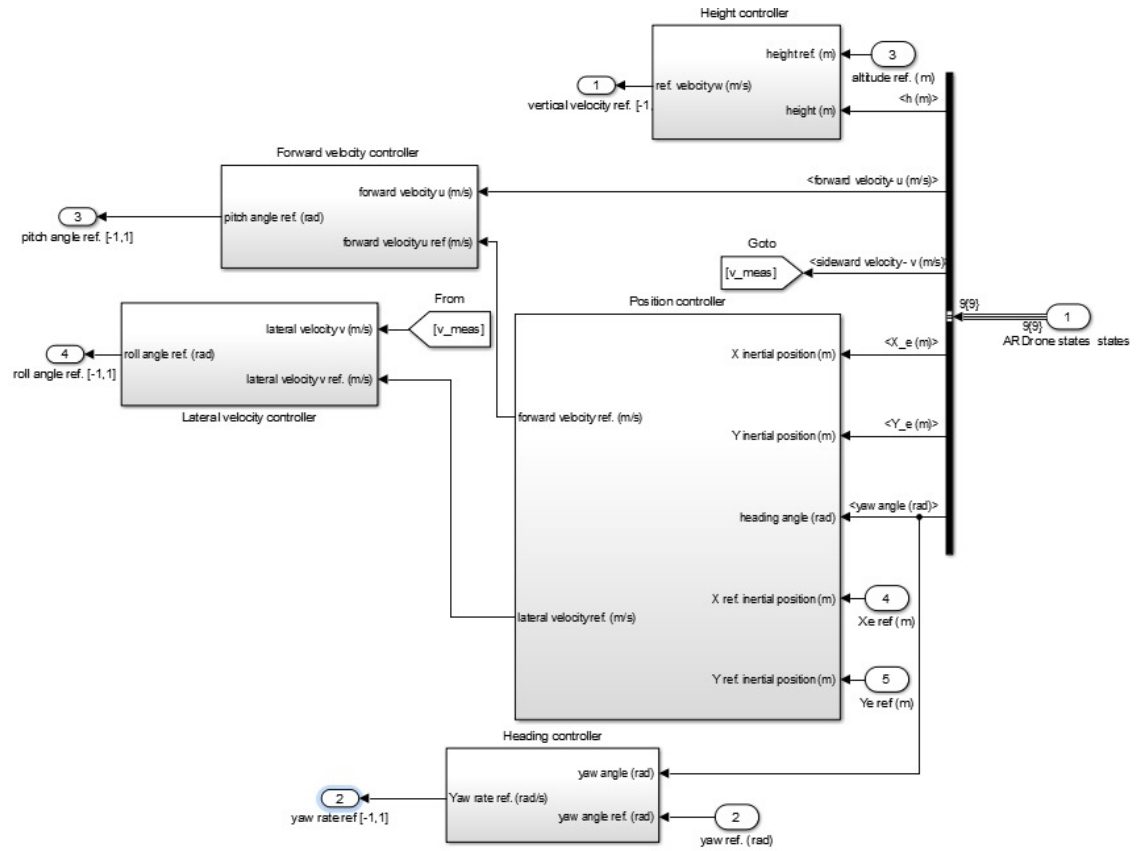
Figura 87: Diagrama interno del bloque Guidance logic



Fuente: Autores

## Anexo E. Diagrama interno del bloque Baseline controller

Figura 88: Diagrama interno del bloque Baseline controller



Fuente: Autores