



EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLÓGICO
NACIONAL DE MÉXICO

Tecnológico Nacional de México

Centro Nacional de Investigación
y Desarrollo Tecnológico

Tesis de Maestría

**Análisis y Diseño de un Método de Control Cooperativo de
Sistemas Autónomos Bajo Degradación de Componentes –
Aplicación a una Flota de Drones en Presencia de
Degradación de batería.**

presentada por

ING. Isay Obet Huicochea Rios

como requisito para la obtención del grado de
Maestro en Ciencias de la Electrónica

Director de tesis

Dr. Manuel Adam Medina

Codirector de tesis

Dr. Carlos Daniel García Beltrán

Cuernavaca, Morelos, México. Enero de 2024.

Cuernavaca, Mor.,

13/diciembre/2023

No. de Oficio:

DIE/0153/2023

Asunto:

Aceptación de documentos de tesis

DR. CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO
PRESENTE

Por este conducto, los integrantes de Comité Tutorial del **C. Isay Obet Huicochea Ríos**, con número de control **M21CE081** de la Maestría en Ciencias en Ingeniería Electrónica, le informamos que hemos revisado el trabajo de tesis profesional titulado “**Análisis y Diseño de un Método de Control Cooperativo de Sistemas Autónomos Bajo Degradación de Componentes-Aplicación a una Flota de Drones en Presencia de Degradación de Batería**”, y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que hemos acordado aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

DIRECTOR DE TESIS



Dr. Manuel Adam Medina
Doctor en Ciencias en Ingeniería Electrónica
Cédula profesional 5992003

CODIRECTOR DE TESIS



Dr. Carlos Daniel García Beltrán
Doctor en Ciencias en Ingeniería Electrónica
Cédula profesional 8699605

REVISOR 1



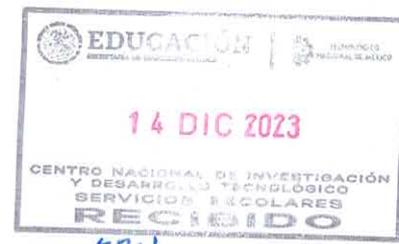
Dr. Enrique Quintero Márquez Márquez
Doctor en Ciencias en Ingeniería Química
Cédula profesional 10259407

REVISOR 2º



Dr. Víctor Manuel Alvarado Martínez
Doctor en Ciencias en Ingeniería Electrónica
Cédula profesional 8031070

C.p. M.T.I. María Elena Gómez Torres, Jefa del Departamento de Servicios Escolares
Estudiante
JGM/kmqh





Cuernavaca, Mor.,
No. De Oficio:
Asunto:

14/diciembre/2023
SAC/208/2023
Autorización de
impresión de tesis

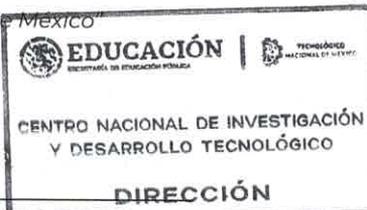
ISAY OBET HUICOHEA RÍOS
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS
EN INGENIERÍA ELECTRÓNICA
P R E S E N T E

Por este conducto, tengo el agrado de comunicarle que el Comité Tutorial asignado a su trabajo de tesis titulado **“Análisis y Diseño de un Método de Control Cooperativo de Sistemas Autónomos Bajo Degradación de Componentes-Aplicación a una Flota de Drones en Presencia de Degradación de Batería”**, ha informado a esta Subdirección Académica, que están de acuerdo con el trabajo presentado. Por lo anterior, se le autoriza a que proceda con la impresión definitiva de su trabajo de tesis.

Esperando que el logro del mismo sea acorde con sus aspiraciones profesionales, reciba un cordial saludo.

ATENTAMENTE

Excelencia en Educación Tecnológica®
“Conocimiento y tecnología al servicio de México”



CARLOS MANUEL ASTORGA ZARAGOZA
SUBDIRECTOR ACADÉMICO

C. c. p. Departamento de Ingeniería Electrónica
Departamento de Servicios Escolares

CMAZ/lmz



Dedicatoria

A mis padres, fuente inagotable de amor y sabiduría, quienes con su inquebrantable apoyo y sacrificio me han brindado las alas para perseguir mis sueños. Cada logro es el reflejo de sus valores y enseñanzas que han iluminado mi camino. A mis hermanos, por su inspiración y motivación constante. Nuestra unión familiar ha sido el motor que impulsa mis metas y ambiciones.

A mis amigos, verdaderos compañeros de este viaje, cuyas risas y palabras de aliento han enriquecido cada etapa. Gracias por compartir este camino y por ser mi red de apoyo incondicional.

A mis profesores y mentores, quienes con su dedicación y conocimiento han dejado una marca imborrable en mi formación académica. Agradezco su guía y enseñanzas que han modelado mi perspectiva y comprensión del mundo.

A mis colegas de estudios, con quienes he compartido desafíos y triunfos, construyendo lazos que van más allá de las aulas. Juntos hemos crecido y aprendido, formando parte de una comunidad académica invaluable.

A todos aquellos que, de alguna manera, han contribuido a este logro, mi más profundo agradecimiento. Este trabajo no solo es el resultado de esfuerzos individuales, sino de una red de personas cuyo apoyo ha sido esencial.

Esta tesis de maestría está dedicada a cada uno de ustedes, con gratitud y cariño.

Agradecimientos

Me gustaría expresar mi más profundo agradecimiento a todas las personas que han sido parte fundamental de este viaje académico y que han contribuido de manera significativa al desarrollo y conclusión de esta tesis. Sus esfuerzos y apoyo han dejado una huella imborrable en este trabajo. Mi agradecimiento se extiende a todos los profesores del Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), cuyas enseñanzas han sido una fuente constante de inspiración y conocimiento. La calidad de la educación que he recibido aquí ha dejado una marca indeleble en mi formación académica. Mención especial al Consejo Nacional de Humanidades, Ciencias y Tecnología (CONAHCYT) por proporcionar el financiamiento para llevar a cabo este proyecto.

Quiero agradecer especialmente a mi director de tesis, el Dr. Manuel Adam Medina, por su inestimable orientación, apoyo y dedicación a lo largo de este proyecto. Sus conocimientos y visión crítica han sido cruciales para dar forma a esta investigación. Agradezco también a mi co-director, el Dr. Carlos Daniel García Beltrán, por sus valiosos aportes y orientación, que han mejorado significativamente la calidad de este trabajo. De igual manera agradezco a Jesús Vázquez, Juan Vázquez y Ricardo Schacht por la paciencia y disposición mostrada.

Agradezco a mis compañeros de clase, Alejandro Peña, Jarib Andrade y Felipe Estrada, por el intercambio constante de ideas y experiencias. Sus aportes en las discusiones académicas han sido enriquecedores y han contribuido al desarrollo de este trabajo. Así como también agradezco a Eduardo Sánchez, Omar Ortiz, Fernando Sánchez, Donaciano Flores, Bernardino Rosales, Roberto Alvarado, Juan Flores, Saúl Olivar, Fausto Lorenzo, Ireti Rojas, Jonathan Arenas, Inri Cruz, Oscar Brito, Arturo Espinosa, Ezequiel Domínguez, Sebastián Valle, Dulce Garcilazo, Katya González, Irbin Carrillo y Rodrigo Velázquez; la conexión que hemos construido trasciende lo académico, y estoy agradecido por tenerlos a mi lado. El aprendizaje conjunto ha sido una parte esencial de este viaje, y estoy agradecido por la colaboración y camaradería compartida.

A mi familia, les estoy profundamente agradecido por su amor incondicional, paciencia y comprensión. Su apoyo constante ha sido un pilar fundamental que me ha dado fuerzas para seguir adelante. Este logro es también suyo, y estoy agradecido por el respaldo que siempre me han brindado mis padres, Rosa Osiris Ríos Rodríguez y Obet Huicochea Rodríguez; mis hermanos, Sergio Elier Huicochea Ríos y Yahir Eliezer Huicochea Ríos; a mi abuelo Cándido Ríos Leyva; y a mi tío Armando Huicochea. A aquellos familiares que ya no están físicamente con nosotros, pero cuyo espíritu y amor perduran en nuestros

corazones, les rindo homenaje y eterna gratitud a Maria Antonia Leticia Rodriguez Corces y a Reynalda Rodriguez Rendón. Su legado continúa inspirándome y guiándome en cada paso que doy.

Mi gratitud se extiende a mis amigos, quienes han estado a mi lado brindando apoyo emocional y aliento durante los momentos desafiantes. Sus palabras de ánimo y su amistad han sido un faro constante, recordándome la importancia de equilibrar el trabajo académico con momentos de alegría y camaradería. Extiendo mis agradecimientos a mi novia, Carla Sofía Ayala Pineda; y a mis amigos Jesús Antonio Ramírez, Zeltzin de la Puente, Sofía Mazón, Carlos Málaga, Ricardo Balbuena, Rodrigo Vergara, Vivek Ornelas, Javier Barrios, Uriel García, Sergio Ortega, Zahir Crump, Jesús Daniel Bisogno, Santiago González, Yamile Pedroza, Karla González, Antonio Mena, Pablo Hernández y Camila Huicochea.

Quisiera expresar mi agradecimiento a todas las personas que, de una manera u otra, han contribuido a este proyecto. Cada aporte, ya sea grande o pequeño, ha dejado una marca importante en esta investigación y en el desarrollo de mi formación personal y académica.

En resumen, este logro no es solo mío, sino de un grupo de personas cuyo apoyo y colaboración han sido esenciales. Gracias a todos.

Índice de figuras

1.1.	Sistemas multiagente.	9
1.2.	Red de procesamiento centralizada [1].	10
1.3.	Red de procesamiento descentralizada [1].	10
1.4.	Mecanismo del autómatas creado por Leonardo da Vinci [2].	12
1.5.	Clasificación de vehículos UAV [3]. a) y b) vehículos <i>fixed-wing</i> o ala fija [4]; c) bicóptero [5]; d) tricóptero [6]; e) cuadricóptero [7]; f) pentacóptero [8]; g) hexacóptero [9]; h) octacóptero [10]; i) y j) vehículos <i>flapping wing</i> [11], [12].	13
1.6.	Sistema multiagente con líder seguidor.	20
1.7.	Sistema de propulsión.	20
1.8.	Diagrama de control de formación líder seguidor.	23
1.9.	Diagrama de control de formación líder seguidor con el sistema de propulsión.	23
2.1.	Esquema del doble integrador.	26
2.2.	Entradas del doble integrador.	26
2.3.	Marco de referencia de un UAV cuadricóptero.	27
2.4.	Modelo no lineal de un UAV con el subsistema rotacional controlado.	29
2.5.	Modelo no lineal de un UAV con el subsistema rotacional controlado, incluyendo la dinámica de la batería y de los BLM.	31
2.6.	Circuito equivalente de una batería de polímero de litio ([13]).	32
2.7.	Posiciones de los casos de estudio.	34
2.8.	Velocidades de los casos de estudio.	35
3.1.	Diagrama de bloques del control de formación basado en consenso sin líder seguidor.	37
3.2.	Topología de comunicación de un sistema multiagente.	38
3.3.	Grafo directo.	38
3.4.	Topología de comunicación sin líder seguidor para un sistema de 3 agentes.	39
3.5.	Formación hexagonal de 6 agentes en el marco de referencia xy	41
3.6.	Control de formación basado en consenso sin líder seguidor.	46
3.7.	Error de formación del sistema multiagente.	47
4.1.	Diagrama de bloques del sistema de propulsión de un UAV.	49
4.2.	Control de formación con la dinámica de la batería y los motores.	50

4.3. Esquema de control para el líder seguidor.	51
4.4. Diagrama de bloques de un control PI-D o PID modificado.	51
4.5. Controladores PI-D para el líder virtual.	52
4.6. Comportamiento de las posiciones del líder al seguir una trayectoria. . .	53
4.7. Trayectoria a seguir del líder seguidor en los ejes xyz	54
4.8. Trayectoria recorridas por el líder a lo largo del tiempo de simulación. .	54
4.9. Trayectoria recorrida por el líder seguidor.	55
4.10. Topología de comunicación del sistema multiagente líder seguidor. . . .	60
4.11. Sistema multiagente bajo el control de formación basado en consenso líder seguidor y sus errores.	62
4.12. Trayectoria A del líder.	63
4.13. Trayectoria B del líder.	63
4.14. Seguimiento de trayectoria del sistema multiagente (simulación A). . . .	64
4.15. Seguimiento de trayectoria del sistema multiagente (simulación B). . . .	64
4.16. Estado de carga de los agentes en la simulación A.	65
4.17. Estado de carga de los agentes en la simulación B.	65
4.18. Fuerza de empuje en el eje z de los agentes del sistema.	66
4.19. Errores de formación del sistema multiagente en la simulación A.	66
4.20. Errores de formación del sistema multiagente en la simulación B.	67
4.21. Errores de sincronización del sistema multiagente en la simulación A. . .	68
4.22. Errores de sincronización del sistema multiagente en la simulación B. . .	69
A.1. Diagrama de flujo del caso 1.	81
A.2. Diagrama de flujo del caso 2.	82
A.3. Diagrama de flujo del caso 3.	83
A.4. Diagrama flujo para el control de formación basado en consenso.	84

Índice general

Nomenclatura	1
Resumen	4
Abstract	5
1. Introducción	9
1.1. Antecedentes	11
1.2. Estado del arte	14
1.2.1. Control de formación basado en líder seguidor	14
1.2.2. Sistema de propulsión	16
1.2.3. Control tolerante a fallas en sistemas multiagente	18
1.2.4. Resumen y análisis del estado del arte	19
1.3. Planteamiento del problema	19
1.4. Justificación	21
1.5. Hipótesis	22
1.6. Objetivos	22
1.6.1. Objetivo general	22
1.6.2. Objetivos específicos	22
1.7. Propuesta de solución	23
2. Representaciones de un UAV	25
2.1. Doble integrador	25
2.2. Modelo no lineal de un UAV con la dinámica rotacional controlada	27
2.2.1. Control PI	28
2.3. Modelo no lineal de un UAV con sistema de propulsión y con la dinámica rotacional controlada	31
2.3.1. Leyes de control en términos de las velocidades angulares de los motores	31
2.3.2. Modelo de la batería	32
2.3.3. Modelo de los motores <i>brushless</i>	33
2.4. Discusión de resultados (conclusión del capítulo)	34
3. Control de formación basado en consenso con sistema de propulsión	37
3.1. Teoría de grafos	38

3.1.1. Topología de comunicación del sistema multiagente	39
3.2. Distancia relativa entre agentes	40
3.2.1. Matriz de distancias relativas del sistema multiagente	41
3.3. Protocolo de consenso basado en observador	42
3.3.1. Ganancias del protocolo de consenso	44
3.4. Control interno del sistema multiagente	44
3.5. Discusión de resultados (conclusión del capítulo)	45
4. Control de formación líder seguidor basado en consenso afectado por el sistema de propulsión	49
4.1. Agente líder	50
4.1.1. Control PI-D	51
4.2. Protocolo de consenso	55
4.2.1. Análisis de estabilidad basado en Lyapunov	57
4.3. Condiciones de simulación	60
4.4. Discusión de resultados (conclusión del capítulo)	61
5. Conclusiones	71
5.1. Contribución	72
5.2. Trabajos futuros	72
Referencias	72
A. Diagramas de flujo	81
B. Resumen del estado del arte	85
C. Parámetros para las simulaciones	87
D. Modelos matemáticos utilizados	89
D.1. Modelo de un UAV	89
D.2. Modelo de la batería de un UAV	90
D.3. Modelo de los motores <i>brushless</i>	91
E. Acrónimos	93

Nomenclatura

ref	Vector de referencia del agente líder
e	Vector de error del agente líder
X_l	Vector de estados del agente líder
U_l	Vector de leyes de control del agente líder
\mathcal{Y}_l	Vector de salida del agente líder
x_i	Posición en el eje x del i -ésimo agente
y_i	Posición en el eje y del i -ésimo agente
z_i	Posición en el eje z del i -ésimo agente
\dot{x}_i	Velocidad en el eje x del i -ésimo agente
\dot{y}_i	Velocidad en el eje y del i -ésimo agente
\dot{z}_i	Velocidad en el eje z del i -ésimo agente
\ddot{x}_i	Aceleración en el eje x del i -ésimo agente
\ddot{y}_i	Aceleración en el eje y del i -ésimo agente
\ddot{z}_i	Aceleración en el eje z del i -ésimo agente
ϕ_i	Posición angular con respecto al eje x del i -ésimo agente
θ_i	Posición angular con respecto al eje y del i -ésimo agente
ψ_i	Posición angular con respecto al eje z del i -ésimo agente
$\dot{\phi}_i$	Velocidad angular con respecto al eje x del i -ésimo agente
$\dot{\theta}_i$	Velocidad angular con respecto al eje y del i -ésimo agente
$\dot{\psi}_i$	Velocidad angular con respecto al eje z del i -ésimo agente
$\ddot{\phi}_i$	Aceleración angular con respecto al eje x del i -ésimo agente
$\ddot{\theta}_i$	Aceleración angular con respecto al eje y del i -ésimo agente
$\ddot{\psi}_i$	Aceleración angular con respecto al eje z del i -ésimo agente

$\Sigma_{\mathcal{T}}$	Subsistema traslacional del UAV
$\Sigma_{\mathcal{R}}$	Subsistema rotacional del UAV
X_B	Vector de estados de la batería
X_i	Vector de estados del i-ésimo agente
\bar{U}_i	Vector de leyes de control teóricas del i-ésimo agente
\bar{T}_i	Fuerza de empuje teórica en el eje z del i-ésimo agente
\bar{R}_i, \bar{P}_i y \bar{Y}_i	Pares angulares teóricos del i-ésimo agente
U_i	Vector de leyes de control suministradas al sistema multiagente del i-ésimo agente
T_i	Fuerza de empuje en el eje z del i-ésimo agente
R_i, P_i y Y_i	Pares angulares del i-ésimo agente
$\bar{\Omega}_i$	Vector de velocidades angulares teóricas de los motores del i-ésimo agente
Ω_i	Vector de velocidades angulares del modelo de los motores del i-ésimo agente
\mathcal{Y}_i	Vector de salida del i-ésimo agente
H	Matriz de distancias relativas entre los agentes
h_i	Vector de distancias relativas del i-ésimo agente
\mathcal{A}	Matriz de adyacencias
\mathcal{L}	Matriz Laplaciana
r_i	Vector de estados del protocolo de consenso del i-ésimo agente
u_i	Entradas de control generadas por el protocolo de consenso del i-ésimo agente
A, B y C	Matrices del doble integrador
p	Vector de posiciones del doble integrador
v	Vector de velocidades del doble integrador
u	Vector de entradas suministradas en la comparación de modelos
ϕ_r, θ_r y ψ_r	Ángulos de referencia <i>roll</i> , <i>pitch</i> y <i>yaw</i>
\tilde{A}, \tilde{B} y \tilde{C}	Matrices de la parte rotacional del modelo lineal de un UAV
\bar{T}	Fuerza de empuje teórica en el eje z
\bar{R}, \bar{P} y \bar{Y}	Pares angulares teóricos en los ángulos <i>roll</i> , <i>pitch</i> y <i>yaw</i>
T	Fuerza de empuje suministrada al UAV
R, P y Y	Pares angulares en los ángulos <i>roll</i> , <i>pitch</i> y <i>yaw</i> suministrados al UAV

\bar{U}_ϕ	Vector de leyes de control del subsistema rotacional
Γ	Matriz que define las entradas de control en términos de las velocidades angulares
N_M	Número de motores
$\bar{\Omega}$	Vector de velocidades angulares teóricas de los motores
Ω	Vector de velocidades angulares del modelo de los motores
V_{SoC}	Estado de carga
V_d	Voltaje de descarga
V_{OCV}	Voltaje de circuito abierto
V_{batt}	Voltaje en línea
Dc	Vector de señales PWM
γ_1, γ_2 y γ_3	Coefficientes de la función que calcula las señales PWM
I_{batt}	Corriente consumida por los motores

Resumen

En el presente trabajo de tesis se plantea el análisis de la batería en los sistemas multi-agente al realizar una misión de largo alcance, donde, por medio del análisis realizado es posible abordar una investigación enfocada a la administración del uso de la batería en vehículos aéreos no tripulados.

En este trabajo se analizó el comportamiento de la batería de un vehículo aéreo no tripulado (UAV) en condiciones adversas. Con la finalidad de conjuntar los distintos subsistemas que componen a un UAV, se comparó el comportamiento de distintas representaciones de los UAVs, las cuales son: doble integrador, modelo lineal Newton-Euler controlado y modelo no lineal Newton-Euler controlado. En dicha comparación se ajustan los parámetros relacionados al sistema de propulsión de un UAV, logrando así, encontrar los parámetros aproximados para conjuntar los diferentes sistemas que representan las dinámicas de un UAV.

Posterior a la obtención de valores viables para la simulación de un sistema multiagente, se realizó un control de formación basado en consenso sin líder seguidor, al cual se añadieron las dinámicas del sistema de propulsión. Los resultados obtenidos del control de formación fueron parcialmente satisfactorios, por lo tanto, se diseñó un control de formación basado en consenso con líder seguidor. El control de formación líder seguidor fue diseñado a partir de el análisis de estabilidad de Lyapunov, las ganancias obtenidas al resolver la desigualdad matricial lineal planteada satisfacen el controlador para una flota de 3 UAVs, donde un agente presenta una caída de voltaje.

Abstract

This thesis introduces an analysis of battery performance within multi-agent systems engaged in long-range missions. The research aims to contribute to the investigation of battery usage management in unmanned aerial vehicles (UAVs) by means of a comprehensive analysis.

The study delves into the battery behavior of a UAV under adverse conditions. To integrate the diverse subsystems comprising a UAV, a comparative analysis was conducted involving different UAV representations, including the double integrator, the controlled linear Newton-Euler model, and the controlled nonlinear Newton-Euler model. Throughout this comparison, adjustments to parameters related to the UAV's propulsion system were made, enabling the identification of approximate parameters to unify the various systems representing UAV dynamics.

Following the acquisition of viable values for simulating a multi-agent system, a consensus-based formation control without leader-follower was implemented, incorporating the dynamics of the propulsion system. While the results of this formation control were partially satisfactory, a leader-follower consensus-based formation control was subsequently designed. This leader-follower formation control was developed based on Lyapunov stability analysis, with gains obtained from solving the linear matrix inequality, thereby satisfying the controller for a fleet of 3 UAVs, even in the presence of a voltage drop in one agent.

Capítulo 1

Introducción

El interés por los sistemas multiagente se ha expandido por su utilidad para completar tareas, las cuales son originalmente realizadas por un sólo agente [14]. Redes de sensores, sistemas de energía eléctrica y formación de vehículos son algunas de las aplicaciones realizadas utilizando el enfoque de los sistemas multiagente (figura 1.1). El control de los sistemas multiagente se realiza mediante el consenso entre agentes, el cual consiste en alcanzar un objetivo en común, mediante el intercambio de información de los estados entre los agentes del sistema [15]; así mismo, los controladores pueden ser centralizados y descentralizados.

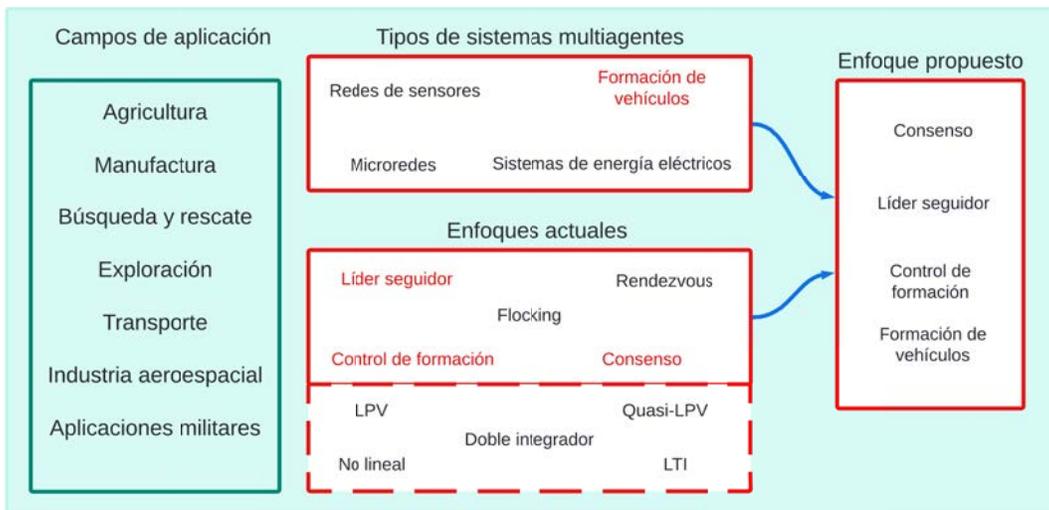


Figura 1.1: Sistemas multiagente.

El intercambio de información entre los estados es puede procesado de manera centralizada o descentralizada. El procesamiento de información centralizado supone que la información de los agentes (posiciones y velocidades) son transmitidas a una unidad de procesamiento, la cual calcula las entradas de control de los agentes, como se observa en la figura 1.2, por lo tanto, los agentes en una red centralizada tienen la característica de

tener una complejidad mínima, los agentes están compuestos únicamente de *hardware*. Al procesar la información medida en una unidad central de procesamiento, es necesario el contar con una unidad adecuada para cada sistema, es decir, los procesos que realiza la unidad de procesamiento de la red dependen de la cantidad de agentes del sistema [1]. La necesidad de tener una unidad de procesamiento adecuada hace que las redes centralizadas estén compuestas por menos agentes y realicen tareas menos complejas.

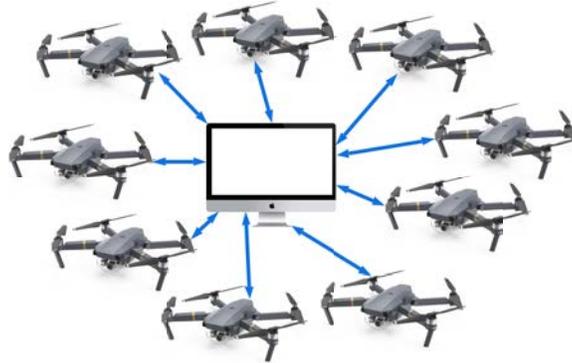


Figura 1.2: Red de procesamiento centralizada [1].

La red descentralizada cuenta con que los agentes transmitan la información de sus estados entre sí (figura 1.3), los agentes procesan la información por sí solos, ya que no cuentan con una unidad de procesamiento principal [16]. Este tipo de red requiere mayor complejidad en los agentes, ya que necesitan unidades con la capacidad de procesamiento suficiente para sentir, comunicar y calcular los estados de sí mismo, recibir los estados de los agentes vecinos y calcular sus propias entradas de control. Por lo tanto, el añadir agentes a las redes descentralizadas sólo representa una dificultad económica.

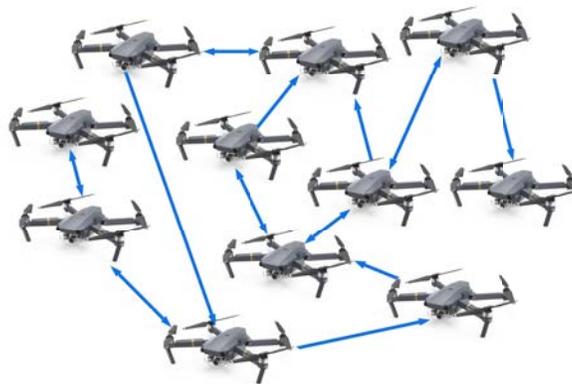


Figura 1.3: Red de procesamiento descentralizada [1].

Las misiones asignadas para un sistema multiagente en formación pueden fallar cuando un agente tiene un funcionamiento defectuoso si su control es descentralizado [14]. La degradación natural de los componentes de un agente compromete el desempeño del

sistema multiagente, ya que, en su mayoría cuentan con actuadores, sensores y fuentes de alimentación, sobre los cuales pueden ocurrir fallas o averías. Debido a esto, este trabajo plantea la formación de vehículos en el enfoque de sistemas multiagente, en el cual se analizará el desempeño de la fuente de alimentación de los agentes, ya que su desempeño tiende a decrecer por los ciclos de carga y descarga, daños provocados por descargas excesivas, altas temperaturas, etc. Dicho decrecimiento en un agente afecta su desempeño y la fiabilidad de la información que este comparte con los agentes vecinos, por lo tanto, la formación y la misión estará comprometida.

1.1. Antecedentes

A lo largo de los años se han desarrollado distintas técnicas, dispositivos y herramientas, con la finalidad de facilitar tareas cotidianas, las cuales van desde las más primitivas y esenciales como las armas de cacería, pasando por las más trascendentes como los vehículos, hasta las más triviales dedicadas al ocio; por lo tanto, los dispositivos complejos que reducen la mano de obra humana y por ende los costos, así como la mortalidad de algunos labores, han sido creados más bien por la necesidad de disminuir el riesgo de lesiones y decesos causados por dichos labores. Hoy en día, gracias al desarrollo tecnológico, muchas de estas creaciones como lo son los automóviles, motocicletas, aviones helicópteros, navíos etc, son invenciones relativamente recientes; el automóvil empezó a comercializarse a finales del siglo XIX y la gran mayoría de los vehículos destinados al transporte han sido ampliamente utilizados y desarrollados a partir del siglo XX. Actualmente, los avances de este siglo se enfocan en los dispositivos inteligentes capaces de realizar tareas con un porcentaje mínimo de intervención humana. Sin embargo, a diferencia de lo que se puede intuir, la búsqueda de la autonomía de las herramientas utilizadas por el ser humano comenzó mucho tiempo atrás. En el año de 1478 (siglo XV) el famoso polímata Leonardo da Vinci desarrolló un autómata mecánico programable controlado por computadora, el cual fue utilizado en distintos festivales en los siglos XVI y XVII y es considerado la primera computadora en las civilizaciones occidentales, a parte de ser un precursor importante en cuanto a los robots actuales. Este autómata fue reconstruido a partir de esbozos realizados por el mismo Leonardo da Vinci, el autor del libro *Leonardo's Lost Robots* [2] relata el proceso de reconstrucción, dicho proceso requirió de cierta interpretación y comparación con otros mecanismos de otros tiempos; el autómata (figura 1.4(a)) se 'programa' a través de levas (figura 1.4(b)) de distintos tamaños, las cuales afectan la trayectoria que sigue la reconstrucción del autómata (figura 1.4(c)).

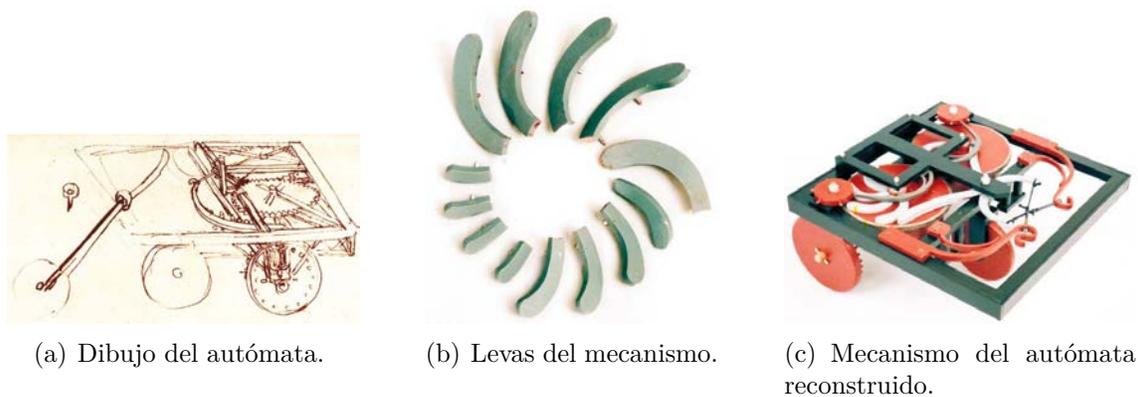


Figura 1.4: Mecanismo del autómeta creado por Leonardo da Vinci [2].

Años después del desarrollo del autómeta, el desarrollo de tecnología autónoma en los siglos posteriores no presentó avances significativos hasta el siglo XX durante la segunda guerra mundial. Donde los ejércitos alemán e italiano enfocaron sus esfuerzos en el diseño de bombarderos controlados a distancia; sin embargo, no fue hasta la guerra de Vietnam que el ejército estadounidense se unió al desarrollo de vehículos aéreos no tripulados a partir de la década de los 60s. En el año de 1962 se produjo el primer dron de propulsión a chorro *Ryan Model 147 Lightning Bug*, cuya familia de UAVs (AQM-34Q, BGM-34C) es la primera que cumple con la definición de vehículo aéreo no tripulado [17], en la figura 1.5 se muestra como se clasifican los vehículos aéreos no tripulados. En tiempos más recientes, los *fixed-wing* UAVs o UAVs de ala fija han sido los vehículos aéreos más utilizados en el campo militar, en especial en misiones o tareas donde se requiere alta velocidad y recorrer distancias mayores, sin embargo, las aplicaciones en labores de rescate y de menor alcance necesitan vehículos cuya maniobrabilidad sea suficiente para completar misiones específicas. En el año 2010 la compañía *Parrot* desarrolló un cuadricóptero controlado remotamente, llamado *Parrot AR.Drone*, diseñado para exteriores, este vehículo cuenta con 6 grados de libertad y con una unidad inercial miniatura, la cual mide sus ángulos de rotación para su alcanzar la estabilización de este.

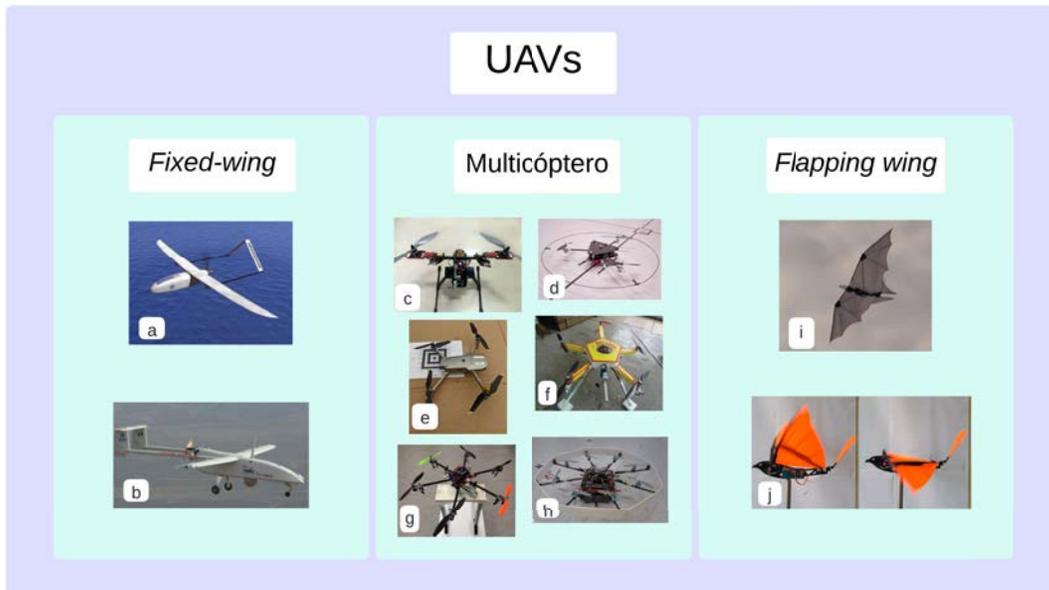


Figura 1.5: Clasificación de vehículos UAV [3]. a) y b) vehículos *fixed-wing* o ala fija [4]; c) bicóptero [5]; d) tricóptero [6]; e) cuadricóptero [7]; f) pentacóptero [8]; g) hexacóptero [9]; h) octacóptero [10]; i) y j) vehículos *flapping wing* [11], [12].

Las aplicaciones de los UAVs han evolucionado gracias a la comercialización de drones de distintos tipos, los cuales son desde recreativos, hasta de rescate o reconocimiento, como se ha mencionado anteriormente. En 2009, se lanzó el proyecto Alcedo [18], donde se desarrolló un cuadricóptero sistemas de localización y navegación sofisticados, con el fin de detectar víctimas de avalanchas; por otra parte, en [19] se modificó un UAV comercial para realizar operaciones de rescate, donde se valida la fiabilidad del sistema de rescate diseñado. En la agricultura, los UAVs tienen varias aplicaciones como la monitorización del crecimiento y la salud de la vegetación, la irrigación y el control de plagas [20]; además, los UAVs permiten economizar en las pruebas de imágenes aéreas y la evaluación de los terrenos, siendo esta aplicación el objetivo de [21] donde se evalúa el terreno que sufrió daños por tornado. A pesar de las aplicaciones novedosas que son capaces de realizar los UAVs, el mismo UAV diseñado específicamente a realizar misiones de largo alcance suele ser insuficiente en estos casos, en consecuencia, surgen los sistemas multiagente. Los sistemas multiagente son sistemas compuestos por dos o más agentes, los cuales se comunican de manera centralizada o descentralizada, es decir, no necesariamente se comunican entre sí, los cuales tienen objetivos locales o individuales; es importante resaltar que no es un sistema en general, sino múltiples sistemas [22]. Las distintas definiciones que se le ha dado al término 'agente' en el ámbito científico en general se comparten conceptos como la noción de un agente, su entorno y autonomía. Wooldridge [23] define a un agente como una entidad (*software* o *hardware*) situada en algún entorno y es capaz de reaccionar de manera autónoma a los cambios suscitados en el entorno. Los sistemas multiagente y sus aplicaciones civiles comenzaron a tomar importancia a partir del año 2000; por ejemplo, en [24] se reportan los experimentos llevados a cabo en Portugal en los años 2003, 2004 y 2005, donde se realiza detección de

incendios y localización de objetos utilizando 3 distintos UAVs comunicados entre sí; en [25] se utilizó la localización y el mapeo simultáneos (SLAM) en sistemas multiagente, para desplegar un sistema multiagente capaz de localizar personas en ambientes urbanos; en [26] se diseña un sistema multiagente con la capacidad de transportar las cargas y evadir obstáculos mientras la carga es transportada. Los distintos tipos de UAVs cuentan con una fuente de alimentación (batería), motores o servomotores dependiendo del UAV, propelas y tarjetas electrónicas para sus diferentes funciones de conectividad, estabilidad, comunicación, etc; estas, para las aplicaciones mencionadas anteriormente son de vital importancia, sobre todo si son misiones con un bajo margen de error. Este trabajo hace énfasis en el sistema de propulsión, el cual está compuesto por los motores, los controladores de velocidad del UAV y la batería o fuente de alimentación; los trabajos sobre la batería de los UAVs en sistemas multiagente suelen ser análisis u optimizaciones como lo presentan [27] y [28].

1.2. Estado del arte

Las tareas asignadas a los sistemas multiagente son aquellas que no pueden ser cumplidas por un agente, por lo tanto, diferentes estrategias de control cooperativo han sido diseñadas, con la finalidad de completar misiones de mayor complejidad e importancia. Algunas de estas estrategias serán referenciadas a continuación.

Un agente es un sistema dinámico, el cual puede ser un dispositivo o vehículo autónomo tal como: vehículos terrestres, acuáticos y aéreos, satélites, sensores, microprocesadores, circuitos eléctricos, etc. El objetivo del control cooperativo es controlar múltiples agentes autónomos de manera eficiente, con el fin de alcanzar un comportamiento colectivo por medio de interacción entre los agentes. El control cooperativo de los sistemas multiagente es capaz de replicar y describir comportamientos observados en la naturaleza como las rebaños y enjambres. En esta sección, la literatura donde se presenta control de formación será analizada, así como los trabajos de investigación de control tolerante a fallas y desempeño de la batería en vehículos aéreos no tripulados (UAVs).

1.2.1. Control de formación basado en líder seguidor

La estructura de líder seguidor ha utilizada recientemente en los controles basados en consenso. Es esta estructura, uno o más agentes son considerados líderes, dichos agentes líderes guían a los seguidores con la finalidad de alcanzar el consenso. El desempeño de los sistemas multiagente se mejora agregando líderes extra a este [29]. El consenso líder seguidor de un sistema multiagente puede ser un líder virtual [30] o un líder real [31].

El control a sistemas multiagente con líder seguidor tiene distintos enfoques de control, por ejemplo: en [32] se diseñaron estrategias de control predictivo basado en el modelo (MPC) con la finalidad de resolver los problemas de control de formación y seguimiento de trayectoria óptima bajo para sistemas multiagente líder seguidor; en [33] se diseñó

una estrategia de control introduciendo una red neuronal de base radial al protocolo de consenso con el objetivo de mejorar el aprendizaje basado en un conjunto de variables auxiliares y filtros de primer orden; en [34] se diseñó un control de modos deslizantes con una ley de control para cada UAV con la finalidad de resolver el problema de control de formación sujeto a restricciones en la velocidad de los agentes; en [35] se diseñó un control de formación variante en el tiempo para un sistema multiagente de discreto y de segundo orden, donde se tienen en cuenta tanto los retardos temporales de comunicación no uniformes como una topología cambiante; en [36] se diseñó un control de formación basado en consenso utilizando un algoritmo de programación de ganancias difusas para la coordinación basada en consenso. El objetivo del control de formación es estabilizar las posiciones entre los agentes hacia un valor definido con anterioridad. Para alcanzar el control de formación se realiza un algoritmo para un grupo de agentes, con la finalidad de alcanzar una formación; los artículos analizados están enfocados principalmente en los UAVs.

En [37] se diseñó un control robusto basado en observador, utilizando una técnica de control de formación basada en consenso líder seguidor, este control fue aplicado a una flotilla de UAVs. Para diseñar el control mencionado, la estimación del error de seguimiento entre los agentes con respecto al líder virtual fue lograda mediante una ley control basada en observador. Para calcular la ley de control basada en observador, el autor centra su trabajo en una optimización con el criterio H_∞ , así como también se centra en el análisis de estabilidad de Lyapunov en lazo cerrado; de esta forma, se proponen LMIs, cuya solución permite el cálculo de las ganancias matriciales del controlador basado en observador. La dinámica del sistema multiagente fue descrita por un doble integrador por agente al proponer las LMIs, sin embargo, el ejemplo numérico presentado fue descrito por el modelo no lineal de un UAV Newton-Euler; además, los resultados fueron obtenidos en simulación.

En [38] un protocolo cuadráticamente acotado basado en observador para sistemas multiagente bajo control de formación líder seguidor fue diseñado, donde los agentes están sujetos a perturbaciones exógenas acotadas. Este trabajo calcula ganancias matriciales para un controlador basado en observador para alcanzar un control de formación bajo perturbaciones externas, satisfaciendo así, las condiciones establecidas del líder virtual y las acotaciones. Para resolver la problemática, se proporcionaron suficientes condiciones para garantizar el consenso líder seguidor cuadráticamente acotado; después, se propusieron LMIs, cuya solución provee las ganancias del controlador y el observador. Los resultados presentados de este artículo fueron obtenidos por medio de simulaciones, donde el caso acotado muestra robustez a la turbulencia ocasionada por el viento, en contraste al caso nominal. Las LMIs fueron propuestas a partir de la representación de un sistema multiagente compuesto de dobles integradores; en el ejemplo numérico los agentes son representados por el modelo no lineal de un UAV.

En [39] un protocolo de consenso líder seguidor para sistemas multiagente fue propuesto, donde el sistema multiagente está sometido a fallas en el intercambio de información y perturbaciones externas. La principal contribución de este artículo es el diseño de un control de formación robusto líder seguidor, utilizando los enfoques de Lyapunov

y el criterio de optimización H_∞ , tal que los agentes sigan un líder virtual con un desempeño óptimo a pesar de presentar fallas en el intercambio de información entre agentes y estar expuesto a la turbulencia del viento. Para alcanzar el consenso en este entorno, LMIs fueron propuestas con la finalidad de obtener unas ganancias capaces de absorber las condiciones a las que está sujeto el sistema. Los resultados presentados son simulaciones, donde se compara un control de formación clásico con el propuesto; la robustez del control diseñado se demuestra utilizando un sistema multiagente compuesto por la representación no lineal de múltiples UAVs.

En [40] una ley de control basada en observador para el consenso líder seguidor para sistemas multiagente fue diseñada, con el objetivo de estimar las posiciones y velocidades de los agentes, lo anterior fue aplicado a representaciones no lineales de un sólo integrador y doble integrador. Para ambos casos, se propusieron LMIs para garantizar la estabilidad de los sistemas y para obtener las ganancias del observador y el controlador líder seguidor. En contraste con los artículos mencionados con anterioridad, los agentes tienen distintas representaciones y se presentaron dos ejemplos numéricos; los resultados de las simulaciones muestran que las posiciones y velocidades de un líder seguidor nominal son más lentas que la estimación presentada por el autor.

1.2.2. Sistema de propulsión

En años recientes, el uso de drones se ha extendido a aplicaciones como monitoreo de actividades o tareas de entrega y logística. Los UAVs integran distintos elementos en su estructura; organizando los subsistemas como el sistema aerodinámico, el sistema de propulsión y el sistema de control o controlador [41]. El sistema de propulsión es el principal subsistema de todo vehículo no tripulado, este subsistema se compone de motores, controladores de velocidad y fuente de alimentación (batería) [42]. Existen distintos enfoques para el diseño de un sistema de propulsión por ejemplo: en [43] se propone un método de caracterización aplicado al sistema de propulsión de un UAV, el cual se basó en métodos experimentales para la obtención de parámetros; en [44] se presentan técnicas de control para el motor de combustión interna de un UAV, el cual cuenta con un sistema de propulsión híbrido; en [45] se presentó el modelado y control de un UAV, cuyo sistema de propulsión es híbrido, los resultados de este trabajo se validaron de manera experimental. Particularizando el enfoque de este trabajo, las baterías de iones de litio (de polímero de litio y de iones de litio) son las empleadas en los sistemas de almacenamiento, medios de transporte eléctricos y en los vehículos aéreos [46]. Sin embargo, el comportamiento de una batería es complejo, debido a los principios internos y las condiciones inciertas de trabajo de esta, por lo tanto, el establecer un modelo que represente de manera precisa el comportamiento de una batería puede complicarse [47].

Como se ha mencionado en las secciones previas, el modelo de la batería y el de los motores que conforman el sistema de propulsión de un UAV será seleccionado, en esta sección se analizará literatura acerca del uso de los sistemas de propulsión para la optimización del consumo, predicción del desempeño de un UAV y análisis de la batería

de un UAV bajo condiciones adversas.

En [27] un algoritmo de generación de trayectoria fue diseñado, dicho algoritmo toma en cuenta el desempeño de la batería a lo largo del tiempo, con el fin de optimizar el consumo de energía mientras se recorre una trayectoria o se realiza una misión; el objetivo es encontrar una trayectoria óptima para un UAV, mientras el consumo de energía es minimizado, donde el tiempo final de la misión depende de la variación del estado de salud de la batería (SoH). Este objetivo fue alcanzado por medio de la predicción de la degradación de la batería para calcular el estado de salud de esta. Los resultados en simulación muestran una comparación entre un algoritmo de generación de trayectoria basado en funciones polinómicas con el algoritmo basado en la optimización del consumo de energía diseñado en este artículo; de esta manera, se probó que el algoritmo propuesto por el autor es capaz de generar trayectorias con un consumo mínimo de energía por el UAV incluso restringido por el estado de salud de la batería. Además, se diseñó un controlador PID en cascada donde se incluye al sistema de propulsión de un UAV, es decir, se incluyen la dinámica de la batería y de los motores.

En [48] se presenta un modelo completo de consumo para UAVs, el es utilizado para predecir requerimientos de energía para generar trayectorias energéticamente eficientes para UAVs. El modelo de este artículo fue obtenido por medio de estudios empíricos acerca del desempeño de la batería en distintas actividades de un UAV. Para el desarrollo del modelo se consideraron distintos escenarios de vuelo y condiciones en el entorno. Se hicieron pruebas experimentales, tales como: el consumo de energía en tierra, consumo de energía cuando las comunicaciones vía GPS y Wi-Fi están funcionando, el impacto del movimiento al realizar una trayectoria y el impacto de la turbulencia del viento; donde se obtuvo información con la finalidad de conocer el consumo del UAV durante las pruebas. Finalmente, se planteó una misión para un UAV con el fin de evaluar el modelo diseñado. En los resultados experimentales se puede observar un error mínimo entre el consumo de energía calculado por el modelo con el medido, probando así, la fiabilidad del modelo presentado por el autor.

En [49] el autor presenta una metodología para estimar el estado de carga (SoC) de la batería de un UAV, utilizando un filtro de Kalman extendido (EKF) durante el vuelo de un UAV. La estimación tiene como finalidad el predecir el tiempo en que se descarga la batería o fin de la descarga (EoD), mientras un UAV realiza una misión. Lo anterior, se logra considerando la representación matemática del sistema de propulsión de un UAV, el cual se compone por los motores sin escobillas (BLM) y la batería, lo cual permite calcular el consumo de energía durante una misión. Los resultados de simulación muestran el estado de carga estimado y el tiempo final en que se descarga la batería.

En [50] se propuso un circuito equivalente para modelar una batería de litio, el cual mantiene una relación entre los parámetros del modelo con los principios electromecánicos de la batería. El autor remarca algunos modelos en la literatura que tienden a fallar en relacionar los parámetros con las propiedades físicas de la batería. Para representar de manera completa el comportamiento de la batería, se presentó una metodología para obtener los parámetros del modelo, el cual consiste en tres pruebas experimentales: la

primera es una descarga total de la batería con una corriente constante, la segunda consiste de aplicar un comportamiento escalonado de carga y descarga y la tercera es una espectroscopía de impedancia electroquímica. Para demostrar la fiabilidad del modelo, este fue sometido a distintas temperaturas, salidas de potencias y frecuencias, además, el modelo se probó experimentalmente en una micro red, demostrando así, la precisión y estabilidad del modelo.

1.2.3. Control tolerante a fallas en sistemas multiagente

La tolerancia a fallas se define como la capacidad de un sistema de continuar operando a pesar de presentar fallas, las cuales se definen como una desviación del valor aceptable de un parámetro [51]. Los controles diseñados para los sistemas multiagente suelen trabajarse en casos libres de fallas en la actualidad, sin embargo, el enfoque multiagente incrementa el número de actuadores, sensores, componentes y enlaces de comunicación a tomar en cuenta para el diseño de controladores [52]. Por lo tanto, recientemente se han desarrollado distintos esquemas de control tolerante a fallas destinados a los sistemas multiagente con la finalidad de aumentar la fiabilidad de estos. En [52] se clasifican las estrategias recientes de control tolerante a fallas de sistemas compuestos por múltiples UAVs de la siguiente manera: control tolerante a fallas individual [53], control tolerante a fallas líder seguidor [54] y control tolerante a fallas distribuido [55]. Los controles tolerantes a fallas (FTC) son diseñados para mantener la integridad de un controlador en un caso de posible falla, así como también, cambios en las condiciones de operación del sistema.

Algunos enfoques de control para los controles tolerantes a fallas se muestran a continuación: en [56] se diseña un control tolerante a fallas adaptativo para sistemas multiagente utilizando un algoritmo de aprendizaje; en [57] se diseñan dos controladores tolerantes a fallas adaptativos basados en observadores distribuidos y activados por eventos; en [58] se diseña un control tolerante a fallas utilizando modos deslizantes para un sistema multiagente con fallas en el actuador. En esta sección se citan y resumen artículos que involucren el control tolerante a fallas en los sistemas multiagente y a los UAVs.

En [59] un control tolerante a fallas para sistemas multiagente con topología cambiante fue diseñado, el sistema multiagente fue sometido a fallas aditivas no lineales y a fallas intermitentes de comunicación entre los agentes y el controlador. Las fallas aditivas fueron añadidas como entradas nominales; además, las fallas intermitentes en el intercambio de información fueron añadidas entre el controlador y los agentes. Los distintos escenarios fueron analizados bajo las topologías centralizada y descentralizada. Finalmente, los resultados de simulación validan que el consenso puede ser alcanzado bajo las condiciones adversas que el autor propuso.

En [60] se diseñó un control tolerante a fallas para mitigar fallas en el actuador y pérdida en la efectividad del mismo, sin la necesidad de aislar y detectar las fallas. Las fallas de este control se definieron como no lineales y variantes en el tiempo, estas logran ser reconfiguradas por medio de un actuador virtual adaptativo no lineal, cuyas

condiciones son consideradas para el cálculo de las ganancias del control. Para absorber las fallas se utiliza un modelo de referencia, el cual es seguido por el actuador virtual. Los resultados de simulación muestran la reducción de los errores de seguimiento de los agentes, probando así, la tolerancia a fallas del control propuesto.

En [61] se diseñaron distintos esquemas de control tolerante a fallas activo, donde las fallas fueron estimadas por tres observadores: adaptativo, proporcional-integral y quasi-LPV. En este artículo se analiza la controlabilidad por medio del método llamado índice de autoridad de control disponible (ACAI), se diseñan observadores utilizando desigualdades matriciales lineales, se diseña un control proporcional-derivativo (PD) para un vehículo de despeje y aterrizaje vertical (VTOL) y finalmente las fallas son reconfiguradas por medio del control nominal y el sistema de detección de fallas para mantenerse cerca del funcionamiento libre de fallas. Los resultados en obtenidos en simulación fueron validados experimentalmente en un vehículo VTOL.

1.2.4. Resumen y análisis del estado del arte

Los artículos revisados muestran una tendencia por la técnica de líder seguidor en el control de formación para los sistemas multiagente, también, en la literatura se tiende a omitir el impacto que puede tener el sistema de propulsión en la formación de vehículos autónomos. El modelo de batería más utilizado para obtener el estado de carga de una batería de litio es el método de conteo de amperios, el cual se representa en forma de ecuaciones lineales en [27] y [49]. Los artículos donde se presentan controles tolerantes a fallas están restringidos a fallas en el actuador y al intercambio de información en sistemas multiagente, por lo tanto, la introducción del sistema de propulsión es un tema no abordado aún.

El involucrar el sistema de propulsión en los artículos relacionados a los sistemas multiagente compuestos por UAVs abre un enfoque interesante, ya que con la información obtenida de los motores, la batería y los controladores de velocidad es posible diseñar controles basados en el consumo de la energía y en técnicas de generación de rutas óptimas, además, la precisión en las simulaciones aumenta al agregar los parámetros, la dinámica y las restricciones que conlleva la adición de un sistema de propulsión. En el apéndice B se muestra una tabla con las características más relevantes para este trabajo de cada artículo revisado.

1.3. Planteamiento del problema

Actualmente, es posible completar distintas tareas de manera remota, utilizando vehículos autónomos, donde la complejidad de las tareas pueden requerir el uso de múltiples agentes trabajando en conjunto (sistemas multiagente) con la finalidad de completar un objetivo en común, por medio del intercambio de información de las posiciones y velocidades entre los agentes del sistema. Para este trabajo, una flotilla de 3 UAVs con

posiciones iniciales aleatorias deben alcanzar una formación bajo un funcionamiento inadecuado por parte de la batería. Todos los agentes deben ser capaces de enviar y recibir información de sus agentes vecinos sin interrupciones, aunque existan cambios en la topología del sistema. A diferencia de los artículos referenciados, el control de formación es alcanzado incluyendo el sistema de propulsión de un UAV (figure 1.7). Este es integrado con la finalidad de analizar que tanto puede ser afectado un UAV por una avería o mal funcionamiento en su batería.

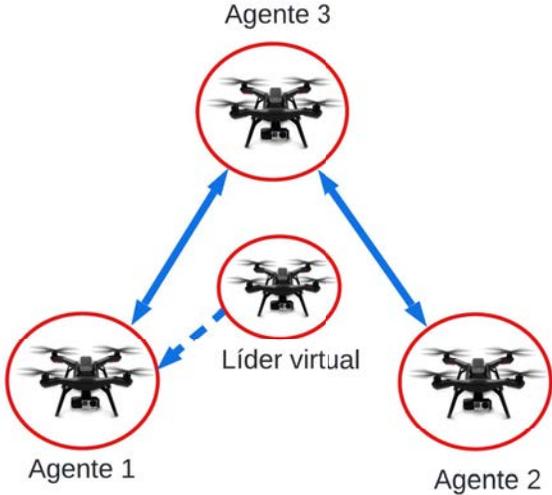


Figura 1.6: Sistema multiagente con líder seguidor.



Figura 1.7: Sistema de propulsión.

Con la finalidad de integrar el sistema de propulsión a los agentes del sistema, se plantea realizar una comparación entre las representaciones de un UAV, de esta manera, se busca encontrar los parámetros del modelo de la batería y de los motores adecuados que comprueben la similitud de las distintas representaciones, permitiendo así que el sistema sea capaz de alcanzar el control de formación. Para controlar el sistema multiagente mencionado se propone un control de formación líder seguidor, con la finalidad de que los agentes sigan al líder mientras realizan una formación predefinida, donde, se

resuelve una LMI basada en observador para calcular las ganancias de control, además, la representación dinámica de los BLM y la batería que componen el sistema de propulsión del UAV es añadida. Finalmente, el sistema será sometido a distintas trayectorias y condiciones de la batería, el sistema multiagente deberá mostrar resistencia a las condiciones adversas y a las misiones realizadas.

1.4. Justificación

Las distintas aplicaciones a sistemas multiagente se han extendido por distintas necesidades, sin embargo, las aplicaciones reales destinadas a tareas de alto riesgo donde se busca implementar un sistema multiagente se han visto reducidas debido al alto costo de implementación y la dificultad de implementar en ambientes adversos las técnicas de control existentes; por lo tanto, en el estado del arte se observa que los artículos relacionados a los sistemas multiagente obtienen sus resultados por medio de simulación, introduciendo perturbaciones, simulando fallas o funcionamientos no óptimos, limitando así las contribuciones de los trabajos relacionados, reduciendo estas sólo a aportaciones teóricas. No obstante, estas aportaciones brindan la fiabilidad necesaria para la implementación de los sistemas multiagente, esto por medio de optimizaciones, controles robustos y controles tolerantes a fallas, cuyas aplicaciones están ampliamente comprobadas en simulación. Por otro lado, los trabajos de sistemas multiagente no cuentan con enfoques que analicen el desempeño de los componentes físicos en las flotillas de UAVs; el enfoque de menor interés en la investigación acerca de sistemas multiagente es el desempeño de la batería, a diferencia de la investigación acerca de UAVs individuales ([49], [62], [63]). En consecuencia, el apuntar los esfuerzos en este campo poco profundizado es llamativo, ya que las optimizaciones, los controles robustos y los controles tolerantes a fallas aplicados sin tomar en cuenta el consumo del sistema multiagente pueden trasladarse a un enfoque de consumo energético, sin embargo, algunas técnicas existentes pudieran ser aplicadas parcialmente, específicamente en el control tolerante a fallas. Es importante tomar en cuenta los trabajos acerca de las baterías de litio, por ejemplo, en [64] se recopilan distintas técnicas de diagnóstico de fallas y control tolerante a fallas para vehículos eléctricos, sin embargo, la técnicas mencionadas se centran en fallas de sensor, por lo tanto el control tolerante a fallas compensa las fallas en los sensores. Las técnicas de diagnóstico de fallas han sido aplicadas de igual manera para sistemas multiagente; [65] muestra una adquisición de datos en diferentes tipos de condiciones de fallo, y se utiliza una red neuronal para clasificar los tipos de datos y tipos de fallas; [66] propone un esquema de diagnóstico de fallas de sensor para un conjunto de baterías de litio en serie, donde se presentan dos esquemas de generación de residuos y la falla es causada por un cambio abrupto en la temperatura de la batería; [67] presenta el diseño de un control tolerante a fallas activo basado en técnicas LPV donde se comparan dos controles, uno nominal y el control propuesto por el autor, el cual es una retroalimentación de estados robusta, los cuales fueron aplicados a UAVs con sistemas LPV y simulados con diferentes porcentajes de efectividad de la batería. La revisión de los artículos permite resaltar el interés por el diagnóstico de fallas de la

batería, el cual se expresa como falla de sensor, sin embargo, el control tolerante a fallas se encuentra limitado en un sistema multiagente compuesto por vehículos de cualquier tipo, ya que a pesar de que el sensor puede ser compensado por medio de estimaciones de parámetros de la batería, no es posible compensar la fuente de poder de un vehículo sin perder su incidencia en el sistema multiagente nominal, es decir, el vehículo que sufra la degradación de los parámetros de su batería tendría que ser retirado para que sea posible retomar el funcionamiento óptimo del sistema considerando condiciones ideales. Esta limitación sobre el enfoque tolerante a fallas nos lleva a buscar enfoques alternos, debido a que el 30 % de los accidentes relacionados con vehículos eléctricos son causadas por fallos en la batería [68]. El análisis de componentes, degradación de la batería, estado de salud de la batería y la planeación de misiones basadas en la predicción de la salud o carga de la batería son algunos de los enfoques alternos al control tolerante a fallas. Al inclinarnos por estos enfoques alternos, este trabajo busca aproximarse al control tolerante a fallas por medio de la obtención del comportamiento de la batería en condiciones nominales y adversas en un sistema multiagente bajo una misión; de esta manera, se logra recabar información acerca de los controladores diseñados y de como incide el desgaste de la batería en su funcionamiento. Posterior a esta explicación, se propone una hipótesis y se plantean los objetivos a seguir para proponer una solución que resuelva el problema planteado con anterioridad.

1.5. Hipótesis

Un sistema multiagente bajo un control de formación basado en consenso es capaz de completar misiones de manera segura bajo la degradación de la batería en un agente.

1.6. Objetivos

1.6.1. Objetivo general

Diseñar un control de formación basado en consenso aplicado a un sistema multiagente dinámico sujeto al comportamiento del sistema de propulsión de un UAV.

1.6.2. Objetivos específicos

- Diseñar una estrategia de control basada en consenso para el manejo de sistemas multiagente compuestos por vehículos aéreos no tripulados.
- Integrar el sistema de propulsión de un UAV a un control de formación basado en consenso.

1.7. Propuesta de solución

Para lograr que un MAS alcance una formación y realice una misión, se diseñó un control de formación líder seguidor, el cual consiste de un líder virtual no lineal, así como los agentes del sistema, los cuales también son no lineales. El agente líder y los agentes seguidores están controlados, el diagrama general propuesto se observa en la figura 1.8.

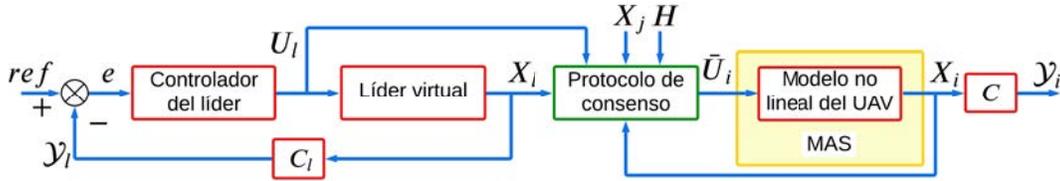


Figura 1.8: Diagrama de control de formación líder seguidor.

El sistema de propulsión está compuesto por la dinámica de la batería y los motores de un UAV; con la finalidad de integrar el sistema de propulsión de un UAV al esquema de control anterior, las leyes de control generadas por el protocolo de consenso son transformadas de forma que las leyes de control puedan ser expresadas en términos directamente relacionados con los motores de los agentes, esta transformación también permite calcular la dinámica de la batería.

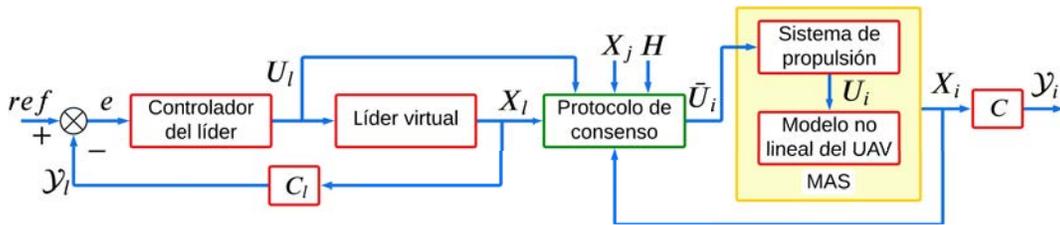


Figura 1.9: Diagrama de control de formación líder seguidor con el sistema de propulsión.

Finalmente, en la figura 1.9 se propone un diagrama de bloques para atender la problemática planteada anteriormente. Como se ha visto en el estado del arte, los trabajos relacionados muestran la falta del sistema de propulsión en los artículos sobre MAS, dichos artículos muestran en su mayoría resultados teóricos, donde las representaciones de los MAS se limitan a la dinámica traslacional y rotacional de los UAVs; además, el efecto de la batería ha sido estudiado con más regularidad únicamente en los UAVs, relacionando el efecto de la batería con el seguimiento de trayectorias y la optimización de estas en un enfoque energético o de consumo.

Capítulo 2

Representaciones de un UAV

En esta sección, se realiza una comparación entre 3 representaciones dinámicas que modelan el comportamiento de un UAV. El comportamiento de las representaciones dinámicas a comparar, deben ser similares para alcanzar el control de formación.

En primer lugar, se analiza una representación doble integrador en lazo abierto como la forma más simple de representar la dinámica de un UAV ([37], [40]); después, el modelo no lineal de un UAV utilizado en [13], [37] y [38] es analizado, el subsistema rotacional del modelo no lineal fue controlada por un controlador PI como se presenta en [15] y [37]. Finalmente, mismo controlador y modelo son el caso de estudio, con la diferencia de la adición de la dinámica de los BLMs [69] y la batería [70] como se realizó en [71] y [72]; llevándonos así, a buscar un comportamiento similar entre las salidas de los 3 casos de estudios descritos anteriormente.

2.1. Doble integrador

La dinámica doble integrador es la representación más sencilla usada para modelar el comportamiento de un UAV, está consiste en 3 ecuaciones lineales de segundo orden, es decir, el UAV en esta representación sólo se traslada en los ejes x , y y z , sin tener una rotación sobre los ángulos ϕ , θ y ψ . El doble integrador se representa de la siguiente manera:

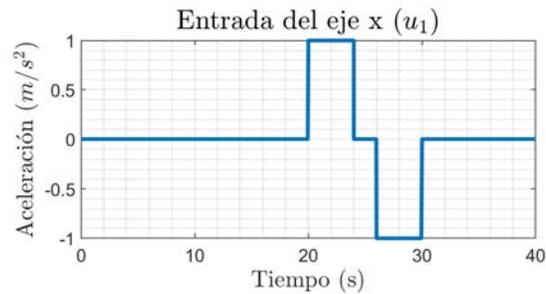
$$\begin{aligned}\dot{p} &= v, \\ \dot{v} &= u, \\ \mathcal{Y} &= p,\end{aligned}\tag{2.1}$$

donde p es el vector de posiciones definido como $p = [x, y, z]^T$, v es el vector de velocidades definido como $v = [\dot{x}, \dot{y}, \dot{z}]^T$, u es el vector de entradas del sistema definido como $u = [u_1, u_2, u_3]^T$ y \mathcal{Y} es el vector de salidas del sistema definido como $\mathcal{Y} = [x, y, z]^T$. El sistema (2.1) fue simulado en lazo abierto, como se observa en la figura 2.1.

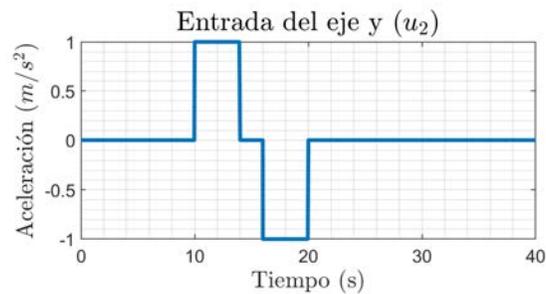


Figura 2.1: Esquema del doble integrador.

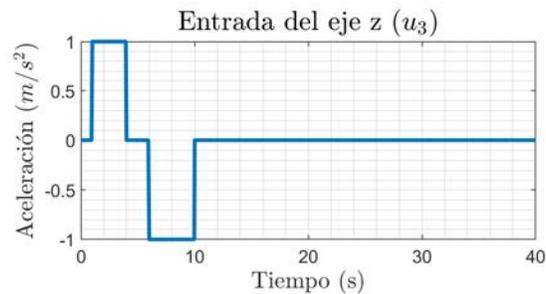
Las entradas de aceleración u del sistema son un pulso positivo seguido de un pulso negativo con amplitud ± 1 . La figura 2.2 muestra las entradas a lo largo del tiempo de simulación.



(a) Pulso positivo en el eje x del segundo 20 hasta el segundo 24, seguido de un pulso negativo del segundo 26 hasta el segundo 30.



(b) Pulso positivo en el eje y del segundo 10 hasta el segundo 14, seguido de un pulso negativo del segundo 16 hasta el segundo 20.



(c) Pulso positivo en el eje z desde el segundo 1 hasta el segundo 4, seguido de un pulso negativo del segundo 6 hasta el segundo 10.

Figura 2.2: Entradas del doble integrador.

Las entradas mostradas en la figura 2.2 son suministradas a los tres casos planteados en esta sección.

2.2. Modelo no lineal de un UAV con la dinámica rotacional controlada

El UAV considerado está compuesto por cuatro motores sin escobillas o *brushless motors* (BLM) anclados a un chasis que cuenta con 4 brazos simétricos. Cada BLM tiene una hélice sujeta al eje, con la finalidad de generar una fuerza de empuje suficiente para levantar el UAV; además, estos rotores son capaces de generar torques angulares capaces de trasladar el UAV en distintas direcciones. El marco de referencia de un UAV que representa los movimientos de este se puede observar en la siguiente figura:

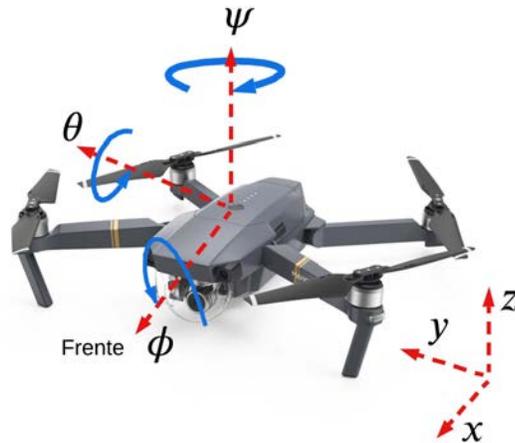


Figura 2.3: Marco de referencia de un UAV cuadricóptero.

Nótese el ángulo ϕ o *roll* hace posible el movimiento del UAV en el eje y , así como el ángulo θ o *pitch* hace posible el movimiento del UAV en el eje x .

El comportamiento dinámico de un UAV es representado comúnmente por el modelo no lineal Newton-Euler [73]. Las ecuaciones diferenciales no lineales de (2.2) representan el comportamiento de un UAV incluyendo la parte rotacional de este a diferencia del

doble integrador.

$$\begin{aligned}
\ddot{x} &= (\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi)\frac{T}{m}, \\
\ddot{y} &= (\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi)\frac{T}{m}, \\
\ddot{z} &= -g + (\cos\phi\cos\theta)\frac{T}{m}, \\
\ddot{\phi} &= \frac{J_y - J_z}{J_x}\dot{\theta}\dot{\psi} + \frac{R}{J_x}, \\
\ddot{\theta} &= \frac{J_z - J_x}{J_y}\dot{\phi}\dot{\psi} + \frac{P}{J_y}, \\
\ddot{\psi} &= \frac{J_x - J_y}{J_z}\dot{\theta}\dot{\phi} + \frac{Y}{J_z}.
\end{aligned} \tag{2.2}$$

El modelo (2.2) se divide en dos subsistemas, donde el subsistema rotacional se compone de los estados ϕ , θ , ψ , $\dot{\phi}$, $\dot{\theta}$ y $\dot{\psi}$, los cuales representan la rotación y la velocidad angular en los ángulos *roll*, *pitch* y *yaw* respectivamente. El subsistema traslacional está compuesto por los estados x , y , z , \dot{x} , \dot{y} y \dot{z} , los cuales representan la traslación y la velocidad traslacional en los ejes x , y y z respectivamente. Los parámetros J_x , J_y y J_z son los momentos inerciales en los ejes x , y y z , además, g es la aceleración de la gravedad y m es la masa del UAV; finalmente, las entradas del sistema T , R , P y Y son la fuerza de empuje en el eje z y los pares rotacionales en los ángulos ϕ , θ y ψ . Nótese que el subsistema traslacional depende de los estados del subsistema rotacional, por lo tanto, para controlar la posición de un UAV, es necesario controlar el subsistema rotacional del UAV.

Para controlar el subsistema rotacional se generaron referencias con las ecuaciones mostradas en (2.2). En [74] las relaciones entre las aceleraciones de entrada con los ángulos de referencia ϕ_r , θ_r y ψ_r se definen como:

$$\begin{aligned}
\phi_r &= \arctan\left(\frac{-u_2}{\sqrt{u_1^2 + (u_3 + g)^2}}\right), \\
\theta_r &= \arctan\left(\frac{u_1}{u_3 + g}\right), \\
\psi_r &= 0,
\end{aligned} \tag{2.3}$$

además, la fuerza de empuje \bar{T} de un UAV se calcula por medio de la siguiente expresión [74]:

$$\bar{T} = m\sqrt{u_1^2 + u_2^2 + (u_3 + g)^2}. \tag{2.4}$$

2.2.1. Control PI

El control proporcional-integral diseñado para seguir los ángulos de referencia se sigue el siguiente esquema de control de la figura 2.4:

donde, el vector de estados se define como: $x = [\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}]^T$, el vector de entradas se define como $\bar{U}_\Phi = [R, P, Y]^T$ y el vector de salidas del sistema se define como $y = [\phi, \theta, \psi]^T$. Las matrices A , B y C se definen como:

$$\tilde{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \tilde{B} = \begin{bmatrix} 0 & 0 & 0 \\ \frac{1}{J_x} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \frac{1}{J_y} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \frac{1}{J_z} \end{bmatrix}; \tilde{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad (2.7)$$

Con el sistema (2.6) y las matrices (2.7) se aumenta el sistema, el modelo aumentado tiene la siguiente forma:

$$\begin{bmatrix} \dot{x} \\ \dot{\tilde{x}} \end{bmatrix} = \begin{bmatrix} \tilde{A} & 0 \\ \tilde{C} & 0 \end{bmatrix} \begin{bmatrix} x \\ \tilde{x} \end{bmatrix} + \begin{bmatrix} \tilde{B} \\ 0 \end{bmatrix} u \quad (2.8)$$

Finalmente, el método de colocación de polos se aplicó al sistema (2.8), donde la ley de control tiene la siguiente forma:

$$\bar{U}_\Phi = K_I \int_0^t (\Phi_r - \Phi) dt - K_P \Phi, \quad (2.9)$$

además, el error y la dinámica del error se definen de la siguiente manera:

$$e = \Phi_r - \Phi, \quad (2.10)$$

$$\dot{\mathcal{I}} = e, \quad (2.11)$$

donde \bar{U}_Φ es el vector de las leyes de control rotacionales definido como $\bar{U}_\Phi = [\bar{R}, \bar{P}, \bar{Y}]^T$, Φ es el vector de los ángulos de rotación de un UAV definido como $\Phi = [\phi, \theta, \psi]^T$, Φ_r es el vector de los ángulos de referencia generados por (2.3) y definidos como $\Phi_r = [\phi_r, \theta_r, \psi_r]^T$, e es el vector de errores, \mathcal{I} es el vector de integrales del error; K_P y K_I son las ganancias matriciales proporcionales e integrales respectivamente del controlador. Las ganancias del controlador se calculan utilizando la función de colocación de polos (*place*) de Matlab, por lo tanto, se propone el siguiente vector:

$$p = [-10, -10,001, -10,002, -10,003, -10,004, -10,005, -100, -10,001, -100,002],$$

el cual contiene los eigenvalores deseados para la matriz \tilde{A} . Las ganancias proporcional e integral del controlador son las siguientes:

$$K_P = \begin{bmatrix} 45,37 & 2,59 & 0 & 0 & 0 & 0 \\ 0 & 0 & 45,38 & 2,59 & 0 & 0 \\ 0,01 & 0 & 0,01 & 0 & 90,74 & 5,18 \end{bmatrix}, \quad (2.12)$$

$$K_I = \begin{bmatrix} 216,1 & -0,03 & 0,04 \\ 0 & 216,15 & 0,01 \\ 0,08 & 0,08 & 432,16 \end{bmatrix}. \quad (2.13)$$

2.3. Modelo no lineal de un UAV con sistema de propulsión y con la dinámica rotacional controlada

En esta sección el caso analizado es similar al de la sección 2.2, sin embargo, la dinámica de la batería y los BLM fueron incluidos en el esquema de control en la figura 2.5, dejando así, el siguiente esquema:

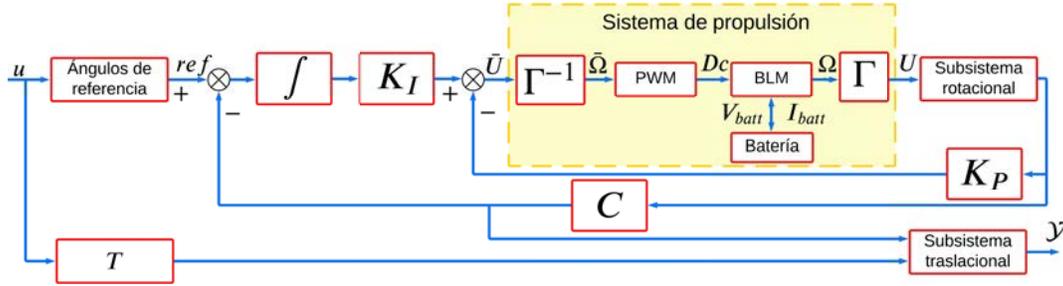


Figura 2.5: Modelo no lineal de un UAV con el subsistema rotacional controlado, incluyendo la dinámica de la batería y de los BLM.

Los bloques encerrados en la figura 2.5 están compuestos por una matriz Γ , que describe las entradas de un UAV en términos de las velocidades angulares de cada BLM; una función que genera una señal de control PWM; el modelo dinámico de la batería y el modelo dinámico de un BLM. La dinámica de cada bloque del sistema de propulsión en la figura 2.5 se describirá en las secciones posteriores.

2.3.1. Leyes de control en términos de las velocidades angulares de los motores

Como fue mostrado en (2.9) las entradas del subsistema rotacional fue controlado; además, las entradas de este pueden ser definidas en términos de las velocidades angulares de los motores BLM de un UAV, estas leyes de control dependen del número de motores del UAV, por lo tanto, las entradas del sistema para un cuádróptero son las siguientes [76]:

$$\begin{aligned}
 T &= b \left(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2 \right), \\
 R &= bl \left(-\Omega_1^2 + \Omega_3^2 \right), \\
 P &= bl \left(-\Omega_2^2 + \Omega_4^2 \right), \\
 Y &= d \left(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2 \right).
 \end{aligned} \tag{2.14}$$

El conjunto de ecuaciones (2.14) se representan en forma matricial con la siguiente expresión:

$$U = \Gamma \Omega^2 \quad (2.15)$$

donde el vector de entradas del sistema se definen como $U = [T, R, P, Y]^T$, el vector de velocidades angulares se define como $\Omega = [\Omega_1, \Omega_2, \Omega_3, \Omega_4]^T$, y la matriz Γ es definida como:

$$\Gamma = \begin{bmatrix} b & b & b & b \\ -bl & 0 & bl & 0 \\ 0 & -bl & 0 & bl \\ -d & d & -d & d \end{bmatrix}, \quad (2.16)$$

donde b es la constante de empuje, d es la constante de arrastre y l es la distancia de cada BLM al centro de masa del UAV.

2.3.2. Modelo de la batería

El comportamiento de los motores del UAV es impactado por el comportamiento de la batería; por lo tanto, el impacto de la batería en un UAV será considerado, así como su modelo matemático. Los modelos de una batería de litio propuestos en la literatura pueden ser basados en mecanismos electroquímicos, circuitos equivalentes y basados en adquisición de datos [77]. En [13] y [70], el comportamiento de la batería es analizado por medio del siguiente circuito equivalente:

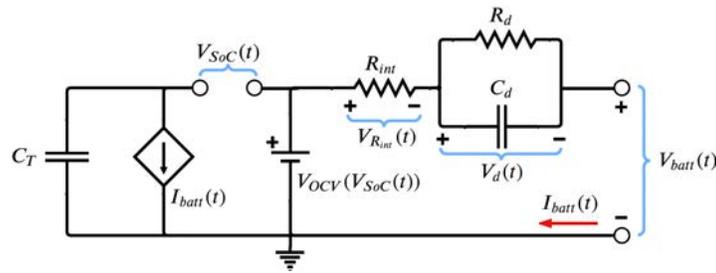


Figura 2.6: Circuito equivalente de una batería de polímero de litio ([13]).

donde V_{SoC} es el estado de carga o *state of charge* (SoC), C_T es la capacidad total de la batería, $V_{OCV}(V_{SoC})$ es el voltaje de circuito abierto, I_{batt} es la corriente total consumida, R_{int} es la resistencia interna de la batería, R_d y C_d es la resistencia y capacitancia del par RC del circuito, V_d es la respuesta transitoria de la batería y V_{batt} es el voltaje de la batería. De acuerdo a la figura 2.6, el comportamiento de la batería es modelado por

las siguientes ecuaciones:

$$\begin{aligned}
\dot{V}_{SoC} &= -\frac{1}{3600C_T}I_{batt}, \\
\dot{V}_d &= -\frac{1}{R_dC_d}V_d + \frac{1}{C_d}I_{batt}, \\
V_{OCV}(V_{SoC}) &= \sum_{\tilde{i}=0}^{\tilde{n}-1} \tilde{\lambda}_{\tilde{i}}V_{SoC}^{\tilde{i}} + \tilde{\lambda}_{\tilde{n}}\ln(V_{SoC})V_{SoC}, \\
V_{batt} &= V_{OCV}(V_{SoC}) - V_d - R_{int}I_{batt},
\end{aligned} \tag{2.17}$$

donde el orden de la función polinómica está dado por \tilde{n} y $\tilde{\lambda}_{\tilde{i}}$ son los coeficientes del polinomio. El método para calcular el SoC de la batería es llamado conteo de coulombs o conteo de amperios-hora (2.17), el cual es el método más utilizado par el cálculo del SoC [78]. En los siguientes artículos se utiliza el método de conteo de coulombs: [27], [49] y [72].

2.3.3. Modelo de los motores *brushless*

Las velocidades angulares de los BLMs son calculadas por medio de la relación entre los representaciones de las partes eléctricas y mecánicas de los BLMs ([69], [79]):

$$\begin{aligned}
\bar{v}_{batt} &= R\bar{i}_{batt} + K_E\Omega, \\
\dot{\Omega} &= \left(K_E\bar{i}_{batt} - d\Omega^2 - D_f\Omega - T_f \right) \frac{1}{J_E},
\end{aligned} \tag{2.18}$$

con el voltaje promedio generado por el controlador de velocidad electrónico (ESC) y la corriente total consumida por los BLMs ([27]):

$$\begin{aligned}
\bar{v}_{batt} &= V_{batt} \cdot Dc, \\
I_{batt} &= \sum_{\mathcal{M}=1}^{N_{\mathcal{M}}} \bar{i}_{batt_{\mathcal{M}}} \cdot Dc_{\mathcal{M}},
\end{aligned} \tag{2.19}$$

donde \mathcal{M} es cada BLM del UAV, $N_{\mathcal{M}}$ es el número de BLMs del UAV, \bar{v}_{batt} y \bar{i}_{batt} son el voltaje y corriente promedio generado por el ESC, Dc es el ciclo de trabajo de cada BLM, I_{batt} es la corriente total consumida por los motores V_{batt} es el voltaje de la batería. Además, los parámetros K_E , J_E , D_f , d y T_f son la fuerza contra electromotriz, el momento inercial de los BLMs, el coeficiente de amortiguamiento viscoso, la constante de arrastre y el par causado por la fricción.

Sin embargo, las ecuaciones de (2.18) y (2.19) fueron manipuladas con el fin de omitir la generación de las señales de la corriente y el voltaje promedio. Finalmente, la dinámica de los BLM se resumen por medio de las siguientes expresiones matemáticas:

$$\begin{aligned}
\dot{\Omega} &= \left[(V_{batt}Dc - K_E\Omega) \frac{K_E}{R} - d\Omega^2 - D_f\Omega - T_f \right] \frac{1}{J_E}, \\
I_{batt} &= \sum_{\mathcal{M}=1}^{N_{\mathcal{M}}} (V_{batt} \cdot Dc_{\mathcal{M}} - K_E \cdot \Omega_{\mathcal{M}}) \frac{Dc_{\mathcal{M}}}{R}.
\end{aligned} \tag{2.20}$$

El ciclo de trabajo (Dc) de la señal modulada o *pulse width modulation* (PWM) que controla a los BLM está dado por la siguiente ecuación algebraica:

$$Dc = \gamma_1 \bar{\Omega}^2 + \gamma_2 \bar{\Omega} + \gamma_3, \quad (2.21)$$

donde γ_1 , γ_2 y γ_3 son los coeficientes de la función que genera la señal PWM, y $\bar{\Omega}$ es el vector de velocidades angulares de referencia definido como $\bar{\Omega} = [\bar{\Omega}_1, \bar{\Omega}_2, \bar{\Omega}_3, \bar{\Omega}_4]^T$, el vector $\bar{\Omega}$ se obtiene por medio de la ecuación inversa de (2.15):

$$\bar{\Omega} = \sqrt{\Gamma^{-1} \bar{U}}, \quad (2.22)$$

donde $\bar{U} = [\bar{T}, \bar{R}, \bar{P}, \bar{Y}]^T$ es el vector de leyes de control teóricas calculadas en (2.9) y Γ^{-1} es la matriz inversa de (2.16).

2.4. Discusión de resultados (conclusión del capítulo)

Como se menciona en las secciones anteriores se realizó una comparación entre 3 casos, para finalizar con la comparación, en esta sección los resultados de las simulaciones son reportados, así como las conclusiones de estas; los resultados de las simulaciones son los siguientes:

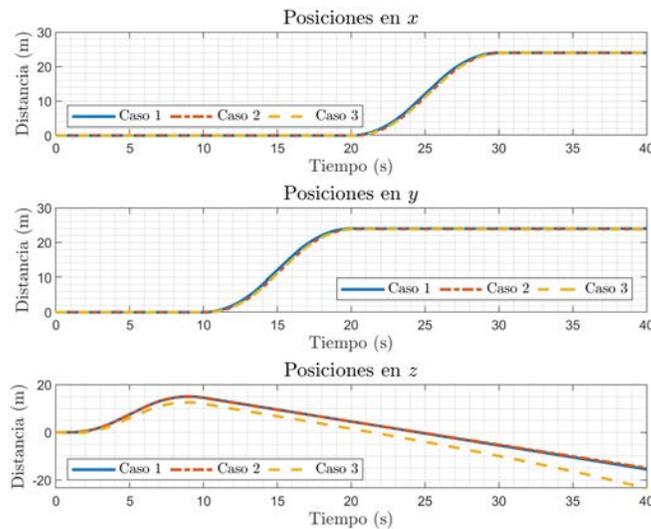


Figura 2.7: Posiciones de los casos de estudio.

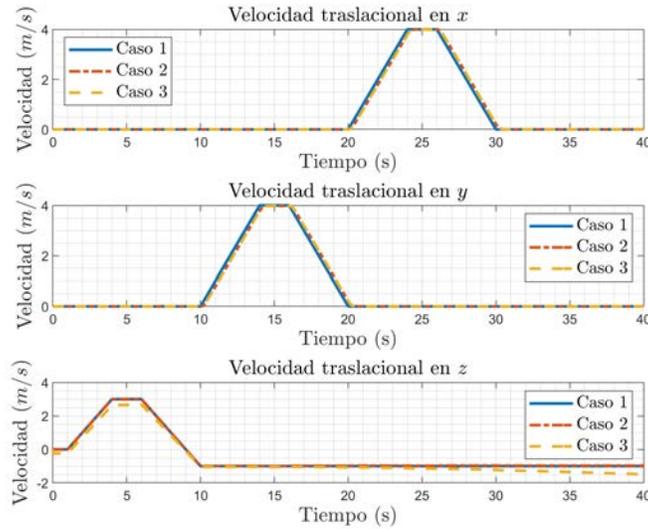


Figura 2.8: Velocidades de los casos de estudio.

Donde la línea continua azul es el doble integrador 2.1, la línea roja punteada hace referencia al modelo no lineal controlado en la sección 2.2 y la línea punteada amarilla representa el modelo no lineal controlado con las dinámicas de la batería y los BLM desarrollado en la sección 2.3. Se hicieron las siguientes observaciones acerca de las figuras 2.7 y 2.8:

- En la figura 2.7 a partir del segundo 20 el UAV se traslada en eje x hasta el segundo 30, donde el UAV alcanza los 24 metros. Este comportamiento se repite en el eje y , pero incrementando a partir de los 10 segundos hasta que el UAV alcanza los 24 metros en el segundo 20, por otro lado, la posición en el eje z tiende a decrecer sin estabilizarse en el tiempo de simulación.
- En la figura 2.8 la velocidad en el eje x incrementa, hasta que esta alcanza los 4 metros por segundo, después, la velocidad se estabiliza durante 2 segundos, finalmente, la velocidad decrece a 0 en el segundo 30. Este comportamiento se repite en el eje y , pero incrementando a partir del segundo 10, estabilizándose durante 2 segundos y decreciendo a 0 en el segundo 20, por otro lado, la velocidad en el eje z incrementa a partir del segundo 1, se estabiliza durante 2 segundos en una velocidad de 3 metros por segundo y decrece a -1 metro por segundo en el segundo 10.
- La diferencia entre los 3 casos en los ejes x y y es mínima. Sin embargo, en las figuras 2.7 y 2.8 el error de las posiciones y velocidades del z entre los 3 casos tiende a incrementar a lo largo del tiempo a partir del segundo 3 y a partir del segundo 1 respectivamente, a pesar del aumento del error el comportamiento de las posiciones y velocidades sigue siendo similar entre los casos comparados.

Como se puede observar en la sección actual, las distintas representaciones de un UAV se

comportan de manera similar a lo largo de la simulación, se puede decir que las diferentes representaciones tienen la misma respuesta a lo largo de tiempo incluso tomando en cuenta la integración del sistema de propulsión de un UAV, por lo tanto, las técnicas de control aplicadas en este trabajo tienen validez debido a que los errores mostrados tienen un valor bajo y permiten suponer que las estrategias de control utilizadas posteriormente son adecuadas. El procedimiento para simular los distintos sistemas de ecuaciones en esta sección se muestra en el algoritmo [A.1](#) en el apéndice [A](#).

Capítulo 3

Control de formación basado en consenso con sistema de propulsión

En la sección actual, se diseña un control de formación basado en consenso sin líder seguidor, siguiendo la metodología presentada en [15], con la adición de los modelos de la batería y los motores, incluyendo los parámetros obtenidos en la comparación sección 2. En la figura 3.1 se muestra el esquema de control de formación sin líder seguidor:

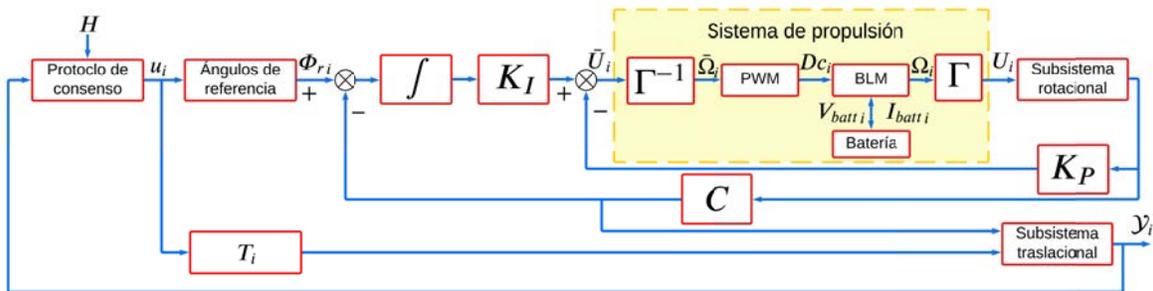


Figura 3.1: Diagrama de bloques del control de formación basado en consenso sin líder seguidor.

El esquema de la figura 3.1 alcanza la formación utilizando el protocolo de consenso basado en observador mostrado en [80], donde se agrega la extensión a control de formación. Este protocolo requiere de una formación predefinida, así como también se requiere transformar la variable calculada por el protocolo de consenso. También existe un control interno, el cual es un controlador por retroalimentación de estados PI la cual controla la parte rotacional de la dinámica del sistema multiagente. Además, el sistema multiagente para este caso de estudio está compuesto de tres agentes homogéneos, es decir, de comportamiento dinámico idéntico.

3.1. Teoría de grafos

Los sistemas multiagente están compuestos por agentes dinámicos individuales, los cuales comunican entre sí sus estados. También, los sistemas multiagente intercambian su información con los agentes vecinos a través de una topología de comunicación. En la figura 3.2 se muestra una topología de comunicación de un sistema multiagente.



Figura 3.2: Topología de comunicación de un sistema multiagente.

Para modelar los sistemas multiagente se utiliza la teoría de grafos. Un grafo es la representación de un grupo de nodos, donde cada nodo representa un agente, así como un enlace representa el flujo de información entre los agentes (figura 3.3). Un grafo directo \mathcal{G} es un par $(\mathcal{V}, \mathcal{E})$, donde $\mathcal{V} = \{v_1, \dots, v_N\}$ es un conjunto de nodos finitos no vacíos y $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ es un conjunto de bordes de pares de nodos, llamados bordes. Un grafo ponderado está asociado a un peso con cada borde en el grafo. El borde (v_i, v_j) en el conjunto de bordes \mathcal{E} denotan que el agente v_j puede obtener información del agente v_i , pero no necesariamente viceversa. Para el nodo (v_i, v_j) , el nodo v_i es llamado nodo padre, v_j es llamado nodo hijo y v_i es el nodo vecino de v_j . El conjunto de vecinos del nodo v_i se denota como \mathcal{N}_i cuya cardinalidad es llamada grado del nodo v_i .

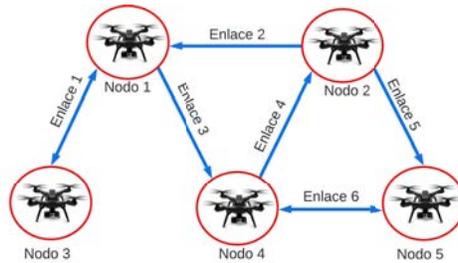


Figura 3.3: Grafo directo.

La matriz de adyacencias se define como $\mathcal{A} = [a_{ij}] \in \mathbb{R}^{N \times N}$, la matriz \mathcal{A} está asociada con el grafo \mathcal{G} ; puede decirse que $a_{ii} = 0$, a_{ij} tiene un valor positivo sí y sólo sí $(i, j) \in \mathcal{E}$, de lo contrario $a_{ij} = 0$. La matriz Laplaciana $\mathcal{L} = [\mathcal{L}_{ij}] \in \mathbb{R}^{N \times N}$ del grafo \mathcal{G} está definida como $\mathcal{L}_{ii} = \sum_{j \neq i} a_{ij}$ y $\mathcal{L}_{ij} = -a_{ij}$, $i \neq j$. La matriz Laplaciana y la matriz de adyacencias del grafo en la figura 3.3 son:

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \quad \mathcal{L} = \begin{bmatrix} 2 & -1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 \\ -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 2 & 0 \\ 0 & -1 & 0 & -1 & 2 \end{bmatrix}.$$

En otras palabras, la matriz de adyacencias es una matriz cuadrada de dimensión N (número de agentes o nodos), la cual tiene ceros en su diagonal, los valores restantes de la matriz tienen un valor positivo si el agente i recibe información del agente j , es decir, $(i, j) \in \mathcal{E}$; donde el agente i representa las filas de la matriz y el agente j representa las columnas de la matriz. Por otro lado, la matriz Laplaciana consta de una diagonal diferente de cero, cuyo valor es la suma de la cantidad de agentes que se comunican con el agente i , es decir, $\mathcal{L}_{ii} \neq 0$ y $\mathcal{L}_{ii} = a_{i1} + a_{i2} + \dots + a_{ij}$. Los valores restantes de la matriz Laplaciana son los mismos que los de la matriz de adyacencias con el signo cambiado cuando $i \neq j$, es decir, $\mathcal{L}_{ij} = -a_{ij}$ cuando $i \neq j$.

3.1.1. Topología de comunicación del sistema multiagente

De manera particular, el caso abordado en esta sección se compone de una flota de 3 UAVs homogéneos, es decir, de parámetros idénticos. El sistema es controlado de manera descentralizada y la comunicación entre sus agentes está descrita por una topología de comunicación (ver figura 3.4). Es importante resaltar que este ejemplo es adaptable a distintos tipos de agente siempre y cuando sean homogéneos. Además, la complejidad de su adaptación depende del tipo de agente seleccionado.

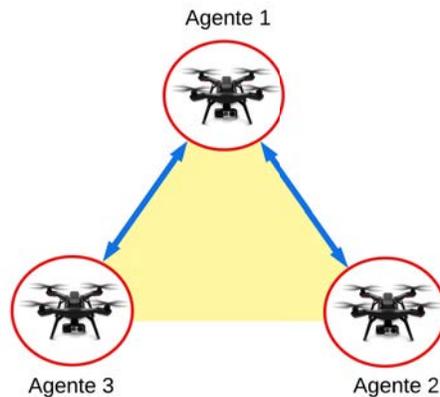


Figura 3.4: Topología de comunicación sin líder seguidor para un sistema de 3 agentes.

Con la topología de comunicación definida es posible obtener las matrices de información que se requieren para obtener las variables generadas por el protocolo de consenso. Por

lo tanto, las matrices de adyacencias (3.1) y Laplaciana (3.2) son las siguientes:

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad (3.1)$$

$$\mathcal{L} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \quad (3.2)$$

donde se puede observar que los agentes 2 y 3 intercambian información en ambos sentidos con el agente 1.

3.2. Distancia relativa entre agentes

Para alcanzar el control de formación el protocolo de consenso necesita una matriz que cuente con las posiciones relativas finales del sistema; esta matriz H restringe la posición de los agentes cuando estos alcanzan la formación. La forma de la matriz H está dada por la siguiente expresión:

$$H = [h_1, h_2, \dots, h_N],$$

donde cada vector h_i contiene las distancias relativas entre los agentes del sistema, con respecto al agente de referencia (agente 1 o agente líder); por ejemplo, el vector h_1 se define como: $h_1 = [X_1, Y_1, Z_1, 0, 0, 0]^T$. Por lo tanto, la matriz H para un sistema multiagente con N número de agentes es representada de la siguiente manera:

$$H = \begin{bmatrix} X_1 & X_2 & \cdots & X_N \\ Y_1 & Y_2 & \cdots & Y_N \\ Z_1 & Z_2 & \cdots & Z_N \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \end{bmatrix} \quad (3.3)$$

donde cada columna de la matriz H representa cada agente i , y cada vector h_i son los estados deseados en la formación final, donde X_i , Y_i y Z_i son las posiciones en los ejes x , y y z ; nótese que la matriz H debe tener las mismas columnas que agentes del sistema (N), así como puede tener el mismo número de filas que de estados del protocolo de consenso. Por ejemplo, la figura 3.5 muestra la distribución de los agentes en una formación hexagonal en los ejes xy :

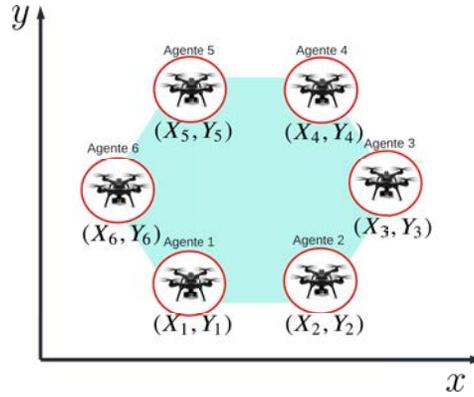


Figura 3.5: Formación hexagonal de 6 agentes en el marco de referencia xy .

Por lo tanto, la matriz de distancias relativas H de la formación en la figura 3.5 es la siguiente:

$$H = \begin{bmatrix} X_1 & X_2 & X_3 & X_4 & X_5 & X_6 \\ Y_1 & Y_2 & Y_3 & Y_4 & Y_5 & Y_6 \\ Z_1 & Z_2 & Z_3 & Z_4 & Z_5 & Z_6 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

3.2.1. Matriz de distancias relativas del sistema multiagente

Para un sistema multiagente compuesto por 3 agentes, para el cual se busca alcanzar una formación triangular, como se observa en la figura 3.4. La matriz de distancias relativas que limita la formación del sistema multiagente:

$$H = \begin{bmatrix} 0 & 3 & 1,5 \\ 0 & 0 & 3 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (3.4)$$

donde el agente 1 es el agente de referencia; el agente 2 se encuentra a 3 metros en el eje x del agente 1 y el agente 3 se encuentra a 1.5 metros en el eje x y a 3 metros en el eje y del agente 1.

3.3. Protocolo de consenso basado en observador

Considérese un sistema de N agentes, el cual se representa en espacio de estados de la siguiente manera:

$$\begin{aligned}\dot{X}_i &= AX_i + BU_i \\ \mathcal{Y}_i &= CX_i, \quad i = 1, 2, \dots, N.\end{aligned}$$

El sistema está compuesto por N dobles integradores, donde x_i es el vector de estados, \mathcal{Y}_i es el vector de salidas, u_i es el vector de entradas, el índice i representa el agente y las matrices A , B y C son las matrices del doble integrador, las cuales son las siguientes:

$$A = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}. \quad (3.5)$$

La comunicación del sistema entre cada agente está dada por la topología de la figura 3.4, el cual es un grafo directo compuesto de un conjunto de nodos (agentes) y un conjunto de bordes (enlaces de comunicación). En cada instante de tiempo, la información disponible del agente i es la medición relativa de otro agente j con respecto a sí mismo, esta medición está dada por la siguiente ecuación:

$$\zeta_i = \sum_{j=1}^N a_{ij}(\mathcal{Y}_i - \mathcal{Y}_j),$$

donde a_{ij} es el valor posicionado en la fila i y la columna j de la matriz de adyacencias A . En [80] se propone el siguiente protocolo de consenso basado en observador:

$$\begin{aligned}\dot{r}_i &= (A + BF)r_i + L \left(\sum_{j=1}^N a_{ij}C(r_i - r_j) - \zeta_i \right), \\ u_i &= Fr_i.\end{aligned}$$

Sustituyendo ζ_i en \dot{r}_i y agregando el control de formación al protocolo propuesto, se obtiene la siguiente ecuación:

$$\begin{aligned}\dot{r}_i &= (A + BF)r_i + L \left(\sum_{j=1}^N a_{ij} [C(r_i - r_j) - (\mathcal{Y}_i - \mathcal{Y}_j - C(h_i - h_j))] \right), \\ u_i &= Fr_i,\end{aligned} \quad (3.6)$$

donde, r es el vector de estados del protocolo de consenso, \mathcal{Y} es el vector de salidas del sistema, h es el vector de posiciones relativas de la matriz H , u es el vector de entradas de control generadas por el protocolo de consenso, i representa al agente actual, j

representa a los agentes vecinos, L y K son las ganancias del protocolo de consenso y A , B y C son las matrices del doble integrador. Como se mencionó, el protocolo de consenso (3.6) fue planteado a partir de la representación de cada agente como doble integrador, por lo tanto, el vector u_i también representa a las entradas del doble integrador, las cuales son 3 aceleraciones traslacionales. Las entradas del doble integrador x , y y z deben ser transformadas a las entradas del modelo Newton-Euler (2.2), las cuales son los pares rotacionales en los ángulos ϕ , θ y ψ y la fuerza de empuje en el eje z (*thrust*); obteniendo así, los ángulos de referencia $\Phi_r = [\phi_r, \theta_r, \psi_r]^T$. Las transformaciones fueron obtenidas considerando $\psi = 0$ en la parte traslacional del modelo no lineal del UAV, obteniendo así, las siguientes expresiones:

$$\begin{aligned}\ddot{x} &= (\cos\phi\text{sen}\theta)\frac{T}{m}, \\ \ddot{y} &= (-\text{sen}\phi)\frac{T}{m}, \\ \ddot{z} &= (\cos\theta\cos\phi)\frac{T}{m} - g,\end{aligned}$$

estas expresiones son aceleraciones en los ejes x , y y z ; estas aceleraciones se redefinen como $\ddot{x} := \tilde{u}_1$, $\ddot{y} := \tilde{u}_2$ y $\ddot{z} := \tilde{u}_3$. Para obtener la primera transformación, se relacionan \tilde{u}_1 y \tilde{u}_3 y se igualan para obtener el ángulo de referencia θ_r :

$$\begin{aligned}\frac{\tilde{u}_1}{\tilde{u}_3 + g} &= \frac{T \cdot m(\cos\phi_r\text{sen}\theta_r)}{T \cdot m(\cos\theta_r\cos\phi_r)} = \frac{\text{sen}\theta_r}{\cos\theta_r} = \tan\theta_r, \\ \theta_r &= \arctan\left(\frac{\tilde{u}_1}{\tilde{u}_3 + g}\right).\end{aligned}$$

Posteriormente, se suman \tilde{u}_1 y \tilde{u}_2 , se elevan al cuadrado y se desarrollan hasta obtener la expresión (3.7):

$$\begin{aligned}\tilde{u}_1^2 + (\tilde{u}_3 + g)^2 &= \left(\frac{T}{m}\right)^2 \cos^2\phi_r\text{sen}^2\theta_r + \left(\frac{T}{m}\right)^2 \cos^2\phi_r\cos^2\theta_r, \\ \tilde{u}_1^2 + (\tilde{u}_3 + g)^2 &= \left(\frac{T}{m}\cos\phi_r\right)^2 (\text{sen}^2\theta_r + \cos^2\theta_r), \\ \frac{T}{m}\cos\phi_r &= \sqrt{\tilde{u}_1^2 + (\tilde{u}_3 + g)^2},\end{aligned}\tag{3.7}$$

finalmente, se relacionan las ecuaciones de \tilde{u}_2 y (3.7) de la siguiente manera con la finalidad de obtener el ángulo de referencia ψ_r :

$$\begin{aligned}\frac{T \cdot m(-\text{sen}\phi_r)}{T \cdot m(\cos\phi_r)} &= \frac{\tilde{u}_2}{\sqrt{\tilde{u}_1^2 + (\tilde{u}_3 + g)^2}}, \\ \tan\phi_r &= -\frac{\tilde{u}_2}{\sqrt{\tilde{u}_1^2 + (\tilde{u}_3 + g)^2}}, \\ \phi_r &= \arctan\left(-\frac{\tilde{u}_2}{\sqrt{\tilde{u}_1^2 + (\tilde{u}_3 + g)^2}}\right).\end{aligned}$$

Estas transformaciones se han utilizado anteriormente y están definidas en las ecuaciones (2.3).

3.3.1. Ganancias del protocolo de consenso

Para obtener la ganancia del observador del protocolo de consenso se resolvieron las siguientes LMIs [15]:

$$\begin{aligned} A^T Q + Q A - 2C^T C + 2\alpha Q &< 0, \\ Q &> 0, \end{aligned}$$

donde Q es una matriz simétrica y definida positiva de dimensión $N \times N$, $\alpha = 1$, $L = -Q^{-1}C^T$ y las matrices A y C son matrices del doble integrador (3.5). Posteriormente, se utilizó la colocación de polos para obtener la ganancia K del protocolo de consenso, dichos polos se cambiaron por el siguiente vector $\tilde{p} = [-1 \ -1 \ -1 \ -1,5 \ -1,5 \ -1,5]$, utilizando la nueva ubicación de los polos y las matrices A , B y C del doble integrador (3.5). Con los métodos de cálculo de ganancias mencionados, las ganancias obtenidas para el protocolo de consenso son las siguientes:

$$L = \begin{bmatrix} -3,6968 & 0 & 0 \\ 0 & -3,6968 & 0 \\ 0 & 0 & -3,6968 \\ -4,9486 & 0 & 0 \\ 0 & -4,9486 & 0 \\ 0 & 0 & -4,9486 \end{bmatrix}, \quad (3.8)$$

$$F = \begin{bmatrix} -1,5 & 0 & 0 & -2,5 & 0 & 0 \\ 0 & -1,5 & 0 & 0 & -2,5 & 0 \\ 0 & 0 & -1,5 & 0 & 0 & -2,5 \end{bmatrix}, \quad (3.9)$$

con las ganancias F y L el protocolo de consenso (3.6) es capaz de alcanzar el consenso y la formación predefinida, sin embargo, debido a que el diseño de las ganancias del protocolo de consenso se realizó a partir de un doble integrador, las entradas de control calculadas por este protocolo son utilizadas como referencias, las cuales deben ser seguidas por un control a posteriori, el cual sea capaz de controlar la dinámica rotacional de cada UAV.

3.4. Control interno del sistema multiagente

Para controlar cada agente de manera individual, es necesario realizar un control para cada UAV. Para este sistema, se diseñó un control por retroalimentación de estados con acción proporcional-integral en la sección 2.2.1, donde se controla la parte rotacional del modelo simplificado de un UAV (2.5) por medio de la ley de control (2.9), dicho control se obtiene a partir de aumentar el espacio de estados (2.6), el sistema aumentado es el

siguiente:

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\psi} \\ \ddot{\psi} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \\ \theta \\ \dot{\theta} \\ \psi \\ \dot{\psi} \\ \tilde{\phi} \\ \tilde{\theta} \\ \tilde{\psi} \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 1/J_x & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1/J_y & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1/J_z \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \bar{U}_\Phi. \quad (3.10)$$

Donde las ganancias de la siguiente ley de control:

$$\bar{U}_\Phi = K_I \int_0^t (\Phi_r - \Phi) dt - K_P \Phi,$$

son calculadas a partir de la colocación de polos utilizando las matrices del sistema (3.10). Las ganancias (2.12) y (2.13) son las ganancias mencionadas en la sección 2.2. Finalmente, al obtener estas ganancias es posible aplicar el control al sistema multi-agente y observar sus resultados.

3.5. Discusión de resultados (conclusión del capítulo)

El control de formación basado en consenso sin líder seguidor fue simulado incluyendo el sistema de propulsión, además, el sistema multiagente se simuló durante 20 segundos y con un tiempo de muestreo (T_s) de 0.001 segundo por iteración; sin embargo, como se puede observar en la diagrama de flujo en la figura A.4. Algunas de las ecuaciones diferenciales mostradas en esta sección fueron resueltas por medio de la función ODE45 de Matlab el cual tiene un paso de integración variable, por otro lado, ciertas ecuaciones diferenciales, como la integral del error o el algoritmo de consenso se resolvieron por medio del método numérico de integración de Euler, el cual utiliza un paso de integración fijo.

La formación del sistema multiagente se puede observar en la figura 3.4, las matrices de adyacencias y Laplaciana del sistema están definidas en (3.1) y (3.2) respectivamente y la matriz de distancias relativas que restringe la formación final del sistema se observa en (3.4). Las posiciones iniciales en los ejes x , y y z de cada agente del sistema son $A_1(0) = [1 \ 2 \ 0]^T$, $A_2(0) = [2 \ 1 \ 0]^T$ y $A_3(0) = [3 \ 4 \ 0]^T$. El control (ver secciones 3.3.1 y 3.4) para el sistema calcula las leyes de control, las cuales son transformadas por (2.3) alimentan al modelo matemático de los agentes del sistema, siendo la siguiente figura la que muestra el comportamiento que alcanza el sistema al realizar una formación triangular:

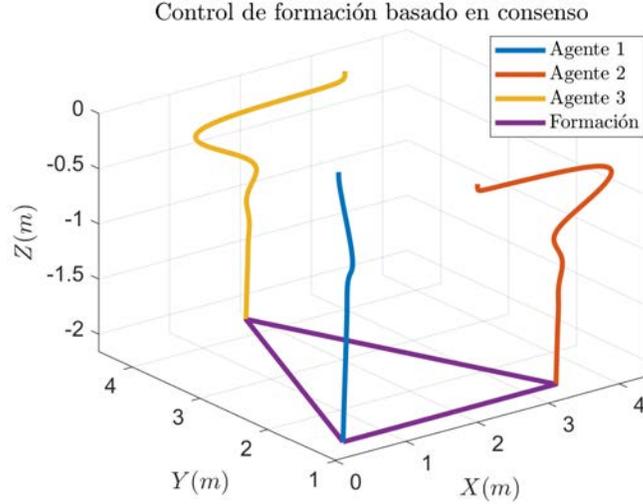


Figura 3.6: Control de formación basado en consenso sin líder seguidor.

En la figura 3.6 se observa que el sistema multiagente alcanza la formación predefinida, sin embargo, también se observa que la formación se alcanza en valores negativos de z , es decir, el control diseñado alcanza el control de formación basado en consenso obteniendo valores no deseados. Es importante resaltar los ajustes de parámetros que se realizaron, esto debido a la falta de parámetros para la matriz Γ (2.16), la cual contiene los parámetros de empuje, arrastre y la longitud de los brazos de un UAV (b , d y l) tomados de [49], además, dicha matriz fue adaptada de manera empírica para obtener los resultados en simulación de esta sección. En la figura 3.7 se observa el error de formación entre los agentes el cual se define como [39]: $d_{ij} = \|(p_i - p_j) - (h_i - h_j)\|$, donde si $d_{ij} = 0$ significa que el sistema multiagente alcanzó la formación. Nótese que el error de formación alcanza el valor de 0 a los 12 segundos aproximadamente, es decir, el sistema realizó la formación deseada a los 12 segundos de simulación.

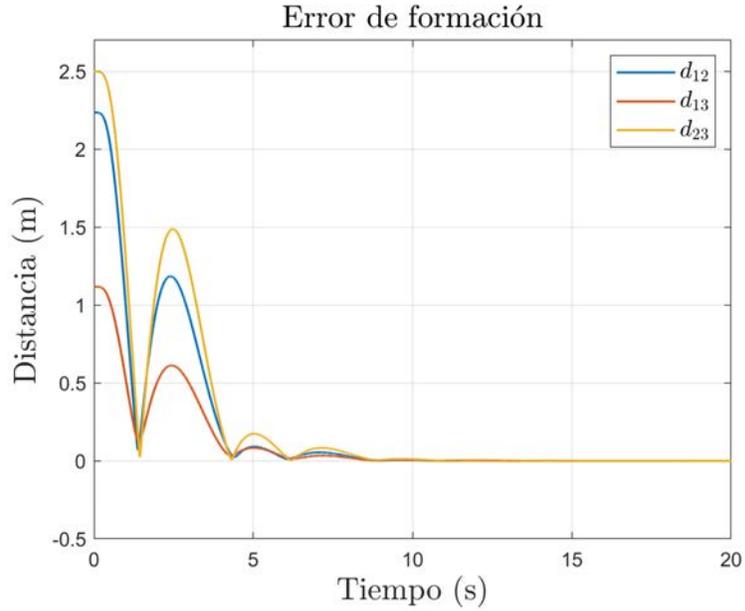


Figura 3.7: Error de formación del sistema multiagente.

La simulación del control de formación basado en consenso sin líder seguidor fue parcialmente exitosa, se observa la formación deseada como resultado final, sin embargo, el trayecto recorrido por los agentes del sistema tiende a decrecer, a pesar de tener el comportamiento que se desea. Por esta razón, el control de formación sin líder seguidor no fue considerado apto para recorrer trayectorias predefinidas debido a que los agentes no se elevan, por lo tanto, posteriormente, se diseñó un control de formación con líder seguidor para buscar la elevación del sistema y la posibilidad de recorrer trayectorias de manera adecuada para el análisis del sistema de propulsión de cada agente.

Capítulo 4

Control de formación líder seguidor basado en consenso afectado por el sistema de propulsión

En esta sección, se diseñó un control de formación aplicado a una flota de UAVs; el control mencionado se diseñó a partir de las técnicas de control presentadas en los artículos [15], [37] y [81]. Se utilizó un protocolo de consenso líder seguidor basado en observador para alcanzar el consenso [82], las leyes de control calculadas por el protocolo de consenso son transformadas, esto debido a la integración de los modelos de la batería y los motores (figura 4.1). Al integrar el sistema de propulsión al esquema general del control de formación para el sistema multiagente se hace posible la obtención de información del estado de carga de la batería, posteriormente, se analizará el comportamiento del sistema multiagente al recorrer trayectorias.

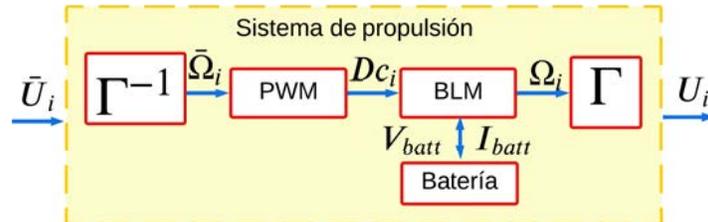


Figura 4.1: Diagrama de bloques del sistema de propulsión de un UAV.

Para alcanzar el control de formación se plantea el siguiente diagrama (figura 4.2), el cual integra los diagramas mostrados en las figuras 1.9 y 4.1:

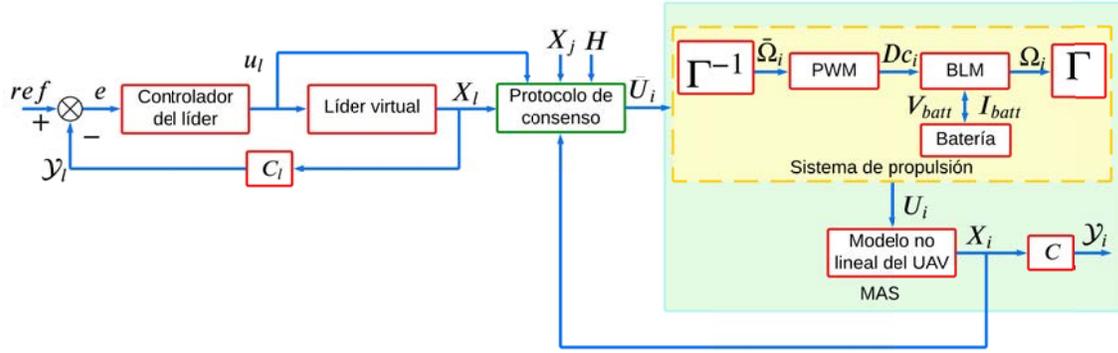


Figura 4.2: Control de formación con la dinámica de la batería y los motores.

En las siguientes secciones se muestra el desarrollo para simular el diagrama de control de la figura 4.2, donde la parte del líder virtual se desarrolla en la sección 4.1, así como las ganancias y los parámetros del protocolo de consenso para controlar al sistema multiagente se definen y se calculan en la sección 4.2.1. Además, en la sección 4.3 se muestran los ejemplos numéricos a los que el sistema multiagente fue sometido, donde se analiza el comportamiento de las leyes de control y el sistema de propulsión a lo largo del tiempo. Finalmente, la sección 4.4 muestra las conclusiones y las diferencias entre los casos de simulación a los que se sometió el sistema.

4.1. Agente líder

Como se ha mencionado anteriormente, los sistemas multiagente pueden tener o no tener un líder, el cual cumple la función de posicionar en el eje z al sistema entero. Este agente líder puede ser representado como un sistema dinámico únicamente con movimiento traslacional, así como también puede ser representado como un agente con dinámica traslacional y rotacional, donde sería necesario controlar estos comportamientos; dicho esto, para este trabajo se utiliza un agente líder representado por el modelo dinámico de Newton-Euler (2.2), el cual es controlado por un control PI-D para realizar formaciones a una altura definida o trayectorias. El esquema de control que se planea implementar es el siguiente:

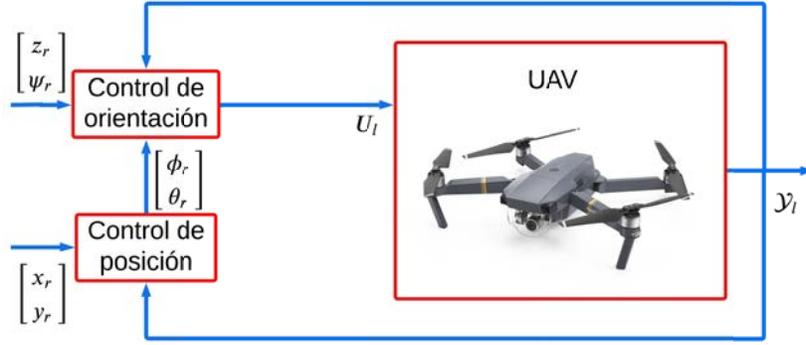


Figura 4.3: Esquema de control para el líder seguidor.

En la figura 4.3 se observa el esquema diseñado para controlar al líder del sistema multiagente, en esta figura se observa un control de posición, el cual genera ángulos de referencia para el controlador de orientación, el cual se encarga de calcular las entradas de control que suministran al agente líder.

4.1.1. Control PI-D

A continuación, el control PID clásico se modifica para obtener un funcionamiento adecuado, el control PI-D o PID modificado tiene el siguiente esquema de control:

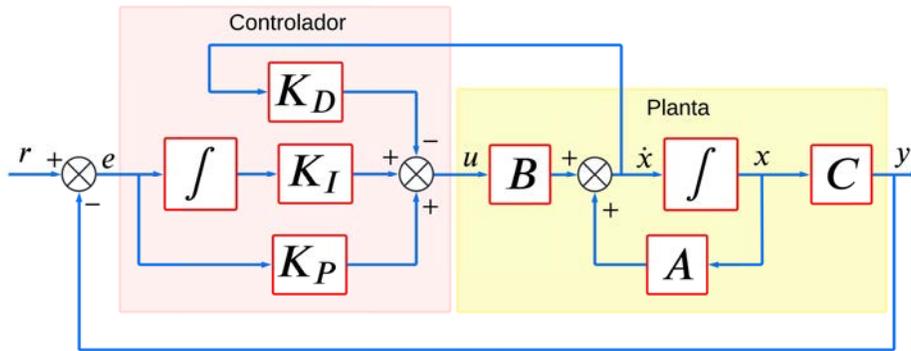


Figura 4.4: Diagrama de bloques de un control PI-D o PID modificado.

donde la diferencia de un control PID clásico, el error referencia-salida no fue sometido a una acción derivativa, siendo la velocidad del sistema la que multiplica a la ganancia derivativa del controlador como se observa en la figura 4.4. Sin embargo, el control PI-D se sintonizó como un control PID. La sintonización se realizó dividiendo el modelo lineal de un UAV (2.5) en 6 subsistemas. Los subsistemas de la parte traslacional del modelo lineal de un UAV son los siguientes:

$$\begin{bmatrix} \dot{x} \\ \dot{\ddot{x}} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ g \end{bmatrix} \theta_r, \quad (4.1)$$

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \end{bmatrix} + \begin{bmatrix} 0 \\ -g \end{bmatrix} \phi_r, \quad (4.2)$$

$$\begin{bmatrix} \dot{z} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z \\ \dot{z} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} T + \begin{bmatrix} 0 \\ -g \end{bmatrix}, \quad (4.3)$$

y los subsistemas de la parte rotacional son los siguientes:

$$\begin{bmatrix} \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J_x} \end{bmatrix} R, \quad (4.4)$$

$$\begin{bmatrix} \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J_y} \end{bmatrix} P, \quad (4.5)$$

$$\begin{bmatrix} \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{J_z} \end{bmatrix} Y. \quad (4.6)$$

Es posible observar en los subsistemas del modelo lineal que los subsistemas (4.1) y (4.2) dependen de la rotación del UAV en los ángulos θ y ϕ respectivamente, es decir, para mover el UAV en los ejes x y y los ángulos θ y ϕ calculados son los ángulos deseados que los subsistemas (4.4) y (4.5) utilizan como referencias. Finalmente, se diseñó un controlador para cada subsistema, el esquema general para controlar al líder virtual del sistema multiagente es el siguiente:

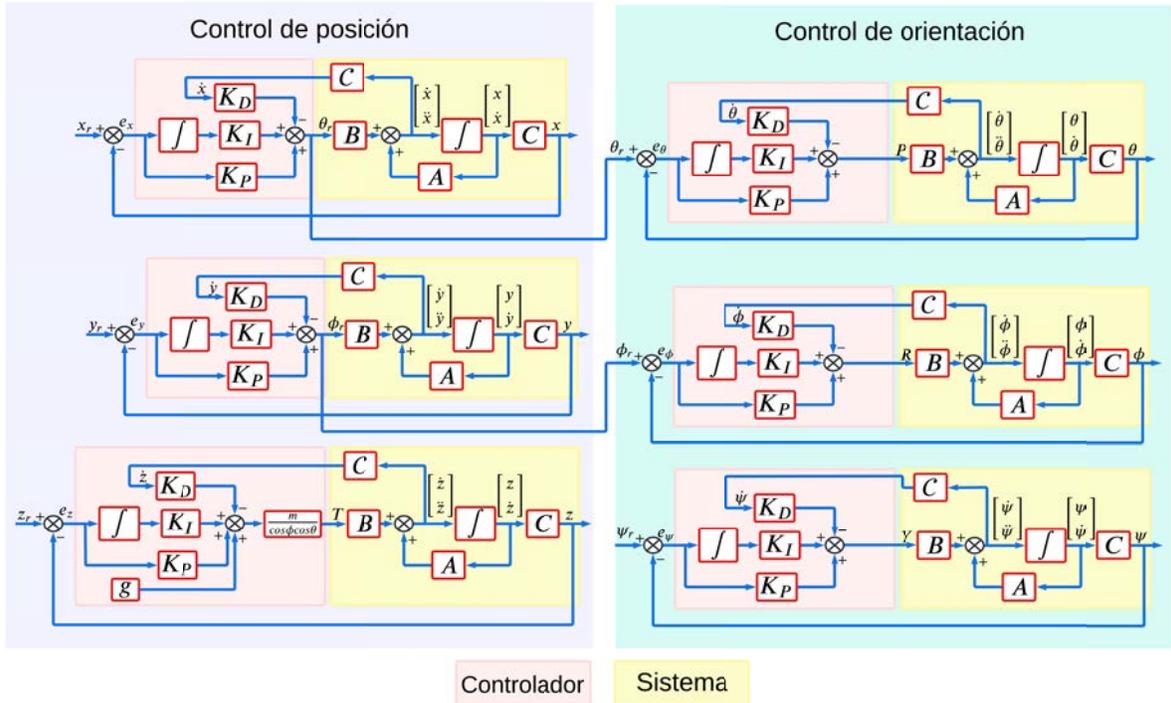


Figura 4.5: Controladores PI-D para el líder virtual.

El control de esta sección se aplicó basándose en [27] y [72], cuyas ganancias fueron sintonizadas de manera heurística, es decir, las ganancias fueron ajustadas hasta encontrar

el comportamiento deseado. Las ganancias mostradas en la tabla 4.1 son las utilizadas para la simulación.

Ganancia	Subsistema	x	y	z	ϕ	θ	ψ
K_P		0,8	-1,2	28	20	6	5
K_I		0,5	-1	3,5	0	0,1	0,1
K_D		0,3673	-0,8	10	0,6	0,75	2

Tabla 4.1: Ganancias del control PI-D.

Para probar el control del líder seguidor se definió una trayectoria de referencia moviendo al agente 5 metros en el eje z , 4 metros en el eje y y 7 metros en el eje x , las posiciones del agente líder en los ejes son las siguientes:

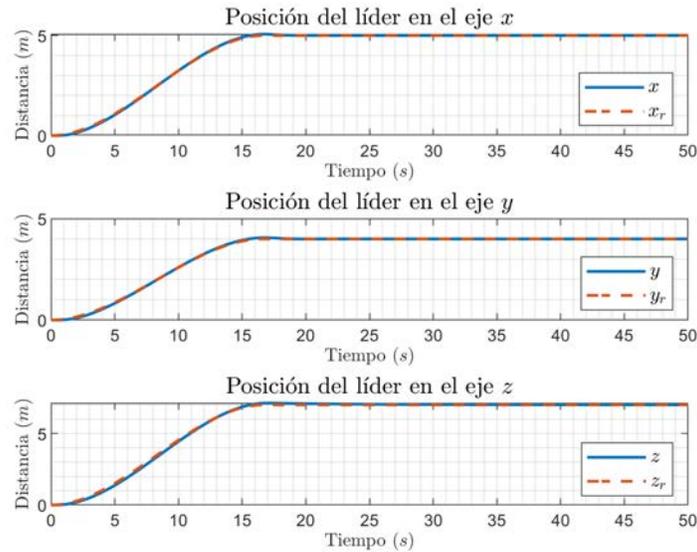


Figura 4.6: Comportamiento de las posiciones del líder al seguir una trayectoria.

En la figura 4.6 se observa que la posición en x y y tienen un error mínimo respecto a la referencia. Sin embargo, se planea probar el control diseñado con una referencia que represente una tarea de mayor dificultad, por lo tanto, se generó una trayectoria donde el agente líder debe elevarse 100 metros, y recorrer una trayectoria en forma de cuadrado con una distancia de 50 metros por lado durante 100 segundos, como se observa en la figura 4.7.

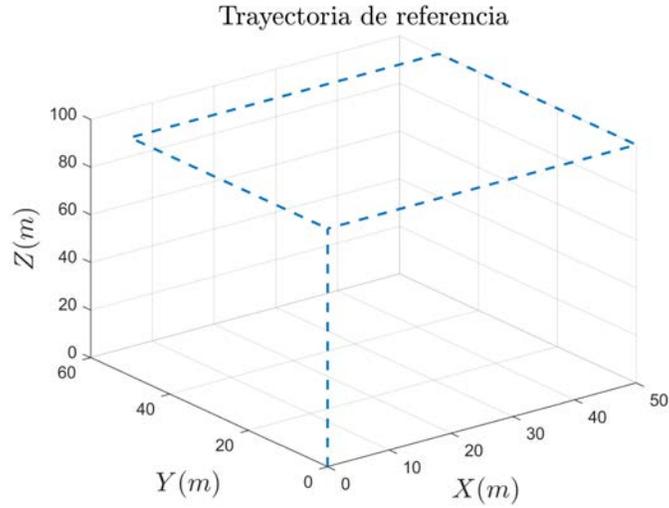


Figura 4.7: Trayectoria a seguir del líder seguidor en los ejes xyz .

Las posiciones del agente líder a lo largo del tiempo se muestran en la siguiente figura:

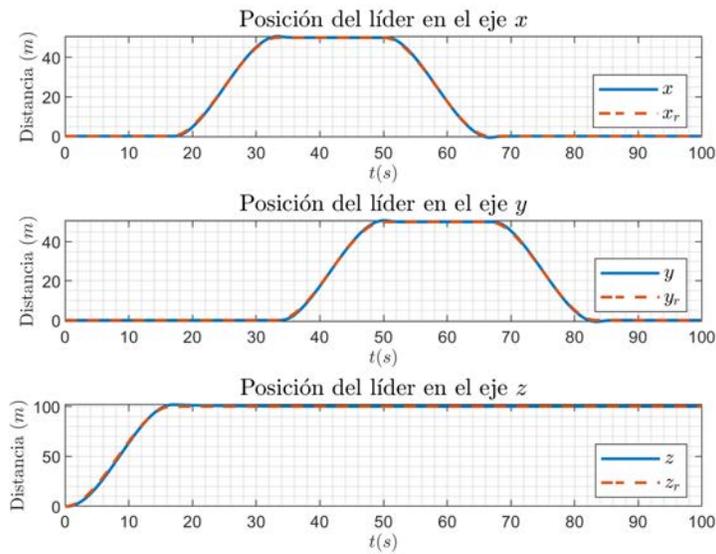


Figura 4.8: Trayectoria recorridas por el líder a lo largo del tiempo de simulación.

En la figura 4.8 se observa que el error es muy pequeño; de igual manera, se graficaron los datos del agente y su referencia en los ejes x , y y z :

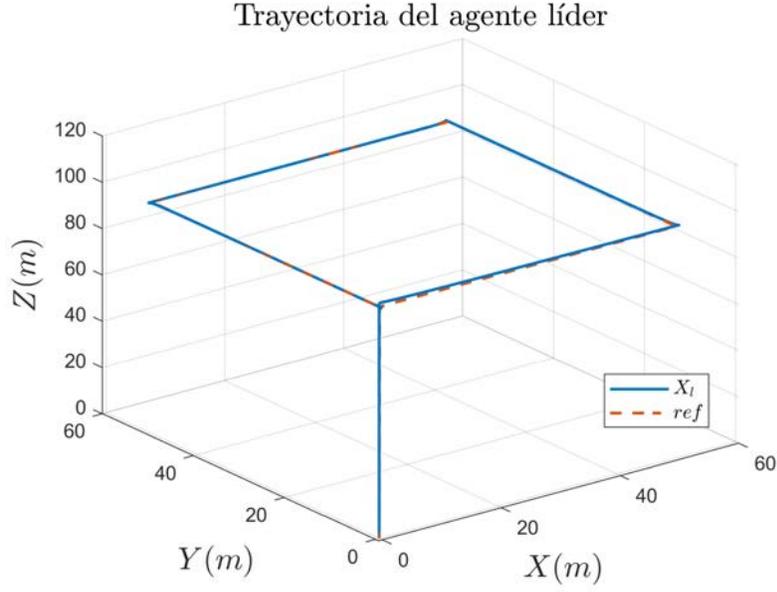


Figura 4.9: Trayectoria recorrida por el líder seguidor.

En la figura 4.9 se observa que el error es bastante menor, y la trayectoria se cumple de manera adecuada y suficiente para el sistema multiagente que se planteará en las secciones posteriores.

4.2. Protocolo de consenso

Considérese una red de N agentes homogéneos, la dinámica de los agentes puede ser descrita como:

$$\begin{aligned} \dot{X}_i &= AX_i + BU_i \\ \mathcal{Y}_i &= CX_i, \quad i = 1, 2, \dots, N. \end{aligned} \tag{4.7}$$

donde X_i es el vector de estados, U_i es el vector de las leyes de control y \mathcal{Y}_i es el vector de salidas del sistema, así como A , B y C son las siguientes matrices del modelo lineal

(2.5):

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & g & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -g & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1/m & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1/J_x & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1/J_y & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1/J_z \end{bmatrix},$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \quad (4.8)$$

El consenso está definido como un estado de convenio para un conjunto de agentes, los cuales alcanzan un objetivo en común. Diferentes protocolos de comunicación han sido desarrollados para alcanzar el consenso en sistemas multiagente, como se mencionó previamente, en este trabajo se utilizó la forma clásica de un protocolo de consenso basado en observador con líder seguidor [30]:

$$u_i = K \left[\sum_{j=1}^N a_{ij}(\hat{x}_i - \hat{x}_j) + \alpha_i(\hat{x}_i - X_l) \right] + u_l. \quad (4.9)$$

Al no contemplarse un observador en este trabajo, el protocolo de consenso usa los estados de cada agente, así como también se incluye la extensión para el control de formación en este. El protocolo de consenso modificado es el siguiente [83]:

$$u_i = K \left[\sum_{j=1}^N a_{ij}(\mathcal{S}_i - \mathcal{S}_j) + \alpha_i(\mathcal{S}_i - X_l) \right] + u_l, \quad (4.10)$$

donde $\mathcal{S}_i = X_i - h_i$, $\mathcal{S}_j = X_j - h_j$ y $H = [h_1, h_2, \dots, h_N]^T$. También, U_i es el vector de las leyes de control de cada agente, U_l es el vector de las leyes de control del agente líder, K es la matriz de las ganancias, a_{ij} es el enlace de comunicación entre el i -ésimo agente con el j -ésimo agente, α_i es el enlace de comunicación entre el líder y los seguidores; si existe comunicación entre el agente i con el líder entonces $\alpha_i = 1$, de lo contrario, $\alpha_i = 0$, i representa cada agente, j es el agente vecino, N es el número de agentes.

4.2.1. Análisis de estabilidad basado en Lyapunov

Considérese el siguiente espacio de estados:

$$\begin{aligned}\dot{x}_i &= Ax_i + Bu_i \\ y_i &= Cx_i, \quad i = 1, 2, \dots, N.\end{aligned}\tag{4.11}$$

donde x_i es el vector de estados, u_i es el vector de leyes de control y y_i es el vector de salidas, así como A , B y C son las matrices del modelo lineal de un UAV (2.5). Además, la dinámica del líder está dada por la siguiente ecuación:

$$\dot{x}_l = Ax_l + Bu_l,\tag{4.12}$$

el error de sincronización es descrito por:

$$\delta_i = x_i - x_l;\tag{4.13}$$

la dinámica del error de sincronización es descrita por:

$$\begin{aligned}\dot{\delta}_i &= \dot{x}_i - \dot{x}_l, \\ \dot{\delta}_i &= Ax_i + Bu_i - (Ax_l + Bu_l), \\ \dot{\delta}_i &= A\delta_i + B(u_i - u_l).\end{aligned}\tag{4.14}$$

Considérese un observador de orden completo (Luenberger) descrito por las siguientes ecuaciones:

$$\begin{aligned}\dot{\hat{x}}_i &= A\hat{x}_i + Bu_i + Le_i, \\ \hat{y}_i &= C\hat{x}_i, \\ e_i &= x_i - \hat{x}_i,\end{aligned}\tag{4.15}$$

donde \hat{x}_i es el vector de estados estimados, \hat{y}_i es el vector de salidas estimadas, e_i es el vector de errores de estimación, L es la ganancia del observador y A , B y C son las matrices del sistema lineal de un UAV (2.5). La dinámica del error está dada por:

$$\begin{aligned}\dot{e}_i &= \dot{x}_i - \dot{\hat{x}}_i, \\ \dot{e}_i &= Ax_i + Bu_i - [A\hat{x}_i + Bu_i + L(Cx_i - C\hat{x}_i)], \\ \dot{e}_i &= A(x_i - \hat{x}_i) - LC(x_i - \hat{x}_i), \\ \dot{e}_i &= (A - LC)e_i,\end{aligned}\tag{4.16}$$

Las ecuaciones (4.14) y (4.16) pueden ser representadas usando el producto Kronecker de la siguiente manera:

$$\dot{e} = (I_N \otimes (A - LC))e,\tag{4.17}$$

$$\dot{\delta} = (I_N \otimes A + \bar{\mathcal{L}} \otimes BK)\delta - (\bar{\mathcal{L}} \otimes BK)e,\tag{4.18}$$

donde, $\bar{\mathcal{L}} = \mathcal{L} + \Lambda$, $\Lambda = \text{diag}(\alpha_1, \alpha_2, \dots, \alpha_N)$, I_N es una matriz identidad de dimensión N y \otimes denota el producto Kronecker. El sistema multiagente en lazo cerrado es expresado por el siguiente sistema:

$$\dot{\xi} = \begin{bmatrix} I_N \otimes A + \bar{\mathcal{L}} \otimes BK & -\bar{\mathcal{L}} \otimes BK \\ 0 & I_N \otimes (A - LC) \end{bmatrix} \xi,\tag{4.19}$$

donde $\xi = [\delta^T \ e^T]^T$. Con la finalidad de analizar la estabilidad del error de estimación y el error de sincronización del sistema descrito en (4.19) se propuso una función candidata de Lyapunov. Dicha función está dada por la siguiente expresión:

$$V(\xi) = \xi^T \begin{bmatrix} I_N \otimes P_1 & 0 \\ 0 & I_N \otimes P_2 \end{bmatrix} \xi > 0, \quad (4.20)$$

donde P_1 y P_2 son matrices simétricas positivas definidas, es decir $P_1 = P_1^T > 0$ y $P_2 = P_2^T > 0$. El análisis de estabilidad de Lyapunov indica que la derivada de la función candidata debe ser negativa definida, es decir $\dot{V}(z) < 0$, por lo tanto, la derivada de $V(z)$ es la siguiente:

$$\dot{V}(\xi) = 2\xi^T \begin{bmatrix} I_N \otimes P_1 & 0 \\ 0 & I_N \otimes P_2 \end{bmatrix} \dot{\xi} < 0, \quad (4.21)$$

Sustituyendo el estado ξ y el sistema de la ecuación (4.19) en la derivada de la función candidata de Lyapunov en (4.21) se obtiene la siguiente ecuación:

$$\begin{aligned} \dot{V}(\xi) &= 2 \begin{bmatrix} \delta^T & e^T \end{bmatrix} \begin{bmatrix} I_N \otimes P_1 & 0 \\ 0 & I_N \otimes P_2 \end{bmatrix} \begin{bmatrix} \dot{\delta} \\ \dot{e} \end{bmatrix} < 0 \\ &= 2 \begin{bmatrix} \delta^T & e^T \end{bmatrix} \begin{bmatrix} I_N \otimes P_1 & 0 \\ 0 & I_N \otimes P_2 \end{bmatrix} \begin{bmatrix} I_N \otimes A + \bar{\mathcal{L}} \otimes BK & -\bar{\mathcal{L}} \otimes BK \\ 0 & I_N \otimes (A - LC) \end{bmatrix} \begin{bmatrix} \delta \\ e \end{bmatrix} < 0 \\ &= 2 \begin{bmatrix} \delta^T & e^T \end{bmatrix} \begin{bmatrix} (I_N \otimes P_1)(I_N \otimes A + \bar{\mathcal{L}} \otimes BK) & -(I_N \otimes P_1)(\bar{\mathcal{L}} \otimes BK) \\ 0 & (I_N \otimes P_2)(I_N \otimes (A - LC)) \end{bmatrix} \begin{bmatrix} \delta \\ e \end{bmatrix} < 0 \\ &= 2\delta^T (I_N \otimes P_1)(I_N \otimes A + \bar{\mathcal{L}} \otimes BK)\delta - 2\delta^T (I_N \otimes P_1)(\bar{\mathcal{L}} \otimes BK)e \dots \\ &\dots + 2e^T (I_N \otimes P_2)(I_N \otimes (A - LC))e < 0 \\ &= 2\delta^T (I_N \otimes P_1 A + \bar{\mathcal{L}} \otimes P_1 BK)\delta - 2\delta^T (\bar{\mathcal{L}} \otimes P_1 BK)e + 2e^T (I_N \otimes P_2 (A - LC))e < 0 \end{aligned} \quad (4.22)$$

Después, se realizó una descomposición espectral de la matriz $\bar{\mathcal{L}}$, tal que $\bar{\mathcal{L}} = T J T^{-1}$, donde T es una matriz invertible y $J = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_N)$ donde λ_j son los eigenvalores o valores propios de la matriz $\bar{\mathcal{L}}$. También, se aplicó el siguiente cambio de coordenadas tal que $\varphi = (T^{-1} \otimes I_N)\delta$ y $\phi = (T^{-1} \otimes I_N)e$, (4.22) puede ser reescrito como la siguiente ecuación:

$$\begin{aligned} \dot{V}(\xi) &= \sum_{j=1}^N \left[\varphi_j^T \text{He} \{ P_1 A + \lambda_j P_1 B K \} \varphi_j - 2\varphi_j^T \lambda_j P_1 B K \phi_j \right. \\ &\quad \left. + \phi_j^T \text{He} \{ P_2 A - P_2 L C \} \phi_j \right] < 0, \end{aligned} \quad (4.23)$$

considerando el vector $[\varphi_j^T \ \phi_j^T]^T$, la ecuación de (4.23) puede ser reescrita como:

$$\begin{aligned} \dot{V}(\xi) &= \sum_{j=1}^N \begin{bmatrix} \varphi_j^T & \phi_j^T \end{bmatrix} \vartheta \begin{bmatrix} \varphi_j \\ \phi_j \end{bmatrix} < 0, \\ \vartheta &= \begin{bmatrix} \text{He} \{ P_1 A + \lambda_j P_1 B K \} & -\lambda_j P_1 B K \\ * & \text{He} \{ P_2 A - P_2 L C \} \end{bmatrix} < 0, \end{aligned} \quad (4.24)$$

pre-multiplicando y post-multiplicando ϑ de la siguiente manera:

$$\begin{aligned}\Theta &= \begin{bmatrix} P_1^{-1} & 0 \\ 0 & I_N \end{bmatrix} \vartheta \begin{bmatrix} P_1^{-1} & 0 \\ 0 & I_N \end{bmatrix} < 0 \\ &= \begin{bmatrix} He\{AP_1^{-1} + \lambda_j BK P_1^{-1}\} & -\lambda_j BK P_1^{-1} \\ * & He\{P_2 A - P_2 LC\} \end{bmatrix} < 0\end{aligned}\quad (4.25)$$

es posible reescribir la matriz Θ de la siguiente manera:

$$\begin{bmatrix} \Theta_{11} & 0 \\ * & \Theta_{22} \end{bmatrix} + He \left\{ \begin{bmatrix} \Theta_{12} \\ 0 \end{bmatrix} \begin{bmatrix} 0 & I_N \end{bmatrix} \right\} < 0. \quad (4.26)$$

Aplicando la relación de Young ([84]) a la ecuación (4.26), la siguiente desigualdad es planteada:

$$\begin{bmatrix} \Theta_{11} & 0 \\ * & \Theta_{22} \end{bmatrix} + \mu \begin{bmatrix} \Theta_{12} \\ 0 \end{bmatrix} P_1 \begin{bmatrix} \Theta_{22}^T & 0 \end{bmatrix} + \mu^{-1} \begin{bmatrix} 0 \\ I_N \end{bmatrix} P_1^{-1} \begin{bmatrix} 0 & I_N \end{bmatrix} < 0, \quad (4.27)$$

después utilizando el complemento de Schur ([30]) a (4.27), la desigualdad anterior puede ser reescrita como:

$$\sum_{j=1}^N \begin{bmatrix} Q_1 & 0 & Q_3 & 0 \\ * & Q_2 & 0 & I_N \\ * & * & -\mu^{-1} \mathcal{P}_1 & 0 \\ * & * & * & -\mu \mathcal{P}_1 \end{bmatrix} < 0, \quad (4.28)$$

donde $Q_1 = He\{\mathcal{P}_1 A + \lambda_j BN\}$, $Q_2 = He\{P_2 A - \mathcal{M}C\}$, $Q_3 = -\lambda_j BN$, $\mathcal{P}_1 = P_1^{-1}$, $\mathcal{N} = K \mathcal{P}_1$, $\mathcal{M} = P_2 L$ y $*$ denota los elementos simétricos de la matriz (4.28).

Con el objetivo de obtener la ganancia K del protocolo de consenso (4.10), se resolvió la LMI obtenida en (4.28) por medio del *toolbox* Yalmip en Matlab. Las ganancias K y L del protocolo de consenso basado en observador líder seguidor son las siguientes:

$$K = \begin{bmatrix} 0 & -0,14 & 0 & -0,02 & -67,43 & -164,71 & -0,32 & -0,13 & -0,56 & -0,32 & 0,26 & 0,47 \\ 0 & 0 & 0,62 & 1,26 & 0 & 0,01 & -5,21 & -1,1 & 0,01 & 0 & 0 & 0 \\ -0,62 & -1,26 & 0 & 0 & 0 & 0 & -0,01 & 0 & -5,22 & -2,01 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0,01 & -0,01 & 0 & 0 & 0 & 0 & -1,53 & -3,74 \end{bmatrix} \quad (4.29)$$

$$L = \begin{bmatrix} 224,74 & -0,02 & -0,01 & -0,05 & 9,48 & 0 \\ 146,9 & 0,02 & 0,03 & -0,06 & 23,5 & -0,01 \\ -0,21 & 225,27 & 0 & -9,42 & -0,01 & -0,01 \\ -0,15 & 147,28 & 0,02 & -23,51 & -0,03 & -0,01 \\ 0,02 & 0,3 & 225,04 & 0,03 & 0 & 0 \\ 0,02 & 0,18 & 145 & 0,02 & 0,02 & 0,01 \\ 0,19 & 13,02 & -0,17 & 229,69 & 0,01 & -0,01 \\ 0,16 & 1,42 & -0,11 & 148,17 & 0,08 & -0,01 \\ -13,17 & 0,48 & 0,04 & 0,02 & 228,35 & -0,01 \\ -1,43 & 0,3 & 0,06 & 0,09 & 147,23 & 0,03 \\ -0,07 & -0,07 & 0,09 & 0,13 & 0,28 & 226,43 \\ -0,03 & -0,04 & 0,03 & 0,11 & 0,16 & 145,91 \end{bmatrix} \quad (4.30)$$

La ganancia L (4.30) se puede utilizar en un observador Luenberger, sin embargo, no se planteó el uso de un observador en este control de formación. Teniendo la matriz de ganancias K (4.29), es posible simular los sistemas planteados en esta sección y aplicar el control de formación que se busca satisfacer (ver figura 4.2).

4.3. Condiciones de simulación

El control de formación diseñado fue aplicado a una flota de 3 agentes homogéneos, es decir, con dinámica idéntica. La dinámica del líder virtual, así como los seguidores, están representadas por el modelo de Newton-Euler mostrado en (2.2). En la figura 4.10 la topología de comunicación del sistema multiagente es mostrada:

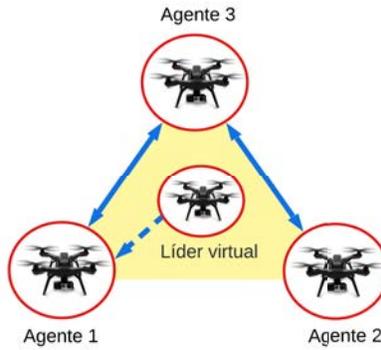


Figura 4.10: Topología de comunicación del sistema multiagente líder seguidor.

Además, la matriz de adyacencias, la matriz Laplaciana, la matriz de distancias relativas y la matriz de intercambio de información entre el líder y el resto del sistema son las siguientes respectivamente:

$$\mathcal{A} = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad (4.31)$$

$$\mathcal{L} = \begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}, \quad (4.32)$$

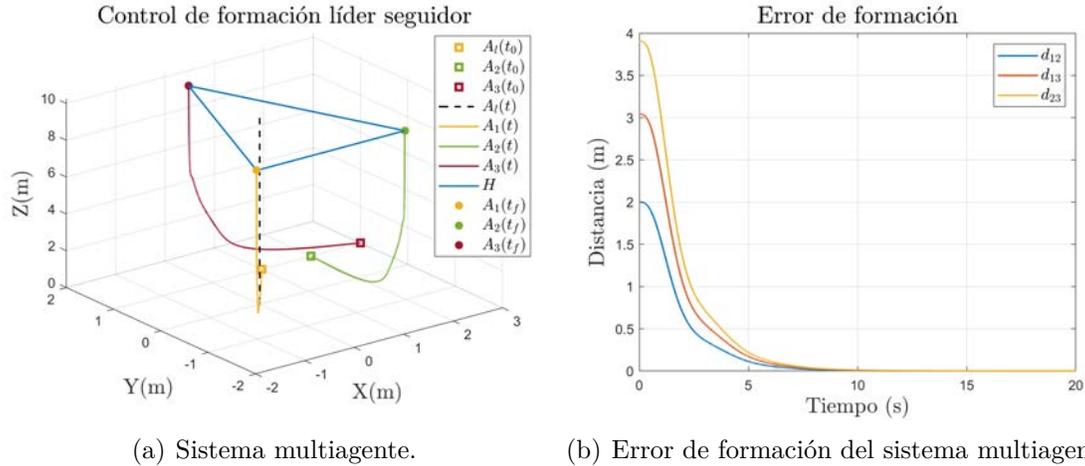
$$H = \begin{bmatrix} -1,5 & 1,5 & 0 \\ 0 & 0 & 0 \\ -1,5 & -1,5 & 1,5 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad (4.33)$$

$$\Lambda = \text{diag}(1, 0, 0), \quad (4.34)$$

dadas las matrices constantes, los parámetros están definidos en el Apéndice C. Finalmente, el sistema multiagente fue simulado en Matlab durante 100 segundos (T_{sim}), el tiempo de muestreo de la simulación fue 0.001 segundos (T_s), sin embargo, las distintas ecuaciones dinámicas fueron resueltas por medio de la instrucción que resuelve ecuaciones diferenciales por medio del método numérico ODE45 en Matlab, el cual trabaja con un paso de integración variable.

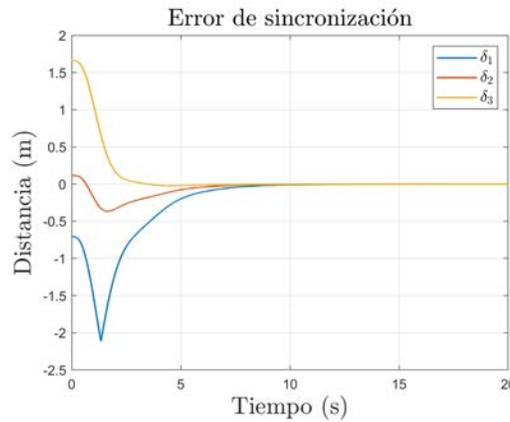
4.4. Discusión de resultados (conclusión del capítulo)

El sistema multiagente sometido al líder seguidor fue simulado con los mismos parámetros (apéndice C) que el control de formación sin líder seguidor; el comportamiento del sistema multiagente sometido al control de formación líder seguidor que busca alcanzar la formación predefinida en (4.33) es el siguiente (ver figura 4.11):



(a) Sistema multiagente.

(b) Error de formación del sistema multiagente.



(c) Error de sincronización del sistema multiagente.

Figura 4.11: Sistema multiagente bajo el control de formación basado en consenso líder seguidor y sus errores.

La figura 4.11(a) muestra que el sistema multiagente alcanza la formación definida anteriormente, las condiciones iniciales de cada agente son $A_l(0) = [0 \ 0 \ 0]^T$, $A_1(0) = [1 \ 1 \ 0]^T$, $A_2(0) = [2 \ 1 \ 0]^T$ y $A_3(0) = [3 \ 1 \ 0]^T$. Se observa que el sistema multiagente alcanza la formación predefinida (ver figura 4.11(b)), además, el error de sincronización ($\delta_i = \|x_i - x_l\| - \|h_i\|$) tiende a 0 (ver figura 4.11(c)), es decir, el sistema es capaz de seguir al líder virtual después de 9 segundos aproximadamente, el cual, únicamente se traslada a lo largo del eje z , alcanzando una altura de 10 metros, a diferencia del primer control de formación, el control de formación líder seguidor es capaz de recorrer una trayectoria de mayor complejidad. Posteriormente, se definieron dos misiones para el sistema multiagente (simulación A y simulación B), donde el líder virtual sigue una trayectoria a través del tiempo, las figuras 4.12 y 4.13 muestran la trayectoria de referencia que el líder recorre.

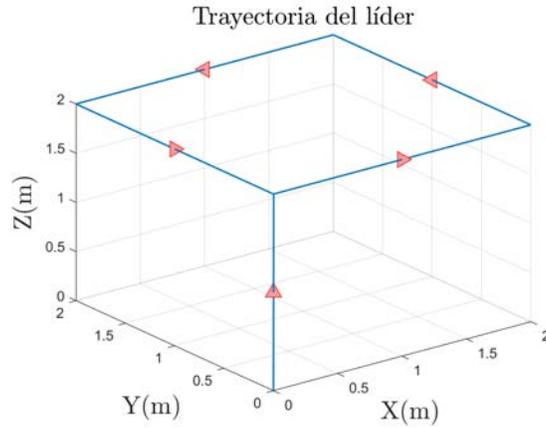


Figura 4.12: Trayectoria A del líder.

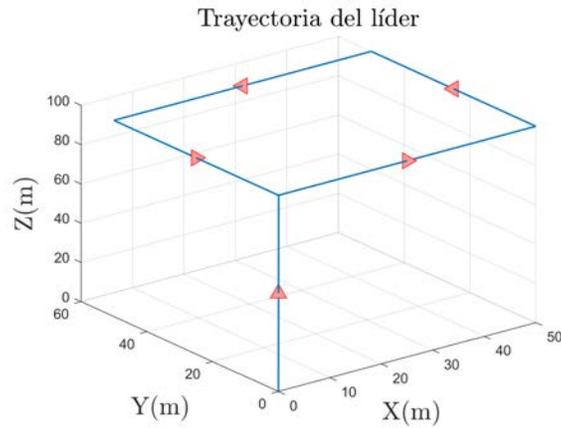


Figura 4.13: Trayectoria B del líder.

Se puede observar que la referencia del líder alcanza una altura de 2 y 100 metros respectivamente, después, la trayectoria recorrida a lo largo de los ejes x y y tiene forma de un cuadrado, la cual se compone de 4 aristas de 2 y 50 metros respectivamente. Se espera que los agentes seguidores sigan al líder virtual sin perder la formación definida, a pesar de que sólo el agente 1 intercambie información con el líder como se muestra en la figura 4.10. El comportamiento del sistema multiagente se muestra en las siguientes figuras (4.14 y 4.15):

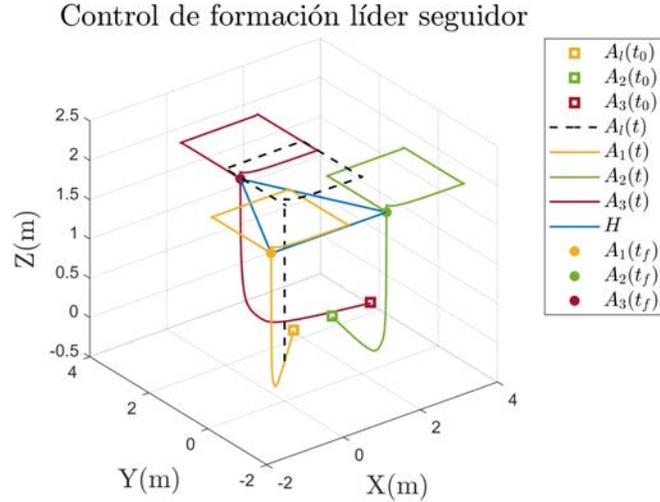
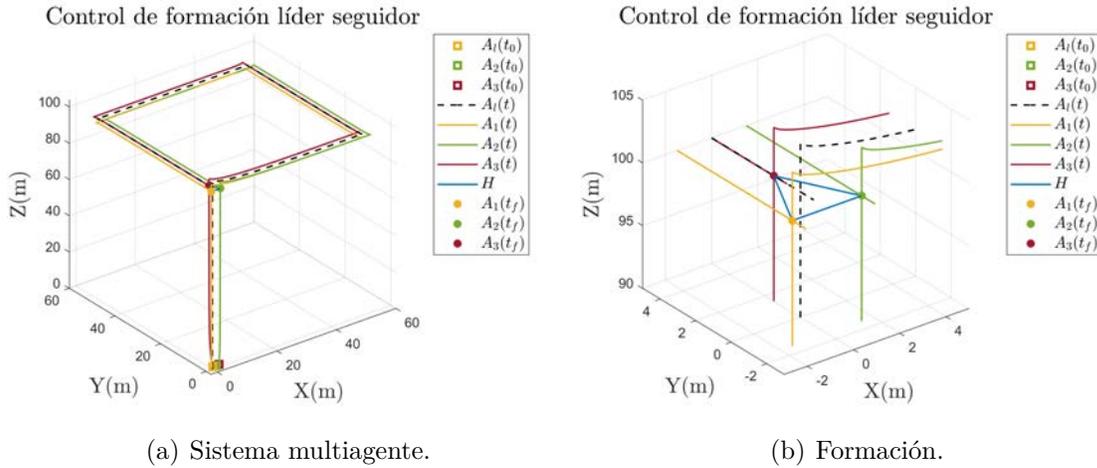


Figura 4.14: Seguimiento de trayectoria del sistema multiagente (simulación A).

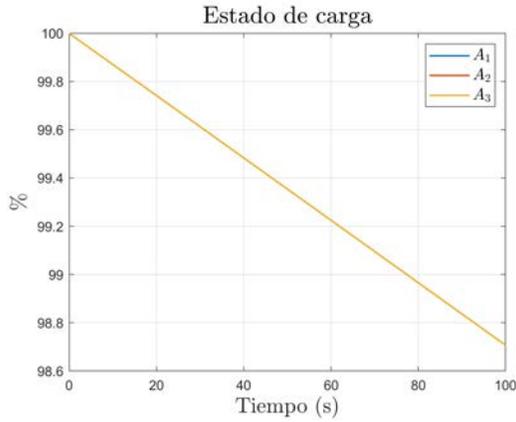


(a) Sistema multiagente.

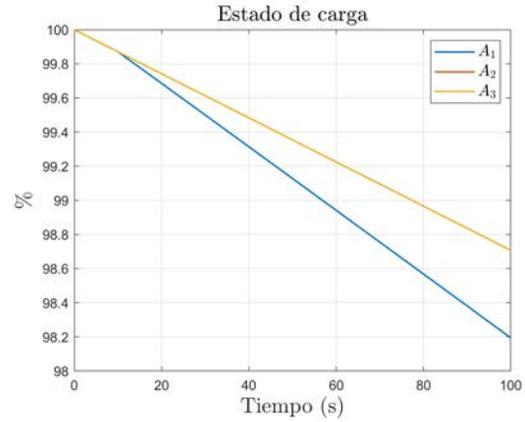
(b) Formación.

Figura 4.15: Seguimiento de trayectoria del sistema multiagente (simulación B).

Nótese que en la figura 4.14 el sistema multiagente alcanza la formación deseada, así como es capaz de seguir la trayectoria del líder virtual del sistema. La figura 4.15(a) muestra una trayectoria a seguir por el sistema multiagente de mayores dimensiones, donde también se alcanza la formación (véase figura 4.15(b)), además, la trayectoria es seguida de manera adecuada. Finalmente, el estado de carga fue medido en ambos casos, proponiendo una caída de voltaje de 5 volts en la batería del agente 1 (A_1), la comparación del estado de carga de los agentes del sistema se observa a continuación:

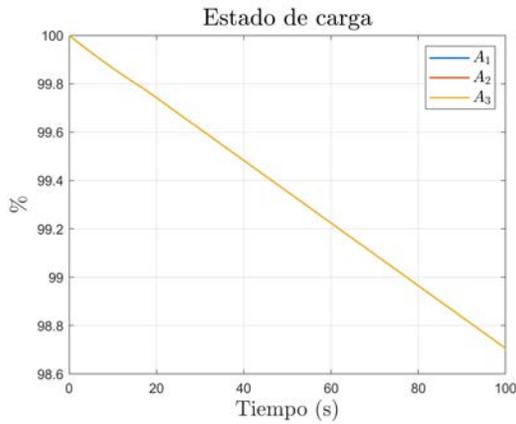


(a) Nominal.

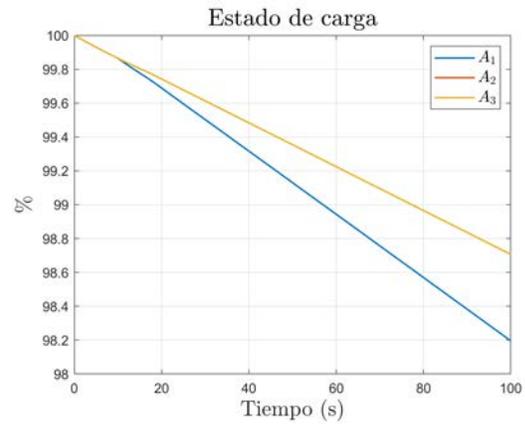


(b) Funcionamiento defectuoso.

Figura 4.16: Estado de carga de los agentes en la simulación A.



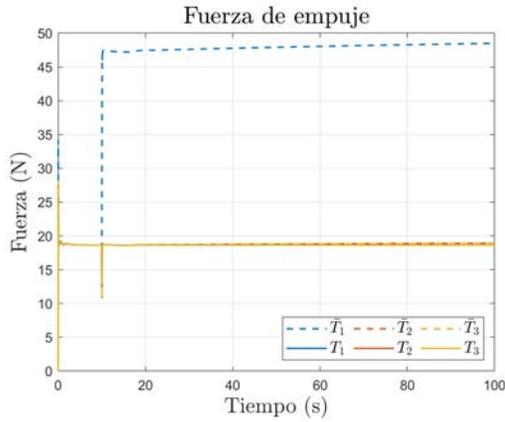
(a) Nominal.



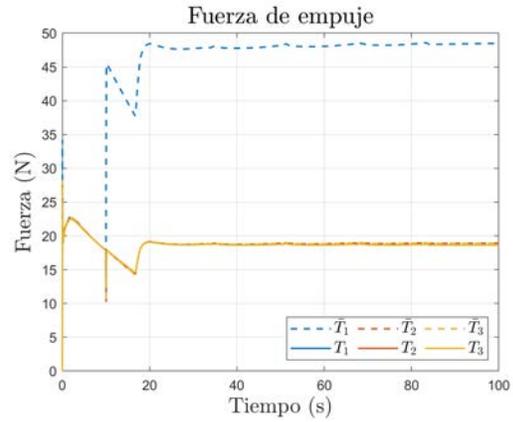
(b) Funcionamiento defectuoso.

Figura 4.17: Estado de carga de los agentes en la simulación B.

Se puede observar que en el estado de carga de ambas simulaciones en las figuras 4.16 y 4.17 las diferencias son mínimas. Cuando la caída de voltaje ocurre, la batería tiende a degradarse más rápido que en el caso nominal, dejando abierta la posibilidad de aplicar técnicas de consenso para alcanzar la sincronización del estado de carga, de esta manera es posible aumentar la fiabilidad de las aplicaciones de los sistemas multiagente. La fuerza de empuje o *thrust* de los agentes es compensada por el controlador. En la figura 4.18 se observan las leyes de control calculadas por el controlador, las cuales están representadas por la línea punteada; y las obtenidas al agregar la dinámica del sistema de propulsión, las cuales se representan por las líneas sólidas. En las figuras 4.18(a) y 4.18(b) se observa que la fuerza de empuje en el eje z es compensada por el controlador.



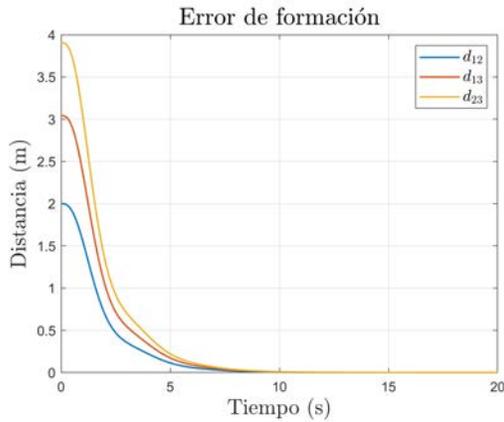
(a) Fuerza de empuja en la simulación A.



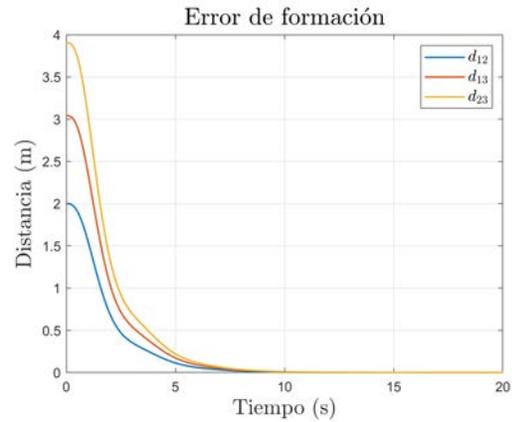
(b) Fuerza de empuja en la simulación B.

Figura 4.18: Fuerza de empuje en el eje z de los agentes del sistema.

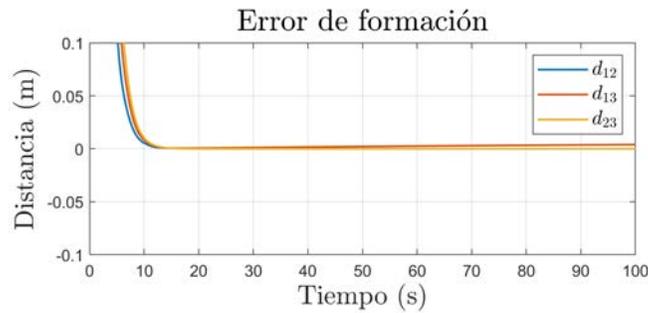
Por otro lado, es posible evaluar el control de formación diseñado por medio de su error de formación y su error de sincronización. Como se ha mencionado anteriormente, el error de formación puede observarse en los dos casos en las figuras 4.19 y 4.20.



(a) Nominal.



(b) Funcionamiento defectuoso.



(c) Error de formación (figura 4.19(b)).

Figura 4.19: Errores de formación del sistema multiagente en la simulación A.

En la figura 4.19(a), se muestra el caso nominal de la simulación A, donde el error de formación es muy cercano a 0 a los 10 segundos. En la figura 4.19(b), se graficó el error de formación cuando ocurre el decremento de voltaje en la batería del agente A_1 ; el gráfico parece mostrar un comportamiento idéntico con respecto al caso nominal (4.19(a)), sin embargo, en el acercamiento en la figura 4.19(c) se observa una ligera diferencia entre los términos d_{12} y d_{13} con d_{23} a partir de la caída de voltaje, dichos términos se refieren al error relativo entre los agentes A_1 con A_2 y A_1 con A_3 . Esto significa que entre más transcurre el tiempo error de formación crecerá hasta el final del tiempo de simulación, por lo tanto, en el caso de misiones de complejidad elevada, el control de formación no garantizará la formación predefinida.

El error de formación del sistema multiagente al realizar la simulación B se muestra en la figura 4.20, la diferencia entre el error de formación en el caso nominal (ver figura 4.20(a)) y el error de formación en el caso defectuoso (ver figura 4.20(b)) parece mínimo, sin embargo, al igual que la simulación A, al observar el acercamiento (ver figura 4.20(c)) se puede notar que el error de formación se aleja de 0, causando así, irregularidades en la formación definida con anterioridad.

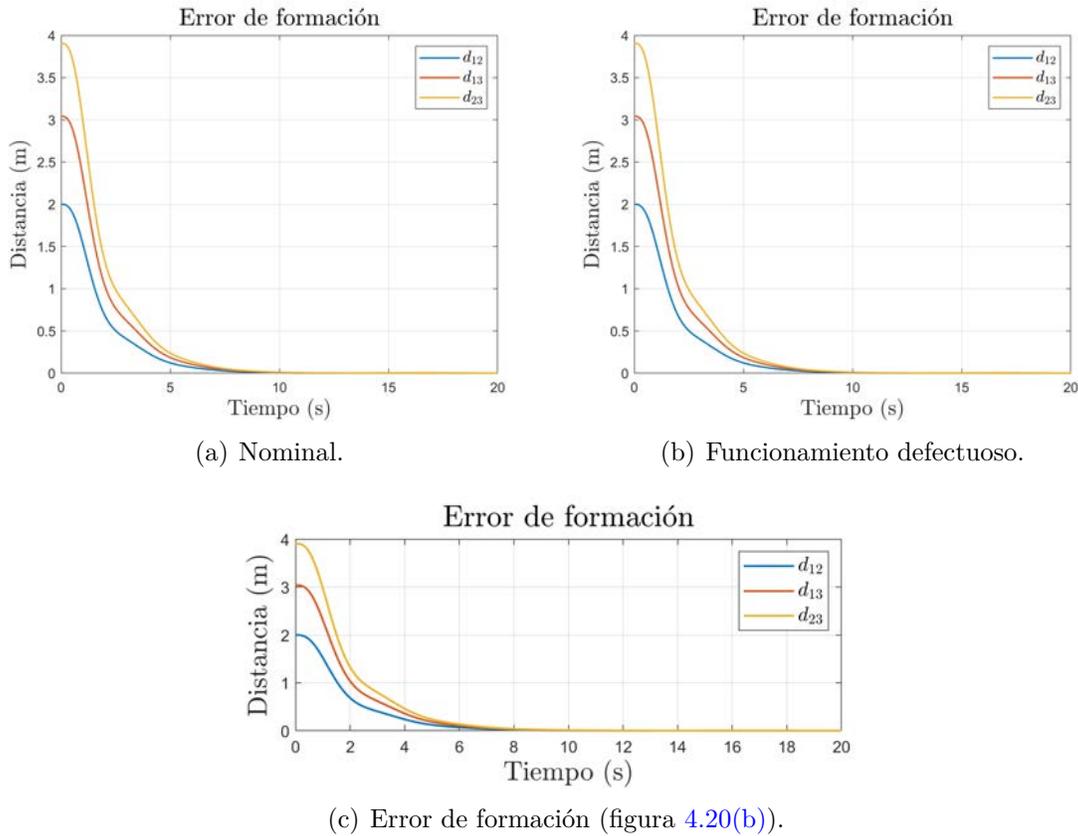


Figura 4.20: Errores de formación del sistema multiagente en la simulación B.

El error de sincronización está dado por [83]: $\delta_i = \|x_i - x_l\| - \|h_i\|$ y es el error relativo de cada agente con el líder seguidor, donde se considera que el agente i sigue al líder

si $\delta_i = 0$, de esta manera es posible evaluar el desempeño del sistema multiagente al recorrer las trayectorias A y B. En la figura 4.21 se observa el error de sincronización del sistema cuando se realiza la simulación A en los casos nominal y defectuoso. El caso nominal (ver figura 4.21(a)) muestra un error de sincronización que se aproxima bastante a 0 a los 10 segundos de transcurrida la formación, a diferencia del caso defectuoso (ver figura 4.21(b)), donde el error de sincronización parece alcanzar el valor de 0 a los 10 segundos, sin embargo, se encuentra más alejado de 0; la figura 4.21(c) permite observar la diferencia que existe entre δ_1 , δ_2 , δ_3 y 0. En otras palabras, el sistema multiagente es capaz de seguir al agente líder parcialmente cuando sucede la caída de voltaje en la batería.

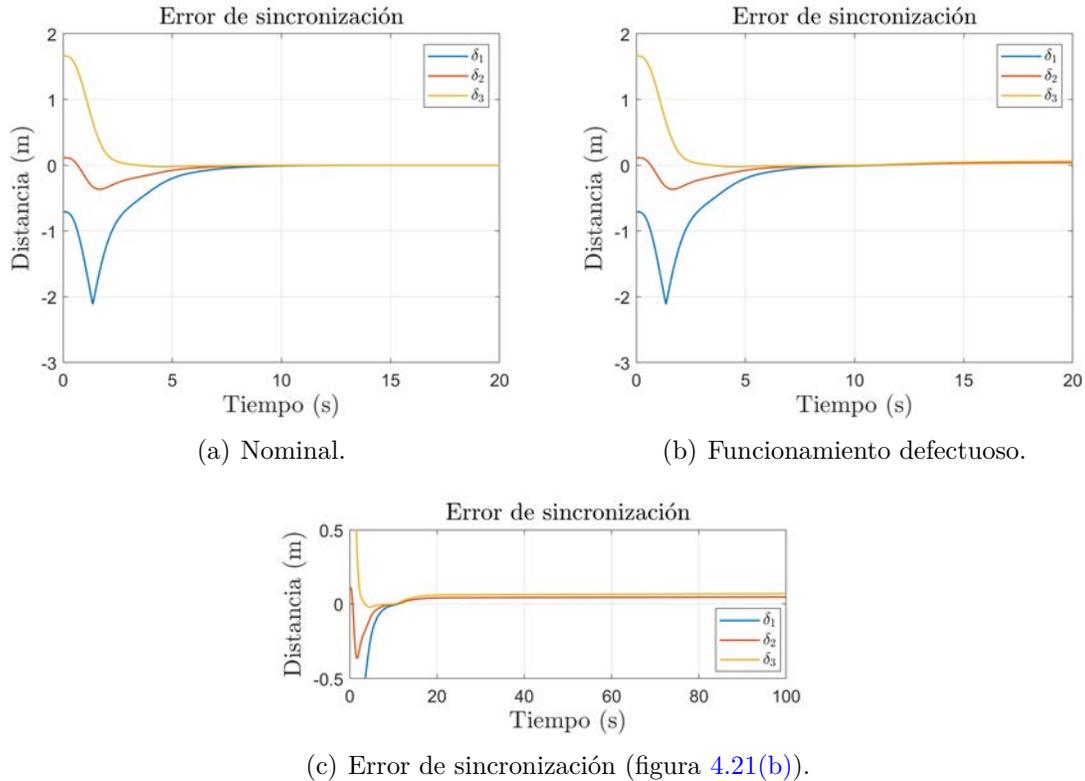


Figura 4.21: Errores de sincronización del sistema multiagente en la simulación A.

Al someter el sistema multiagente al control de formación basado en consenso líder seguidor se obtuvieron los gráficos de la figura 4.22, en el caso nominal (ver figura 4.22(a)) se observa que el sistema multiagente sigue al líder sin problema alguno, es decir, el error de sincronización se aproxima al valor de 0 de manera adecuada. En el caso defectuoso (ver figura 4.22(b)), se observa una ligera diferencia en la aproximación,

al igual que en la simulación A (ver figura 4.21(b)). Dicha diferencia se observa de manera clara en el acercamiento de la figura 4.22(c). Esto nos lleva a que el control del sistema multiagente sigue parcialmente la trayectoria del líder.

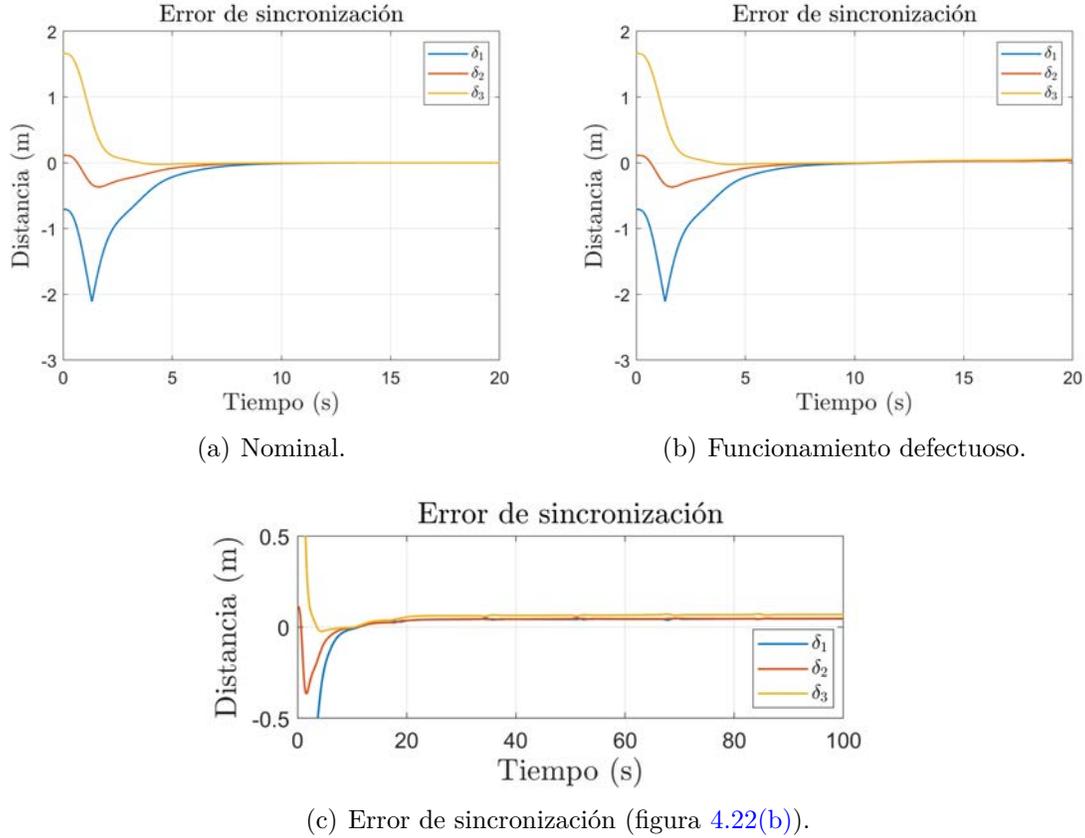


Figura 4.22: Errores de sincronización del sistema multiagente en la simulación B.

El control de formación basado en consenso líder seguidor es capaz de realizar las misiones asignadas en este capítulo, sin embargo, el controlador sufre un declive de efectividad al realizar formaciones con agentes defectuosos. Dicho lo anterior, el sistema multiagente sigue siendo capaz de mantener la formación y seguir al líder durante una misión, esto debido a las características del diseño de las ganancias obtenidas en este capítulo que permiten cierta robustez o resistencia al funcionamiento del control de formación presentado.

Capítulo 5

Conclusiones

A lo largo del tema de tesis se han investigado distintas maneras de control de formación, como lo son: el control de formación basado en consenso con y sin líder seguidor ([15], [81]), control de formación para sistemas LPV ([67], [83]), control de formación robusto ([37], [39]), así como también se encuentran vertientes tales como: el control tolerante a fallas ([59], [60], [67], [76]), diagnóstico de fallas ([61], [65], [66]), optimizaciones por medio de observadores robustos ([70]), filtro de Kalman ([49]) o predicción del deterioro de componentes ([27], [72]), sin embargo, la mayoría de estos enfoques no involucran al sistema de propulsión de un UAV, este sistema es tomado en cuenta para los FTC que absorben fallas de actuador, aun así, estos no incluyen a la batería dentro de la reconfiguración.

El análisis del desempeño de la batería en los sistemas multiagente carece de profundidad, esto debido a la robustez que presentan los controladores que se diseñan actualmente, esta robustez es capaz de absorber el desgaste de la batería en los trabajos que presentan resultados de simulación, los cuales son mayoría en el enfoque multiagente, además, los autores únicamente se enfocan en la dinámica de los agentes, los cuales suelen ser no actuados, por lo tanto, los subsistemas no incluidos en estos sistemas no actuados son ignorados; el control tolerante a fallas en el enfoque multiagente suele incluir el sistema de propulsión, donde las ganancias que se calculan para la reconfiguración del sistema compensan en su mayoría fallas de actuador o en el intercambio de información. Este tema de tesis de maestría presenta un control de formación basado en consenso líder seguidor para un sistema multiagente, donde, a diferencia de otros trabajos, se analiza el comportamiento de un agente cuando presenta un deterioro en la batería, el cual causa una caída de voltaje, limitando así, la fiabilidad del sistema multiagente. Para alcanzar al control de formación basado en consenso líder seguidor se realizó una comparación entre las diferentes formas de representar la dinámica de un UAV, la cual tuvo como objetivo el encontrar los parámetros adecuados para agregar el sistema de propulsión y poder analizar el comportamiento del sistema bajo un funcionamiento defectuoso, posteriormente se realizó un control de formación sin líder seguidor, sin embargo, al no tener una referencia, no era capaz de elevarse, por lo tanto los casos de simulación fueron únicamente aplicados al control de formación basado en

consenso líder seguidor.

Se concluye que el control diseñado para el sistema multiagente en cuestión es capaz de realizar trayectorias incluso con el funcionamiento inadecuado de la batería que alimenta a los agentes, de igual manera, los controles de formación se diseñaron de manera exitosa, logrando un error de sincronización y formación muy cercano a 0 en los casos nominales y un error pequeño en los casos defectuosos, finalmente, la inclusión del sistema de propulsión fue exitosa a pesar de no tener la posibilidad de medir experimentalmente los parámetros de los agentes.

5.1. Contribución

En este trabajo se logró el diseño de un controlador basado en consenso líder seguidor para sistemas multiagente, cuyo líder cuenta con una dinámica no lineal; además, el sistema de propulsión (motores, baterías y controladores de velocidad) fue incluido para lograr analizar el comportamiento de la batería en particular, este sistema necesitó el uso de nuevos parámetros los cuales fueran adecuados a los demás parámetros de los agentes.

5.2. Trabajos futuros

El tema realizado en este trabajo de maestría deja abierta la posibilidad de distintos trabajos futuros, entre los cuales se propone la predicción del estado de carga de la batería por medio de técnicas de optimización para sistemas multiagente; otro trabajo que se propone es la estimación de parámetros de los motores de un UAV cuadricóptero, así como también la obtención experimental de estos.

Referencias

- [1] Danny Espín-Sarzosa, Rodrigo Palma-Behnke, and Oscar Núñez-Mata. Energy management systems for microgrids: Main existing trends in centralized control architectures. *Energies*, 13(3):547, 2020.
- [2] Mark Eiling Rosheim. *Leonardo's Programmable Automaton and Lion*, pages 21–68. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.
- [3] Omar I Dallah Bashi, WZ Hasan, N Azis, S Shafie, and Hiroaki Wagatsuma. Unmanned aerial vehicle quadcopter: A review. *Journal of Computational and Theoretical Nanoscience*, 14(12):5663–5675, 2017.
- [4] Zairil Zaludin and Ezanee Gires. Automatic flight control requirements for transition flight phases when converting long endurance fixed wing uav to vtol aircraft. In *2019 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS)*, pages 273–278. IEEE, 2019.
- [5] Qimin Zhang, Zihe Liu, Jieru Zhao, and Shuguang Zhang. Modeling and attitude control of bi-copter. In *2016 IEEE International Conference on Aircraft Utility Systems (AUS)*, pages 172–176. IEEE, 2016.
- [6] D. Kastelan, M. Konz, and J. Rudolph. Fully actuated tricopter with pilot-supporting control. *IFAC-PapersOnLine*, 48(9):79–84, 2015.
- [7] Cody Yoakum and Joseph Cerreta. A review of dji's mavic pro precision landing accuracy. *International Journal of Aviation, Aeronautics, and Aerospace*, 7(4):5, 2020.
- [8] L. N. Santhosh and T. K. Chandrashekar. Anti-collision pentacopter for ndt applications. In *2015 IEEE International Transportation Electrification Conference (ITEC)*, pages 1–13. IEEE, 2015.
- [9] Jeroen AJ Ligthart, Pakorn Poksawat, Liuping Wang, and Henk Nijmeijer. Experimentally validated model predictive controller for a hexacopter. *IFAC-PapersOnLine*, 50(1):4076–4081, 2017.
- [10] Meng Joo Er, Shenghai Yuan, and Ning Wang. Development control and navigation of octocopter. In *2013 10th IEEE International Conference on Control and Automation (ICCA)*, pages 1639–1643. IEEE, 2013.

- [11] Dawei Bie, Daochun Li, Jinwu Xiang, Huadong Li, Zi Kan, and Yi Sun. Design, aerodynamic analysis and test flight of a bat-inspired tailless flapping wing unmanned aerial vehicle. *Aerospace Science and Technology*, 112:106557, 2021.
- [12] Matthew Ng Rongfa, Teppatat Pantuphag, and Sutthiphong Srigrarom. Analysis of kinematics of flapping wing uav using optitrack systems. *Aerospace*, 3(3):23, 2016.
- [13] Ricardo Schacht-Rodríguez, Jean-Christophe Ponsart, Carlos D García-Beltrán, and Carlos-M Astorga-Zaragoza. Analysis of energy consumption in multirotor uav under actuator fault effects. In *2019 4th Conference on Control and Fault Tolerant Systems (SysTol)*, pages 104–109. IEEE, 2019.
- [14] Zhongkui Li and Zhisheng Duan. *Cooperative control of multi-agent systems: a consensus region approach*. CRC press, 2017.
- [15] Jesus Avelino Vazquez Trejo. *Control de formación distribuido para una flotilla de vehículos autónomos basado en sistemas multi-agentes*. Tesis de maestría, Centro Nacional de Investigación y Desarrollo Tecnológico, Cuernavaca, Morelos, Junio 2020.
- [16] Lubomir Bakule and Martin Papik. Decentralized control and communication. *Annual Reviews in Control*, 36(1):1–10, 2012.
- [17] Konstantinos Dalamagkidis, Kimon P Valavanis, Les A Piegl, Konstantinos Dalamagkidis, Kimon P Valavanis, and Les A Piegl. *Aviation History and Unmanned Flight*, pages 11–42. Springer Netherlands, Dordrecht, 2012.
- [18] Alcedo. <https://www.alcedo.ethz.ch/#>. Revisado: 2.09.2023.
- [19] Mario Silvagni, Andrea Tonoli, Enrico Zenerino, and Marcello Chiaberge. Multi-purpose uav for search and rescue operations in mountain avalanche events. *Geomatics, Natural Hazards and Risk*, 8(1):18–33, 2017.
- [20] Dimosthenis C Tsouros, Stamatia Bibi, and Panagiotis G Sarigiannidis. A review on uav-based applications for precision agriculture. *Information*, 10(11):349, 2019.
- [21] Saket Gowravaram, Pengzhi Tian, Harold Flanagan, Jacksen Goyer, and Haiyang Chao. Uas-based multispectral remote sensing and ndvi calculation for post disaster assessment. In *2018 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 684–691. IEEE, 2018.
- [22] Stephen DJ McArthur, Euan M Davidson, Victoria M Catterson, Aris L Dimeas, Nikos D Hatziargyriou, Ferdinanda Ponci, and Toshihisa Funabashi. Multi-agent systems for power engineering applications—part i: Concepts, approaches, and technical challenges. *IEEE Transactions on Power systems*, 22(4):1743–1752, 2007.
- [23] Michael Wooldridge. *Intelligent Agents*, chapter 1, pages 1–50. The MIT press, London, England, 1999.

- [24] JR Martínez-de Dios, Luis Merino, Aníbal Ollero, Luis M Ribeiro, and Xavier Viegas. Multi-uav experiments: application to forest fires. *Multiple heterogeneous unmanned aerial vehicles*, pages 207–228, 2007.
- [25] O. Malaschuk and A. Dyumin. Intelligent multi-agent system for rescue missions. In *Advanced Technologies in Robotics and Intelligent Systems: Proceedings of ITR 2019*, pages 89–97. Springer, 2020.
- [26] Sami El Ferik and Abdulrahman Aliyu. Adaptive backstepping multi-agent control of multiple uavs conveying load; a graph rigidity approach. *IEEE Access*, 2023.
- [27] R Schacht-Rodríguez, J-C Ponsart, C-D García-Beltrán, C-M Astorga-Zaragoza, Didier Theilliol, and Youmin Zhang. Path planning generation algorithm for a class of uav multicopter based on state of health of lithium polymer battery. *Journal of Intelligent & Robotic Systems*, 91:115–131, 2018.
- [28] Won Joon Yun, Soohyun Park, Joongheon Kim, MyungJae Shin, Soyi Jung, David A Mohaisen, and Jae-Hyun Kim. Cooperative multiagent deep reinforcement learning for reliable surveillance via autonomous multi-uav control. *IEEE Transactions on Industrial Informatics*, 18(10):7086–7096, 2022.
- [29] Andrew Clark, Linda Bushnell, and Radha Poovendran. On leader selection for performance and controllability in multi-agent systems. In *2012 IEEE 51st IEEE conference on decision and control (CDC)*, pages 86–93. IEEE, 2012.
- [30] Juan Antonio Vazquez Trejo, Damiano Rotondo, Manuel Adam Medina, and Didier Theilliol. Robust observer-based leader-following consensus for a class of nonlinear multi-agent systems: application to uav formation control. In *2021 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1565–1572. IEEE, 2021.
- [31] Ruiwen Liao, Liang Han, Xiwang Dong, Qingdong Li, and Zhang Ren. Finite-time formation-containment tracking for second-order multi-agent systems with a virtual leader of fully unknown input. *Neurocomputing*, 415:234–246, 2020.
- [32] Tianlai Xu, Jinlong Liu, Zexu Zhang, Guodong Chen, Di Cui, and Huiping Li. Distributed mpc for trajectory tracking and formation control of multi-uavs with leader-follower structure. *IEEE Access*, 2023.
- [33] Alejandro Morfin-Santana, Filiberto Muñoz, Sergio Salazar, and José Manuel Valdovinos. Robust neural network consensus for multiagent uass based on weights’ estimation error. *Drones*, 6(10):300, 2022.
- [34] Xiangke Wang, Yangguang Yu, and Zhongkui Li. Distributed sliding mode control for leader-follower formation flight of fixed-wing unmanned aerial vehicles subject to velocity constraints. *International journal of robust and nonlinear control*, 31(6):2110–2125, 2021.
- [35] Jianhua Wang, Liang Han, Xiaoduo Li, Xiwang Dong, Qingdong Li, and Zhang Ren. Time-varying formation of second-order discrete-time multi-agent systems

- under non-uniform communication delays and switching topology with application to uav formation flying. *IET Control Theory & Applications*, 14(14):1947–1956, 2020.
- [36] E. G. Rojo-Rodriguez, E. J. Ollervides, P. Zambrano-Robledo, and O. Garcia. A fuzzy gain scheduling control algorithm for formation flight of multi-uavs. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 712–720. IEEE, 2019.
- [37] Jesus Avelino Vazquez Trejo, Manuel Adam Medina, Juan Antonio Vázquez Trejo, Carlos Daniel García Beltrán, and Jarniel García Morales. Robust observer-based leader-following consensus for multi-agent systems. *IEEE Latin America Transactions*, 19(11):1949–1958, 2021.
- [38] J. A. Vazquez Trejo, D. Rotondo, D. Theilliol, and M. Adam Medina. Observer-based quadratic boundedness leader-following control for multi-agent systems. *International Journal of Control*, 0(0):1–10, 2022.
- [39] Juan Antonio Vazquez Trejo, Manuel Adam-Medina, Carlos Daniel Garcia-Beltran, Gerardo Vicente Guerrero Ramirez, Betty yolanda lópez Zapata, Eduardo-Mael Sanchez-Coronado, and Didier Theilliol. Robust formation control based on leader-following consensus in multi-agent systems with faults in the information exchange: Application in a fleet of unmanned aerial vehicles. *IEEE Access*, 9:104940–104949, 2021.
- [40] Mahsa Mohammadi, Mahdi Baradarannia, and Ali Farzamnia. Leader-following consensus of nonlinear multi-agent systems based on position and velocity estimations. In *2021 7th International Conference on Control, Instrumentation and Automation (ICCIA)*, pages 1–7, 2021.
- [41] Cinzia Amici, Federico Ceresoli, Marco Pasetti, Matteo Saponi, Monica Tiboni, and Simone Zanoni. Review of propulsion system design strategies for unmanned aerial vehicles. *Applied Sciences*, 11(11):5209, 2021.
- [42] Grzegorz Szafranski, Roman Czyba, and Marian BŁachuta. Modeling and identification of electric propulsion system for multirotor unmanned aerial vehicle design. In *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 470–476. IEEE, 2014.
- [43] Petar Piljek, Denis Kotarski, and Matija Krznar. Method for characterization of a multirotor uav electric propulsion system. *Applied Sciences*, 10(22):8229, 2020.
- [44] Danijel Pavković, Matija Krznar, Mihael Cipek, Davor Zorc, and Maja Trstenjak. Internal combustion engine control system design suitable for hybrid propulsion applications. In *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1614–1619. IEEE, 2020.
- [45] Ye Xie, Al Savvaris, Antonios Tsourdos, Jason Laycock, and Andrew Farmer. Modelling and control of a hybrid electric propulsion system for unmanned aerial vehicles. In *2018 IEEE aerospace conference*, pages 1–13. IEEE, 2018.

- [46] Ming Zhang, Dongfang Yang, Jiaxuan Du, Hanlei Sun, Liwei Li, Licheng Wang, and Kai Wang. A review of soh prediction of li-ion batteries based on data-driven algorithms. *Energies*, 16(7):3167, 2023.
- [47] Xin Sui, Shan He, Søren B Vilsen, Jinhao Meng, Remus Teodorescu, and Daniel-Ioan Stroe. A review of non-probabilistic machine learning-based state of health estimation techniques for lithium-ion battery. *Applied Energy*, 300:117346, 2021.
- [48] Hasini Viranga Abeywickrama, Beeshanga Abewardana Jayawickrama, Ying He, and Eryk Dutkiewicz. Comprehensive energy consumption model for unmanned aerial vehicles, based on empirical studies of battery performance. *IEEE access*, 6:58383–58394, 2018.
- [49] Ricardo Schacht-Rodriguez, Gerardo Ortiz-Torres, Carlos Daniel García-Beltrán, Carlos Manuel Astorga-Zaragoza, Jean-Cristophe Ponsart, and Didier Theilliol. Soc estimation using an extended kalman filter for uav applications. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 179–187, 2017.
- [50] Alberto Berrueta, Andoni Urtasun, Alfredo Ursúa, and Pablo Sanchis. A comprehensive model for lithium-ion batteries: From the physical principles to an electrical model. *Energy*, 144:286–300, 2018.
- [51] Arslan Ahmed Amin and Khalid Mahmood Hasan. A review of fault tolerant control systems: advancements and applications. *Measurement*, 143:58–68, 2019.
- [52] Yu Ziquan, Youmin Zhang, Bin Jiang, FU Jun, and JIN Ying. A review on fault-tolerant cooperative control of multiple unmanned aerial vehicles. *Chinese Journal of Aeronautics*, 35(1):1–18, 2022.
- [53] Ziquan Yu, Youmin Zhang, Bin Jiang, Xiang Yu, Jun Fu, Ying Jin, and Tianyou Chai. Distributed adaptive fault-tolerant close formation flight control of multiple trailing fixed-wing uavs. *ISA transactions*, 106:181–199, 2020.
- [54] Fujiao Zhou and Kecai Cao. Formation fault tolerant control of multi-uav based on virtual actuator. In *2018 5th IEEE International Conference on Cloud Computing and Intelligence Systems (CCIS)*, pages 28–32. IEEE, 2018.
- [55] Wanzhang Liu, Ke Zhang, Bin Jiang, and Xinggang Yan. Adaptive fault-tolerant formation control for quadrotors with actuator faults. *Asian Journal of Control*, 22(3):1317–1326, 2020.
- [56] Hongyi Li, Ying Wu, and Mou Chen. Adaptive fault-tolerant tracking control for discrete-time multiagent systems via reinforcement learning algorithm. *IEEE Transactions on Cybernetics*, 51(3):1163–1174, 2020.
- [57] Yong Xu and Zheng-Guang Wu. Distributed adaptive event-triggered fault-tolerant synchronization for multiagent systems. *IEEE Transactions on Industrial Electronics*, 68(2):1537–1547, 2020.

- [58] Chun Liu, Bin Jiang, Ron J Patton, and Ke Zhang. Decentralized output sliding-mode fault-tolerant control for heterogeneous multiagent systems. *IEEE transactions on cybernetics*, 50(12):4934–4945, 2019.
- [59] Maryam Fattahi and A Afshar. Fault tolerant consensus control of multi-agent systems with switching topology. *SN Applied Sciences*, 1(6):559, 2019.
- [60] Meysam Yadegar and Nader Meskin. Fault-tolerant control of nonlinear heterogeneous multi-agent systems. *Automatica*, 127:109514, 2021.
- [61] Gerardo Ortiz-Torres, Pedro Castillo, Felipe DJ Sorcia-Vázquez, Jesse Y Rumbo-Morales, Jorge A Brizuela-Mendoza, Javier De La Cruz-Soto, and Mario Martínez-García. Fault estimation and fault tolerant control strategies applied to vtol aerial vehicles with soft and aggressive actuator faults. *IEEE Access*, 8:10649–10661, 2020.
- [62] Niansi Li, Xiaoyong Liu, Bendong Yu, Liang Li, Jianqiang Xu, and Qiong Tan. Study on the environmental adaptability of lithium-ion battery powered uav under extreme temperature conditions. *Energy*, 219:119481, 2021.
- [63] Shiqin Jiao, Guiyang Zhang, Mei Zhou, and Guoqi Li. A comprehensive review of research hotspots on battery management systems for uavs. *IEEE Access*, 2023.
- [64] Rui Xiong, Quanqing Yu, and Weixiang Shen. Review on sensors fault diagnosis and fault-tolerant techniques for lithium ion batteries in electric vehicles. In *2018 13th IEEE Conference on Industrial Electronics and Applications (ICIEA)*, pages 406–410. IEEE, 2018.
- [65] Yimin Zhou, Lina Zhu, Jianjia Yi, Tom Hao Luan, and Changle Li. On vehicle fault diagnosis: A low complexity onboard method. In *GLOBECOM 2020-2020 IEEE Global Communications Conference*, pages 1–6. IEEE, 2020.
- [66] Rui Xiong, Quanqing Yu, Weixiang Shen, Cheng Lin, and Fengchun Sun. A sensor fault diagnosis method for a lithium-ion battery pack in electric vehicles. *IEEE Transactions on Power Electronics*, 34(10):9709–9718, 2019.
- [67] Zhixiang Liu, Chi Yuan, and Youmin Zhang. Active fault-tolerant control of unmanned quadrotor helicopter using linear parameter varying technique. *Journal of Intelligent & Robotic Systems*, 88:415–436, 2017.
- [68] Suwarna Shete, Pranjal Jog, R. Kamalakannan, J T Arun Raghesh, S Manikandan, and R. K. Kumawat. Fault diagnosis of electric vehicle’s battery by deploying neural network. In *2022 Sixth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, pages 346–351, 2022.
- [69] Ung Hee Lee, Chen-Wen Pan, and Elliott J Rouse. Empirical characterization of a high-performance exterior-rotor type brushless dc motor and drive. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8018–8025. IEEE, 2019.

- [70] Wooyong Kim, Pyeong-Yeon Lee, Jonghoon Kim, and Kyung-Soo Kim. A robust state of charge estimation approach based on nonlinear battery cell model for lithium-ion batteries in electric vehicles. *IEEE Transactions on Vehicular Technology*, 70(6):5638–5647, 2021.
- [71] Ricardo Schacht-Rodríguez, Jean-Cristophe Ponsart, Carlos Daniel García-Beltrán, and Carlos Manuel Astorga-Zaragoza. Prognosis and health management for the prediction of uav flight endurance. *IFAC-PapersOnLine*, 51(24):983–990, 2018.
- [72] Ricardo Schacht Rodriguez. *Planeación de misión de vehículos aéreos no tripulados basado en la condición de salud de la batería*. Tesis de doctorado, Centro Nacional de Investigación y Desarrollo Tecnológico, Cuernavaca, Morelos, 2019.
- [73] Hansi Fernando, Ashen De Silva, M.D.C. De Zoysa, K.A.D.C. Dilshan, and S.R. Munasinghe. Modelling, simulation and implementation of a quadrotor uav. In *2013 IEEE 8th International Conference on Industrial and Information Systems*, pages 207–212. IEEE, 2013.
- [74] José-Fermi Guerrero-Castellanos, Argel Vega-Alonzo, Nicolas Marchand, Sylvain Durand, Jesús Linares-Flores, and G Mino-Aguilar. Real-time event-based formation control of a group of vtol-uavs. In *2017 3rd International Conference on Event-Based Control, Communication and Signal Processing (EBCCSP)*, pages 1–8. IEEE, 2017.
- [75] Abbas Chamseddine, Youmin Zhang, Camille Alain Rabbath, Cedric Join, and Didier Theilliol. Flatness-based trajectory planning/replanning for a quadrotor unmanned aerial vehicle. *IEEE Transactions on Aerospace and Electronic Systems*, 48(4):2832–2848, 2012.
- [76] Wang Jun, Yin Xiong-Dong, and Tian Yu-Yang. Fault-tolerant control design of quadrotor uav based on cpso. In *2018 IEEE 4th International Conference on Control Science and Systems Engineering (ICCSSE)*, pages 279–283. IEEE, 2018.
- [77] Wenlu Zhou, Yanping Zheng, Zhengjun Pan, and Qiang Lu. Review on the battery model and soc estimation method. *Processes*, 9(9):1685, 2021.
- [78] Ingvild B Espedal, Asanthi Jinasena, Odne S Burheim, and Jacob J Lamb. Current trends for state-of-charge (soc) estimation in lithium-ion battery electric vehicles. *Energies*, 14(11):3284, 2021.
- [79] M. Poovizhi, M. Senthil Kumaran, P. Ragul, L. Irene Priyadarshini, and R. Logambal. Investigation of mathematical modelling of brushless dc motor(blcd) drives by using matlab-simulink. In *2017 International Conference on Power and Embedded Drive Control (ICPEDC)*, pages 178–183, 2017.
- [80] Zhongkui Li, Zhisheng Duan, and Guanrong Chen. Consensus of discrete-time linear multi-agent systems with observer-type protocols. *Discrete and Continuous Dynamical Systems - B*, 16(2):489–505, 2011.

- [81] Juan Antonio Vazquez Trejo. *Fault tolerant methods design for a fleet of autonomous vehicles against faults/failures based on multi-agent systems*. Tesis de doctorado, Université de Lorraine ; Centro Nacional de Investigación y Desarrollo Tecnológico (Cuernavaca, Mor., México), Julio 2021.
- [82] Damiano Rotondo, Jean-Christophe Ponsart, and Didier Theilliol. Gain-scheduled observer-based consensus for linear parameter varying multi-agent systems. *Automatica*, 135:109979, 2022.
- [83] Jesus A Vazquez Trejo, Jean-Christophe Ponsart, Manuel Adam-Medina, Guillermo Valencia-Palomo, and Didier Theilliol. Distributed observer-based leader-following consensus control for lpv multi-agent systems: Application to multiple vtol-uavs formation control. In *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1316–1323. IEEE, 2023.
- [84] Houria Kheloufi, Ali Zemouche, Fazia Bedouhene, and Mohamed Boutayeb. On lmi conditions to design observer-based controllers for linear systems with parameter uncertainties. *Automatica*, 49(12):3700–3704, 2013.

Apéndice A

Diagramas de flujo

En este apéndice se muestran los diagramas de flujo seguidos para realizar las simulaciones presentadas en este tema de tesis. Los diagramas en las figuras A.1, A.2 y A.3 representan la simulación del capítulo 2 y el diagrama A.4 corresponde a la simulación del capítulo 3.

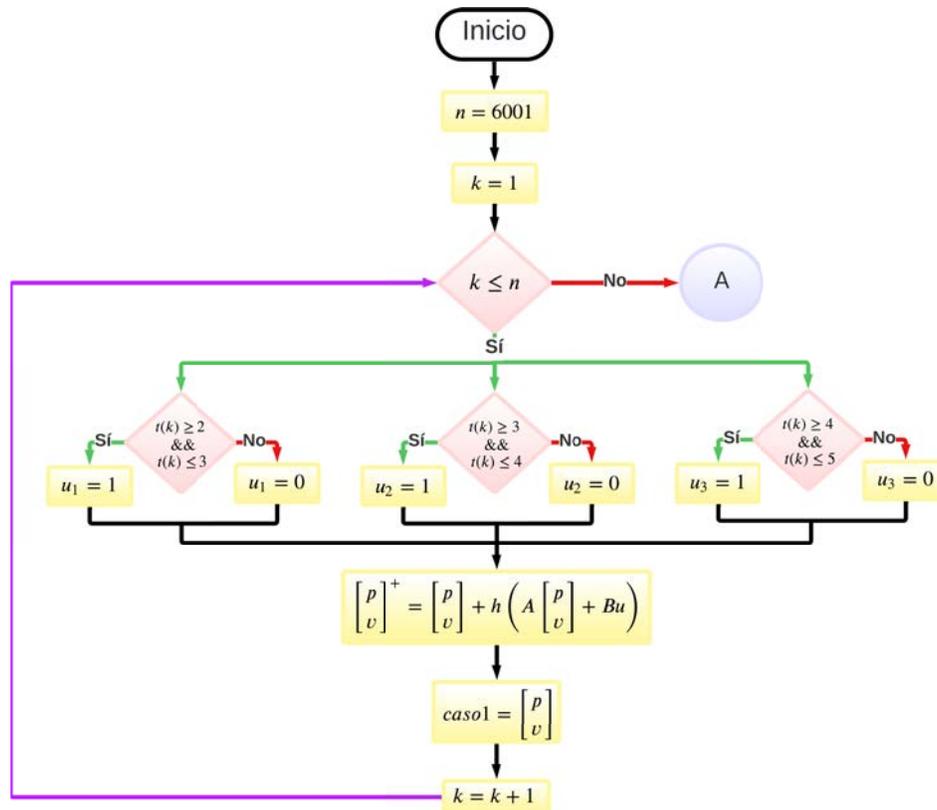


Figura A.1: Diagrama de flujo del caso 1.

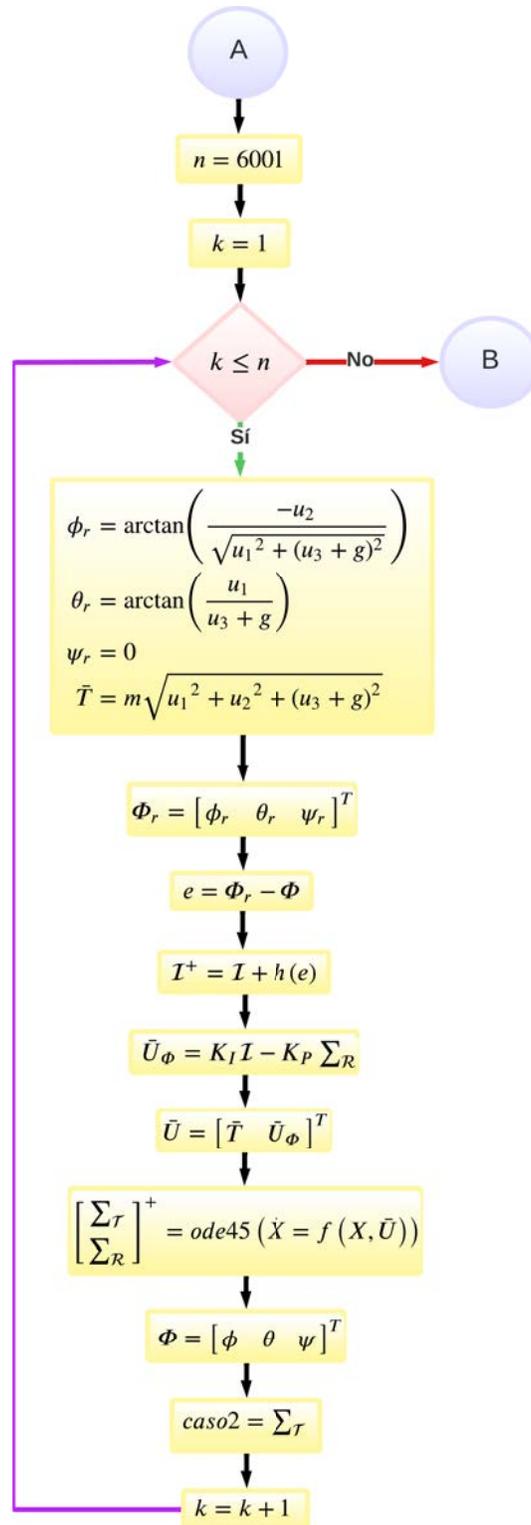


Figura A.2: Diagrama de flujo del caso 2.

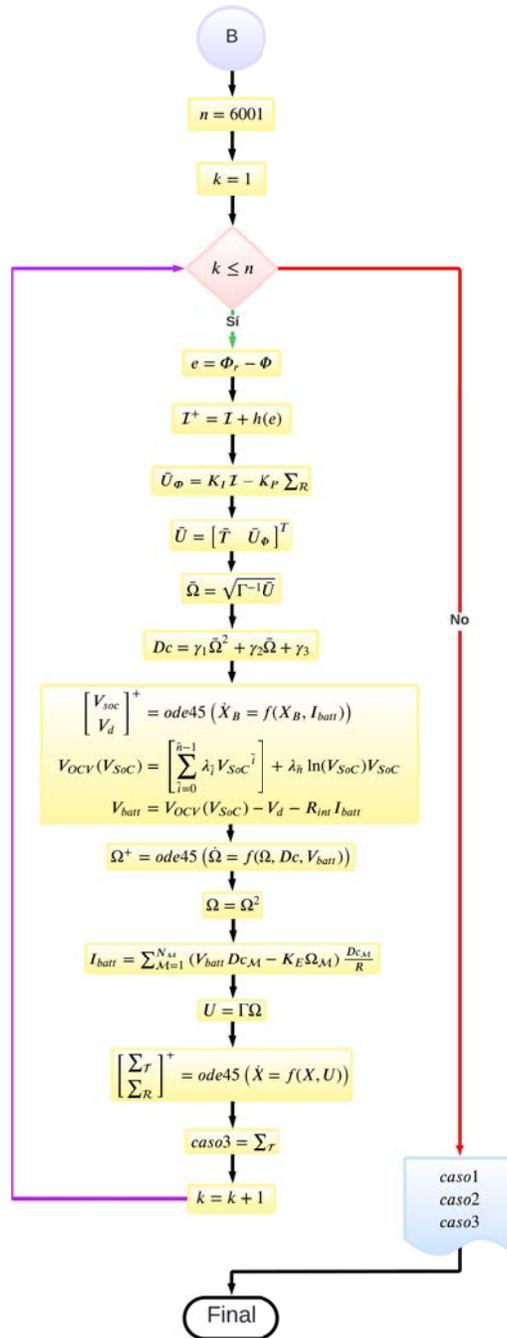


Figura A.3: Diagrama de flujo del caso 3.

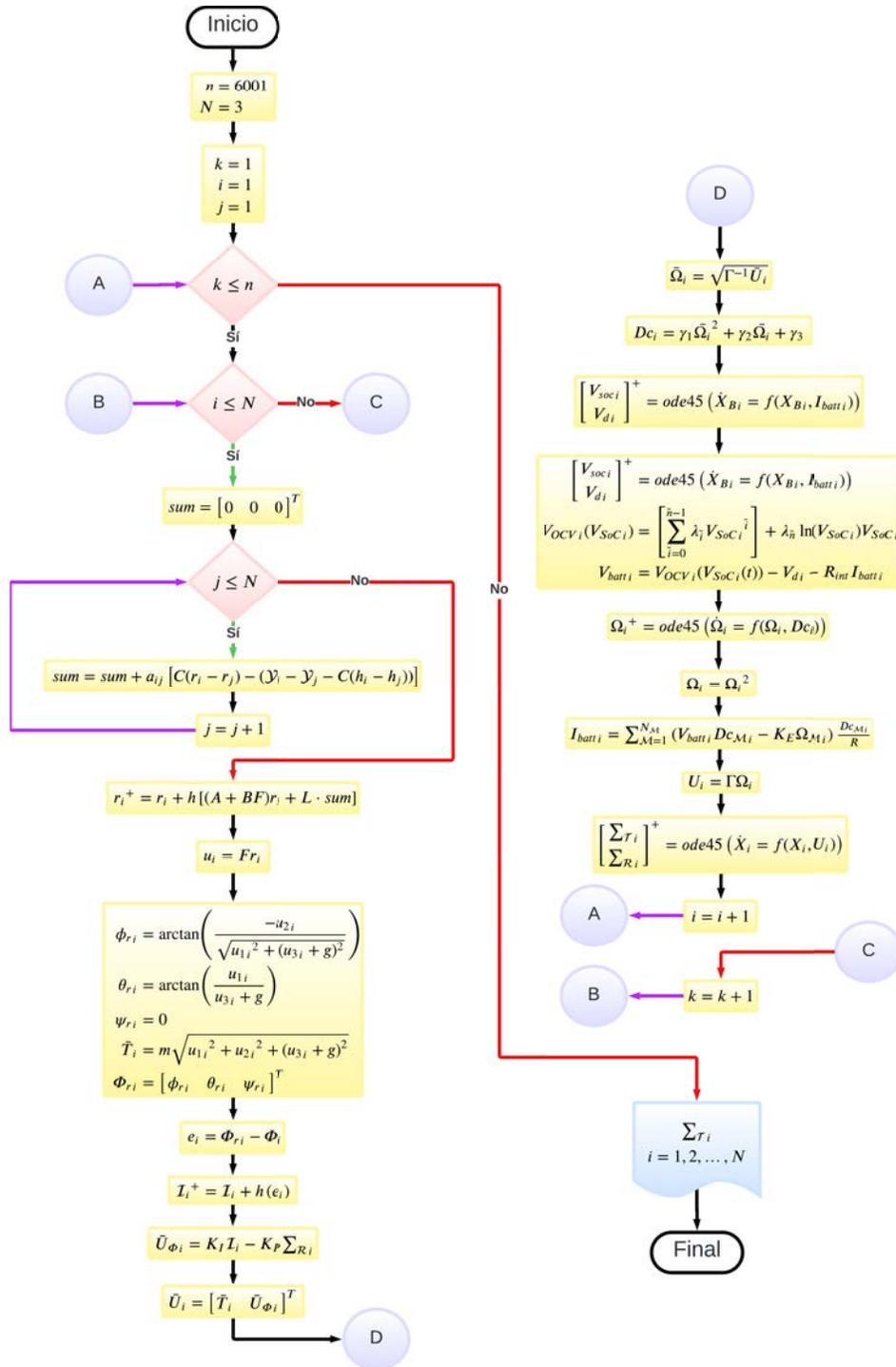


Figura A.4: Diagrama flujo para el control de formación basado en consenso.

Apéndice B

Resumen del estado del arte

La información de los artículos considerados de mayor relevancia del estado del arte fue sustraída y categorizada en la siguiente tabla (ver tabla [B.1](#)):

Referencia	Control	MAS	P	Tipo de perturbación	Resultados	FTC	Tipo de falla	Lugar de la falla	Reconfig	SP
[Vazquez et. Al., 2021]	Control robusto basado en observador	Si	Si	Turbulencia del viento	Simulación	No	-	-	-	No
[Vazquez et. Al. 2022]	Control basado en observador robusto	Si	Si	Turbulencia del viento	Simulación	No	-	-	-	No
[Vazquez et. Al., 2021]	Control robusto	Si	Si	Turbulencia del viento	Simulación	Si	Aditiva	Actuador	Actuador virtual	No
[Mohammadi et. Al., 2021]	Control basado en observador	Si	No	-	Simulación	No	-	-	-	No
[Schacht et. Al., 2018]	PID control	No	Si	Degradación de la batería	Simulación	No	-	-	-	Si
[Abeywickrama et Al, 2018]	No especificado	No	Si	Turbulencia del viento	Experimental	No	-	-	-	Si
[Schacht et Al, 2017]	PID control	No	Si	Degradación de la batería	Simulación	No	-	-	-	Si
[Berrueta et Al, 2018]	-	-	-	-	Experimental	-	-	-	-	-
[Fattahi et. Al., 2019]	Control distribuido	Si	No	-	Simulación	Si	Aditiva, no lineal e intermitente	Actuador y comunicación	Consenso tolerante a falla	No
[Yadegar et. Al., 2021]	Control Tolerante a fallas	Si	No	-	Simulation	Si	Aditiva variante en el tiempo	Actuador	Actuador virtual adaptativo no lineal	-
[Ortiz et. Al., 2020]	Control Tolerante a fallas	No	Si	No especificado	Experimental	Si	Aditiva externa	Actuador	Ganancias de control reajustadas	No

Tabla B.1: Aspectos importantes de los artículos del estado del arte.

P → Perturbación.

Reconfig → Reconfiguración.

SP → Sistema de propulsión.

Apéndice C

Parámetros para las simulaciones

Parámetro	Valor	Unidad
m	1,9	Kg
J_x	$21,6 \times 10^{-3}$	$Kg \times m^2$
J_y	$21,6 \times 10^{-3}$	$Kg \times m^2$
J_z	$43,2 \times 10^{-3}$	$Kg \times m^2$
g	9,81	m/s^2
l	0,3	m

Tabla C.1: Parámetros de los agentes.

Parámetro	Valor	Unidad
J_E	$8,3808 \times 10^{-6}$	Nms^2
b	$1,5027 \times 10^{-5}$	-
d	$1,6497 \times 10^{-5}$	-
K_E	0,0104	Nm/A
R	0,2	Ω
T_f	4×10^{-2}	Nm
D_f	2×10^{-6}	$Nm s/rad$

Tabla C.2: Parámetros de los BLM.

Parámetro	Valor	Unidad
C_T	22320	F
R_{int}	0,0421	Ω
R_d	0,0237	Ω
C_d	$1,6497 \times 10^{-5}$	F

Tabla C.3: Parámetros de la batería.

Apéndice D

Modelos matemáticos utilizados

En el presente tema de tesis se utilizan distintos modelos matemáticos para representar la dinámica de un UAV y algunos de sus subsistemas. El objetivo de este apéndice es representar de manera concisa los modelos utilizados en este trabajo para facilitar la lectura del mismo.

D.1. Modelo de un UAV

Para representar el comportamiento dinámico de un UAV se utilizan las siguientes ecuaciones diferenciales no lineales [73]:

$$\begin{aligned}\ddot{x} &= (\cos\phi\text{sen}\theta\cos\psi + \text{sen}\phi\text{sen}\psi)\frac{T}{m}, \\ \ddot{y} &= (\cos\phi\text{sen}\theta\text{sen}\psi - \text{sen}\phi\cos\psi)\frac{T}{m}, \\ \ddot{z} &= -g + (\cos\phi\cos\theta)\frac{T}{m}, \\ \ddot{\phi} &= \frac{J_y - J_z}{J_x}\dot{\theta}\dot{\psi} + \frac{R}{J_x}, \\ \ddot{\theta} &= \frac{J_z - J_x}{J_y}\dot{\phi}\dot{\psi} + \frac{P}{J_y}, \\ \ddot{\psi} &= \frac{J_x - J_y}{J_z}\dot{\theta}\dot{\phi} + \frac{Y}{J_z}.\end{aligned}$$

Así mismo, el modelo lineal de un UAV se representa de la siguiente manera [75]:

$$\begin{aligned}\ddot{x} &= g\theta, \\ \ddot{y} &= -g\phi, \\ \ddot{z} &= -g + \frac{T}{m}, \\ \ddot{\phi} &= \frac{R}{J_x}, \\ \ddot{\theta} &= \frac{P}{J_y}, \\ \ddot{\psi} &= \frac{Y}{J_z}.\end{aligned}$$

Donde los estados ϕ , θ , ψ , $\dot{\phi}$, $\dot{\theta}$ y $\dot{\psi}$, representan la rotación y la velocidad angular en los ángulos *roll*, *pitch* y *yaw* respectivamente; los estados x , y , z , \dot{x} , \dot{y} y \dot{z} , representan la traslación y la velocidad traslacional en los ejes x , y y z respectivamente. Los parámetros J_x , J_y y J_z son los momentos inerciales en los ejes x , y y z , además, g es la aceleración de la gravedad y m es la masa del UAV; finalmente, las entradas del sistema T , R , P y Y son la fuerza de empuje en el eje z y los pares rotacionales en los ángulos ϕ , θ y ψ .

D.2. Modelo de la batería de un UAV

El comportamiento de la batería es modelado por las siguientes ecuaciones [13]:

$$\begin{aligned}\dot{V}_{SoC} &= -\frac{1}{3600C_T}I_{batt}, \\ \dot{V}_d &= -\frac{1}{R_dC_d}V_d + \frac{1}{C_d}I_{batt}, \\ V_{OCV}(V_{SoC}) &= \sum_{\tilde{i}=0}^{\tilde{n}-1} \tilde{\lambda}_{\tilde{i}}V_{SoC}^{\tilde{i}} + \tilde{\lambda}_{\tilde{n}}\ln(V_{SoC})V_{SoC}, \\ V_{batt} &= V_{OCV}(V_{SoC}) - V_d - R_{int}I_{batt},\end{aligned}$$

donde V_{SoC} es el estado de carga o *state of charge* (SoC), C_T es la capacidad total de la batería, $V_{OCV}(V_{SoC})$ es el voltaje de circuito abierto, I_{batt} es la corriente total consumida, R_{int} es la resistencia interna de la batería, R_d y C_d es la resistencia y capacitancia del par RC del circuito, V_d es la respuesta transitoria de la batería, V_{batt} es el voltaje de la batería. El orden de la función polinómica está dado por \tilde{n} y $\tilde{\lambda}_{\tilde{i}}$ son los coeficientes del polinomio. El método para calcular el SoC de la batería es llamado conteo de coulombs o conteo de amperios-hora, el cual es el método más utilizado par el cálculo del SoC [78].

D.3. Modelo de los motores *brushless*

Las velocidades angulares de los motores se obtienen por medio de las siguientes ecuaciones ([69], [79]):

$$\begin{aligned}\dot{\Omega} &= \left[(V_{batt} Dc - K_E \Omega) \frac{K_E}{R} - d\Omega^2 - D_f \Omega - T_f \right] \frac{1}{J_E}, \\ I_{batt} &= \sum_{\mathcal{M}=1}^{N_{\mathcal{M}}} (V_{batt} \cdot Dc_{\mathcal{M}} - K_E \cdot \Omega_{\mathcal{M}}) \frac{Dc_{\mathcal{M}}}{R}, \\ Dc &= \gamma_1 \bar{\Omega}^2 + \gamma_2 \bar{\Omega} + \gamma_3,\end{aligned}$$

donde Ω es el vector de velocidades angulares de cada motor, γ_1 , γ_2 y γ_3 son los coeficientes de la función polinómica, $N_{\mathcal{M}}$ es el número de motores, \mathcal{M} representa cada motor, I_{batt} es la corriente total consumida por los motores V_{batt} es el voltaje de la batería. Además, los parámetros K_E , J_E , D_f , d y T_f son la fuerza contra electromotriz, el momento inercial de los BLMs, el coeficiente de amortiguamiento viscoso, la constante de arrastre y el par causado por la fricción respectivamente.

Apéndice E

Acrónimos

BLM Motor *brushless* (sin escobillas)

EKF Filtro de Kalman extendido

EoD Fin de la descarga

ESC Controlador de velocidad electrónico

FTC Control tolerante a fallas

LMI Desigualdad matricial lineal

MAS Sistema multiagente

MPC Control predictivo basado en el modelo

PID Proporcional-integral-derivativo

PI Proporcional-integral

PI-D Proporcional-integral-derivativo modificado

PWM Modulación de ancho de pulso

SoC Estado de carga

SoH Estado de salud

UAV Vehículo aéreo no tripulado