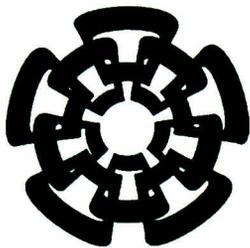


CT-928-551

DOJ. 2016



Centro de Investigación y de Estudios Avanzados
del Instituto Politécnico Nacional
Unidad Guadalajara

Sistema de navegación de UAV's en formación

Tesis que presenta:

David Fernández Flores

para obtener el grado de:

Maestro en Ciencias

en la especialidad de:

Ingeniería Eléctrica

Directores de Tesis

Dr. Bernardino Castillo Toledo

Dr. Antonio Ramírez Treviño

**CINVESTAV
IPN
ADQUISICION
LIBROS**

CLASS. CT00829
AGENCY CT-928-501
RECORD 23-05-2016
PROCESSED Down 2016
\$

Sistema de navegación de UAV's en formación

**Tesis de Maestría en Ciencias
Ingeniería Eléctrica**

Por:

David Fernández Flores

Ingeniero en Mecatrónica

Centro de Enseñanza Técnica Industrial 2009-2013

Becario de Conacyt, expediente no. 300887

Directores de Tesis

Dr. Bernardino Castillo Toledo

Dr. Antonio Ramírez Treviño

CINVESTAV del IPN Unidad Guadalajara, Septiembre de 2015.

Resumen

Los vehículos aéreos no tripulados o UAV's (Unmanned Aerial Vehicles) cada vez son más populares en el sector aeronáutico debido a la diversidad de ventajas que presentan respecto a las aeronaves tripuladas. Entre ellas cabe destacar la reducción de peso, dimensiones y costes. De entre todas las clases de UAV's se toma al quadrotor de la familia de los multicopteros, este en particular cuenta con 4 motores, con los beneficios de despegue y aterrizaje vertical capaz de aterrizar sin necesidad de pista. Al tener un grupo de UAV's en formación la eficacia del grupo aumenta, como por ejemplo, crece la capacidad de carga, el área de visión en una situación de búsqueda con una formación orientada se multiplica dependiendo del número de agentes que componen la formación.

En este trabajo se propone un control distribuido para que un grupo predeterminado de vehículos aéreos no tripulados mantengan una formación en el seguimiento de una trayectoria con topologías de comunicación cambiantes, mediante un algoritmo de consenso con el enfoque líder-seguidor. Para ello se presentan primeramente algunos preliminares en algoritmos de consenso y topologías de comunicación mediante teoría de grafos. El control se realiza con un algoritmo de consenso, las técnicas de control a bloques (*Block Control*) y el algoritmo de modos deslizantes *Super Twisting*, con lo cual se logra un control robusto, los ángulos de euler en este control están limitados sólo por cuestiones físicas, *Pitch* y *Roll* pueden ir de $-\frac{\pi}{2}$ a $\frac{\pi}{2}$ pero el ángulo *Yaw* está libre de singularidades.

Es propuesto un algoritmo para la generación de una trayectoria capaz de evadir obstáculos en un espacio de trabajo conocido, mediante la técnica reactiva de campos potenciales, con lo cual se va trazando la trayectoria al mismo tiempo de ir avanzando, cabe destacar que este algoritmo se encarga de la generación de la trayectoria independientemente del modelo matemático de los agentes en la formación.

Abstract

The Unmanned Aerial Vehicles or UAV's are becoming more popular in the aviation sector due to the diversity of advantages compared to manned aircraft. These include reducing weight, size and cost. From among all kinds of UAV's, of the family of multi-copters the quadrotor is taken, this in particular has 4 engines, with the benefits of vertical takeoff and landing able to land without track. Having a group of UAV's in formation, the efficiency of the group increases, such as the capacity grows, the viewing area in a search situation with a targeted training is multiplied depending on the number of agents that make up the formation.

In this work a distributed control is proposed, to maintain a predetermined group of unmanned aerial vehicles in a path tracking even with changing communication topologies, using an consensus algorithm involving a leader-follower approach. For this, first some preliminary consensus algorithms and communication topologies are presented using graph theory. The control is performed with an consensus algorithm, the techniques of Block Control and the sliding mode algorithm Super Twisting, whereby a robust control is achieved. The Euler angles in this control are limited only by physical issues, *Pitch* and *Roll* can go only from $-\frac{\pi}{2}$ to $\frac{\pi}{2}$ but the *Yaw* angle is free of singularities.

In addition, it is proposed an algorithm for path generation capable of obstacles avoidance in a known work space, using a reactive technique based on potential fields, which traces the path while advancing. It is noteworthy that this algorithm generate a path independently of the mathematical model of the agents in the formation.

Agradecimientos

Antes que nada expreso mi gratitud hacia mis padres Narciso Fernández González y Ma Mercedes Flores Cortés por todo el apoyo que siempre me han brindado en todo el transcurso de mis estudios y a lo largo de mi vida, sin ellos este trabajo nunca hubiera sido posible. También agradezco a toda mi familia por todo el apoyo que de alguna u otra forma siempre me han dado, una palabra de aliento o una frase de ánimo, gracias por estar ahí, en especial a Diana Paula Montaña Castro, sin tu apoyo esta aventura ni siquiera hubiera comenzado.

Quiero agradecer a mis asesores, el Dr. Bernardino Castillo Toledo y el Dr. Arturo Ramírez Treviño, por la ayuda y atención prestadas y la confianza dada para la realización de este trabajo. A mis sinodales, el Prof. Eduardo José Bayro Corrochano por su ayuda en la revisión de la tesis y al Dr. Carlos Aurelio López Limón por su ayuda con la resolución de muchas de mis dudas en la elaboración de la misma.

Agradezco a todos mis amigos por su compañía y los buenos momentos que compartimos durante estos años, sobre todo por esa unión que se mostraba en nuestro grupo. Gracias Cont-13.

Un especial agradecimiento al Dr. Luis F. Luque Vega en quien basé mi trabajo y siempre estuvo abierto a resolver mis dudas, al igual que a mi compañero Abraham Villanueva por sus comentarios y trabajo en equipo en la comprensión de los quadrotors.

Finalmente quiero agradecer a Conacyt por el apoyo económico y al Cinvestav Guadalajara por darme las facilidades de terminar este trabajo.

Índice general

1. Introducción	1
1.1. Estado del Arte	2
1.2. Motivación	3
1.3. Estructura de la Tesis	5
2. Preliminares	7
2.1. Modelado del Quadrotor	7
2.1.1. Funcionamiento Básico	7
2.1.2. Análisis de Cuerpo Libre	9
2.1.3. Modelo en Espacio de Estados	12
2.2. Block Control	14
2.2.1. Representación a Bloques de una Clase de Sistemas no Lineales	14
2.2.2. Transformación Recursiva a Bloques	15
2.3. Modos Deslizantes	16
2.3.1. Algoritmo de Super-Twisting	17
2.4. Consenso	18
2.4.1. Teoría de Grafos	19
2.4.2. Topología de Comunicación	20
2.4.3. Enfoque Líder-Seguidor	23
3. Diseño de Control Para un Quadrotor	31
3.1. Ley de Control Para un Quadrotor	31

3.2. Estabilidad de Control Para un Quadrotor	35
3.2.1. Estabilidad en Errores de Posición	36
3.2.2. Estabilidad del Modo Deslizante	36
3.3. Simulación de Control Para un Quadrotor	37
3.3.1. Sin Perturbaciones	38
3.3.2. Con Perturbaciones	41
4. Diseño de Control Distribuido	45
4.1. Control Distribuido	45
4.2. Estabilidad de Control Distribuido	49
4.2.1. Estabilidad en Errores de Posición	50
4.2.2. Estabilidad del Modo Deslizante	52
4.3. Simulación de Control Distribuido	52
4.3.1. Formación Fija	53
4.3.2. Formación Orientada	65
5. Generación de Trayectoria	69
5.1. Navegación	69
5.2. Evasión de Obstáculos	70
5.3. Simulación de Evasión de Obstáculos en Formación Orientada	73
6. Conclusiones y Trabajo Futuro	79
6.1. Conclusiones	79
6.2. Trabajo Futuro	80
Bibliografía	81

Índice de tablas

2.1. Ejemplo, grafo dirigido y sus matrices respectivas.	20
2.2. Matrices de adyacencia respectivas a los ejemplos en Fig. 2.7.	21
2.3. Matrices Laplacianas respectivas a los ejemplos en Fig. 2.7.	22
2.4. Matrices de adyacencia respectivas a los ejemplos en Fig. 2.9.	25
2.5. Matrices Laplacianas respectivas a los ejemplos en Fig. 2.9.	25
3.1. Parámetros	37
4.1. Matrices de adyacencia respectivas a los ejemplos en Fig. 4.13.	61
4.2. Matrices Laplacianas respectivas a los ejemplos en Fig. 4.13.	62

Índice de figuras

2.1. Tipo de configuración.	8
2.2. <i>Pitch, Roll, Yaw</i> .	8
2.3. Fuerzas en el quadrotor	10
2.4. Marcos de referencia principales.	11
2.5. Función relé.	17
2.6. Gráfica de error.	17
2.7. Topologías de comunicación para 3 agentes.	21
2.8. Consenso.	22
2.9. Topologías de comunicación líder-seguidor.	24
2.10. Seguimiento en consenso 1	26
2.11. Seguimiento en consenso 2	27
2.12. Seguimiento en consenso 3	28
3.1. Seguimiento de trayectoria 1 quadrotor.	38
3.2. Plano (x,y) 1 quadrotor.	39
3.3. Gráfica de posición (x, y, z) , 1 quadrotor.	39
3.4. Gráfica de ángulos (<i>Pitch, Roll, Yaw</i>), 1 quadrotor.	40
3.5. Gráfica de fuerzas, 1 quadrotor.	40
3.6. Seguimiento de trayectoria 1 quadrotor, con perturbaciones.	41
3.7. Plano (x,y) 1 quadrotor, con perturbaciones.	42
3.8. Gráfica de posición (x, y, z) , 1 quadrotor, con perturbaciones.	42
3.9. Gráfica de ángulos (<i>Pitch, Roll, Yaw</i>), 1 quadrotor, con perturbaciones.	43

3.10. Gráfica de fuerzas, 1 quadrotor, con perturbaciones.	43
4.1. Topología de comunicación donde todos conocen la referencia.	54
4.2. Seguimiento, formación fija, todos conocen referencia	55
4.3. Plano (x,y), formación fija, todos conocen referencia	55
4.4. Gráfica de posición, formación fija, todos conocen referencia	56
4.5. Gráfica de ángulos, formación fija, todos conocen referencia	56
4.6. Gráfica de fuerzas, formación fija, todos conocen referencia	57
4.7. Topología de comunicación con Árbol de Expansión.	58
4.8. Seguimiento, formación fija, árbol de expansión	59
4.9. Plano (x,y), formación fija, árbol de expansión	59
4.10. Gráfica de posición, formación fija, árbol de expansión	60
4.11. Gráfica de ángulos, formación fija, árbol de expansión	60
4.12. Gráfica de fuerzas, formación fija, árbol de expansión	61
4.13. Topologías de comunicación cambiantes.	62
4.14. Seguimiento, formación fija, topología cambiante	63
4.15. Plano (x,y), formación fija, topología cambiante	63
4.16. Gráfica de posición, formación fija, topología cambiante	64
4.17. Gráfica de ángulos, formación fija, topología cambiante	64
4.18. Gráfica de fuerzas, formación fija, topología cambiante	65
4.19. Seguimiento, formación orientada	66
4.20. Plano (x,y), formación orientada .	67
4.21. Gráfica de posición, formación orientada	67
4.22. Gráfica de ángulos, formación orientada .	68
4.23. Gráfica de fuerzas, formación orientada	68
5.1. Esquema de distancias.	72
5.2. Línea singular.	73
5.3. Retrato de fase	75
5.4. Seguimiento, Evasión	76

5.5. Plano (x,y), evasión	76
5.6. Gráfica de posición, evasión	77
5.7. Gráfica de ángulos, evasión	77
5.8. Gráfica de fuerzas, evasión	78

Capítulo 1

Introducción

Los vehículos aéreos no tripulados o UAV's (Unmanned Aerial Vehicles) cada vez son más populares en el sector aeronáutico debido a la diversidad de ventajas que presentan respecto a las aeronaves tripuladas. Entre ellas cabe destacar la reducción de peso, dimensiones y costes, así como la capacidad para volar en entornos complejos.

Existen variedad de aplicaciones para los UAV's, como son, en el campo militar para labores de espionaje, reconocimiento o búsqueda y rescate, en alguna zona forestal, por ejemplo, en esta situación en especial, el uso de un grupo de UAV's en formación es más eficiente en cuanto al área de visión que se puede abarcar, en un único viaje. Otra ventaja muy importante en este rubro militar es que los UAV's evitan poner en peligro la vida de quienes los manejan o supervisan, con un plan de misión para definir la trayectoria del grupo, el personal encargado estaría solamente supervisando el funcionamiento. Dentro del campo industrial, el manejo de materiales peligroso se realiza mediante UAV's, en una situación como esta, un grupo en formación tendría mayor eficacia operativa, al poder cargar más peso en un solo viaje. Como algunas aplicaciones civiles y comerciales, también son usados para vídeos y fotografías aéreas, extinción y ubicación de incendios y reconocimiento en el caso de accidentes y catástrofes, en las cuales, como ya se ha mencionado, con un grupo se puede abarcar más área de visión e inspeccionar cualquier zona con mayor rapidez.

En función del tipo de despegue, los UAV's se pueden clasificar en dos grupos, despegue vertical y no vertical. Lo cual infiere directamente con su capacidad de despegue y aterrizaje de cualquier zona, los no verticales son todos aquellos aeroplanos que necesitan de una pista. Por otro lado los helicópteros convencionales pertenecen al primer grupo. De la misma forma los multicopteros pertenece al grupo de despegue vertical, a diferencia de un helicóptero convencional estos cuentan con dos o más rotores.

De entre la familia de los multicopteros tomamos el *quadcopter* o quadrotor el cual es un UAV que se considera equilibrado en sus características debido a su sencillez mecánica, coste de los materiales necesarios para su construcción y eficiencia en cuanto a peso y consumo de corriente.

Un trirotor, a diferencia, requiere de un servomotor en uno de los brazos para hacer movimientos de guiñada, disminuyendo así su sencillez mecánica y capacidad de carga con respecto al quadrotor, un hexarotor por otro lado tiene mayor capacidad de carga que el quadrotor pero su estructura también aumenta de peso con mayor numero de brazos y motores necesarios, aumentando así, su consumo de energía y costo por construcción con respecto a un quadrotor.

El control de movimiento en los helicópteros convencionales se realiza a través del cambio de ángulo de paso en la pala, manteniendo velocidad de giro constante, en cambio el control de movimiento en el quadrotor se consigue a través de la variación en las velocidades de los rotores, como se verá más adelante. Por lo que en el helicóptero convencional, se requieren elementos mecánicos que el quadrotor no necesita. Sin embargo, el control del quadrotor es mucho más complejo por el efecto de acoplamiento que aparece en la dinámica de los rotores.

1.1. Estado del Arte

En el ámbito del control de UAV's, uno de los sistemas más estudiados han sido los aviones. Sin embargo, un tipo de aeronave que actualmente está siendo muy referenciada en el ámbito de control es el quadrotor. En comparación con los aeroplanos, este tipo de aeronave posee una mayor agilidad para maniobrar. Sin embargo, su control se hace mucho más complejo, entre otros motivos, por la mayor inestabilidad de su dinámica.

Desde el punto de vista de control, la construcción de este tipo de UAV está lejos de simplificar el problema, en realidad sucede lo contrario. Esto se debe a que los pares y fuerzas necesarios para controlar el sistema son aplicados no sólo a través de efectos aerodinámicos, sino también a través del efecto de acoplamiento que aparece entre la dinámica de los rotores y la del cuerpo de la maqueta.

Estos efectos de acoplamiento tienen implicaciones en el control de la dinámica del sistema. Por ejemplo, si se consideran como salidas a controlar la posición y el ángulo de guiñada, una linealización por realimentación de estados de la dinámica completa del quadrotor da lugar a una matriz singular haciendo que el desacoplamiento entrada-salida sea inviable, por lo

que no se puede emplear esta técnica directamente [1]. Este hecho, unido con las incertidumbres de modelado, hace que el sistema sea incluso más difícil de controlar que un helicóptero convencional, al menos empleando técnicas básicas de control.

Varios métodos de control se han aplicado a esta aeronave como linealización por retroalimentación de estados [1], [2] y *backstepping* [3], entre otros. Sin embargo, la aplicación de estos enfoques requiere un conocimiento exacto de la dinámica de la planta. Es necesario entonces un control de vuelo robusto capaz de resistir el efecto de las incertidumbres paramétricas, dinámicas no modeladas, y la presencia de viento o turbulencia. Se han logrado resultados satisfactorios tanto en la estabilización [4] como en el control de seguimiento de referencia [5], [6], [7]. Con el fin de hacer cada vez más robusto este tipo de sistemas se han generado estimadores para el viento [8].

Por otro lado, en la parte de la formación existen diversidad de enfoques para lograr consenso en alguna variable de interés y así avanzar en formación. En especial un enfoque, independiente del algoritmo de consenso utilizado, es el enfoque líder-seguidor el cual ha sido aplicado a sistemas de robots móviles [9] o simplemente a algún sistema no lineal [10]. Este enfoque ha sido aplicado a quadrotors con resultados satisfactorios pero con poca robustez [11]. El reto en los últimos años ha sido el lograr un seguimiento de trayectoria en formación, con topologías cambiantes y un control robusto [12].

1.2. Motivación

En general, a diferencia de los sistemas con un solo agente, el trabajo que puede realizar un sistema de vehículos autónomos permite una mayor eficacia operativa a través de trabajo en equipo cooperativo. Por ejemplo en una situación en la cual sea necesario mover residuos peligrosos, un grupo en formación puede cargar más peso que un solo agente, en una situación de búsqueda y rescate, un grupo con una formación orientada puede cubrir más área con una sola trayectoria de referencia, esto indica que se necesita un solo algoritmo de generación de trayectoria para llevar varios agentes con menos cálculos realizados y menor información transmitida que si fueran agentes independientes.

Para controlar a un grupo de agentes existen dos enfoques de control, centralizado y distribuido, en un sistema de control centralizado debe existir una computadora o un cerebro encargado de medir y controlar todo el sistema al mismo tiempo, este tiene que ser capaz de analizar las mediciones recibidas y enviar señales de control a cada uno de los agentes, el flujo de información debe ser en ambas direcciones, mediciones hacia el cerebro y control para los agentes. En un sistema de control distribuido cada agente se encarga de realizar sus

propias mediciones y calcula el control, simplemente con un plan de misión que pueden ser tanto una trayectoria en función del tiempo como puntos de camino (*way points*) por los cuales se desea pasar.

Por otra parte una generación de trayectoria que lleve al grupo a través de estos puntos de camino y sea capaz de evadir obstáculos dentro de un ambiente conocido es muy importante. En el ámbito de control automático, la evasión de obstáculos ha estado conjunta con el control de seguimiento de trayectoria y por ende dependiente del modelo matemático del robot móvil en cuestión. En el ámbito de inteligencia artificial el problema se ha atacado de manera diferente, generando primeramente la trayectoria independientemente del robot móvil en cuestión y después siguiéndola. En este caso para el seguimiento de trayectoria con un grupo de agentes es preferible mantener el modelo matemático independiente del algoritmo de generación de trayectoria, de esta manera si cambia el número de agentes no se modificaría el algoritmo de generación de trayectoria.

Para esto se plantean los siguientes objetivos:

▪ **Objetivo General**

Realizar el control para un grupo de UAV's con una topología de comunicación cambiante, mediante una ley de control distribuido, para mantener una formación determinada en la generación y seguimiento de una trayectoria de referencia capaz de evadir obstáculos en un espacio de trabajo conocido.

Objetivos Específicos

- Proponer una ley de control para seguimiento de una trayectoria de referencia con un solo quadrotor.
- Extender la ley de control enfocada para un solo quadrotor a un control distribuido mediante los algoritmos de consenso para un grupo con n agentes y topologías cambiantes.
- Proponer un algoritmo para la generación de una trayectoria capaz de evadir obstáculos en un espacio de trabajo conocido, el cual sea independiente de la dinámica del grupo.

Realizar simulaciones para verificar el funcionamiento de las leyes de control y los algoritmos de generación de trayectoria propuestos.

1.3. Estructura de la Tesis

Esta tesis está organizada de la siguiente manera, en el Capítulo 2 se encuentran los temas preliminares para la comprensión y elaboración de la ley de control distribuido, comenzando el capítulo con el modelado del quadrotor hasta su representación en espacio de estados, siguiendo con las técnicas de control no lineal *Block Control* y *Super-Twisting* y por último la teoría de grafos y los algoritmos necesarios para modelar y con controlar un sistema multiagentes.

En el Capítulo 3 se propone la ley de control para seguimiento de trayectoria con un quadrotor presentando posteriormente el análisis de estabilidad y terminando el capítulo con una simulación para verificar el funcionamiento de dicha ley de control.

Para el Capítulo 4 se extiende la ley de control propuesta para un quadrotor a una ley de control distribuido para controlar un grupo de quadrotors presentando posteriormente el análisis de estabilidad para la ley de control distribuido y terminando con algunas simulaciones para verificar y comparar el funcionamiento de la ley de control bajo diferentes topologías de comunicación y tipos de formaciones.

Siguiendo con el Capítulo 5 se presentan los conceptos de navegación y se propone un algoritmo para la generación de una trayectoria capaz de evadir obstáculos en un espacio de trabajo conocido mediante la técnica reactiva de campos potenciales armónicos artificiales, terminando el capítulo con una simulación para verificar el funcionamiento del algoritmo de generación de trayectoria junto con la ley de control distribuido.

Por último en el Capítulo 6 se presentan algunas conclusiones y trabajo futuro.

Capítulo 2

Preliminares

Como ya se mencionó en la introducción, el quadrotor tiene un control muy complejo por el efecto de acoplamiento que aparece en la dinámica de los rotores, por lo cual se tiene un sistema sub-actuado, no lineal y acoplado.

Para aplicar estas técnicas de control es necesario un modelo matemático que describa la dinámica tanto traslacional como rotacional del quadrotor, con el acoplamiento que conlleva el movimiento en los rotores como entrada [13][14].

Además serán necesarias dos técnicas de control para sistemas no lineales que se utilizarán en el diseño del control de posición y seguimiento de una trayectoria para el quadrotor. Por último en la parte del consenso se abordan algunos conceptos de teoría de grafos necesarios para el modelado de la comunicación entre agentes, prosiguiendo con los algoritmos necesarios para llevar un grupo de agentes a consenso.

2.1. Modelado del Quadrotor

2.1.1. Funcionamiento Básico

Entonces, en la elección de un UAV que ofrezca capacidad de despegue y aterrizaje eficiente sin necesidad de pista, la mejor elección es uno con despegue vertical, de una sencillez mecánica y bajo coste de los materiales para su construcción, por todo lo anterior la mejor opción es el quadrotor, otra ventaja para el quadrotor es que los motores a pares permiten directamente llevar a cero la sumatoria de los torques, colocando en pares giros a favor o en contra de las manecillas del reloj, esto significa que es posible mantener un frente deseado.

El quadrotor está construido con cuatro motores, girando así, dos a favor de las manecillas del reloj y dos en contra, se cuenta con dos configuraciones dependiendo de hacia donde se requiere el frente, estas configuraciones se muestran en Fig. 2.1:

Estas configuraciones son muy similares una de otra, pero la forma en que se consiguen sus

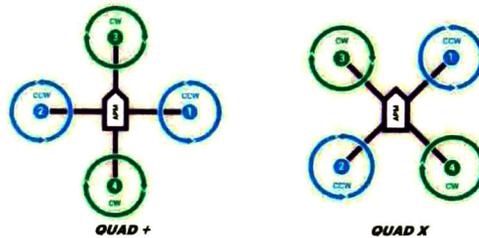


Figura 2.1: Tipo de configuración.

movimientos varia un poco, la configuración dependerá entonces de la aplicación y el frente deseado para el quadrotor.

Dentro del argot aeronáutico se les nombra a los movimientos angulares en los ejes del quadrotor como en Fig. 2.2

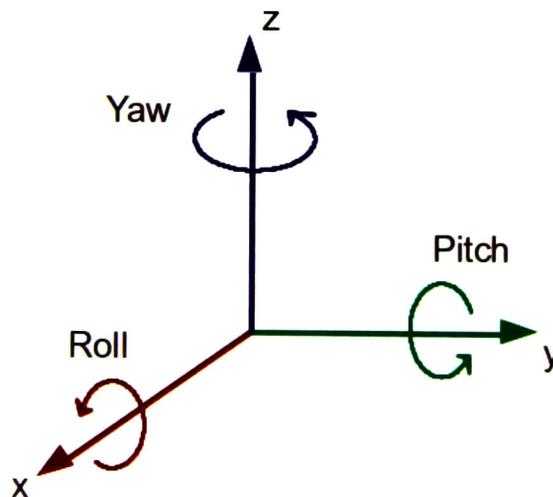


Figura 2.2: *Pitch, Roll, Yaw.*

Se le llama *Pitch* al ángulo del eje hacia la izquierda (levantar o bajar la "nariz"), *Roll* el ángulo con respecto al eje hacia el frente y *Yaw* a la orientación en el eje hacia arriba del quadrotor.

El movimiento del quadrotor es a través del aumento o disminución simultáneo en las velocidades angulares de los motores, lo que se traduce directamente en una fuerza siempre dirigida hacia el eje z en el sistema de coordenadas local acoplado al centro de masa del quadrotor, la cual es una fuerza resultante de la sumatoria de las 4 fuerzas que entregan los motores.

Para despegar se aumenta la velocidad simultáneamente en los 4 motores a iguales cantidades hasta que la fuerza resultante sea capaz de mantener el peso del quadrotor y una vez superado el peso por la fuerza generada por los motores se logra despegar. De la misma manera disminuir simultáneamente las 4 fuerzas en iguales cantidades para poder aterrizar.

Los motores están enumerados como en la Fig. 2.3 (a), y los ejes como en Fig. ??, entonces, para generar un movimiento en el eje y es necesario rotar en *Roll* un cierto ángulo, el cual entregará una componente en el eje z para mantener altura y una componente en el eje y para avanzar, lo cual se logra aumentando en el motor 1 la misma cantidad que se disminuye en el motor 3 y viceversa para avanzar hacia el lado contrario. De la misma manera se puede avanzar en el eje x pero rotando en *Pitch* aumentando en 2 la misma cantidad que se disminuye en 4 y viceversa para avanzar hacia el lado contrario.

Como en todo momento un par de motores gira a favor de las manecillas del reloj y un par en contra, la sumatoria de torques resultante es cero, lo cual indica que para poder rotar en *Yaw* se tiene que aumentar el giro en los dos motores hacia donde se quiera girar y disminuir los contrarios simultáneamente en las mismas cantidades.

2.1.2. Análisis de Cuerpo Libre

Para el análisis de cuerpo libre son considerados dos marcos de referencia principales: el marco de referencia global $E^E(O^E, e_1^E, e_2^E, e_3^E)$, fijo en tierra y el marco de referencia local $E^B(O^B, e_1^B, e_2^B, e_3^B)$, fijo en el centro de masa del quadrotor, como en Fig. 2.4.

La posición absoluta del quadrotor está expresada con respecto al marco de referencia global E^E como $X_E = [x, y, z]^T$ y su pose por los ángulos de Euler como $\Theta = [\phi, \theta, \psi]^T$. Los ángulos de Euler también son llamados *Roll* $\equiv \phi$, *Pitch* $\equiv \theta$, *Yaw* $\equiv \psi$.

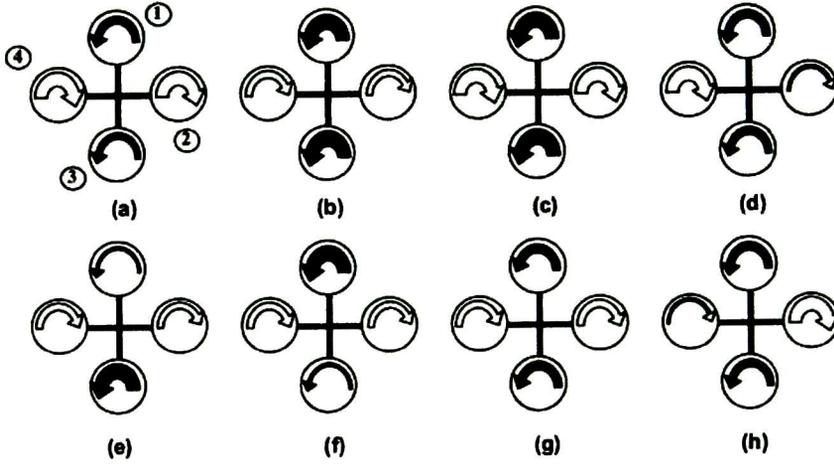


Figura 2.3: a) *Yaw* (A favor), b) *Yaw* (En contra), c) *Arriba*, d) *Roll* (A Favor), e) *Pitch* (A favor), f) *Pitch* (En contra), g) *Abajo*, h) *Roll* (En contra).

El quadrotor es un cuerpo rígido con 6 grados de libertad de acuerdo a: 3 traslacionales $\nu = [u, v, w]^T$ (velocidad lineal) y 3 rotacionales $\Omega = [p, q, r]^T$ (velocidad rotacional), con la orientación del marco de referencia local a global dado por las matrices de rotación $R(\Theta)^1$ y $M(\Theta)$, las cuales son matrices de transformación de velocidad lineal y rotacional respectivamente.

$$R(\Theta) = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\theta & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\theta & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}$$

$$M(\Theta) = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix}$$

El modelo dinámico se obtiene usando el formalismo Newton-Euler en el marco de referencia local E^B , en cuanto a la sumatoria de fuerzas y momentos externos aplicados al centro

¹ $C_x = \text{Cos}(x)$ y $S_x = \text{Sen}(x)$

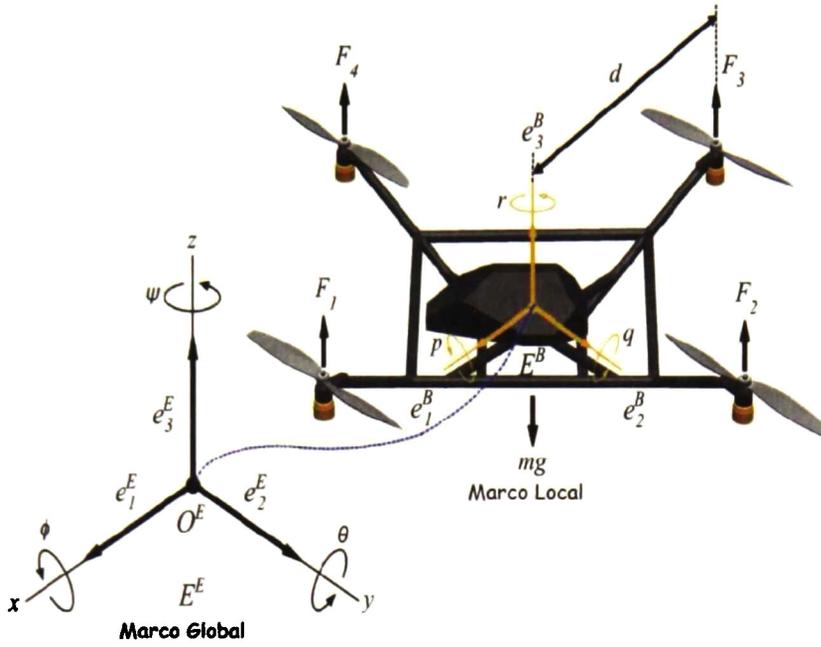


Figura 2.4: Marcos de referencia principales.

de masa. Por lo tanto, las ecuaciones dinámicas de movimiento se describen por:

$$\begin{aligned} \sum F_{ext} &= m\dot{\nu} + \Omega \times m\nu, \\ \sum T_{ext} &= I\dot{\Omega} + \Omega \times I\Omega. \end{aligned} \tag{2.1}$$

Donde $I \equiv \text{diag}(I_x, I_y, I_z)$ y m son la matriz de inercia y la masa de quadrotor, respectivamente. Con las fuerzas y torques externos representados por:

$$\begin{aligned} \sum F_{ext} &= F_{prop} - F_{aero} - F_{grav}, \\ \sum T_{ext} &= T_{prop} - T_{aero} - T_{gyro}. \end{aligned} \tag{2.2}$$

Donde F_{prop} y T_{prop} son las fuerzas y torques producidos por las hélices o propelas, respectivamente. F_{grav} es el efecto de la fuerza de gravedad con $G = [0, 0, 9.81]^T m/s^2$. F_{aero} y T_{aero} son las fuerzas y torques aerodinámicos actuando sobre el UAV en ese momento, respectivamente. Y T_{gyro} define los efectos giroscópicos resultantes de las rotaciones en las hélices. Estas variables son definidas como:

$$F_{prop} = \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^4 F_i \end{bmatrix}, T_{prop} = \begin{bmatrix} d(F_4 - F_2) \\ d(F_3 - F_1) \\ c \sum_{i=1}^4 (-1)^{i+1} \omega_i^2 \end{bmatrix}, F_{grav} = mR^T G,$$

$$F_{aero} = \begin{bmatrix} A_u \\ A_v \\ A_w \end{bmatrix}, T_{aero} = \begin{bmatrix} A_p \\ A_q \\ A_r \end{bmatrix}, T_{gyro} = \sum_{i=1}^4 J_R(\Omega x e_3) (-1)^{i+1} \omega_i.$$

Donde d es la distancia del centro de masa al centro del rotor, c es el factor de arrastre y J_R es el factor de inercia. Las funciones aerodinámicas $A_F = [A_u, A_v, A_w]^T$ y $A_T = [A_p, A_q, A_r]^T$ son calculadas como $A_F = K_t(V - V_{air})$ y $A_T = K_r(\Omega - \Omega_{air})$ donde $\{K_t, K_r\}$ son matrices diagonales afectadas por la densidad del aire y $\{V_{air}, \Omega_{air}\}$ son las velocidades del aire traslacionales y rotacionales respectivamente.

Usando (2.1) y (2.2) se obtienen las ecuaciones que describen la dinámica del quadrotor con respecto al marco de referencia E^E

$$\begin{aligned} \ddot{X} &= \frac{1}{m} R[F_{prop} - F_{aero} - G], \\ \ddot{\Theta} &= (IM)^{-1}(T_{prop} - T_{aero} - T_{gyro} - M\dot{\Theta}xIM\dot{\Theta} - I \left(\frac{\partial M}{\partial \theta} \dot{\theta} + \frac{\partial M}{\partial \phi} \dot{\phi} \right) \dot{\Theta}). \end{aligned} \quad (2.3)$$

2.1.3. Modelo en Espacio de Estados

Antes de seguir a la representación en espacio de estados, algo muy importante a destacar es que haciendo la consideración de que los ángulos *Pitch* y *Roll* son muy pequeños al estar flotando, la matriz $M(\Theta)$ puede ser considerada como la identidad.

$$M(\Theta) = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\phi S_\theta \\ 0 & -S_\phi & C_\phi C_\theta \end{bmatrix} \approx I$$

Esto debido a que $\text{Cos}(0) = 1$ y $\text{Sen}(0) = 0$. Así se obtiene un modelo en espacio de estados de la siguiente manera:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= f_1(x_2, x_3, x_4) + g_1(x_3)\varphi_1, \\ \dot{x}_3 &= x_4, \\ \dot{x}_4 &= f_2(x_2, x_3, x_4) + g_2\varphi_2. \end{aligned} \quad (2.4)$$

Donde:

$x_1 = [x, y, z]^T$ Vector de posición lineal,
 $x_2 = [\dot{x}, \dot{y}, \dot{z}]^T$ Vector de velocidad lineal,
 $x_3 = [\phi, \theta, \psi]^T$ Vector de posición rotacional,
 $x_4 = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ Vector de velocidad rotacional,
 $X = [x_1, x_2, x_3, x_4]^T$ Vector de estado.

Con:

$$\begin{aligned} f_1 &= \begin{bmatrix} A_u \\ A_v \\ A_w - g \end{bmatrix}, & g_1 &= \begin{bmatrix} \frac{U_1}{m} & 0 & 0 \\ 0 & \frac{U_1}{m} & 0 \\ 0 & 0 & \frac{C_\phi C_\theta}{m} \end{bmatrix}, & \varphi_1 &= \begin{bmatrix} u_x \\ u_y \\ U_1 \end{bmatrix} \\ f_2 &= \begin{bmatrix} \dot{\theta}\psi \frac{I_y - I_z}{I_x} - \dot{\theta} \frac{J_R}{I_x} w + A_p \\ \dot{\phi}\psi \frac{I_z - I_x}{I_y} + \dot{\phi} \frac{J_R}{I_y} w + A_q \\ \dot{\theta}\dot{\phi} \frac{I_z - I_y}{I_x} + A_r \end{bmatrix}, & g_2 &= \begin{bmatrix} \frac{1}{I_x} & 0 & 0 \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{I_z} \end{bmatrix}, & \varphi_2 &= \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix} \end{aligned} \quad (2.5)$$

Para configuración *quad+* se definen las entradas como:

$$\begin{aligned} U_1 &= F_1 + F_2 + F_3 + F_4, \\ U_2 &= d(F_4 - F_2), \\ U_3 &= d(F_3 - F_1), \\ U_4 &= -\frac{c}{b}F_1 + \frac{c}{b}F_2 - \frac{c}{b}F_3 + \frac{c}{b}F_4, \\ w &= -w_1 + w_2 - w_3 + w_4. \end{aligned} \quad (2.6)$$

Y los pseudo controles:

$$\begin{aligned} u_x &= C_\psi S_\theta C_\phi + S_\psi S_\phi, \\ u_y &= S_\psi S_\theta C_\phi - C_\psi S_\phi. \end{aligned} \tag{2.7}$$

Entonces la fuerza proporcionada por el i -ésimo rotor está dada por $F_i = b\omega_i^2$ con b siendo el factor de empuje proporcionado por la hélice.

En las entradas (2.6) se tiene que U_1 impacta directamente en el movimiento para subir o bajar en z , siempre positiva o igual a cero, U_2 para avanzar en x , dependiendo de su signo, U_3 de la misma manera pero en el eje y y U_4 para el cabeceo que es la orientación.

2.2. Block Control

En el Control a Bloques o *BlockControl* [15] es necesario tener el sistema a controlar en una forma específica, a bloques. Este es un principio que nos permite imponer una dinámica deseada bajo un control virtual entre los bloques, esto limita la aplicación a sólo cierta clase de sistemas bajo una representación definida que se muestra continuación.

2.2.1. Representación a Bloques de una Clase de Sistemas no Lineales

Se analizará la descomposición a bloques, considerando el siguiente sistema no lineal sujeto a incertidumbre:

$$\dot{x} = f(x, t) + B(x, t) + g(x, t). \tag{2.8}$$

Donde $x \in X \subset \mathbb{R}^n$ es el vector de estado, $u \in U \subset \mathbb{R}^m$ es el vector de entradas, que está acotado por:

$$\|u_i\| \leq U_0 \tag{2.9}$$

Con $U_0 > 0$, $u = (u_1, \dots, u_m)$.

El mapeo desconocido $g(x, t)$ está formado por perturbaciones externas y variaciones paramétricas. Suponemos que los campos vectores $f(x, t)$, $g(x, t)$ y las columnas de $B(x, t)$ son mapeos suaves y acotados de clase $C_{[0, \infty)}^\infty$, $f(0, t) = 0$ y $\text{rango}(B(x, t)) = m$ para todo $x \in X$ y $t \geq 0$.

La característica esencial del método propuesto es la conversión del sistema (2.8) a la configuración a bloques que consiste de r bloques:

$$\begin{aligned} \dot{x}_1 &= f_1(x_1, t) + B_1(x_1, t)x_2 + g_1(x_1, t), \\ \dot{x}_i &= f_i(\bar{x}_i, t) + B_i(\bar{x}_i, t)x_{i+1} + g_i(\bar{x}_i, t), \\ \dot{x}_r &= f_r(\bar{x}_r, t) + B_r(\bar{x}_r, t)u + g_r(\bar{x}_r, t), \quad i = 2, \dots, r-1. \end{aligned} \quad (2.10)$$

Donde el vector x es descompuesto como $x = (x_1, x_2, \dots, x_r, x_{r+1})^T$ $\bar{x}_i = (x_1, \dots, x_i)^T$ $i = 2, \dots, r$, x_i es un vector de $n_i \times 1$ y los índices (n_1, \dots, n_r) definen la estructura del sistema y satisfacen la siguiente relación:

$$n_1 \leq \dots \leq n_r \leq m \quad (2.11)$$

Y haciendo la sumatoria $\sum_{i=1}^r n_i = n$, obtenemos el estado completo.

La matriz B_i , multiplicada por x_{i+1} en cada i -ésimo bloque de (2.10), tiene rango completo, que es:

$$\text{rango}(B_i) = n_i \quad (2.12)$$

para todo $x \in X \subset \mathbb{R}^n$ y $t \in [0, \infty)$, $i = 1, \dots, r$.

El procedimiento de reducir el sistema (2.8) a la forma de bloques (2.10) basado en el método de transformación integral es presentado en [16] y [15].

La relación (2.11) nos indica $n_i = n_{i+1}$ o $n_i < n_{i+1}$ primero se considerara la planta con la estructura:

$$n_1 < \dots < n_r < m \quad (2.13)$$

2.2.2. Transformación Recursiva a Bloques

Para comenzar se expresan las siguiente suposiciones sobre los límites en los términos desconocidos en (2.8):

Suposición 2.1. Existen constantes positivas $\bar{q}_{i,j}$ y \bar{d}_i tal que:

$$\begin{aligned} \|g_1(\bar{x}_1, t)\| &\leq \bar{q}_{1,1} \|x_1\| + \bar{d}_1, \\ \|g_2(\bar{x}_2, t)\| &\leq \bar{q}_{2,1} \|x_1\| + \bar{q}_{2,2} \|x_2\| + \bar{d}_2, \\ \|g_i(\bar{x}_i, t)\| &\leq \sum_{j=1}^i \bar{q}_{i,j} \|x_j\| + \bar{d}_i, \quad i = 3, \dots, r-1 \end{aligned}$$

Considerando la estructura (2.13), se presenta la siguiente transformación recursiva.

$$\begin{aligned} z_1 &= x_1 := \Phi_1(x_1, t), \\ z_2 &= \tilde{B}_2(\bar{x}_1, t)x_2 + \begin{bmatrix} f_1(x_1, t) + k_1\Phi_1(x_1, t) \\ 0 \end{bmatrix} := \Phi_2(\bar{x}_2, t), \\ z_{i+1} &= \tilde{B}_{i+1}(\bar{x}_i, t)x_{i+1} + \begin{bmatrix} f_i(x_i, t) + k_i\Phi_i(x_i, t) \\ 0 \end{bmatrix} := \Phi_{i+1}(\bar{x}_{i+1}, t). \end{aligned} \quad (2.14)$$

Con $i = 3, \dots, r-1$, donde z_i es un vector de $n_i \times 1$ con nuevas variables $k_i > 0$, $\tilde{B}_{i+1} = \begin{bmatrix} \bar{B}_i \\ E_{i,2} \end{bmatrix}$, $E_{i,2} = [0 \quad I_{n_{i+1}-n_i}]$, $E_{i,2} \in \mathbb{R}^{n_{i+1}-n_i \times n_{i+1}}$. $I_{n_{i+1}-n_i}$ es la matriz identidad.

La transformación (2.14) reduce el sistema (2.8) a la forma deseada siguiente:

$$\begin{aligned} \dot{z}_1 &= -k_1 z_1 + E_{11}z_2 + \bar{g}_1(z_1, t), \\ \dot{z}_i &= -k_i z_i + E_{i1}z_{i+1} + \bar{g}_i(z_i, t), \\ \dot{z}_r &= \bar{f}_r(z, t) + \bar{B}_r(z, t)u + \bar{g}_r(z, t), \quad i = 2, \dots, r-1. \end{aligned} \quad (2.15)$$

Donde $z = (z_1, \dots, z_r)^T$ $\bar{f}_r(z, x_{r+1}, t)$ función acotada, $\text{rango}(\bar{B}_r) = n_1$, $\bar{B}_r = \tilde{B}_{r-1}B_r$.

2.3. Modos Deslizantes

El enfoque de control por modos deslizantes o *Sliding Modes* es reconocido como una de las herramientas más eficientes para el diseño de controladores robustos para sistemas dinámicos complejos de orden superior, no lineales y que operan bajo condiciones de incertidumbre [17]. El término modo deslizante apareció por primera vez en el contexto de los sistemas con relé. Puede suceder que el control, como una función del estado del sistema, cambia a alta frecuencia (teóricamente infinita) y este movimiento se conoce como modo deslizante. Esto puede ser aplicado en su forma más simple al sistema de relé de primer orden con la variable de estado $x(t)$:

$$\dot{x}(t) = f(x) + u \quad |f(x)| < f_0 = \text{constante} \quad (2.16)$$

Con el control u como la función relé (Fig. 2.5), el error de seguimiento $e(t) = r(t) - x(t)$ y $r(t)$ la referencia, $u = u_0 \text{sign}(e)$ con u_0 constante.

$$u = \begin{cases} u_0 & e > 0 \\ -u_0 & e < 0 \end{cases} \quad \text{ó } u = u_0 \text{sign}(e) \quad (2.17)$$

Los valores de $e(t) = r(t) - x(t)$ y $\dot{e}(t) = \dot{r}(t) - f(x, t) + u_0 \text{sign}(e)$ tienen diferentes signos dependiendo de si $u_0 > f_0 + |\dot{r}|$. Esto significa que la magnitud del error de seguimiento decae

a una tasa finita y el error es igual a cero de forma idéntica después de un intervalo de tiempo finito T (Fig. 2.6). Al movimiento en $t > T$ se le llama modo deslizante.

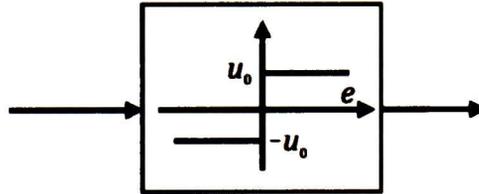


Figura 2.5: Función relé.

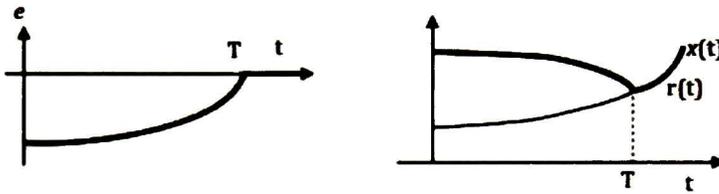


Figura 2.6: Gráfica de error.

En general, condición en (2.17) se clasifica como control discontinuo, el cambio (en teoría) es en frecuencia infinita. Para evitar *chattering*¹ se propusieron algunos enfoques. La idea principal era cambiar la dinámica en una pequeña vecindad de la superficie discontinua con el fin de evitar la discontinuidad en tiempo real y al mismo tiempo para preservar las propiedades principales de todo el sistema. En particular, los modos deslizantes de orden superior (HOSM, Hight Order Sliding Modes) generalizan la idea básica de modo deslizante actuando sobre las derivadas con respecto al tiempo del sistema y no sólo sobre la superficie. Mantiene las principales ventajas del enfoque original, al mismo tiempo que eliminar totalmente el *chattering* y prevé incluso una mayor precisión en la realización. La siguiente subsección presenta el algoritmo de *Super-Twisting* que se utilizará para fines de control.

2.3.1. Algoritmo de Super-Twisting

Un algoritmo importante para el control y la observación usando un modo deslizante de segundo orden (SOSM, Second Order Sliding Modes) es el denominado algoritmo de *Super*

¹ En español se traduce como cascabeleo, es la oscilación o cambio continuo como vibración en la salida del sistema.

Twisting [18], que es descrito por la inclusión diferencial:

$$\begin{aligned} \dot{x}_1 &= -k_1|x_1|^{1/2}\text{sign}(x_1) + x_2 + \varrho_1(x, t) \\ \dot{x}_2 &= -k_3\text{sign}(x_1) + \varrho_2(x, t) \end{aligned} \quad (2.18)$$

donde x_i son las variables de estado escalares, k_i son ganancias para ser diseñadas y ϱ_i son términos de perturbación.

El siguiente teorema asegura la robustez de la estabilidad asintótica a nivel global de (2.18) en un tiempo finito cuando se seleccionan las ganancias suficientemente altas.

Teorema 2.1. Supóngase que las perturbaciones en (2.18) son globalmente acotadas por:

$$\varrho_1 \leq \delta_1|x_1|^{1/2}, \varrho_2 \leq \delta_2 \quad (2.19)$$

por alguna constantes $\delta_1, \delta_2 \geq 0$. Entonces el origen $x = 0$ es un punto de equilibrio que es globalmente asintóticamente estable si las ganancias satisfacen:

$$\begin{aligned} k_1 &> 2\delta_1 \\ k_3 &> k_1 \frac{5\delta_1 k_1 + 6\delta_2 + 4(\delta_1 + \delta_2/k_1)^2}{2(k_1 - 2\delta_1)} \end{aligned} \quad (2.20)$$

Además, todas las trayectorias convergen en un tiempo finito al origen, delimitada por la parte superior $T = \frac{2V^{\frac{1}{2}}(x_0)}{\tilde{\gamma}}$, donde x_0 es el estado inicial y $\tilde{\gamma}$ es una constante que depende de las ganancias k_1, k_3 y los coeficientes de perturbación δ_1, δ_2 . Esto, junto con la función de Lyapunov V se pueden ver en la demostración del teorema 2.1 en [19].

2.4. Consenso

En esta sección se analizaran los axiomas, teoremas y algoritmos necesarios para modelar y controlar un sistema multiagentes. La demostración de los mismos se puede ver en [20]. Así como algunos conceptos básicos de teoría de grafos, necesarios para el modelado de la comunicación entre agentes.

Dentro de un sistema multiagentes es necesario un control cooperativo, ya sea este centralizado o distribuido, un problema fundamental e inherente a este tipo de sistemas es abordado por el siguiente axioma:

Axioma 2.1. [20] La información compartida es una condición necesaria para la cooperación.

Cuando varios agentes están de acuerdo en el valor de una variable de interés se dice que hay *Consenso*.

2.4.1. Teoría de Grafos

En esta sección se abordaran algunos conceptos básicos en *Teoría de Grafos*, es natural modelar el intercambio de información entre vehículos mediante grafos dirigidos o no dirigidos [21]. En un grafo dirigido está indicada la dirección en la que fluye la información y en uno no dirigido simplemente se toma que cada conexión tiene el flujo de la información en ambos sentidos.

Un *Grafo Dirigido* es el par (v_n, ε_n) con $v_n = (1 \dots n)$ siendo un conjunto finito y no vacío de nodos, $\varepsilon_n \subseteq v_n \times v_n$ siendo un conjunto de aristas (edge). La expresión $edge(i, j)$ denota que el vehículo j puede obtener información del vehículo i , con la expresión (i, j) se denota que i es el *Nodo Padre* de j , si $(i, j) \in \varepsilon_n$ entonces i es *Vecino* de j . Cada nodo representa a un agente.

Un *Grafo Ponderado* le da un peso a cada arista, dando mayor importancia al flujo de cierta información.

Un *Camino Dirigido* es una secuencia de aristas de la forma $(i_1, i_2)(i_2, i_3) \dots$, en un grafo dirigido un *Ciclo* es un camino dirigido que comienza y termina en el mismo nodo. Un grafo *Fuertemente Conectado* es en aquel que existe un camino dirigido de cada nodo a todos los otros nodos.

Un *Árbol Dirigido* es un grafo dirigido en el que cada nodo tiene exactamente un padre a excepción de un nodo, llamado la raíz, que no tiene padre y que tiene un camino dirigido a todos los demás nodos. Tenga en cuenta que un árbol dirigido no tiene ciclos porque cada arista está orientada para alejarse de la raíz.

Un *Subgrafo* (v_n^s, ε_n^s) de (v_n, ε_n) es un grafo tal que $v_n^s \subseteq v_n$ y $\varepsilon_n^s \subseteq \varepsilon_n \cap (v_n^s \times v_n^s)$.

Un *Árbol de Expansión Dirigido* (v_n^s, ε_n^s) de un grafo dirigido (v_n, ε_n) es un subgrafo de (v_n, ε_n) tal que (v_n^s, ε_n^s) es un árbol de expansión y $v_n^s = v_n$.

La *Matriz de Adyacencia* $\mathcal{A}_n = [a_{ij}] \in \mathbb{R}^{n \times n}$ de un grafo dirigido (v_n, ε_n) es definida tal que a_{ij} es un peso positivo si $(i, j) \in \varepsilon_n$ y $a_{ij} = 0$ si $(i, j) \notin \varepsilon_n$. Auto aristas no están permitidas, es decir $a_{i,i} = 0$.

Se define a la *Matriz Laplaciana* $\mathcal{L}_n = [\ell_{ij}] \in \mathbb{R}^{n \times n}$ como:

$$\begin{aligned} \ell_{ii} &= \sum_{j=1, j \neq i}^n a_{ij}, \\ \ell_{ij} &= -a_{ij}, \quad i \neq j. \end{aligned} \quad (2.21)$$

Note que si $(i, j) \notin \varepsilon_n$ entonces $\ell_{ij} = 0$ no estaban conectados. La matriz \mathcal{L}_n siempre cumple con las siguientes características:

$$\begin{aligned} \ell_{ij} &\leq 0, \quad i \neq j, \\ \sum_{j=1}^n \ell_{ij} &= 0, \quad i = 1, \dots, n. \end{aligned} \quad (2.22)$$

Para un grafo no dirigido \mathcal{A}_n es simétrica.

Otra forma de obtener la matriz laplaciana directamente con matrices es:

$$\mathcal{L}_n = D - \mathcal{A}_n. \quad (2.23)$$

Donde:

$$\begin{aligned} D &= [d_{ij}] \in \mathbb{R}^{n \times n}, \\ d_{ij} &= 0 \text{ si } i \neq j, \\ d_{ii} &= \sum_{j=1}^n a_{ij}, \quad i = 1 \dots n. \end{aligned}$$

La matriz $\mathcal{L}_n \geq 0$ es semidefinida positiva.

Como ejemplo se muestra el siguiente grafo dirigido con sus respectivas matrices de adyacencia y Laplaciana.

Tabla 2.1: Ejemplo, grafo dirigido y sus matrices respectivas.



$$\mathcal{A}_3 = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathcal{L}_3 = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix}$$

2.4.2. Topología de Comunicación

Para modelar esta interacción entre agentes es necesario definir una *Topología de Comunicación* la cual es modelada mediante un *Grafo*. Una topología de comunicación o *red*, se

define como el conjunto de reglas de interacción y especifican el intercambio de información entre un agente y todos sus vecinos de la red.

Supóngase una red con n agentes y su topología de comunicación representada por el grafo $\mathcal{G}_n \equiv (v_n, \varepsilon_n)$ donde $v_n = 1, \dots, n$ es el conjunto de nodos, y $\varepsilon_n \subseteq \varepsilon_n \times \varepsilon_n$ es el conjunto de aristas. Por ejemplo en la Fig. 2.7 se muestran 3 diferentes topologías de comunicación para 3 agentes.

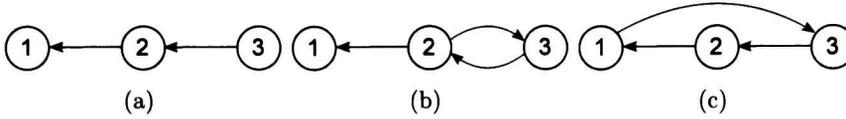


Figura 2.7: Topologías de comunicación para 3 agentes.

El algoritmo de consenso más común y simple es:

$$\dot{x}_i(t) = \sum_{j=1}^n a_{ij}[x_i(t) - x_j(t)], \quad i = 1 \dots n \tag{2.24}$$

Donde a_{ij} es el elemento (i, j) de la matriz de adyacencia $\mathcal{A}_n \in \mathbb{R}^{n \times n}$ asociada a \mathcal{G}_n y x_i es el estado del i -ésimo agente. Si $a_{ij} = 0$ se denota que el agente i no puede recibir información del agente j . La cuestión fundamental es la convergencia, ¿bajo que condiciones los estados de todos los vehículos convergen a un valor común?

Tabla 2.2: Matrices de adyacencia respectivas a los ejemplos en Fig. 2.7.

$$\mathcal{A}_{3,a} = \begin{matrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ \text{(a)} \end{matrix} \quad \mathcal{A}_{3,b} = \begin{matrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix} \\ \text{(b)} \end{matrix} \quad \mathcal{A}_{3,c} = \begin{matrix} \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \\ \text{(c)} \end{matrix}$$

El algoritmo de consenso (2.24) es representado en forma matricial como:

$$\dot{x}(t) = -\mathcal{L}_n x(t). \tag{2.25}$$

Donde $x = [x_1 \dots x_n]^T$ es el vector de estados y $\mathcal{L}_n = [\ell_{ij}] \in \mathbb{R}^{n \times n}$ es el Laplaciano asociado a \mathcal{G}_n .

El consenso es logrado o alcanzado por un equipo de agentes si:

$$\forall x_i(0) \text{ y } \forall i, j = 1, \dots, n, |x_i(t) - x_j(t)| \longrightarrow 0 \text{ conforme } t \longrightarrow \infty. \quad (2.26)$$

Tabla 2.3: Matrices Laplacianas respectivas a los ejemplos en Fig. 2.7.

$$\mathcal{L}_{3,a} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathcal{L}_{3,b} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & -1 & 1 \end{bmatrix} \quad \mathcal{L}_{3,c} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ -1 & 0 & 1 \end{bmatrix}$$

(a) \qquad\qquad\qquad (b) \qquad\qquad\qquad (c)

Para este tipo de sistemas (2.25) donde la topología de comunicación no cambia, es decir el Laplaciano es constante, una condición suficiente pero no necesaria para tener convergencia y por lo tanto consenso, es que el grafo dirigido asociado a \mathcal{L}_n sea fuertemente conectado. La Fig. 2.8 muestra la convergencia para cada uno de los grafos en la Fig. 2.7.

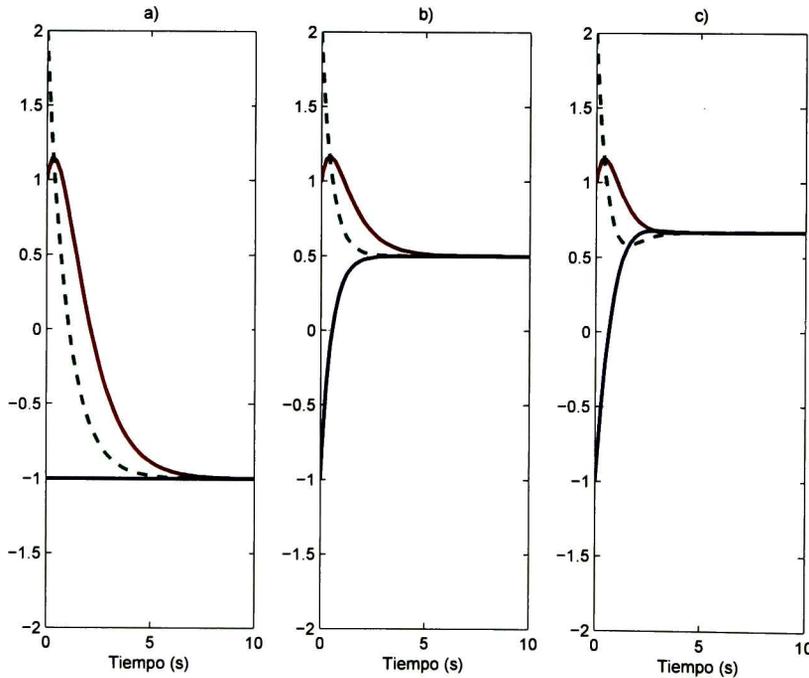


Figura 2.8: Consenso.

Como se puede ver en la Fig. 2.8, los tres ejemplos convergen, pero no todos son fuertemente conectados. La condición que es tanto suficiente como necesaria es que el grafo dirigido asociado a \mathcal{L}_n tenga un árbol de expansión dirigido [20].

La convergencia en este tipo de algoritmo es llevar el estados de todos los agentes a un mismo valor constante y así que estén en consenso. Pero si se busca seguir una trayectoria en un estado deseado para un grupo de agentes es necesario realizar un algoritmo de consenso como el que se analizará a continuación.

2.4.3. Enfoque Líder-Seguidor

En esta sección, el problema de convergencia incluyen tanto la convergencia a un valor común, así como la convergencia del estado común a un valor de referencia variable.

Supóngase un sistema con los mismos n agentes que en la sección anterior, junto con un agente adicional (agente virtual) $n + 1$, el cual actúa como el único líder, entonces se nombra al agente $n + 1$ *líder* del equipo y a los agentes $1, \dots, n$ *seguidores*.

El estado del agente $n + 1$ está definido como $x_{n+1} \equiv x^r \in \mathbb{R}^m$ donde x^r representa el *estado de referencia de consenso*. El estado de referencia de consenso satisface:

$$\dot{x}^r = f(t, x^r). \quad (2.27)$$

Donde $f(t, x^r)$ es acotada, parcialmente continua en t y localmente Lipschitz en x^r . Entonces el problema de seguimiento de una referencia se soluciona si $x_i(t) \rightarrow x^r$ $i = 1, \dots, n$, conforme $t \rightarrow \infty$.

Se usa el grafo $\mathcal{G}_{n+1} \equiv (v_{n+1}, \varepsilon_{n+1})$ para modelar la topología de comunicación entre los $n + 1$ agentes, donde $v_{n+1} = 1, \dots, n + 1$ es el conjunto de nodos, y $\varepsilon_{n+1} \subseteq \varepsilon_{n+1} \times \varepsilon_{n+1}$ es el conjunto de aristas. Sea $\mathcal{A}_{n+1} = [a_{ij}] \in \mathbb{R}^{(n+1) \times (n+1)}$ la matriz de adyacencia asociada al grafo \mathcal{G}_{n+1} , donde $[a_{ij}] > 0$ si $(j, i) \in \varepsilon_{n+1}$ y $[a_{ij}] = 0$ de otra manera, para todo $i = 1, \dots, n$ y $j = 1, \dots, n + 1$ y $a_{(n+1)j} = 0$ para todo $j = 1, \dots, n + 1$. Esto quiere decir que nadie se puede comunicar con el líder puesto que este es virtual ya que es la función de referencia. Sea $\mathcal{L}_{n+1} = [\ell_{ij}] \in \mathbb{R}^{(n+1) \times (n+1)}$ el laplaciano asociado al grafo \mathcal{G}_{n+1} , donde $[\ell_{ij}] = -a_{ij}$ para $i \neq j$, $\ell_{ii} = \sum_{j=1, i \neq j}^{n+1} a_{ij}$ para $i, j = 1, \dots, n + 1$.

Estado de Referencia Constante

En esta sección es considerada una referencia x^r constante, es decir $f(t, x^r) = 0$ en (2.27). Un algoritmo de seguimiento en consenso con un estado de referencia constante se da como:

$$u_i = - \sum_{j=1}^n (x_i - x_j) - a_{i(n+1)}(x_i - x^r), \quad i = 1, \dots, n. \quad (2.28)$$

Donde $x_i \in \mathbb{R}^m$ es el i -ésimo estado y a_{ij} , $i = 1, \dots, n$, $j = 1, \dots, n+1$, es el elemento (i, j) de la matriz de adyacencia $\mathcal{A}_{n+1} \in \mathbb{R}^{(n+1) \times (n+1)}$

Se tiene entonces el siguiente teorema para seguimiento en consenso con un estado de referencia constante:

Teorema 2.2. [20] Sea \mathcal{A}_{n+1} una matriz constante. El problema de seguimiento en consenso con un estado de referencia constante es resuelto con el algoritmo (2.28) si y sólo si el grafo dirigido \mathcal{G}_{n+1} tiene un árbol de expansión dirigido.

El teorema 2.2 es ilustrado por el siguiente ejemplo, en la Fig. 2.9 se muestran 4 topologías de comunicación distintas para el mismo número de agentes.

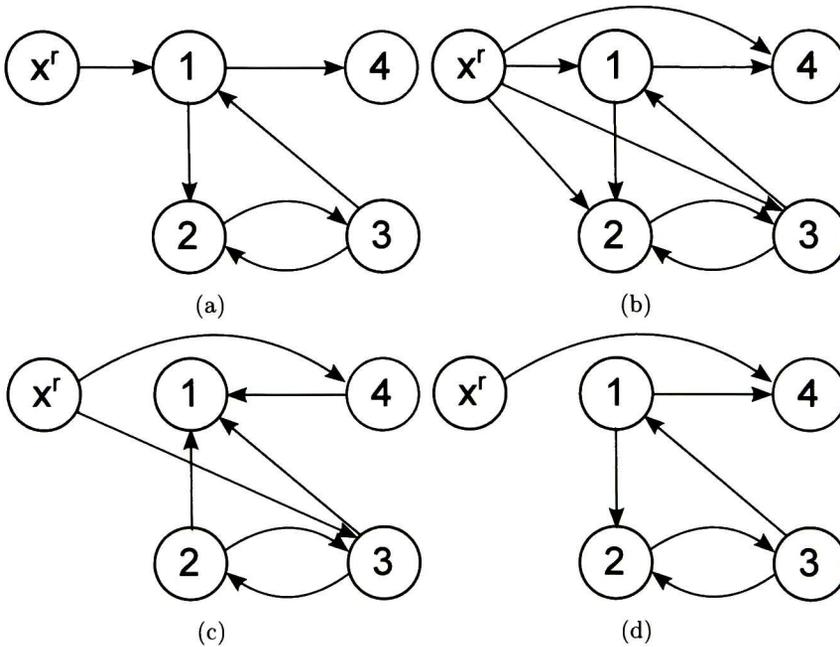


Figura 2.9: Topologías de comunicación líder-seguidor.

Tabla 2.4: Matrices de adyacencia respectivas a los ejemplos en Fig. 2.9.

$$\begin{array}{cc}
\mathcal{A}_{5,a} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \mathcal{A}_{5,b} = \begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\text{(a)} & \text{(b)} \\
\mathcal{A}_{5,c} = \begin{bmatrix} 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \mathcal{A}_{5,d} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\text{(c)} & \text{(d)}
\end{array}$$

Tabla 2.5: Matrices Laplacianas respectivas a los ejemplos en Fig. 2.9.

$$\begin{array}{cc}
\mathcal{L}_{5,a} = \begin{bmatrix} 2 & 0 & -1 & 0 & -1 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \mathcal{L}_{5,b} = \begin{bmatrix} 2 & 0 & -1 & 0 & -1 \\ -1 & 3 & -1 & 0 & -1 \\ 0 & -1 & 2 & 0 & -1 \\ -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\text{(a)} & \text{(b)} \\
\mathcal{L}_{5,c} = \begin{bmatrix} 3 & -1 & -1 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ 0 & -1 & 2 & 0 & -1 \\ 0 & 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} & \mathcal{L}_{5,d} = \begin{bmatrix} 1 & 0 & -1 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \\ -1 & 0 & 0 & 2 & -1 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \\
\text{(c)} & \text{(d)}
\end{array}$$

Considérese un equipo de $n = 4$ agentes, junto con el líder virtual del equipo. El líder virtual del equipo tiene una referencia a consenso constante $x^r = 1$. La Fig. 2.9 muestra, respectivamente, las topologías de interacción entre los cinco vehículos para subcasos (a)-(d), donde nodo $n + 1$ es el líder virtual del equipo. En el subcaso (a), $a_{15} = 1$ y $a_{j5} = 0$, $\forall j \neq 1$, lo que corresponde al caso en que x^r está disponible sólo para el agente 1. En el subcaso (b), $a_{j5} = 1$, $j = 1, \dots, 4$, que corresponde al caso en que x^r está disponible para todos los seguidores. En el subcaso (c), $a_{35} = a_{45} = 1$ y $a_{j5} = 0$, $\forall j \notin \{3, 4\}$, que corresponde al caso en que x^r está disponible sólo para los agentes 3 y 4. En el subcaso (d), $a_{45} = 1$ y $a_{j5} = 0$, $\forall j \neq 4$, lo que corresponde al caso en que x^r está disponible sólo para agente 4.

La Fig. 2.10 muestra, respectivamente, los estados de información de todos los vehículos usando el teorema 2.2. Téngase en cuenta que x_i converge a x^r en cada subcaso excepto en el subcaso (d). Téngase en cuenta también que el nodo *lider* tiene un camino dirigido a todos los demás nodos en los subcasos (a), (b) y (c). Sin embargo, no existe un camino dirigido desde el nodo *lider* a todos los demás nodos en el subcaso (d). Téngase en cuenta que en el subcaso (c), ni 3 ni 4 tiene un camino dirigido a todos los otros seguidores. Sin embargo, como se dijo anteriormente, x_i llega a x^r también en este caso.

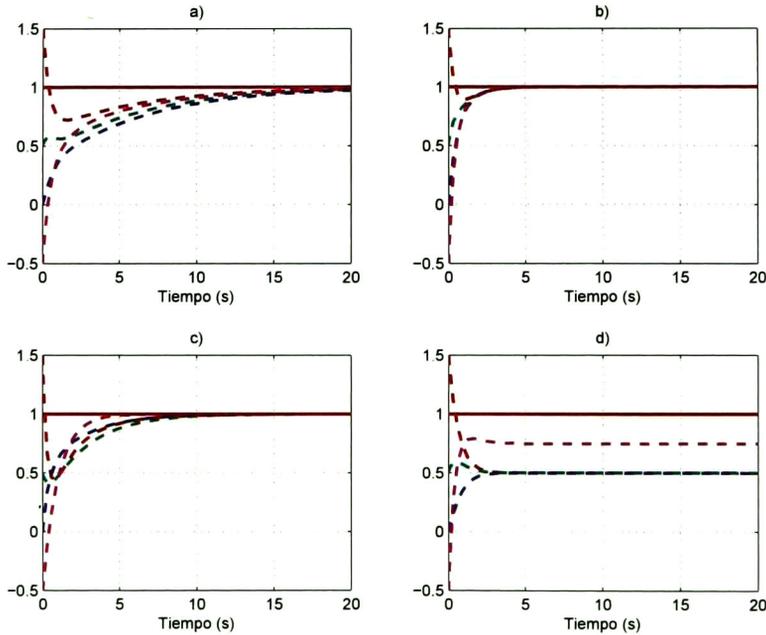


Figura 2.10: Seguimiento en consenso, con una referencia constante, usando algoritmo (2.28).

Estado de Referencia Variante en el Tiempo

En esta sección, se supone que el estado de referencia en consenso podría ser una señal variante en el tiempo. Se supone también que x^r satisface la dinámica dada en (2.27).

Primero se muestra que el algoritmo (2.28) no es suficiente para el seguimiento en consenso con un estado de referencia variante en el tiempo, como se ilustra mediante el siguiente ejemplo.

Sea $x^r = \cos(t)$ y considérese los cuatro subcasos en la Fig. 2.9. Como se muestra en la Fig. 2.11, con el algoritmo (2.28), ninguno de los subcasos resuelve el problema de seguimiento de consenso con un estado de referencia a consenso variante en el tiempo.

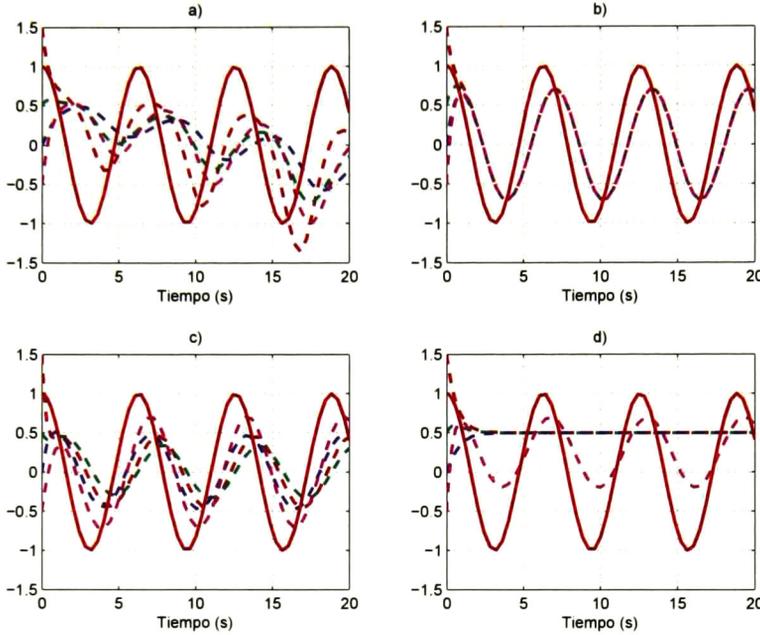


Figura 2.11: Seguimiento en consenso, con una referencia variante en el tiempo, usando algoritmo (2.28).

Para esto se considera el siguiente algoritmo:

$$u_i = a_{i(n+1)}\dot{x}^r - a_{i(n+1)}k_i(x_i - x^r) - \sum_{j=1}^n a_{ij}(x_i - x_j), \quad i = 1, \dots, n. \quad (2.29)$$

Donde k_i es una constante escalar positiva. El algoritmo (2.29) se ilustra mediante el siguiente ejemplo. Sea $x^r = \cos(t)$ y considere los cuatro subcasos en Fig. 2.9. Como se muestra en la Fig. 2.12, con el algoritmo (2.29), el problema de seguimiento de consenso se resuelve sólo en subcaso (b).

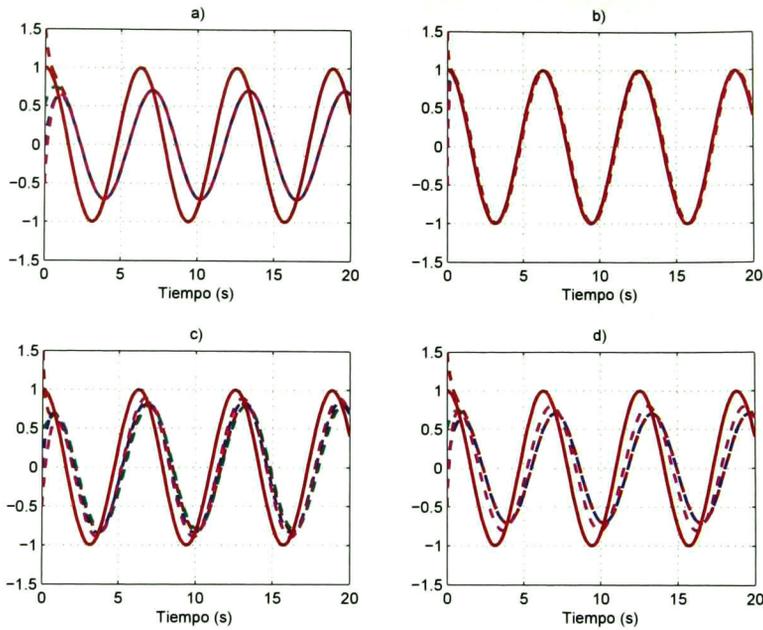


Figura 2.12: Seguimiento en consenso, con una referencia variante en el tiempo, usando algoritmo (2.29).

Se tiene entonces el siguiente teorema para seguimiento en consenso con un estado de referencia variante en el tiempo:

Teorema 2.3. [20] Sea \mathcal{A}_{n+1} una matriz constante. Si $a_{i(n+1)} \equiv 1$, $i = 1, \dots, n$, entonces el problema de seguimiento en consenso con un estado de referencia variante en el tiempo es resuelto con el algoritmo (2.29).

El Teorema 2.3 no se basa en la topología de la interacción entre los seguidores. Todavía Incluso si no hay un intercambio de información entre los seguidores (es decir, $\mathcal{A}_n = 0_{n \times n}$), el algoritmo (2.29) resuelve el problema de seguimiento en consenso con un estado de referencia variante en el tiempo, siempre y cuando x^r y \dot{x}^r estén disponibles para todos los seguidores. Sin embargo, este requisito es bastante restrictivo.

Algoritmo Fundamental de Seguimiento en Consenso

En este apartado, se considera el caso en que x^r y \dot{x}^r están disponibles sólo para un subgrupo de los seguidores. Se propone un algoritmo para seguimiento de referencia en consenso:

$$u_i = \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\dot{x}_j - k_i(x_i - x_j)] + \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\dot{x}^r - k_i(x_i - x^r)], \quad i = 1, \dots, n. \quad (2.30)$$

Donde a_{ij} , $i = 1, \dots, n$, $j = 1, \dots, n + 1$, es el elemento (i, j) de la matriz de adyacencia \mathcal{A}_{n+1} en el tiempo t , k_i es una constante escalar positiva y $\eta_i(t) \equiv \sum_{j=1}^{n+1} a_{ij}(t)$. En este algoritmo el control se depende tanto de la información de los vecinos como de su derivada.

A continuación, se mostraran las condiciones bajo las cuales existe una solución única para cada entrada de control y se realiza el seguimiento de referencia en consenso, sobre una topología de comunicación fija.

Teorema 2.4. [20] Sea \mathcal{A}_{n+1} una matriz constante. Con (2.30), existe una solución única para u_i , $i = 1, \dots, n$ y el problema de seguimiento en consenso es resuelto si y sólo si el grafo dirigido \mathcal{G}_{n+1} tiene un árbol de expansión dirigido.

El algoritmo (2.30) con el teorema 2.4 nos dan condiciones un poco menos restrictivas que el algoritmo (2.29), sin embargo se tiene que comunicar o calcular la derivada también de los vecinos. Pero las condiciones mismas son entregadas por el tipo de trabajo que se vaya a realizar, el siguiente teorema habla de condiciones para una topología cambiante pero constante por partes.

Teorema 2.5. [20] Sea $\mathcal{A}_{n+1}(t)$ una matriz constante por partes y cada no cero y por lo tanto elemento positivo de $\mathcal{A}_{n+1}(t)$ es elegido de un conjunto compacto $[a, \bar{a}]$, donde a y \bar{a} son constantes positivas. Sea t_0 el tiempo inicial. Sea también t_1, t_2, \dots los tiempos de conmutación para $\mathcal{A}_{n+1}(t)$. Con (2.30), existe una solución única para u_i , $i = 1, \dots, n$ y el problema de seguimiento en consenso es resuelto si y sólo si el grafo dirigido cambiante \mathcal{G}_{n+1} tiene un árbol de expansión dirigido a través de cada intervalo $[t_i, t_{i+1})$, $i = 0, 1, \dots$.

Tenga en cuenta que la naturaleza de seguimiento de algoritmo (2.30) requiere condiciones más estrictas para la convergencia sobre la topología de comunicación que aquellas para el algoritmo (2.28), donde el consenso final es una constante. En contraste con el algoritmo (2.29), que requiere que el estado de referencia en consenso variante en el tiempo esté disponible para cada seguidor, el algoritmo (2.30) permite que el estado de referencia en consenso variante en el tiempo esté disponible sólo para un subgrupo de los seguidores y este mismo para una topología cambiante en el tiempo constante por partes.

En este caso de análisis y aplicación a un grupo de UAV's en el capítulo 4 se aplica en simulación los dos algoritmos para referencias variantes en el tiempo haciendo una comparación con ventajas y desventajas de cada uno de los algoritmos.

Capítulo 3

Diseño de Control Para un Quadrotor

En este capítulo se propone una ley de control para realizar el seguimiento de una trayectoria variante en el tiempo con un solo UAV, donde las variables de interés a controlar son la posición lineal $x_1 = [x, y, z]^T$ y el ángulo *Yaw*. Una vez propuesta la ley de control se realiza un análisis de estabilidad en lazo cerrado para la ley de control propuesta. Para terminar el capítulo, se presenta una simulación, con el fin de probar el funcionamiento de la ley de control propuesta.

3.1. Ley de Control Para un Quadrotor

Una vez modelado el sistema físico-mecánico en variables del espacio de estados, definidas las entradas al sistema y las salidas deseadas, se propone la ley de control. Para comenzar se toma el modelo matemático desarrollado en el Capítulo 2:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= f_1(x_2, x_3, x_4) + g_1(x_3)\varphi_1, \\ \dot{x}_3 &= x_4, \\ \dot{x}_4 &= f_2(x_2, x_3, x_4) + g_2\varphi_2. \end{aligned} \tag{3.1}$$

Donde:

- $x_1 = [x, y, z]^T$ Vector de posición lineal,
- $x_2 = [\dot{x}, \dot{y}, \dot{z}]^T$ Vector de velocidad lineal,
- $x_3 = [\phi, \theta, \psi]^T$ Vector de posición rotacional,
- $x_4 = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ Vector de velocidad rotacional,
- $X = [x_1, x_2, x_3, x_4]^T$ Vector de estado.

Con:

$$\begin{aligned}
 f_1 &= \begin{bmatrix} A_u \\ A_v \\ A_w - g \end{bmatrix}, & g_1 &= \begin{bmatrix} \frac{U_1}{m} & 0 & 0 \\ 0 & \frac{U_1}{m} & 0 \\ 0 & 0 & \frac{C_\phi C_\theta}{m} \end{bmatrix} & \varphi_1 &= \begin{bmatrix} u_x \\ u_y \\ U_1 \end{bmatrix} \\
 f_2 &= \begin{bmatrix} \dot{\theta}\dot{\psi}\frac{I_y - I_z}{I_x} - \dot{\theta}\frac{J_R}{I_x}w + A_p \\ \dot{\phi}\dot{\psi}\frac{I_x - I_z}{I_y} + \dot{\phi}\frac{J_R}{I_y}w + A_q \\ \dot{\theta}\dot{\phi}\frac{I_x - I_y}{I_z} + A_r \end{bmatrix}, & g_2 &= \begin{bmatrix} \frac{1}{I_x} & 0 & 0 \\ 0 & \frac{1}{I_y} & 0 \\ 0 & 0 & \frac{1}{I_z} \end{bmatrix} & \varphi_2 &= \begin{bmatrix} U_2 \\ U_3 \\ U_4 \end{bmatrix}
 \end{aligned} \tag{3.2}$$

Se definen las entradas como:

$$\begin{aligned}
 U_1 &= F_1 + F_2 + F_3 + F_4, \\
 U_2 &= d(F_4 - F_2), \\
 U_3 &= d(F_3 - F_1), \\
 U_4 &= -\frac{c}{b}F_1 + \frac{c}{b}F_2 - \frac{c}{b}F_3 + \frac{c}{b}F_4, \\
 w &= -w_1 + w_2 - w_3 + w_4.
 \end{aligned} \tag{3.3}$$

Definiendo los pseudo controles:

$$\begin{aligned}
 u_x &= C_\psi S_\theta C_\phi + S_\psi S_\phi, \\
 u_y &= S_\psi S_\theta C_\phi - C_\psi S_\phi.
 \end{aligned} \tag{3.4}$$

Se hacen las siguientes suposiciones para análisis futuro:

Suposición 3.1. Posición y velocidad tanto lineales como angulares pueden ser medidas o estimadas mediante sensores abordo del UAV.

Suposición 3.2. Los ángulos de Euler están limitados a $-\frac{\pi}{2} < \phi, \theta < \frac{\pi}{2}$.

Suposición 3.3. El término u_1 representa el empuje total de la aeronave en el eje z positivo y es estrictamente positiva para contrarrestar el efecto de la gravedad.

Para este caso el objetivo de control es lograr el seguimiento de una trayectoria para un quadrotor, es decir, el controlador tiene que garantizar la convergencia asintótica de la salida a una trayectoria de referencia definida.

El controlador se dividirá en dos etapas, en la primera etapa, se diseñará un algoritmo de control basado en la combinación de las técnicas de *BlockControl* (BC) y *SuperTwisting* (ST) para la posición absoluta del quadrotor. Una vez obtenidos los pseudo controles u_x y u_y , son utilizados para obtener los ángulos de *Pitch* y *Roll* deseados para el fin de lograr el

seguimiento a lo largo de las posiciones x y y . En la segunda etapa, se diseñará un controlador también basado en la técnica BC y ST para conducir los ángulos de *Pitch* y *Roll* a sus valores deseados, y para seguir el ángulo de referencia en *Yaw*.

Etapa A-Dinámica de Traslación

Se define el error de seguimiento de traslación como:

$$e_1 = x_1 - x^r \quad (3.5)$$

Donde x^r es el vector de referencia en posición lineal absoluta la cual debe ser una función C^2 acotada. Obteniendo la derivada de (3.5):

$$\dot{e}_1 = \dot{x}_1 - \dot{x}^r = x_2 - \dot{x}^r \quad (3.6)$$

Para estabilizar la dinámica de e_1 en (3.6) se sigue con la técnica de BC [16], la entrada de control virtual x_2^* se toma como:

$$x_2^* = \dot{x}^r - k_1 e_1 \quad (3.7)$$

Con $k_1 \in \mathbb{R}^{3 \times 3}$ una matriz constante definida positiva. Entonces se define la superficie deslizante $s_1 = [s_{1,1}, s_{1,2}, s_{1,3}]^T$ como:

$$s_1 = x_2 - x_2^* = x_2 - \dot{x}^r + k_1 e_1 \quad (3.8)$$

Derivando la superficie se obtiene:

$$\dot{s}_1 = \dot{x}_2 - \dot{x}_2^* = \bar{f}_1 + \vartheta_1 \quad (3.9)$$

Con $\bar{f}_1 = f_1 - \ddot{x}^r + k_1 \dot{e}_1 = [\bar{f}_{1,1}, \bar{f}_{1,2}, \bar{f}_{1,3}]^T$ y $\vartheta_1 = g_1 \varphi_1$. Es la entrada φ_1 donde se encuentran los pseudo controles u_x^* , u_y^* y el control u_1 , si las suposiciones 3.2 y 3.3 se mantienen entonces la matriz g_1 es invertible. El paso final para esta etapa es diseñar el vector de control de entrada $\vartheta_1 = [\vartheta_{1,1}, \vartheta_{1,2}, \vartheta_{1,3}]^T$ aplicando el algoritmo de ST[22].

$$\begin{aligned} \vartheta_{1,k} &= -\lambda_{1,k} |s_{1,k}|^{\frac{1}{2}} \text{sigm}(\sigma_{1,k} s_{1,k}) + \dot{\vartheta}_{3,k}, \\ \dot{\vartheta}_{3,k} &= -\lambda_{3,k} \text{sigm}(\sigma_{3,k} s_{1,k}), \quad k = 1, 2, 3. \end{aligned} \quad (3.10)$$

Con $\lambda_1 = [\lambda_{1,1}, \lambda_{1,2}, \lambda_{1,3}]^T$ y $\lambda_3 = [\lambda_{3,1}, \lambda_{3,2}, \lambda_{3,3}]^T$, de igual manera el termino $\sigma_1 = [\sigma_{1,1}, \sigma_{1,2}, \sigma_{1,3}]^T$ y $\sigma_3 = [\sigma_{3,1}, \sigma_{3,2}, \sigma_{3,3}]^T$. Como ϑ_1 se usa para diseñar una superficie deslizante en la siguiente etapa, se usara la función sigmoial $\text{sigm}(\sigma_{1,k} s_{1,k})$ en lugar de la función signo $\text{sign}(\sigma_{1,k} s_{1,k})$ para garantizar la suavidad en la entrada de control. Una vez encontrando φ_1 acomodando la ecuación (3.4) de forma matricial:

$$\begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} S_\psi & C_\psi \\ -C_\psi & S_\psi \end{bmatrix} \begin{bmatrix} S_\phi \\ C_\phi S_\theta \end{bmatrix} \quad (3.11)$$

Los ángulos deseados *Pitch* y *Roll* son despejados de la siguiente forma:

$$\begin{bmatrix} S_\psi & -C_\psi \\ C_\psi & S_\psi \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix} = \begin{bmatrix} S_\phi \\ C_\phi S_\theta \end{bmatrix}$$

Se obtiene:

$$\begin{aligned} \phi^* &= S^{-1}(S_\psi u_x - C_\psi u_y) \\ \theta^* &= S^{-1}\left(\frac{C_\psi u_x + S_\psi u_y}{C_\phi^*}\right) \end{aligned} \quad (3.12)$$

El C_ϕ^* no sera cero siempre que se cumpla la suposición 3.2. Estos ángulos son los deseados para llegar con rapidez al punto de trayectoria deseado resultando en movimientos bruscos, para algunas aplicaciones es necesario limitar estos ángulos, esto se logra saturando los valores obtenidos en (3.12), resultando en movimientos con mayor suavidad pero menor velocidad en el movimiento lineal, situaciones que se deben tomar en cuenta para la generación de la trayectoria. Se define el vector de posición rotacional con los valores en ángulos deseados y la referencia del ángulo *Yaw* de la forma $x_3^* = [\phi^*, \theta^*, \psi^*]^T$ este vector será la entrada para la siguiente etapa.

Etapa B-Dinámica de Rotación

Se define el error de seguimiento de rotación como:

$$e_2 = x_3 - x_3^* \quad (3.13)$$

Obteniendo la derivada de (3.13):

$$\dot{e}_2 = \dot{x}_3 - \dot{x}_3^* = x_4 - \dot{x}_3^* \quad (3.14)$$

Para estabilizar la dinámica de e_2 en (3.14) la entrada de control virtual x_4^* se toma como:

$$x_4^* = \dot{x}_3^* - k_2 e_2 \quad (3.15)$$

Con $k_2 \in \mathbb{R}^{3 \times 3}$ una matriz constante definida positiva. Entonces se define la superficie deslizando $s_2 = [s_{2,1}, s_{2,2}, s_{2,3}]^T$ como:

$$s_2 = x_4 - x_4^* = x_4 - \dot{x}_3^* + k_2 e_2 \quad (3.16)$$

Derivando la superficie se obtiene:

$$\dot{s}_2 = \dot{x}_4 - \dot{x}_4^* = \bar{f}_2 + \vartheta_2 \quad (3.17)$$

Con $\bar{f}_2 = f_2 - \ddot{x}_3^* + k_2 \dot{e}_2 = [\bar{f}_{2,1}, \bar{f}_{2,2}, \bar{f}_{2,3}]^T$ y $\vartheta_2 = g_2 \varphi_2$. Es la entrada φ_2 donde se encuentran los controles u_2, u_3 y u_4 , la matriz g_2 es siempre invertible. Por ultimo se diseña el vector de control de entrada $\vartheta_2 = [\vartheta_{2,1}, \vartheta_{2,2}, \vartheta_{2,3}]^T$ aplicando el algoritmo de ST.

$$\begin{aligned}\vartheta_{2,k} &= -\lambda_{2,k} |s_{2,k}|^{\frac{1}{2}} \text{sigm}(\sigma_{2,k} s_{2,k}) + \vartheta_{4,k}, \\ \dot{\vartheta}_{4,k} &= -\lambda_{4,k} \text{sigm}(\sigma_{4,k} s_{2,k}), \quad k = 1, 2, 3.\end{aligned}\tag{3.18}$$

Con $\lambda_2 = [\lambda_{2,1}, \lambda_{2,2}, \lambda_{2,3}]^T$ y $\lambda_4 = [\lambda_{4,1}, \lambda_{4,2}, \lambda_{4,3}]^T$, de igual manera el termino $\sigma_2 = [\sigma_{2,1}, \sigma_{2,2}, \sigma_{2,3}]^T$ y $\sigma_4 = [\sigma_{4,1}, \sigma_{4,2}, \sigma_{4,3}]^T$. Con el vector de control $U = [u_1, u_2, u_3, u_4]^T$ está completa la ley de control, es necesario obtener los valores de cada una de las velocidades angulares requeridas en los motores del quadrotor, las cuales se obtienen directamente de las entradas de control, colocando la ecuación (3.3) en forma matricial:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -db & 0 & db \\ -db & 0 & db & 0 \\ -c & c & -c & c \end{bmatrix} \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix}\tag{3.19}$$

Despejando las velocidades angulares de (3.19):

$$\begin{bmatrix} \frac{1}{4b} & 0 & -\frac{1}{2db} & -\frac{1}{4c} \\ \frac{1}{4b} & -\frac{1}{2db} & 0 & \frac{1}{4c} \\ \frac{1}{4b} & 0 & \frac{1}{2db} & -\frac{1}{4c} \\ \frac{1}{4b} & \frac{1}{2db} & 0 & \frac{1}{4c} \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} w_1^2 \\ w_2^2 \\ w_3^2 \\ w_4^2 \end{bmatrix}\tag{3.20}$$

De esta forma se obtienen los valores de velocidad angular en cada uno de los motores del quadrotor.

3.2. Estabilidad de Control Para un Quadrotor

En esta sección se realiza el análisis de estabilidad de la ley de control propuesta, para un quadrotor. La estabilidad para este sistema es cuando se realiza un seguimiento de referencia satisfactorio, esto implica que los errores tanto de posición en traslación como en rotación, para esto, se debe demostrar estabilidad en la dinámica tanto del error de posición en traslación como para el error de posición en rotación.

3.2.1. Estabilidad en Errores de Posición

Se obtiene la dinámica del error de posición en traslación de las ecuaciones (3.6) y (3.8), una vez con la superficie $s_1 = 0$ se tiene:

$$\begin{aligned} \dot{e}_1 &= x_2 - \dot{x}^r \\ 0 &= x_2 - \dot{x}^r + k_1 e_1, \\ \dot{e}_1 &= -k_1 e_1. \end{aligned} \tag{3.21}$$

Una vez que la superficie s_1 llegue a cero, el error de posición en traslación es asintóticamente estable, siempre que $k_1 > 0$.

De la misma manera para el error de posición en rotación se usan las ecuaciones (3.14) y (3.16) se tiene:

$$\begin{aligned} \dot{e}_2 &= x_4 - \dot{x}_3^*, \\ 0 &= x_4 - \dot{x}_3^* + k_2 e_2, \\ \dot{e}_2 &= -k_2 e_2. \end{aligned} \tag{3.22}$$

Una vez que la superficie s_2 llegue a cero, el error de posición en rotación es asintóticamente estable, siempre que $k_2 > 0$.

De esta manera quedan establecidas condiciones bajo las cuales se tienen dinámicas en ambos errores de posición estables. Sólo falta obtener condiciones bajo las cuales se lleva las superficies deslizantes s_1 y s_2 a cero.

3.2.2. Estabilidad del Modo Deslizante

Con el fin de establecer condiciones bajo las cuales el modo deslizante es estable, se reescriben las ecuaciones (3.9) y (3.17) con (3.10) y (3.18) respectivamente, como:

$$\begin{aligned} \dot{s}_{j,k} &= \bar{f}_{j,k} - \lambda_{j,k} |s_{j,k}|^{1/2} \text{sigm}(\sigma_{j,k} s_{j,k}) + \vartheta_{j+2,k}, \\ \dot{\vartheta}_{j+2,k} &= -\lambda_{j+2,k} \text{sigm}(\sigma_{j+2,k} s_{j,k}), \\ j &= 1, 2 \text{ y } k = 1, 2, 3. \end{aligned} \tag{3.23}$$

Ahora se introduce la siguiente suposición:

Suposición 3.4. El término de perturbación $\bar{f}_{j,k}$ es acotado en una región de operación admisible D por:

$$|\bar{f}_{j,k}| \leq \delta_{j,k} |s_{j,k}|^{1/2} \tag{3.24}$$

Con $\delta_{j,k}$ una constante positiva.

Teorema 3.1. Bajo la suposición 3.4 el origen $s_{j,k} = 0$ es un punto de equilibrio que es globalmente asintóticamente estable si las ganancias $\lambda_{j,k}$ y $\lambda_{j+2,k}$ satisfacen:

$$\begin{aligned} \lambda_{j,k} &> 2\delta_{j,k} \\ \lambda_{j+2,k} &> \lambda_{j,k} \frac{5\delta_{j,k}\lambda_{j,k} + 4\delta_{j,k}^2}{2(\lambda_{j,k} - 2\delta_{j,k})} \end{aligned} \quad (3.25)$$

Además, todas las trayectorias convergen en un tiempo finito al origen, delimitada por la parte superior $T = \frac{2V^{\frac{1}{2}}(s_{j,k}(0))}{\tilde{\gamma}}$, donde $s_{j,k}(0)$ es el estado inicial y $\tilde{\gamma}$ es una constante que depende de las ganancias $\lambda_{j,k}$, $\lambda_{j+2,k}$ y el coeficiente de perturbación $\bar{f}_{j,k}$. Esto, junto con la función de Lyapunov V se pueden ver en la demostración del teorema 3.1 en [19].

De esta manera, tomando ganancias adecuadas tanto para el algoritmo de modos deslizantes como para la dinámica en los errores de posición, tendremos un sistema estable. Además, dependiendo de que tan grandes sean estos valores de ganancias sera la velocidad de respuesta.

3.3. Simulación de Control Para un Quadrotor

Con el fin de verificar la validez y la eficiencia del algoritmo de control propuesto, se realizan simulaciones en Matlab/Simulink utilizando parámetros tomados de [14], que se enumeran en la Tabla 3.1. Se muestran dos simulaciones, la primera el seguimiento de trayectoria de un quadrotor sin perturbaciones y la segunda, con fines comparativos, el seguimiento de la misma trayectoria de un quadrotor con perturbaciones.

Tabla 3.1: Parámetros

Parámetro	Valor	Unidad
m	0.468	kg
g	9.81	m/s^2
d	0.225	m
J_R	3.357×10^{-5}	$kg \ m^2$
I_x	4.856×10^{-3}	$kg \ m^2$
I_y	4.856×10^{-3}	$kg \ m^2$
I_z	8.801×10^{-3}	$kg \ m^2$
b	2.98×10^{-6}	$N \ s^2/rad^2$
c	1.14×10^{-7}	$Nm \ s^2/rad^2$

3.3.1. Sin Perturbaciones

Se tienen como condiciones iniciales $X(0) = [5, 0, 0.2, 0, 0, 0, 0, 0, 0, 0, 0]^T$. La trayectoria de referencia será la siguiente: en los primeros 10 segundos de la simulación, el quadrotor sube hasta una altura deseada 0.5 m, después de los 10 segundos se tiene como referencia para x y y , $x = 5 \cos(\frac{2\pi(t-10)}{50})$ y $y = 2.5 \sin(\frac{4\pi(t-10)}{50})$. En la Fig. 3.1 se muestra el resultado en la posición tridimensional de una simulación de 60 segundos. En esta simulación el ángulo de *Yaw* no fue modificado, se mantiene siempre en cero, como se puede apreciar en la Fig. 3.2 que muestra las posiciones del quadrotor a los 10s, 20s, 30s y 35s siempre en la misma posición con respecto a *Yaw*. En las Figs. 3.3, 3.4 y 3.5 se muestran las gráficas de posición (x, y, z), ángulos (*Pitch*, *Roll*, *Yaw*) y Fuerzas (F_1, F_2, F_3, F_4) respectivamente. En la gráfica de los ángulos Fig. 3.4 se puede apreciar con claridad en el segundo 10 el impulso que el arranque tiene sobre el ángulo de *roll* ya que este es el que produce movimiento hacia adelante del quadrotor.

Los parámetros de control $k_1 = \text{diag}(1, 0.9, 0.9)$, $\lambda_{1,k} = \lambda_{2,k} = 9$ ($k = 1, 2, 3$), $\lambda_{3,k} = \lambda_{4,k} = 0.9$, $\sigma_{1,k} = \sigma_{2,k} = \sigma_{3,k} = \sigma_{4,k} = 1$ y $k_2 = \text{diag}(2, 2, 2)$. Los resultados dieron un seguimiento de trayectoria satisfactorio.

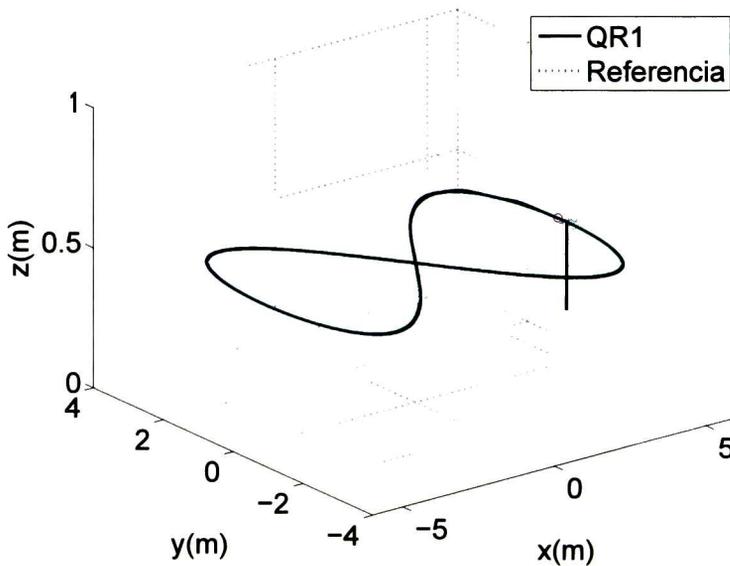


Figura 3.1: Seguimiento de trayectoria 1 quadrotor.

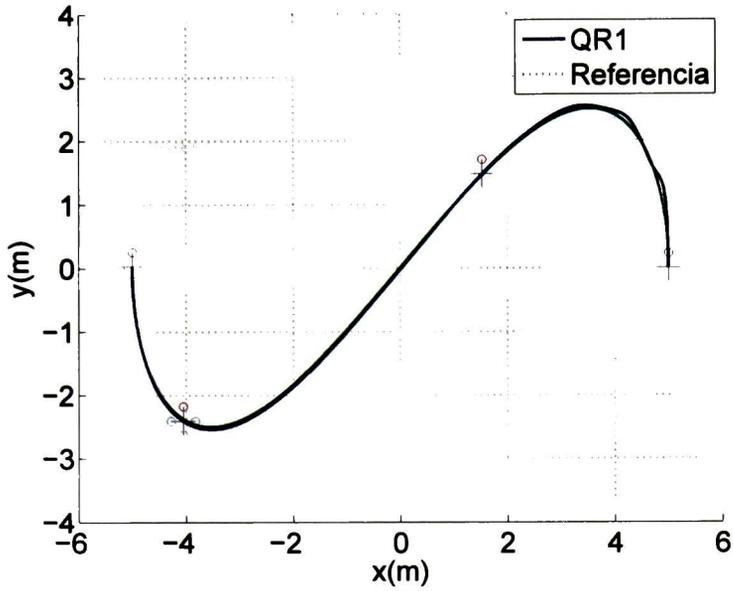


Figura 3.2: Plano (x,y) 1 quadrotor.

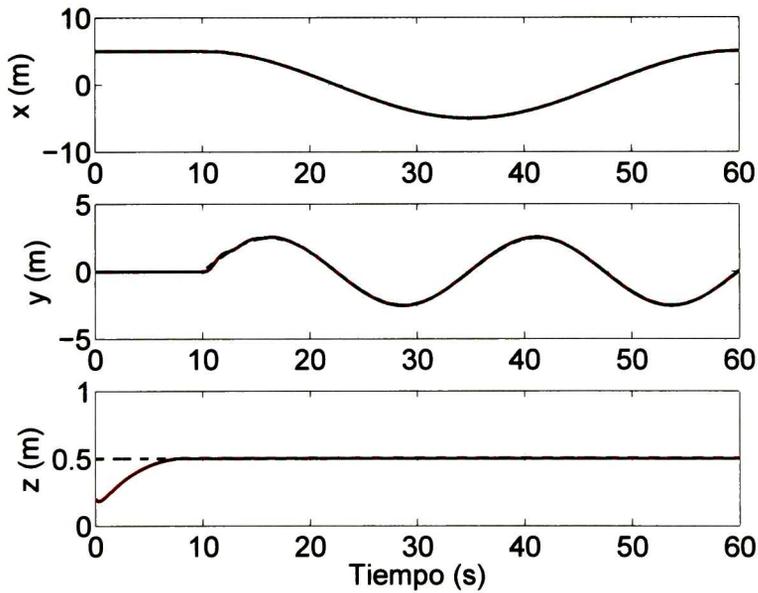


Figura 3.3: Gráfica de posición (x, y, z), 1 quadrotor.

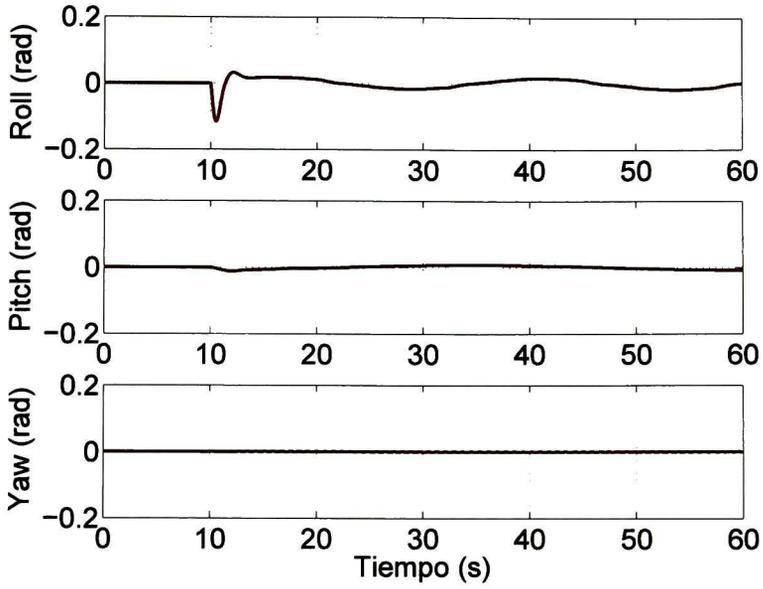


Figura 3.4: Gráfica de ángulos (*Pitch*, *Roll*, *Yaw*), 1 quadrotor.

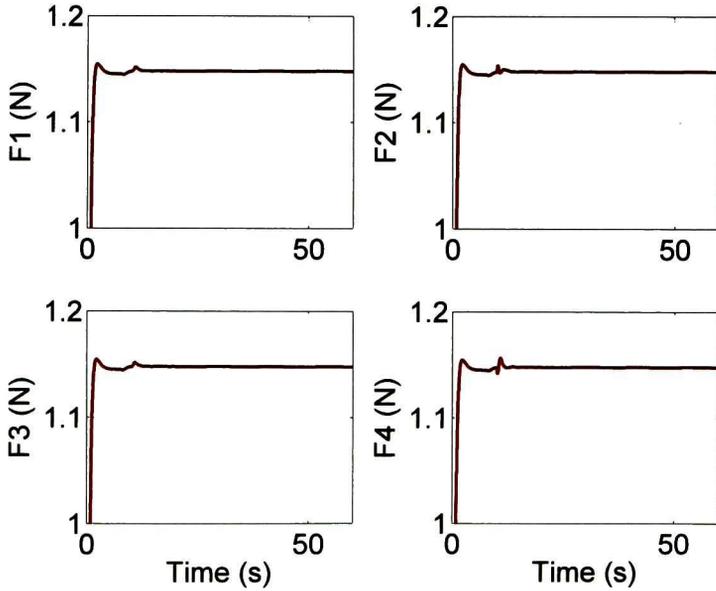


Figura 3.5: Gráfica de fuerzas, 1 quadrotor.

3.3.2. Con Perturbaciones

Entonces, para este ejemplo se tienen dos perturbaciones, la primera a los 25 segundos la cual tiene repercusión directa con las fuerzas de la siguiente forma:

$$A_F = [0.3\sin(0.1t), 0.1\cos(0.1t), 0.1\sin(0.1t)]^T,$$

y la segunda a los 45 segundos con repercusión a los torques:

$$A_T = [0.3\sin(0.1t), 0.3\cos(0.1t), 0.15\sin(0.1t)]^T$$

Haciendo entonces la comparación, se puede notar el efecto de la perturbación al segundo 25 en las gráficas de posición Figs. 3.6 y 3.7, siendo la gráfica de fuerzas donde se nota con mayor claridad la reacción que se tiene sobre dicha perturbación Fig. 3.10, logrando contrarrestarla de manera satisfactoria. Los ángulos de *Pitch* y *Roll* también se vieron modificados debido a esta perturbación Fig. 3.9.

Con respecto a la perturbación en los torques al segundo 45, se nota con mayor claridad la acción de contrarrestarla en la gráfica de fuerzas aumentando así F_2 y F_4 para mantener un $Yaw = 0$ aun con la perturbación.

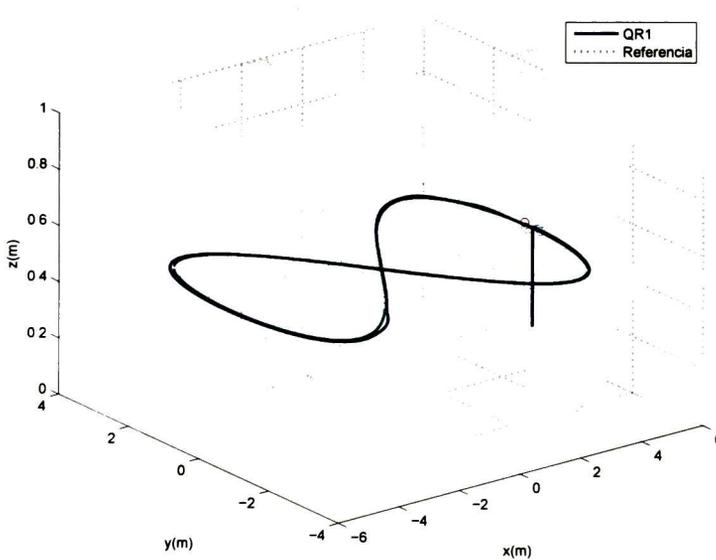


Figura 3.6: Seguimiento de trayectoria 1 quadrotor, con perturbaciones.

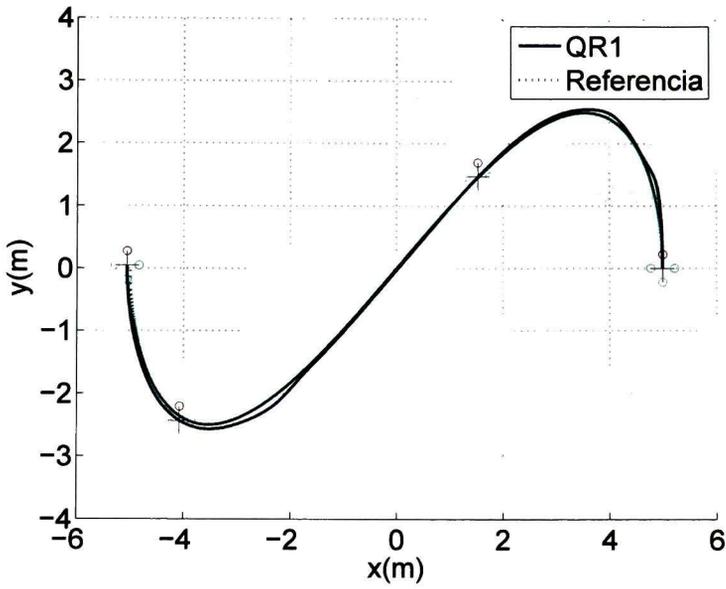


Figura 3.7: Plano (x, y) 1 quadrotor, con perturbaciones.

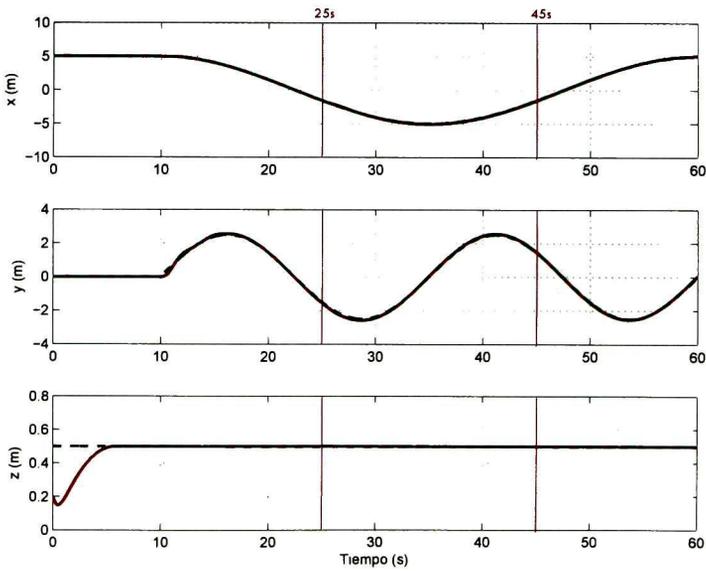


Figura 3.8: Gráfica de posición (x, y, z) , 1 quadrotor, con perturbaciones.

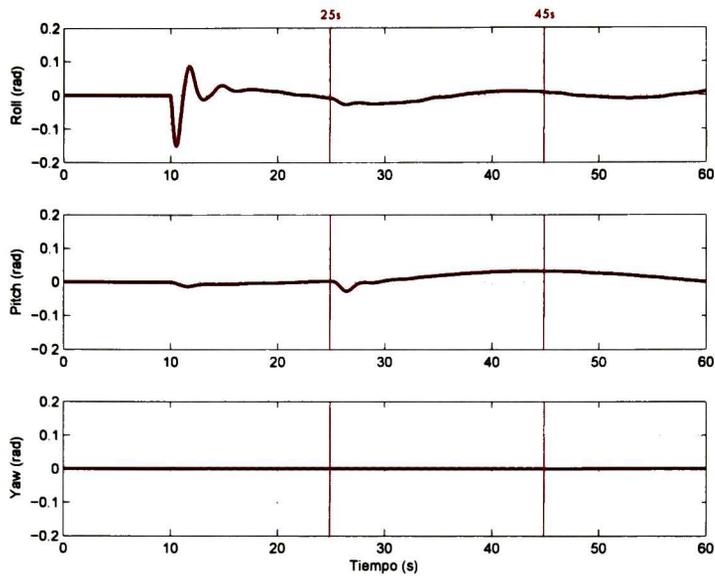


Figura 3.9: Gráfica de ángulos (*Pitch, Roll, Yaw*), 1 quadrotor, con perturbaciones.

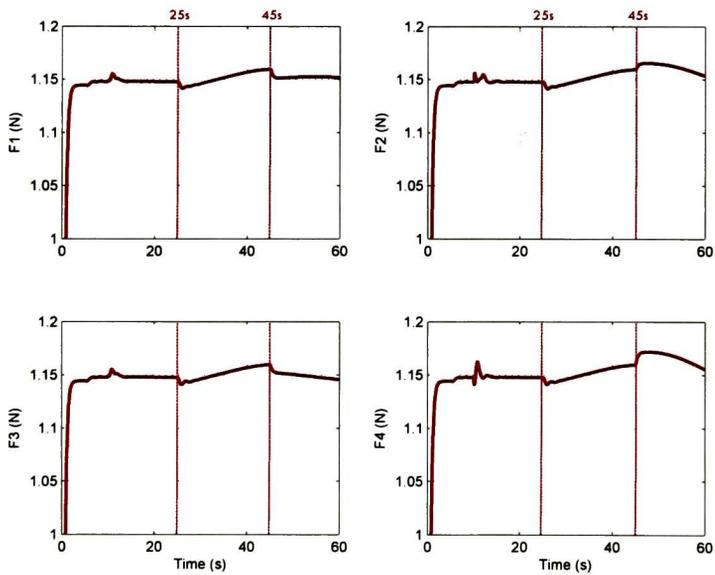


Figura 3.10: Gráfica de fuerzas, 1 quadrotor, con perturbaciones.

Capítulo 4

Diseño de Control Distribuido

En este capítulo se propone una ley de control distribuido para realizar el seguimiento de una trayectoria variante en el tiempo con un grupo de UAV's, siendo las salidas de cada i -ésimo quadrotor las variables de interés a controlar, la posición lineal $x_{1,i} = [x_i, y_i, z_i]^T$ y el ángulo Yaw_i . Una vez propuesta la ley de control distribuido se realiza un análisis de estabilidad en lazo cerrado para la ley de control distribuido propuesta. Para terminar el capítulo, se presenta una simulación, con el fin de probar el funcionamiento de la ley de control distribuido propuesta.

4.1. Control Distribuido

En el capítulo anterior se propuso una ley de control para el seguimiento de una trayectoria variante en el tiempo para un solo quadrotor, en esta sección se extiende el control para un grupo de quadrotors [10], aplicando el algoritmo fundamental para seguimiento de referencia en consenso (2.30). Para modelar la comunicación entre el grupo se usa una topología de comunicación la cual debe cumplir con las condiciones dichas en cualquiera de los dos teoremas 2.4 y 2.5.

El consenso se hace en las variables de interés que en este caso es el vector de posición en traslación. El modelo del i -ésimo quadrotor dentro del grupo se define de la siguiente manera:

$$\begin{aligned}\dot{x}_{1,i} &= x_{2,i}, \\ \dot{x}_{2,i} &= f_{1,i}(x_{2,i}, x_{3,i}, x_{4,i}) + g_{1,i}(x_{3,i})\varphi_{1,i}, \\ \dot{x}_{3,i} &= x_{4,i}, \\ \dot{x}_{4,i} &= f_{2,i}(x_{2,i}, x_{3,i}, x_{4,i}) + g_{2,i}\varphi_{2,i}.\end{aligned}\tag{4.1}$$

Para $i = 1, \dots, n$. Donde:

$x_{1,i} = [x_i, y_i, z_i]^T$ Vector de posición lineal del i -ésimo quadrotor,
 $x_{2,i} = [\dot{x}_i, \dot{y}_i, \dot{z}_i]^T$ Vector de velocidad lineal del i -ésimo quadrotor,
 $x_{3,i} = [\phi_i, \theta_i, \psi_i]^T$ Vector de posición rotacional del i -ésimo quadrotor,
 $x_{4,i} = [\dot{\phi}_i, \dot{\theta}_i, \dot{\psi}_i]^T$ Vector de velocidad rotacional del i -ésimo quadrotor,
 $X_i = [x_{1,i}, x_{2,i}, x_{3,i}, x_{4,i}]^T$ Vector de estado del i -ésimo quadrotor.

Con $f_{1,i}$, $g_{1,i}$, $\varphi_{1,i}$, $f_{2,i}$, $g_{2,i}$ y $\varphi_{1,i}$ como en (3.2) para cada quadrotor. Se definen las entradas del i -ésimo quadrotor como:

$$\begin{aligned}
 u_{1,i} &= F_{1,i} + F_{2,i} + F_{3,i} + F_{4,i}, \\
 u_{2,i} &= d(F_{4,i} - F_{2,i}), \\
 u_{3,i} &= d(F_{3,i} - F_{1,i}), \\
 u_{4,i} &= -\frac{c}{b}F_{1,i} + \frac{c}{b}F_{2,i} - \frac{c}{b}F_{3,i} + \frac{c}{b}F_{4,i}, \\
 w_i &= -w_{1,i} + w_{2,i} - w_{3,i} + w_{4,i}.
 \end{aligned} \tag{4.2}$$

Y los pseudo controles:

$$\begin{aligned}
 u_{x,i} &= C_{\psi_i} S_{\theta_i} C_{\phi_i} + S_{\psi_i} S_{\phi_i}, \\
 u_{y,i} &= S_{\psi_i} S_{\theta_i} C_{\phi_i} - C_{\psi_i} S_{\phi_i}.
 \end{aligned} \tag{4.3}$$

Se hacen las suposiciones 3.1, 3.2 y 3.3 para cada uno de los quadrotors.

Para este caso el objetivo de control es lograr el seguimiento de una trayectoria para un grupo de quadrotors manteniendo una formación predeterminada, es decir, el controlador tiene que garantizar la convergencia asintótica de la salida del *lider* (virtual) a una trayectoria de referencia definida.

El controlador se dividirá en dos etapas, en la primera etapa, se usara el algoritmo fundamental para seguimiento de referencia en consenso y el algoritmo de control ST para la posición absoluta del quadrotor. Una vez obtenidos los pseudo controles $u_{x,i}$ y $u_{y,i}$ de cada i -ésimo quadrotor, son utilizados para obtener los ángulos de *Pitch* y *Roll* deseados para cada uno de los quadrotors con el fin de lograr el seguimiento en consenso a lo largo de las posiciones x y y . En la segunda etapa, se diseñará un controlador también basado en la técnica BC y ST para conducir los ángulos de *Pitch* y *Roll* de cada quadrotor a sus valores deseados.

Etapas A-Dinámica de Traslación en Consenso

La idea básica es llevar a consenso la posición en traslación de todos los quadrotors, comenzando con el algoritmo fundamental para seguimiento de referencia en consenso:

$$\begin{aligned}
x_{2,i}^* &= \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\dot{x}_{1,j} - k_{1,i} \underbrace{((x_{1,i} - \Delta_i) - (x_{1,j} - \Delta_j))}_{e_{i,j}}] + \dots \\
&\dots + \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\dot{x}^r - k_{1,i} \underbrace{((x_{1,i} - \Delta_i) - x^r)}_{e_{i,(n+1)}}].
\end{aligned} \tag{4.4}$$

Donde a_{ij} , $i = 1, \dots, n$, $j = 1, \dots, n+1$, es el elemento (i, j) de la matriz de adyacencia \mathcal{A}_{n+1} en el tiempo t , $k_{1,i} \in \mathbb{R}^{3 \times 3}$ es una matriz constante definida positiva y $\eta_i(t) \equiv \sum_{j=1}^{n+1} a_{ij}(t)$, x^r es el vector de referencia en posición lineal absoluta la cual debe ser una función C^2 acotada. El vector $\Delta_i \in \mathbb{R}^{3 \times 1}$ es la separación que tendrá el i -ésimo quadrotor con respecto al líder, para así avanzar en formación ($\Delta_{n+1} \equiv 0$).

Cualquier algoritmo para seguimiento de referencia en consenso entrega una velocidad de traslación deseada $x_{2,i}^*$ con la cual se obtiene una estabilidad asintótica para el consenso en posición de traslación, siempre y cuando se cumpla cualquiera de los teoremas 2.4 y 2.5. Derivando (4.4):

$$\dot{x}_{2,i}^* = \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\ddot{x}_{1,j} - k_{1,i} \dot{e}_{i,j}] + \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\ddot{x}^r - k_{1,i} \dot{e}_{i,(n+1)}] \tag{4.5}$$

Con

$$\begin{aligned}
\dot{e}_{i,j} &= \dot{x}_{1,i} - \dot{x}_{1,j} = x_{2,i} - x_{2,j} \\
\dot{e}_{i,(n+1)} &= \dot{x}_{1,i} - \dot{x}^r = x_{2,i} - \dot{x}^r
\end{aligned} \tag{4.6}$$

Se define entonces la superficie deslizante

$s_{1,i} = [s_{1,i,1}, s_{1,i,2}, s_{1,i,3}]^T$ como:

$$s_{1,i} = x_{2,i} - x_{2,i}^* = x_{2,i} - \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\dot{x}_{1,j} - k_{1,i} e_{i,j}] - \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\dot{x}^r - k_{1,i} e_{i(n+1)}] \tag{4.7}$$

Derivando la superficie se obtiene:

$$\dot{s}_{1,i} = \dot{x}_{2,i} - \dot{x}_{2,i}^* = \bar{f}_{1,i} + \vartheta_{1,i} \tag{4.8}$$

Con $\bar{f}_{1,i} = f_{1,i} - \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\ddot{x}_{1,j} - k_{1,i} \dot{e}_{i,j}] - \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\ddot{x}^r - k_{1,i} \dot{e}_{i,(n+1)}] = [\bar{f}_{1,i,1}, \bar{f}_{1,i,2}, \bar{f}_{1,i,3}]^T$ y $\vartheta_{1,i} = g_{1,i} \varphi_{1,i}$. Es la entrada $\varphi_{1,i}$ donde se encuentran los pseudo controles $u_{x,i}^*$, $u_{y,i}^*$ y el

control $u_{1,i}$, si las suposiciones 3.2 y 3.3 se mantienen entonces la matriz $g_{1,i}$ es invertible. El paso final para esta etapa es diseñar el vector de control de entrada $\vartheta_{1,i} = [\vartheta_{1,i,1}, \vartheta_{1,i,2}, \vartheta_{1,i,3}]^T$ aplicando el algoritmo de ST.

$$\begin{aligned}\vartheta_{1,i,k} &= -\lambda_{1,i,k} |s_{1,i,k}|^{\frac{1}{2}} \text{sigm}(\sigma_{1,i,k} s_{1,i,k}) + \vartheta_{3,i,k}, \\ \dot{\vartheta}_{3,i,k} &= -\lambda_{3,i,k} \text{sigm}(\sigma_{3,i,k} s_{1,i,k}), \quad i = 1, \dots, n \text{ y } k = 1, 2, 3.\end{aligned}\quad (4.9)$$

Con $\lambda_{1,i} = [\lambda_{1,i,1}, \lambda_{1,i,2}, \lambda_{1,i,3}]^T$ y $\lambda_{3,i} = [\lambda_{3,i,1}, \lambda_{3,i,2}, \lambda_{3,i,3}]^T$ de igual manera el termino $\sigma_{1,i} = [\sigma_{1,i,1}, \sigma_{1,i,2}, \sigma_{1,i,3}]^T$ y $\sigma_{3,i} = [\sigma_{3,i,1}, \sigma_{3,i,2}, \sigma_{3,i,3}]^T$. De igual manera que en la seccion anterior $\vartheta_{1,i}$ se usa para diseñar una superficie deslizante en la siguiente etapa, y también se usara la función sigmoidal $\text{sigm}(\sigma_{1,i,k} s_{1,i,k})$ en lugar de la función signo $\text{sign}(\sigma_{1,i,k} s_{1,i,k})$ para garantizar la suavidad en la entrada de control. Una vez encontrando $\varphi_{1,i}$ acomodando la ecuación (4.3) de forma matricial:

$$\begin{bmatrix} u_{x,i} \\ u_{y,i} \end{bmatrix} = \begin{bmatrix} S_{\psi_i} & C_{\psi_i} \\ -C_{\psi_i} & S_{\psi_i} \end{bmatrix} \begin{bmatrix} S_{\phi_i} \\ C_{\phi_i} S_{\theta_i} \end{bmatrix}\quad (4.10)$$

Los ángulos deseados *Pitch* y *Roll* son despejados de la siguiente forma:

$$\begin{bmatrix} S_{\psi_i} & -C_{\psi_i} \\ C_{\psi_i} & S_{\psi_i} \end{bmatrix} \begin{bmatrix} u_{x,i} \\ u_{y,i} \end{bmatrix} = \begin{bmatrix} S_{\phi_i} \\ C_{\phi_i} S_{\theta_i} \end{bmatrix}$$

Se obtiene:

$$\begin{aligned}\phi_i^* &= S^{-1}(S_{\psi_i} u_{x,i} - C_{\psi_i} u_{y,i}) \\ \theta_i^* &= S^{-1}\left(\frac{C_{\psi_i} u_{x,i} + S_{\psi_i} u_{y,i}}{C_{\phi_i^*}}\right)\end{aligned}\quad (4.11)$$

El $C_{\phi_i^*}$ no sera cero siempre que se cumpla la suposición 3.2. Algunas veces es necesario limitar aun más estos ángulos debido a la aplicación a realizar, esto se logra saturando los valores obtenidos en (4.11). Se define el vector de posición rotacional con los valores en ángulos deseados y la referencia del ángulo *Yaw* de la forma $x_{3,i}^* = [\phi_i^*, \theta_i^*, \psi_i^*]^T$ este vector será la entrada para la siguiente etapa.

Etapa B-Dinámica de Rotación en Consenso

Se define el error de seguimiento de rotación para cada i -esimo quadrotor como:

$$e_{2,i} = x_{3,i} - x_{3,i}^* \quad (4.12)$$

Obteniendo la derivada de (4.12):

$$\dot{e}_{2,i} = \dot{x}_{3,i} - \dot{x}_{3,i}^* = \dot{x}_{4,i} - \dot{x}_{3,i}^* \quad (4.13)$$

Para estabilizar la dinámica de $e_{2,i}$ en (4.13) la entrada de control virtual $x_{4,i}^*$ se toma como:

$$x_{4,i}^* = \dot{x}_{3,i}^* - k_{2,i}e_{2,i} \quad (4.14)$$

$k_{2,i} \in \mathbb{R}^{3 \times 3}$ es una matriz constante definida positiva. Entonces se define la superficie deslizando $s_{2,i} = [s_{2,i,1}, s_{2,i,2}, s_{2,i,3}]^T$ como:

$$s_{2,i} = x_{4,i} - x_{4,i}^* = x_{4,i} - \dot{x}_{3,i}^* + k_{2,i}e_{2,i} \quad (4.15)$$

Derivando la superficie se obtiene:

$$\dot{s}_{2,i} = \dot{x}_{4,i} - \dot{x}_{4,i}^* = \bar{f}_{2,i} + \vartheta_{2,i} \quad (4.16)$$

Con $\bar{f}_{2,i} = f_{2,i} - \ddot{x}_{3,i}^* + k_{2,i}\dot{e}_{2,i} = [\bar{f}_{2,i,1}, \bar{f}_{2,i,2}, \bar{f}_{2,i,3}]^T$ y $\vartheta_{2,i} = g_{2,i}\varphi_{2,i}$. Es la entrada $\varphi_{2,i}$ donde se encuentran los controles de cada i -ésimo quadrotor $u_{2,i}$, $u_{3,i}$ y $u_{4,i}$, la matriz $g_{2,i}$ es siempre invertible. Por último se diseña el vector de control de entrada $\vartheta_{2,i} = [\vartheta_{2,i,1}, \vartheta_{2,i,2}, \vartheta_{2,i,3}]^T$ aplicando el algoritmo de ST.

$$\begin{aligned} \vartheta_{2,i,k} &= -\lambda_{2,i,k}|s_{2,i,k}|^{\frac{1}{2}} \text{sigm}(\sigma_{2,i,k}s_{2,i,k}) + \vartheta_{4,i,k}, \\ \dot{\vartheta}_{4,i,k} &= -\lambda_{4,i,k} \text{sigm}(\sigma_{4,i,k}s_{2,i,k}), \quad i = 1, \dots, n \text{ y } k = 1, 2, 3. \end{aligned} \quad (4.17)$$

Con $\lambda_{2,i} = [\lambda_{2,i,1}, \lambda_{2,i,2}, \lambda_{2,i,3}]^T$ y $\lambda_{4,i} = [\lambda_{4,i,1}, \lambda_{4,i,2}, \lambda_{4,i,3}]^T$ de igual manera el término $\sigma_{2,i} = [\sigma_{2,i,1}, \sigma_{2,i,2}, \sigma_{2,i,3}]^T$ y $\sigma_{4,i} = [\sigma_{4,i,1}, \sigma_{4,i,2}, \sigma_{4,i,3}]^T$. Con el vector de control $U_i = [u_{1,i}, u_{2,i}, u_{3,i}, u_{4,i}]^T$ está completa la ley de control para cada i -ésimo quadrotor, de la misma manera que en la sección anterior, es necesario obtener los valores de cada una de las velocidades angulares requeridas en los motores de cada uno de los quadrotors, estas se obtienen aplicando la ecuación (3.20) para cada uno de los quadrotors.

Si las restricciones de la aplicación permiten el uso de cualquiera de los otros algoritmos, funcionaria la misma ley de control, simplemente sustituyendo el algoritmo al comienzo de la ley de control.

4.2. Estabilidad de Control Distribuido

En esta sección se realiza el análisis de estabilidad de la ley de control distribuido propuesta, para un grupo de quadrotors. La estabilidad para este sistema es cuando se realiza un seguimiento de referencia en formación satisfactorio, esto implica que los errores tanto de posición en traslación como en rotación de cada i -ésimo quadrotor, para esto, se debe demostrar estabilidad en la dinámica tanto del error de posición en traslación como para el error de posición en rotación.

4.2.1. Estabilidad en Errores de Posición

Primeramente, del algoritmo de consenso (2.30), tomando en cuenta que para cumplir cualquiera de los teoremas 2.4 y 2.5, la topología de comunicación debe tener un árbol de expansión. Con respecto a la comunicación se analizaran tres casos posibles, uno donde el nodo padre sea unicamente el líder, otro donde el nodo padre sea algún vecino, y por ultimo uno donde tengamos como nodos padres al líder y a algún vecino.

Para el primer caso, la ecuación (4.4) con el nodo padre siendo unicamente el líder sería:

$$x_{2,i}^* = \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\dot{x}^r - k_{1,i} e_{i,(n+1)}] \quad (4.18)$$

Siendo el nodo padre unicamente el líder, la expresión $\frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) = 0$ y la expresión $\frac{1}{\eta_i(t)} a_{i(n+1)}(t) = 1$, entonces usando las ecuaciones (4.6), (4.7) y (4.18) se tiene:

$$\begin{aligned} \dot{e}_{i,(n+1)} &= x_{2,i} - \dot{x}^r \\ 0 &= x_{2,i} - \dot{x}^r + k_{1,i} e_{i,(n+1)}, \\ \dot{e}_{i,(n+1)} &= -k_{1,i} e_{i,(n+1)}. \end{aligned} \quad (4.19)$$

Lo cual indica que, una vez que la superficie $s_{1,i}$ llegue a cero, el error de posición en traslación con unicamente el líder como padre es asintóticamente estable, siempre que $k_{1,i} > 0$.

Para el segundo caso, la ecuación (4.4) con el nodo padre siendo unicamente algún vecino sería:

$$x_{2,i}^* = \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\dot{x}_{1,j} - k_{1,i} e_{i,j}] \quad (4.20)$$

Siendo el nodo padre unicamente algún vecino, la expresión $\frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) = 1$ y la expresión $\frac{1}{\eta_i(t)} a_{i(n+1)}(t) = 0$, entonces usando las ecuaciones (4.6), (4.7) y (4.20) se tiene:

$$\begin{aligned} \dot{e}_{i,j} &= x_{2,i} - \dot{x}_{1,j}, \\ 0 &= x_{2,i} - \dot{x}_{1,j} + k_{1,i} e_{i,j}, \\ \dot{e}_{i,j} &= -k_{1,i} e_{i,j}. \end{aligned} \quad (4.21)$$

Lo cual indica que, una vez que la superficie $s_{1,i}$ llegue a cero, el error de posición en traslación con unicamente algún vecino como padre es asintóticamente estable, siempre que

$k_{1,i} > 0$.

Para el tercer caso, la ecuación (4.4) con los nodos padres siendo el líder y algún vecino sería:

$$x_{2,i}^* = \frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) [\dot{x}_{1,j} - k_{1,i} e_{i,j}] + \frac{1}{\eta_i(t)} a_{i(n+1)}(t) [\dot{x}^r - k_{1,i} e_{i,(n+1)}] \quad (4.22)$$

Siendo los nodos padres el líder y algún vecino, las expresiones $\frac{1}{\eta_i(t)} \sum_{j=1}^n a_{ij}(t) = \frac{1}{\eta_i(t)} a_{i(n+1)}(t) = \frac{1}{2}$ y se formaría el error total:

$$e_t = e_{i,j} + e_{i,(n+1)} \quad (4.23)$$

Derivando este error total:

$$\dot{e}_t = \dot{e}_{i,j} + \dot{e}_{i,(n+1)} \quad (4.24)$$

Entonces usando las ecuaciones (4.6), (4.7), (4.22), (4.23) y (4.24) se tiene:

$$\begin{aligned} \dot{e}_{i,j} &= x_{2,i} - \dot{x}_{1,j}, \\ \dot{e}_{i,(n+1)} &= x_{2,i} - \dot{x}^r, \\ 0 &= x_{2,i} + \frac{1}{2}(-\dot{x}_{1,j} + k_{1,i} e_{i,j}) + \frac{1}{2}(-\dot{x}^r + k_{1,i} e_{i,(n+1)}), \\ 0 &= \dot{e}_{i,j} + k_{1,i} e_{i,j} + \dot{e}_{i,(n+1)} + k_{1,i} e_{i,(n+1)}, \\ e_t &= e_{i,j} + e_{i,(n+1)}, \\ \dot{e}_t &= \dot{e}_{i,j} + \dot{e}_{i,(n+1)}, \\ \dot{e}_t &= -k_{1,i} e_t. \end{aligned} \quad (4.25)$$

Lo cual indica que, una vez que la superficie $s_{1,i}$ llegue a cero, el error de posición en traslación con el líder y algún vecino como padre es asintóticamente estable, siempre que $k_{1,i} > 0$.

La expresión $\eta_i(t)$ es el numero total de padres que tiene el i -esimo nodo, de esta manera, el error total siempre se divide sin importar el numero de nodos padres, siendo entonces estable para cualquier numero de nodos padres, siempre que $k_{1,i} > 0$.

Por otra parte, para el error de posición en rotación se usan las ecuaciones (4.13) y (4.15) se tiene:

$$\dot{e}_{2,i} = -k_{2,i} e_{2,i} \quad (4.26)$$

Una vez que la superficie $s_{2,i}$ llegue a cero, el error de posición en rotación es asintóticamente estable, siempre que $k_{2,i} > 0$.

4.2.2. Estabilidad del Modo Deslizante

Con el fin de establecer condiciones bajo las cuales el modo deslizante es estable, se reescriben las ecuaciones (4.8) y (4.16) con (4.9) y (4.17) respectivamente, como:

$$\begin{aligned}\dot{s}_{j,i,k} &= \bar{f}_{j,i,k} - \lambda_{j,i,k}|s_{j,i,k}|^{1/2} \text{sigm}(\sigma_{j,i,k}s_{j,i,k}) + \vartheta_{j+2,i,k}, \\ \vartheta_{j+2,i,k} &= -\lambda_{j+2,i,k} \text{sigm}(\sigma_{j+2,i,k}s_{j,i,k}), \quad i = 1, \dots, n, j = 1, 2 \text{ y } k = 1, 2, 3.\end{aligned}\quad (4.27)$$

Ahora se introduce la siguiente suposición:

Suposición 4.1. El término de perturbación $\bar{f}_{j,i,k}$ es acotado en una región de operación admisible D por:

$$|\bar{f}_{j,i,k}| \leq \delta_{j,i,k}|s_{j,i,k}|^{1/2} \quad (4.28)$$

Con $\delta_{j,i,k}$ una constante positiva.

Teorema 4.1. Bajo la suposición 4.1 el origen $s_{j,i,k} = 0$ es un punto de equilibrio que es globalmente asintóticamente estable si las ganancias $\lambda_{j,i,k}$ y $\lambda_{j+2,i,k}$ satisfacen:

$$\begin{aligned}\lambda_{j,i,k} &> 2\delta_{j,i,k} \\ \lambda_{j+2,i,k} &> \lambda_{j,i,k} \frac{5\delta_{j,i,k}\lambda_{j,i,k} + 4\delta_{j,i,k}^2}{2(\lambda_{j,i,k} - 2\delta_{j,i,k})}\end{aligned}\quad (4.29)$$

Además, todas las trayectorias convergen en un tiempo finito al origen, delimitada por la parte superior $T = \frac{2V^{\frac{1}{2}}(s_{j,i,k}(0))}{\tilde{\gamma}}$, donde $s_{j,i,k}(0)$ es el estado inicial y $\tilde{\gamma}$ es una constante que depende de las ganancias $\lambda_{j,i,k}$, $\lambda_{j+2,i,k}$ y el coeficiente de perturbación $\bar{f}_{j,i,k}$. Esto, junto con la función de lyapunov V , que se pueden ver en la demostración del teorema 4.1 en [19].

De esta manera, tomando ganancias adecuadas tanto para el algoritmo de modos deslizantes como para la dinámica en los errores de posición, tendremos un sistema estable. Además, dependiendo de que tan grandes sean estos valores de ganancias sera la velocidad de respuesta.

4.3. Simulación de Control Distribuido

Con el fin de verificar la validez y la eficiencia del algoritmo de control distribuido propuesto, se realizan simulaciones en Matlab/Simulink utilizando los mismos parámetros que en la simulación para un solo quadrotor, estos parámetros son los mismo para cada i -esimo

quadrotor, que se enumeran en la Tabla 3.1.

El grupo está compuesto por 3 agentes seguidores más el líder virtual, se toma primeramente una topología de comunicación como en Fig. 4.1 para aplicar la ley de control con el algoritmo (2.29) en una formación fija, prosiguiendo con una topología de comunicación como en Fig. 4.7 para aplicar el algoritmo (2.30) también en una formación fija, con fines comparativos. Por último en esta sección se introduce una formación orientada hacia un frente deseado [23].

4.3.1. Formación Fija

Todos Conocen la Referencia

Para este ejemplo se tienen 3 agentes más el líder virtual con la topología de comunicación donde todos conocen la referencia como en Fig. 4.1 y su matriz de adyacencia es:

$$\mathcal{A}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.30)$$

Y su matriz laplaciana:

$$\mathcal{L}_4 = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.31)$$

Los parámetros de cada uno de los agentes se toman como en la Tabla 3.1. La trayectoria de referencia será la siguiente: en los primeros 10 segundos de la simulación, el quadrotor sube hasta una altura deseada 0.5 m, después de los 10 segundos se tiene como referencia para x y y , $x = 5 \cos(\frac{2\pi(t-10)}{50})$ y $y = 2.5 \sin(\frac{4\pi(t-10)}{50})$. La formación se llevara acabo con los siguientes vectores de separación:

$$\begin{aligned} \Delta_1 &= [0 \ 0 \ 0]^T \\ \Delta_2 &= [\frac{1}{2} \ -\frac{1}{2} \ 0]^T \\ \Delta_3 &= [-\frac{1}{2} \ -\frac{1}{2} \ 0]^T \end{aligned} \quad (4.32)$$

Las condiciones iniciales en cada uno de los quadrotors, deben de ser acorde con su formación, esto para que no haya colisiones. Siendo de esta manera, las condiciones iniciales son $X_1(0) = [5, 0, 0.2, 0, 0, 0, 0, 0, 0, 0, 0]^T$ $X_2(0) = [5.5, -0.5, 0.2, 0, 0, 0, 0, 0, 0, 0, 0]^T$ y

$$X_3(0) = [4.5, -0.5, 0.2, 0, 0, 0, 0, 0, 0, 0, 0, 0]^T$$

En la Fig. 4.2 se muestra el resultado en la posición tridimensional de una simulación de 60 segundos. En esta simulación el ángulo de *Yaw* y la formación no fueron modificados, El ángulo *Yaw* se mantiene siempre en cero y la formación permanece durante toda la simulación, como se puede apreciar en la Fig. 4.3 que muestra las posiciones de los quadrotors a los 10s, 20s, 30s y 35s. En las Figs. 4.4, 4.5 y 4.6 se muestran las gráficas de posición (x_i, y_i, z_i) , ángulos $(Pitch_i, Roll_i, Yaw_i)$ y Fuerzas $(F1_i, F2_i, F3_i, F4_i)$ respectivamente, (para $i = 1, 2, 3$).

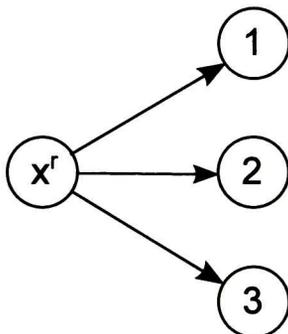


Figura 4.1: Topología de comunicación donde todos conocen la referencia.

Los parámetros de control $k_{1,i} = \text{diag}(1, 0.9, 0.9)$, $\lambda_{1,i,k} = \lambda_{2,i,k} = 9$ ($k = 1, 2, 3$), $\lambda_{3,i,k} = \lambda_{4,i,k} = 0.9$, $\sigma_{1,i,k} = \sigma_{2,i,k} = \sigma_{3,i,k} = \sigma_{4,i,k} = 1$ y $k_{2,i} = \text{diag}(2, 2, 2)$. Los resultados dieron un seguimiento de trayectoria en formación fija satisfactorio.

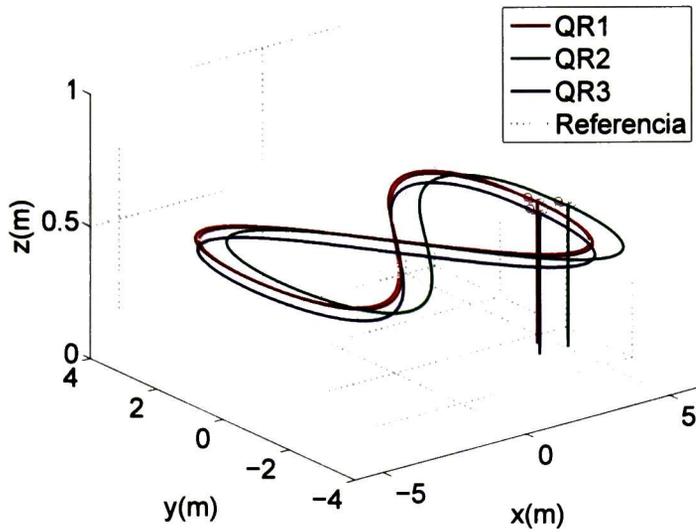


Figura 4.2: Seguimiento de trayectoria en formación fija, con topología donde todos conocen la referencia.

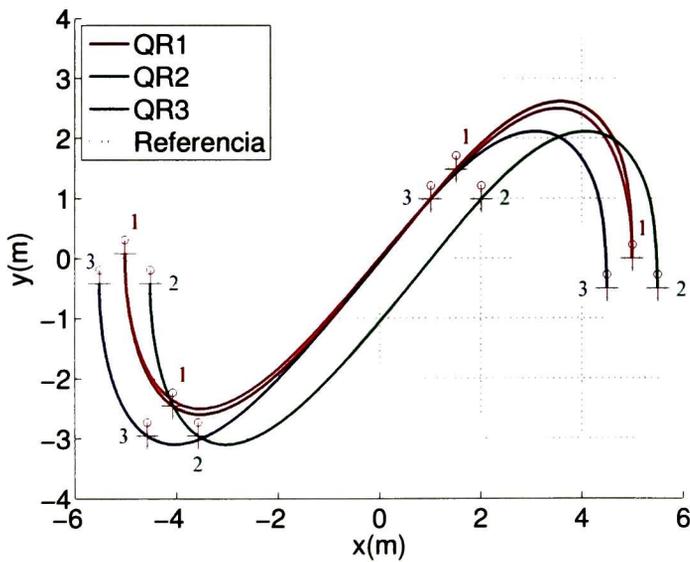


Figura 4.3: Plano (x,y) avance, formación fija, con topología donde todos conocen la referencia.

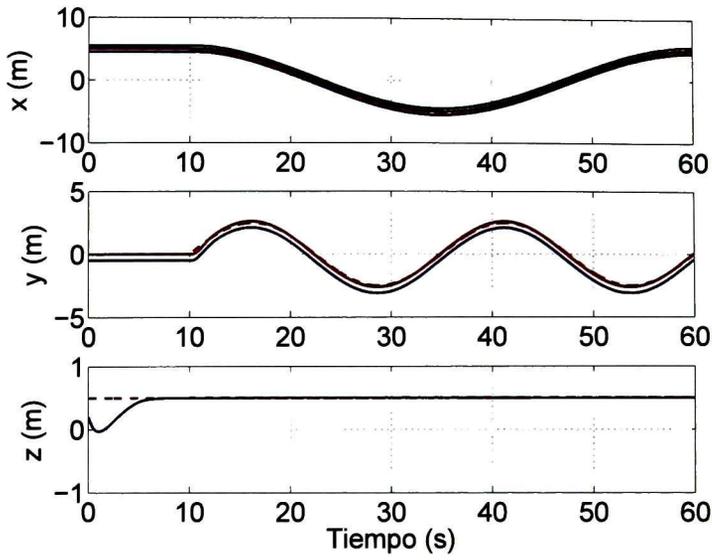


Figura 4.4: Gráfica de posición (x_i, y_i, z_i) , formación fija, con topología donde todos conocen la referencia.

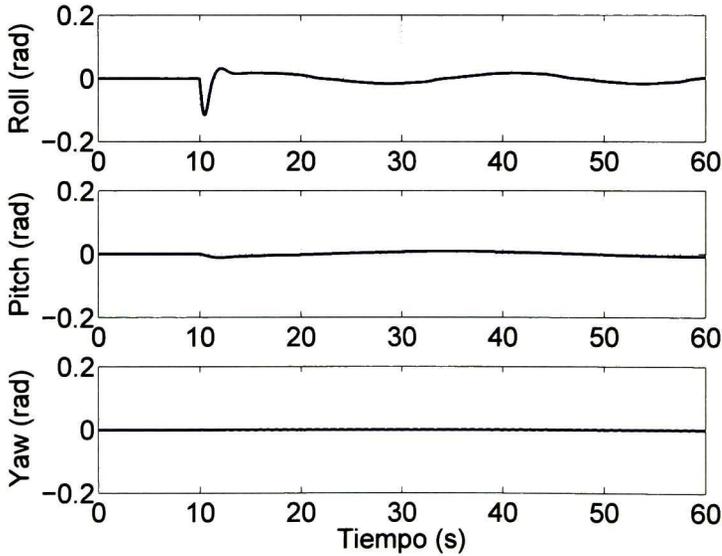


Figura 4.5: Gráfica de ángulos $(Pitch_i, Roll_i, Yaw_i)$, formación fija, con topología donde todos conocen la referencia.

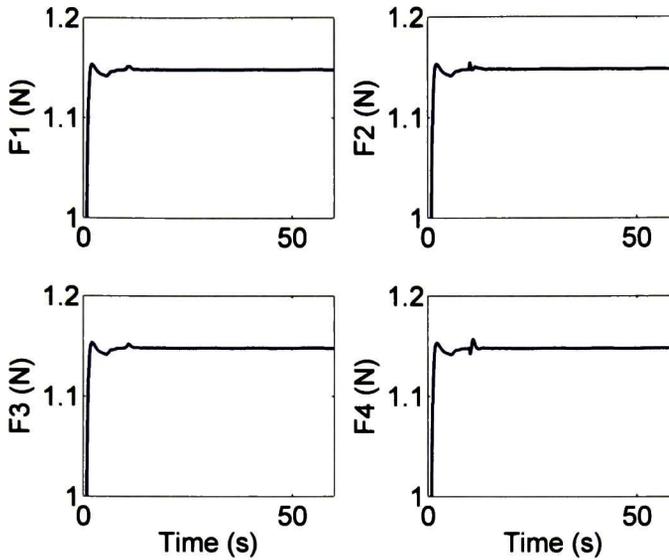


Figura 4.6: Gráfica de fuerzas, formación fija, con topología donde todos conocen la referencia.

Topología con Árbol de Expansión

Este ejemplo es similar al anterior, sólo se cambia la topología de comunicación donde todos conocen la referencia a una topología de comunicación con árbol de expansión y por ende el algoritmo de consenso. Con fines de comparación los parámetros de cada uno de los agentes, la trayectoria de referencia, los vectores de separación, las condiciones iniciales y hasta los parámetros de control son los mismos que en el ejemplo anterior. Se tiene entonces una topología de comunicación como en 4.7 y su matriz de adyacencia es:

$$\mathcal{A}_4 = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.33)$$

Y su matriz laplaciana:

$$\mathcal{L}_4 = \begin{bmatrix} 1 & 0 & 0 & -1 \\ -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad (4.34)$$

En la Fig. 4.8 se muestra el resultado en la posición tridimensional de una simulación de 60 segundos. En esta simulación el ángulo de *Yaw* y la formación no fueron modificados, El

ángulo Yaw se mantiene siempre en cero y la formación permanece durante toda la simulación, como se puede apreciar en la Fig. 4.9 que muestra las posiciones de los quadrotors a los 10s, 20s, 30s y 35s. En las Figs. 4.10, 4.11 y 4.12 se muestran las gráficas de posición (x_i, y_i, z_i) , ángulos $(Pitch_i, Roll_i, Yaw_i)$ y Fuerzas $(F1_i, F2_i, F3_i, F4_i)$ respectivamente, (para $(i = 1, 2, 3)$).

En comparativa de este ejemplo con el anterior, se puede observar que la dinámica de $QR1$ permanece igual pues sigue en comunicación directa con el líder, sin embargo la dinámica de $QR2$ se ve un poco más lenta, y el caso más crítico es el de $QR3$ que está aun más lejos de la comunicación con el líder, estos retardos se pueden observar con mayor claridad en las gráficas de posición en z y en el ángulo de $Roll$ al comienzo de la trayectoria, después alcanzan una dinámica similar a $QR1$ tanto $QR2$ como $QR3$. Aun así los resultados dieron un seguimiento de trayectoria en formación fija satisfactorio.

Para tomar a consideración, este sería el caso más crítico para un grupo de 3 seguidores, dependiendo del tipo de comunicación que se pueda garantizar en la aplicación, de la cantidad de emisores y receptores se puede hacer una topología de comunicación más estable.

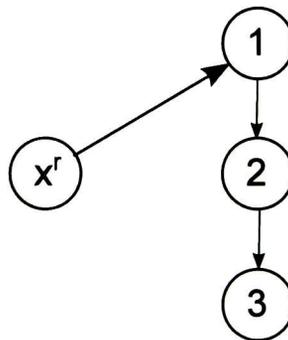


Figura 4.7: Topología de comunicación con Árbol de Expansión.

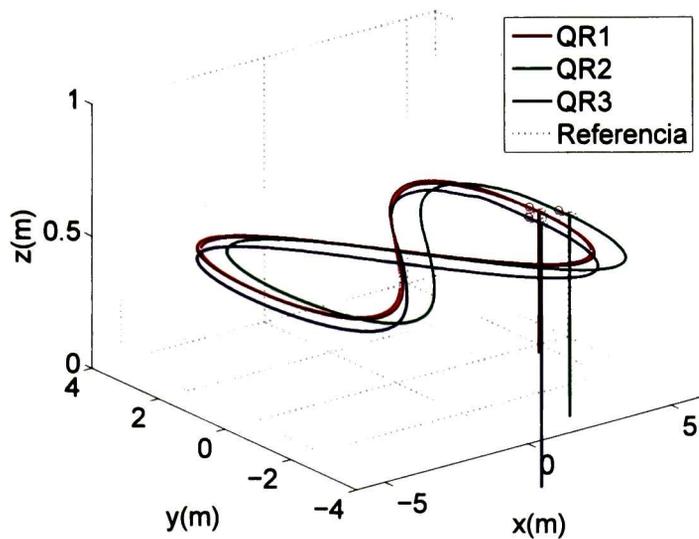


Figura 4.8: Seguimiento de trayectoria en formación fija, con topología con árbol de expansión.

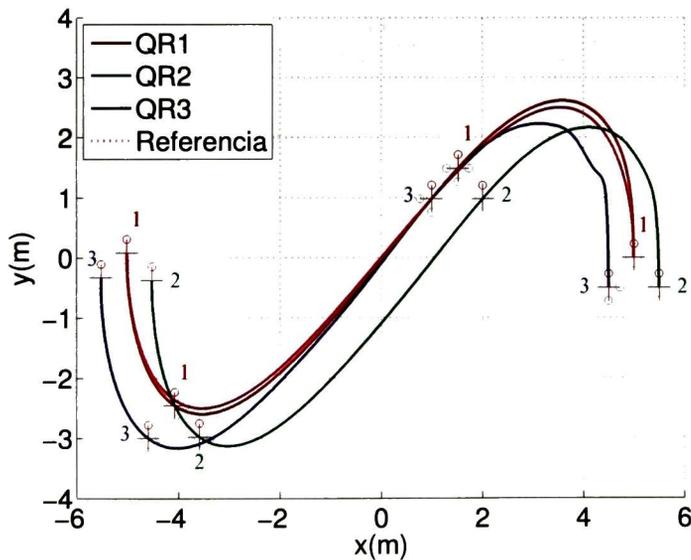


Figura 4.9: Plano (x,y) avance, formación fija, con topología con árbol de expansión.

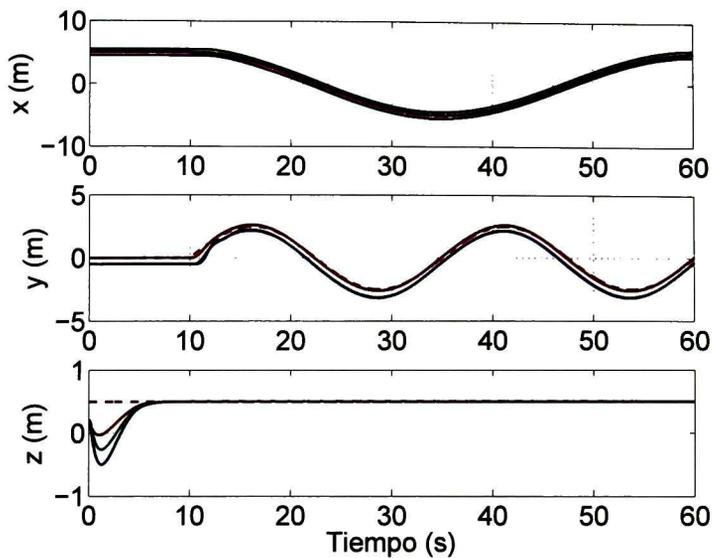


Figura 4.10: Gráfica de posición (x_i, y_i, z_i) , formación fija, con topología con árbol de expansión.

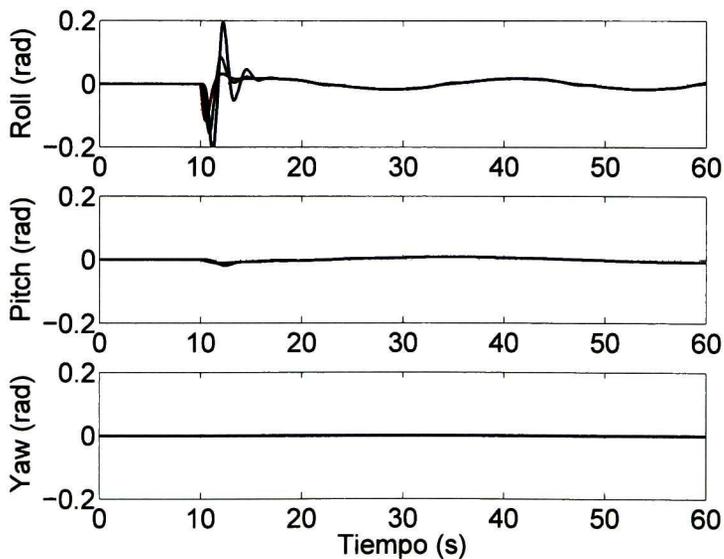


Figura 4.11: Gráfica de ángulos $(Pitch_i, Roll_i, Yaw_i)$, formación fija, con topología con árbol de expansión.

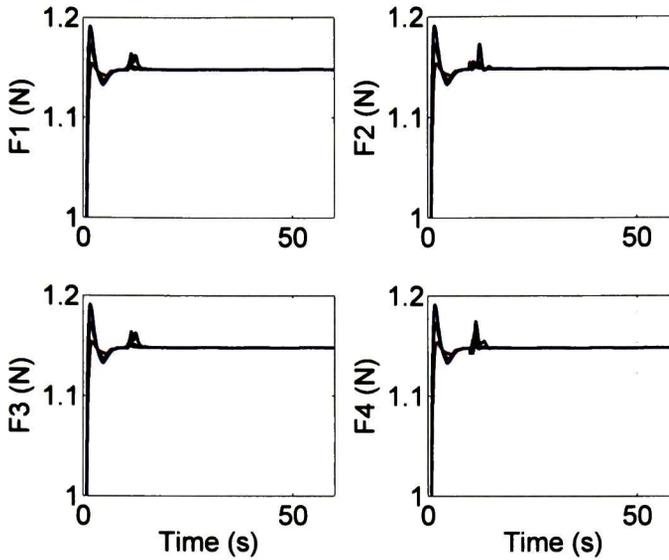


Figura 4.12: Gráfica de fuerzas, formación fija, con topología con árbol de expansión.

Topología Cambiante

Para este ejemplo, con el fin de probar el funcionamiento bajo topologías cambiantes en el tiempo, se tienen 3 agentes más el líder virtual con tres topologías de comunicación entre las cuales se va cambiando a determinado tiempo, la primer topología de comunicación sucede en los segundos 0-20s donde todos conocen la referencia, la siguiente topología sucede en los segundos 20-40s siendo el agente 2 el único que conoce la referencia y a su vez siendo padre de los agentes 1 y 3, la ultima topología sucede en los segundos 40-60s donde el agente 3 conoce la referencia, es padre del agente 2 y este a su vez es padre del agente 1 como se ilustra en Fig. 4.13.

Tabla 4.1: Matrices de adyacencia respectivas a los ejemplos en Fig. 4.13.

$$\begin{aligned}
 \mathcal{A}_4 &= \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} & \mathcal{A}_4 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} & \mathcal{A}_4 &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \\
 & \text{(a)} & \text{(b)} & \text{(c)}
 \end{aligned}$$

En cuanto a los parámetros de cada uno de los agentes, la trayectoria de referencia,

Tabla 4.2: Matrices Laplacianas respectivas a los ejemplos en Fig. 4.13.

$$\mathcal{L}_4 = \begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathcal{L}_4 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathcal{L}_4 = \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

(a)
(b)
(c)

los vectores de separación y las condiciones iniciales, serán las mismas que en los ejemplos anteriores.

En la Fig. 4.14 se muestra el resultado en la posición tridimensional de una simulación de 60 segundos. En esta simulación el ángulo de *Yaw* y la formación no fueron modificados, El ángulo *Yaw* se mantiene siempre en cero y la formación permanece durante toda la simulación, como se puede apreciar en la Fig. 4.15 que muestra las posiciones de los quadrotors a los 10s, 20s, 30s y 35s. En las Figs. 4.16, 4.17 y 4.18 se muestran las gráficas de posición (x_i, y_i, z_i) , ángulos $(Pitch_i, Roll_i, Yaw_i)$ y Fuerzas $(F1_i, F2_i, F3_i, F4_i)$ respectivamente, (para $i = 1, 2, 3$).

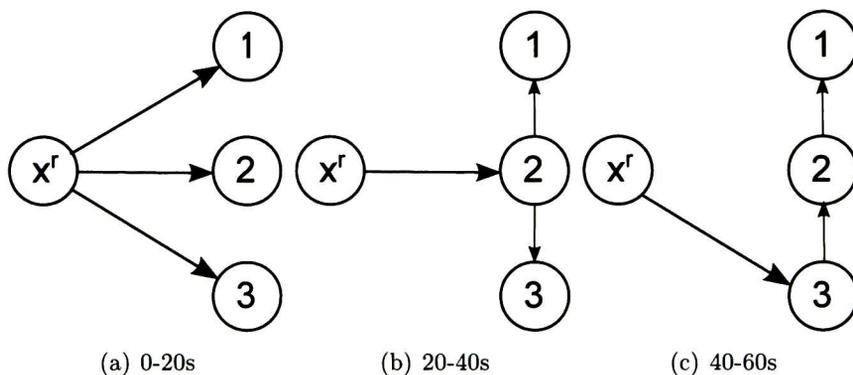


Figura 4.13: Topologías de comunicación cambiantes.

Los parámetros de control son los mismos que en los ejemplos anteriores. Se puede notar que al inicio de la trayectoria, siendo una topología de comunicación con todos conociendo la referencia, la respuesta es inmediata en los 3 agentes, a los 20s, el primer cambio de topología no se nota ninguna diferencia en el seguimiento de trayectoria, sin embargo a los 40s, el segundo cambio de topología, apenas y se nota una pequeña variación en el ángulo de *Roll* que no es significativa para dejar de seguir la trayectoria, siendo así un resultado satisfactorio en el seguimiento de trayectoria en formación fija con topología de comunicación cambiante.

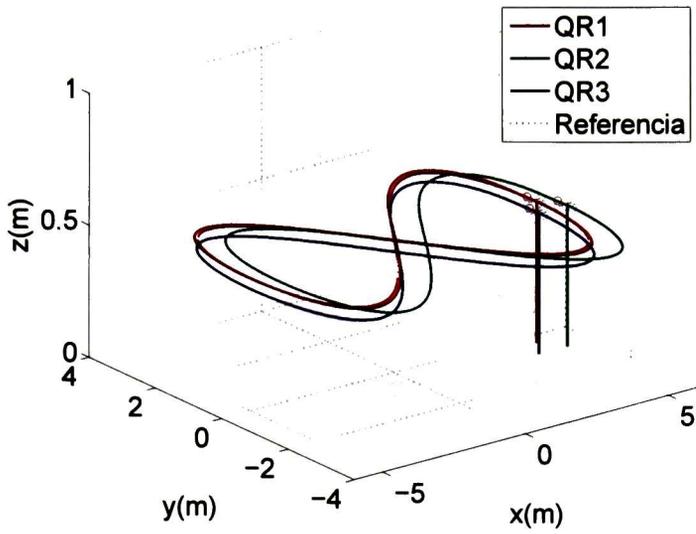


Figura 4.14: Seguimiento de trayectoria en formación fija, con topología cambiante.

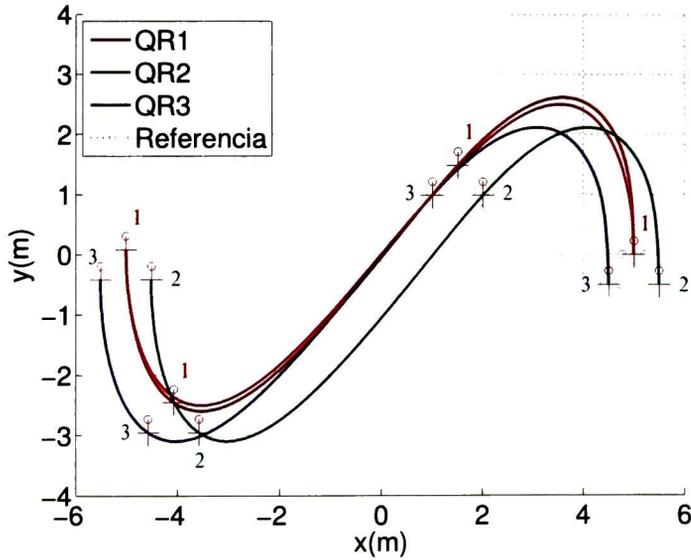


Figura 4.15: Plano (x,y) avance, formación fija, con topología cambiante.

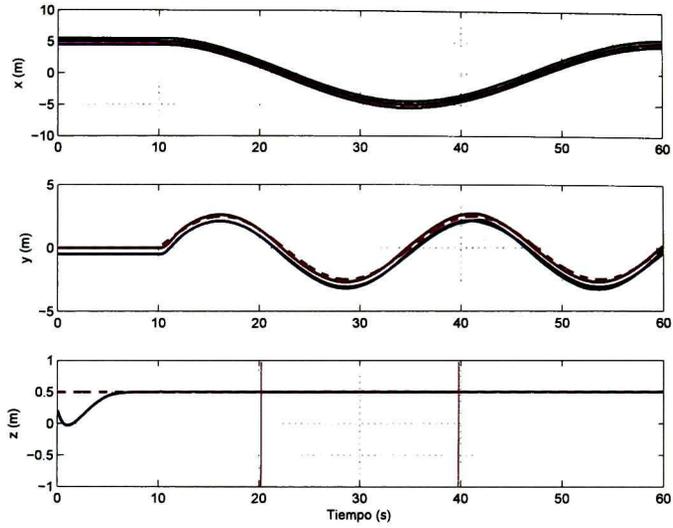


Figura 4.16: Gráfica de posición (x_i, y_i, z_i) , formación fija, con topología cambiante.

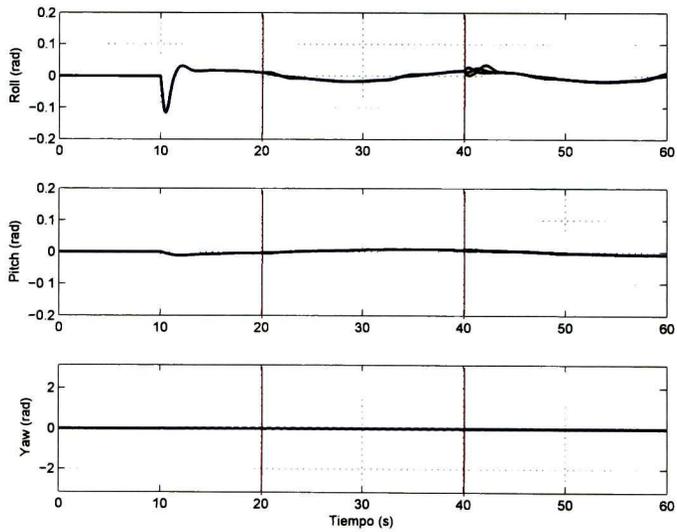


Figura 4.17: Gráfica de ángulos $(Pitch_i, Roll_i, Yaw_i)$, formación fija, con topología cambiante.

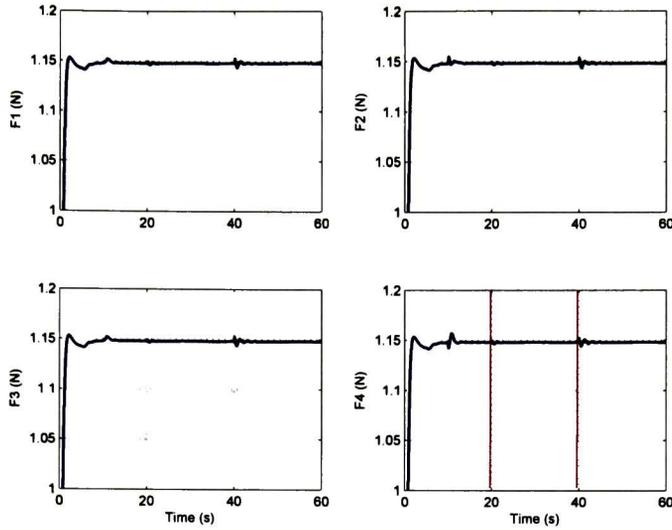


Figura 4.18: Gráfica de fuerzas, formación fija, con topología cambiante.

4.3.2. Formación Orientada

Con el fin de tener una formación orientada variable en el tiempo con $Q1$ al frente el ángulo de rotación deseado Yaw se deriva de:

$$\psi^* = \arctan\left(\frac{\dot{y}^r}{\dot{x}^r}\right) \quad (4.35)$$

Sin embargo el vector de separación $\bar{\Delta}_i$ es variable, cambiando con la matriz de rotación $R(\psi^*)$ la cual se usa como sigue:

$$\bar{\Delta}_i = \begin{bmatrix} C_{\psi^*} & -S_{\psi^*} & 0 \\ S_{\psi^*} & C_{\psi^*} & 0 \\ 0 & 0 & 1 \end{bmatrix} \Delta_i$$

Donde Δ_i es el vector de separación estático y $\bar{\Delta}_i$ es el vector de separación variante en el tiempo del i -th quadrotor. En fines de simulación o implementación se puede usar la función $atan2$ para evitar singularidades.

Para este ejemplo se toma la topología de comunicación de árbol como en Fig. 4.1, los vectores de separación estáticos como en los ejemplos anteriores al igual que los parámetros,

la trayectoria de referencia, condiciones iniciales y los parámetros de control.

Los resultados de la simulación se pueden apreciar en la Fig. 4.19 se muestra el resultado en la posición tridimensional de una simulación de 60 segundos. En esta simulación el ángulo de *Yaw* y la formación van cambiando en el tiempo, el ángulo de orientación *Yaw* va cambiando conforme la trayectoria lo va necesitando y la formación estática permanece durante toda la simulación, como se puede apreciar en la Fig. 4.20 que muestra las posiciones de los quadrotors a los 10s, 20s, 30s y 35s. En las Figs. 4.21, 4.22 y 4.23 se muestran las gráficas de posición (x_i, y_i, z_i) , ángulos $(Pitch_i, Roll_i, Yaw_i)$ y Fuerzas $(F1_i, F2_i, F3_i, F4_i)$ respectivamente, (para $i = 1, 2, 3$). Los resultados dieron un seguimiento de trayectoria en formación orientada satisfactorio.

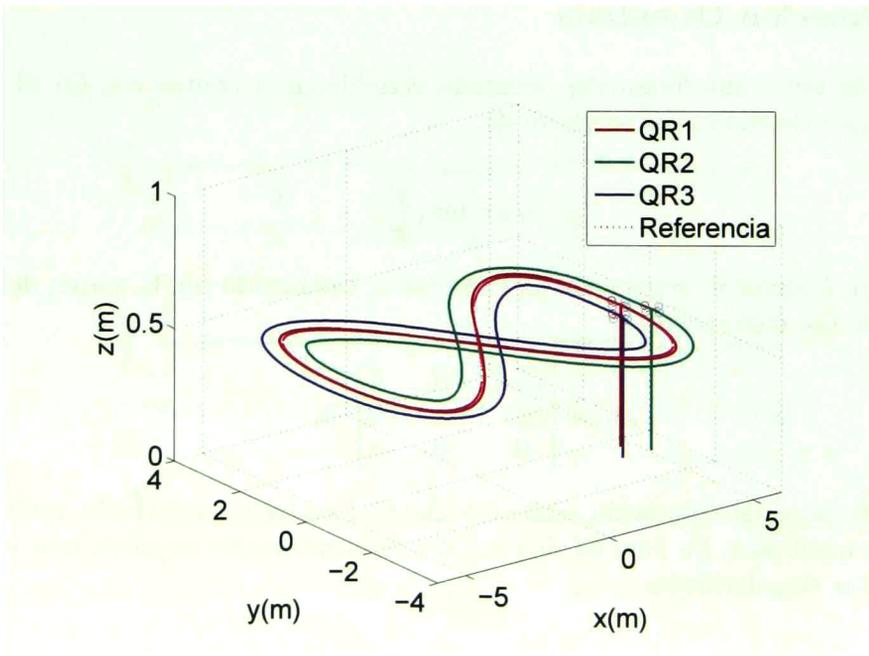


Figura 4.19: Seguimiento de trayectoria en formación orientada, todos conocen la referencia.

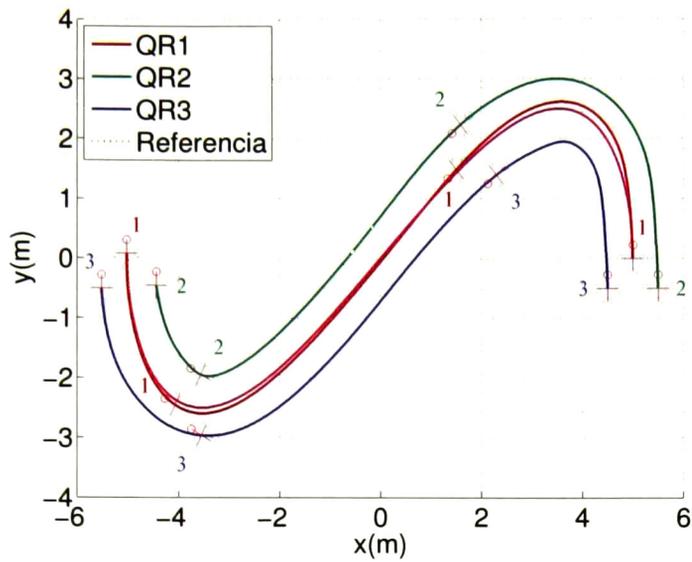


Figura 4.20: Plano (x,y) avance, formación orientada, todos conocen la referencia.

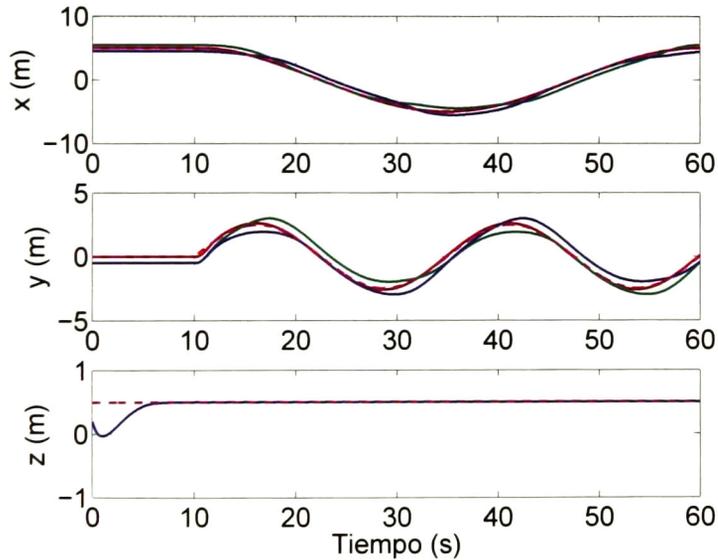


Figura 4.21: Gráfica de posición (x_i, y_i, z_i) , formación orientada, todos conocen la referencia.

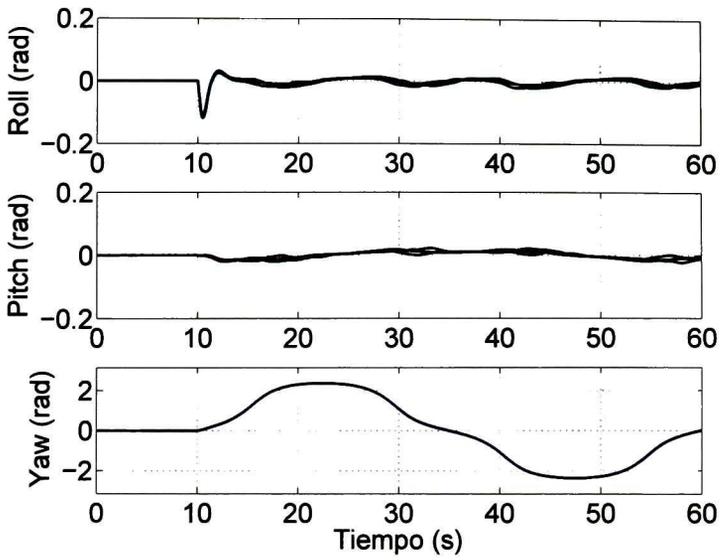


Figura 4.22: Gráfica de ángulos ($Pitch_i$, $Roll_i$, Yaw_i), formación orientada, todos conocen la referencia.

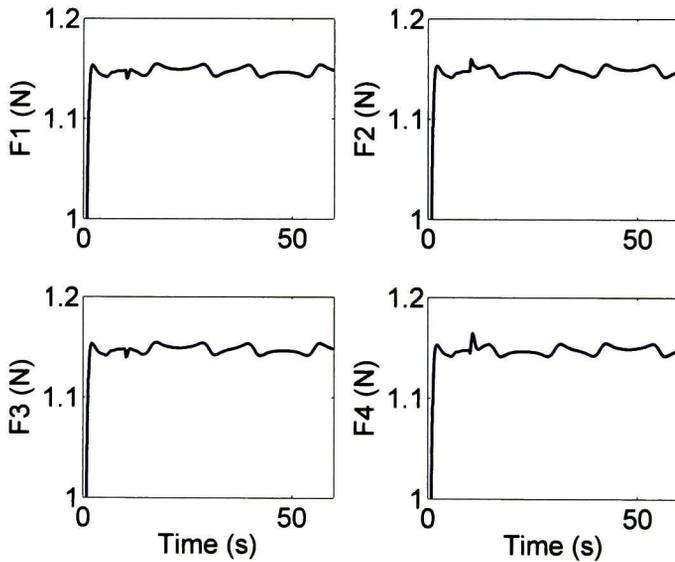


Figura 4.23: Gráfica de fuerzas, formación orientada, todos conocen la referencia.

Capítulo 5

Generación de Trayectoria

Una vez propuesta la ley de control para el seguimiento de una trayectoria con un grupo de UAV's en formación, en este capítulo se presentan primeramente conceptos básicos de navegación en la robótica móvil, siguiendo con el uso de una técnica de navegación con campos potenciales armónicos artificiales donde el grupo de UAV's se consideran como una carga puntual para evitar obstáculos. Esta técnica no tiene en cuenta la dinámica de los robots. El procedimiento se divide en dos partes, la primera es la generación de trayectoria y la segunda parte es aplicar la ley de control para seguirla.

5.1. Navegación

En el campo de estudio de los robots móviles, existe diversidad entre las técnicas de navegación que se pueden implementar, tales como descomposición de celdas [24], diagramas de Voronoi [25], campos potenciales [26] por mencionar algunas. Esta última es una técnica reactiva, lo que significa que se va trazando la trayectoria mientras está avanzando el robot, la cual es la gran diferencia entre las otras técnicas que primero generan la trayectoria y después se va moviendo el robot móvil. En particular cabe destacar que esta técnica no depende de la dinámica de los robots, y su movimiento es guiado por el gradiente del campo potencial artificial.

Navegación: Se llama navegación al conjunto de métodos y técnicas usados para dirigir el curso de un robot móvil a medida que éste atraviesa su entorno. Se supone que se debe llegar a algún destino pedido, sin perderse y sin chocar ni con obstáculos fijos, ni con otros móviles que eventualmente puedan aparecer en el camino. Para efectuar navegación lo más común es disponer de un *mapa*, el cual puede haberse generado previamente.

Dentro de la navegación existen dos tipos de vehículos, *vehículos guiados* los cuales recurren para el guiado a sistemas externos pre-programados, tales como un raíl, cables eléctricos enterrados que crean un campo magnético, etc. y *vehículos autónomos* que se mueven de un punto a otro sin necesidad de ayudas externas (al menos, no en todo momento), lo que los hace capaces de navegación genérica en un entorno dado a partir de órdenes de alto nivel; a la secuencia de tales órdenes se la suele llamar el *plan de misión*.

Un sistema de navegación autónomo se divide en cuatro etapas: *la percepción*, el robot debe interpretar sus sensores para obtener datos significativos, *la localización*, el robot debe determinar su posición en el medio ambiente, *la cognición*, el robot debe decidir cómo actuar para lograr su objetivo, y *el control de movimiento*, el robot debe modular sus salidas de motor para lograr la trayectoria deseada.

Para la navegación en un espacio de trabajo conocido y sin cambios constantes, es decir un espacio de trabajo en el cual no existan obstáculos móviles, es suficiente con la obtención de las características principales de los obstáculos fijos dentro de este espacio de trabajo.

5.2. Evasión de Obstáculos

En esta sección, se abordara la generación de una trayectoria dentro de un espacio de trabajo conocido, es decir, se conocen la posición y el radio de los obstáculos a una altitud constante. El movimiento del grupo de UAV's sería entonces a una altura constante y siguiendo puntos de trayectoria puestos por un plan de la misión. Para ello se utilizara una técnica de navegación con campos potenciales armónicos artificiales.

La teoría de campos potenciales considera al grupo como una partícula bajo la influencia de un campo potencial artificial que se está llevando por un campo resultante de fuerzas repulsivas (objetos) y sólo un atractivo (objetivo). Entonces el campo potencial completo está formado por:

$$U(x) = U_a(x) + U_r(x) \quad (5.1)$$

Donde x es un punto en el espacio de trabajo, U_a potencial atractivo y U_r potencial repulsivo. Sea el gradiente de un campo potencial artificial $U(x)$ y se representa por:

$$E(x) = -\nabla U(x) \quad (5.2)$$

Para cada punto x en el espacio de trabajo, la ecuación de gradiente (5.2) define un vector hacia una dirección de movimiento deseada. Moviéndose a lo largo de este vector gradiente se obtiene una trayectoria que depende del punto de partida (x_0, y_0) , a esta trayectoria se le

denomina "línea de gradiente" El problema con estos métodos es la aparición de los mínimos locales, estos son aquellos lugares que no son la posición de destino donde el potencial es cero. En tal situación, la ruta puede ser atrapada en una posición distinta de la de destino, o debido a la naturaleza discreta del método girar en torno a esta posición. Esto se resuelve en [26] con un "Campo Potencial Armónico Artificial" que es un tipo de campo potencial sin puntos mínimos locales.

Se necesita un campo potencial armónico artificial para la generación de la trayectoria, el dipolo en \mathbb{R}^2 se toma como ejemplo de un posible campo de potencial armónico artificial, sugerido por Utkin y Guldner [27]. Los obstáculos conocidos en el espacio de trabajo están protegidos por las zonas de seguridad. En el caso más simple con una carga puntual sin obstáculos y el objetivo en el origen, el campo potencial armónico artificial se define en las coordenadas polares planas (r, φ) como:

$$U(r, \varphi) = q \ln \frac{1}{r} \quad (5.3)$$

Donde q denota la "fuerza" de la singularidad, y r es la distancia al origen. Entonces su gradiente que es el vector de la dirección deseada es:

$$E = -\nabla U(r, \varphi) = \begin{bmatrix} -\frac{\partial U(r, \varphi)}{\partial r} \\ -\frac{1}{r} \frac{\partial U(r, \varphi)}{\partial \varphi} \end{bmatrix} = \begin{bmatrix} \frac{q}{r} \\ 0 \end{bmatrix} \quad (5.4)$$

Se puede observar que en este caso sin obstáculos el vector deseado solamente está cambiando la magnitud y no el ángulo, lo que significa que la ruta va directamente a la meta. Otra observación importante es que conforme el r va decreciendo, la magnitud en el vector deseado va aumentando, sin embargo al llegar a la meta, la generación de la trayectoria puede darse por terminada. Por otra parte, cuando existe algún obstáculo, este debe cumplir con las siguientes restricciones:

$$0 < q = \frac{R}{R + D} < 1 \quad (5.5)$$

Donde R es el radio del círculo de seguridad, y D es la distancia entre las dos singularidades, obstáculo y meta. Se supone que no se superponen las zonas de seguridad en los obstáculos con el punto de meta, es decir el punto de meta debe estar situado en el exterior de todas las zonas de seguridad. El campo potencial artificial total en un sistema de coordenadas polares con la meta en el origen y un solo obstáculo está dado por:

$$U(r, \varphi) = \frac{q}{2} \ln \frac{1}{r^2 - 2Dr \cos(\varphi) + D^2} - \ln \frac{1}{r} \quad (5.6)$$

Con el gradiente asociado derivado de la ecuación (5.2) se puede escribir:

$$E = \begin{bmatrix} -\frac{\partial U(r, \varphi)}{\partial r} \\ -\frac{1}{r} \frac{\partial U(r, \varphi)}{\partial \varphi} \end{bmatrix} = \begin{bmatrix} q \left(\frac{r - D \cos(\varphi)}{r^2 - 2Dr \cos(\varphi) + D^2} \right) - \frac{1}{r} \\ q \left(\frac{Dr \sin(\varphi)}{r^2 - 2Dr \cos(\varphi) + D^2} \right) \end{bmatrix} \quad (5.7)$$

Se puede observar que ahora en este caso el vector deseado está cambiando tanto en magnitud y ángulo. Con el fin de obtener un movimiento uniforme se considera un vector unitario deseado. Para múltiples obstáculos, se toma una sumatoria de campos potenciales repulsivos y sólo un campo potencial atractivo que es la meta. En este caso el campo de potencial total es:

$$U(r, \varphi) = \frac{1}{2} \sum_{i=1}^n q_i \ln \frac{1}{T_i} - \ln \frac{1}{r} \quad (5.8)$$

Donde n es el numero total de obstáculos, q_i es la fuerza del i -esimo obstáculo y T_i es la distancia entre el punto de carga y el i -esimo obstáculo, esto se representa esquemáticamente en la Figura 5.1. La distancia T_i se obtiene mediante la ley de cosenos, se deriva de la siguiente manera $T_i = r^2 - 2D_i r \cos(\varphi) + D_i^2$.

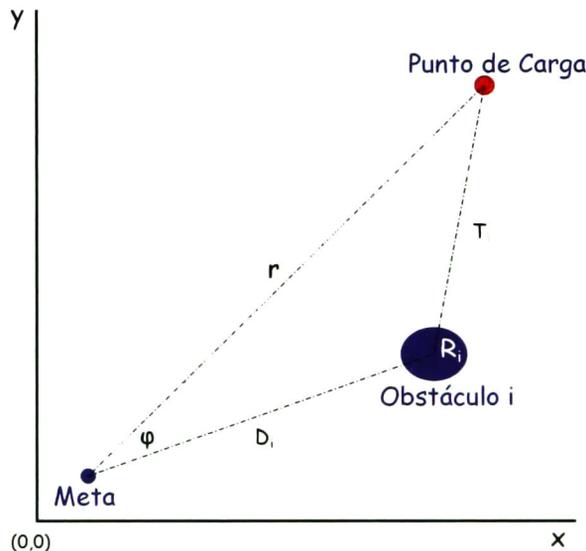


Figura 5.1: Esquema de distancias.

Existe una *línea singular* que se presenta cuando $\varphi = 0$ es decir, cuando el obstáculo se interpone directamente entre el punto de carga que son los vehículos y la meta, esto puede verse gráficamente en la Fig. 5.2. Esto debido a que en el numerador del gradiente del campo potencial en la parte de el ángulo deseado tiene un $\sin(\varphi)$ y $\sin(0) = 0$, físicamente nos podemos dar cuenta que es la línea en la cual da lo mismo irse por arriba que por abajo, esto se puede solucionar con una previa decisión de que cuando esto suceda, es decir, si $\varphi = 0$

aumentarlo o disminuirlo un poco, con el fin de sacarlo de esta línea singular y la generación de trayectoria siga haciendo su trabajo.

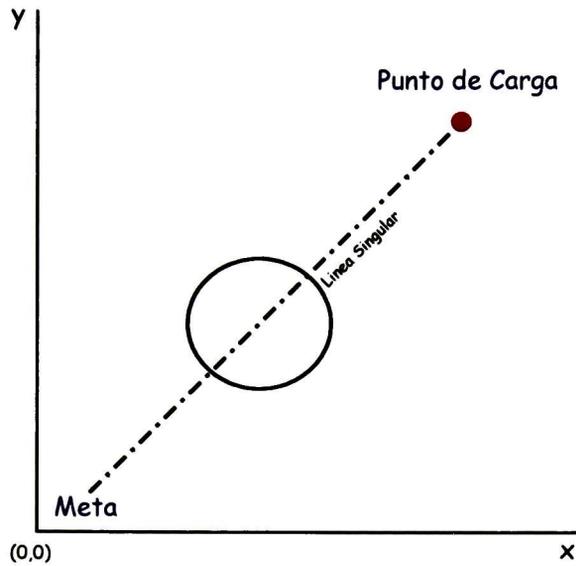


Figura 5.2: Línea singular.

5.3. Simulación de Evasión de Obstáculos en Formación Orientada

En esta sección se realiza una simulación con un ejemplo que aborda la problemática propuesta en este Capítulo, donde aparte de tener una ley de control para seguimiento de una trayectoria en consenso, se genera esta trayectoria dentro de un espacio de trabajo conocido. Esto es, conociendo una posición destino, generar una trayectoria por la cual se evadan obstáculos, independientemente de la posición inicial (nunca dentro de un obstáculo).

Con el fin de verificar el funcionamiento del esquema de generación de trayectoria y control cooperativo propuesto, se lleva a cabo una simulación en el que un grupo de tres quadrotors ($QR_i, i = 1, 2, 3$) tienen que generar y seguir la referencia de trayectoria deseada en una formación orientada con la topología de comunicación de árbol que se muestra en la Fig. 4.1.

Se supone que las condiciones iniciales de los quadrotors y las posiciones finales requeridas en la formación son conocidos a priori para evitar los cruces entre ellos. En esta topología de

comunicación la trayectoria deseada es conocida por todos los quadrotors. Para la generación de trayectoria, el espacio de trabajo plano (x, y) a una altitud deseada tiene que ser conocido a priori, en este caso, hay dos obstáculos cilíndricos con la posición y el radio conocidos, además, una posición de meta se propone con anterioridad. Los vectores de separación tanto estáticos como dinámicos, son adoptados del ejemplo en la sección 4.3.2.

Para este ejemplo se tiene un espacio de trabajo conocido, a una altitud constante de 1 metro con dos obstáculos propuestos con las siguientes características:

1m altitud	x	y	Radio
Obstáculo1	5	8	2
Obstáculo2	16	10	2

Con las siguientes condiciones iniciales y meta:

	x	y
Inicial	0	0
Meta	20	20

Las condiciones iniciales de cada quadrotor en este ejemplo son $X_1(0) = [0, 0, 0.2, 0, 0, 0, 0, 0, -0.8, 0, 0]$, $X_2(0) = [0, -0.5, 0.2, 0, 0, 0, 0, 0, -0.8, 0, 0, 0]^T$ y $X_3(0) = [-0.5, 0, 0.2, 0, 0, 0, 0, 0, -0.8, 0, 0]^T$. El retrato fase del campo potencial armónico con los obstáculos y trayectoria se muestra en la figura 5.3.

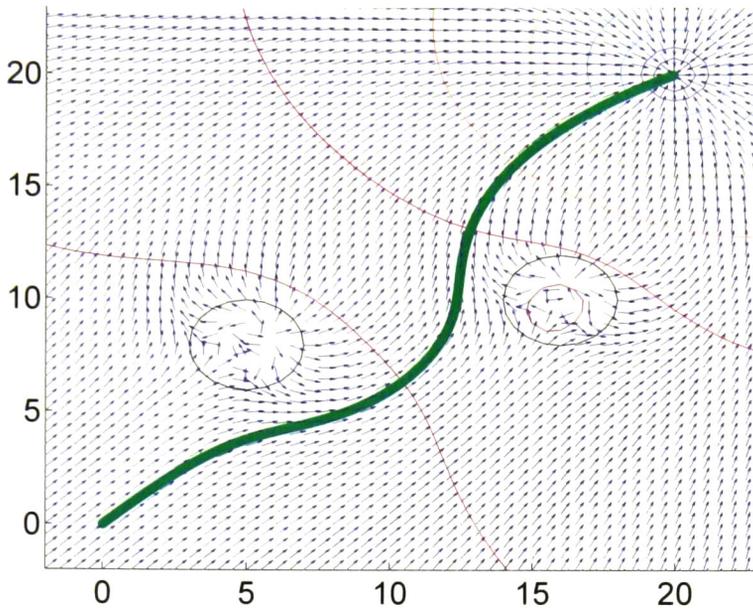


Figura 5.3: Retrato de fase

Los resultados de la simulación se pueden apreciar en la Fig. 5.4 se muestra el resultado en la posición tridimensional de una simulación de 70 segundos. En esta simulación el ángulo de Yaw y la formación van cambiando en el tiempo, el ángulo de orientación Yaw va cambiando conforme la trayectoria lo va necesitando y la formación estática permanece durante toda la simulación, como se puede apreciar en la Fig. 5.5 que muestra las posiciones de los quadrotors a los 10s, 20s, 30s, 40s, 50s y 60s. En las Figs. 5.6, 5.7 y 5.8 se muestran las gráficas de posición (x_i, y_i, z_i) , ángulos $(Pitch_i, Roll_i, Yaw_i)$ y Fuerzas $(F1_i, F2_i, F3_i, F4_i)$ respectivamente, (para $(i = 1, 2, 3)$). Los resultados dieron una generación de trayectoria con evasión de obstáculos y un seguimiento de trayectoria en formación orientada satisfactorio.

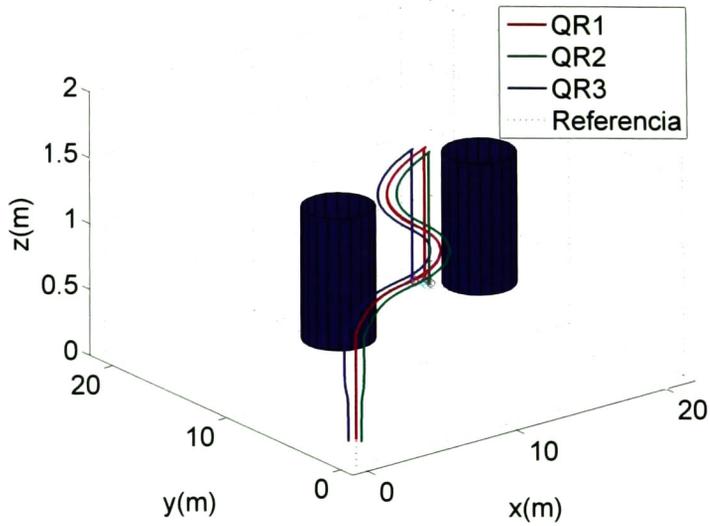


Figura 5.4: Seguimiento de trayectoria en formación orientada, con topología de árbol y evasión de obstáculos.

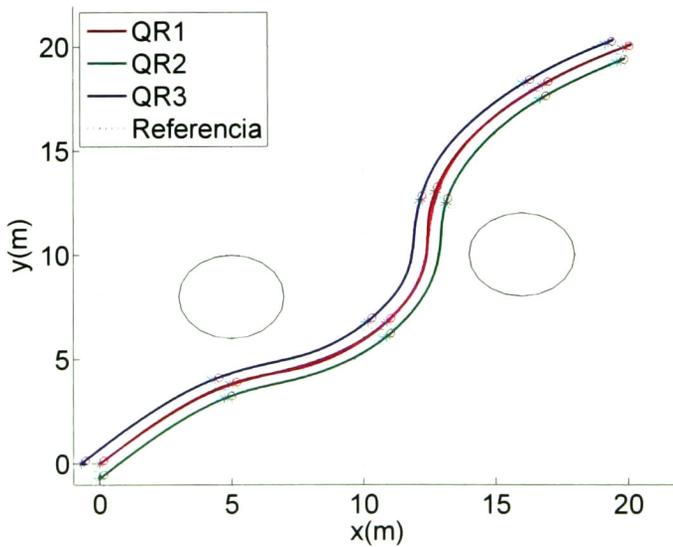


Figura 5.5: Plano (x,y) avance, formación orientada, con topología de árbol y evasión de obstáculos.

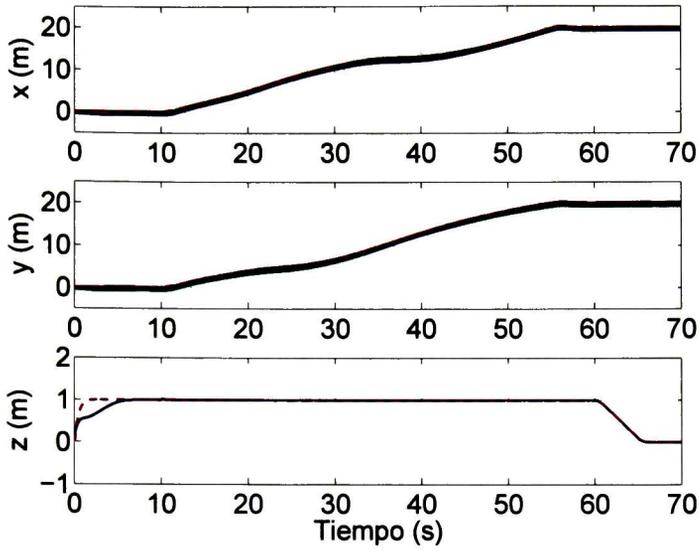


Figura 5.6: Gráfica de posición (x_i, y_i, z_i) , formación orientada, con topología de árbol y evasión de obstáculos.

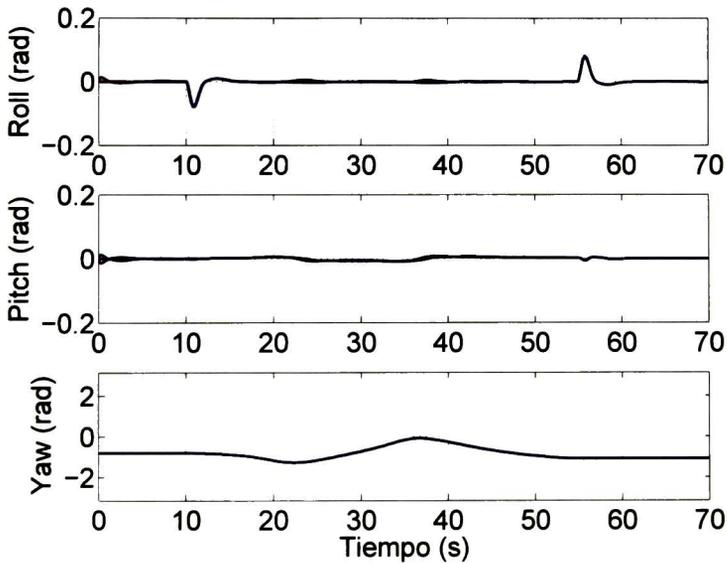


Figura 5.7: Gráfica de ángulos $(Pitch_i, Roll_i, Yaw_i)$, formación orientada, con topología de árbol y evasión de obstáculos.

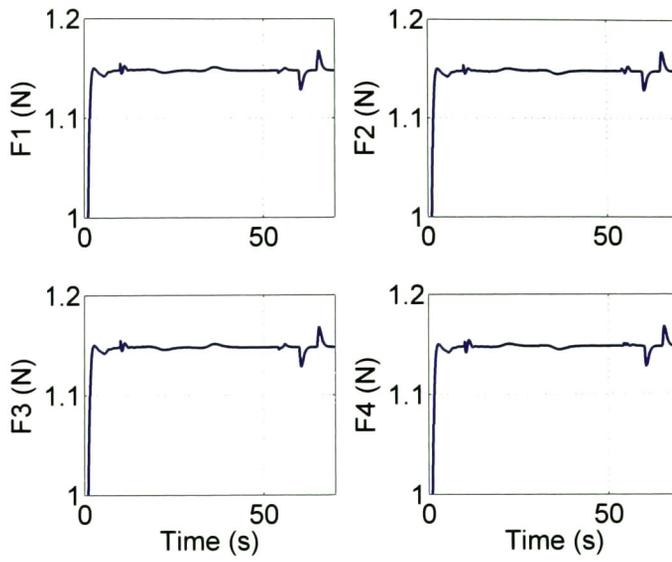


Figura 5.8: Gráfica de fuerzas, formación orientada, con topología de árbol y evasión de obstáculos.

Capítulo 6

Conclusiones y Trabajo Futuro

6.1. Conclusiones

En este trabajo se presentó un sistema de navegación de UAV's en formación, este sistema consiste en una ley de control para el seguimiento de trayectorias con el UAV quadrotor, junto con un algoritmo de generación de trayectoria para la evasión de obstáculos en un espacio de trabajo conocido, mediante el uso de algoritmos de consenso con el enfoque líder-seguidor.

Los algoritmos son capaces de llegar a consenso bajo topologías de comunicación tanto fijas como cambiantes en el tiempo, los parámetros de las aeronaves no deben ser idénticos pero si similares, con capacidades físicas de seguir el mismo avance y velocidad. Dependiendo del tipo de trabajo a realizar y la infraestructura con la que se cuente se elige la comunicación entre agentes, siendo la mejor de las topologías una comunicación directa del líder con todos los seguidores.

Para la ley de control de seguimiento de trayectoria con el UAV quadrotor se utilizaron las técnicas de control de *Block Control* y *Super Twisting*, se eligieron estas por sus cualidades de robustez, sin embargo los algoritmos de consenso son compatibles con cualquier tipo de control y son independientes también del modelo matemático a controlar. Para este caso con la variable a consenso siendo la posición, el algoritmo entrega una velocidad deseada y es trabajo de cualquier control llevar la velocidad física a esa velocidad deseada.

El algoritmo para la generación de trayectoria es también independiente del modelo matemático del sistema móvil en cuestión, en este caso se hace evasión en dirección al eje (x, y) pero es posible hacer la evasión por otro plano diferente definido en 3D.

Los resultados en las simulaciones fueron satisfactorios.

6.2. Trabajo Futuro

Como trabajo futuro, en la parte del consenso, se pretende analizar el sistema con retardos en la comunicación o incluso ausencia de información durante algún instante así como llegar a obtener una topología con un numero de agentes variante, llegando así a un algoritmo útil para enjambres.

Por parte de la formación, analizar con el mismo principio de funcionamiento de los campos potenciales, un algoritmo para evasión de colisiones entre los agentes de la formación, evadiendo obstáculos en el eje (x, y) y evadiendo colisiones entre agentes ya sea en el eje (x, z) o en el eje (y, z) .

Usar el principio de los algoritmos de consenso para un proyecto muy citado en la literatura que es el de un quadrotor aterrizando en un carrito, se podría hacer consenso en la variables de interés $[x, y]$ y dejar libre la variable z del quadrotor para disminuirla hasta aterrizar sobre el carrito. Con el líder siendo el carrito, la trayectoria que debe seguir el quadrotor es posible.

Bibliografía

- [1] V Mistler, A Benallegue, and NK M'sirdi. Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback. In *Robot and Human Interactive Communication, 2001. Proceedings. 10th IEEE International Workshop on*, pages 586–593. IEEE, 2001.
- [2] E Altuğ, JP Ostrowski, and R Mahony. Control of a quadrotor helicopter using visual feedback. In *Robotics and Automation, 2002. Proceedings. ICRA '02. IEEE International Conference on*, volume 1, pages 72–77. IEEE, 2002.
- [3] B Bijmens, QP Chu, GM Voorsluijs, and JA Mulder. Adaptive feedback linearization flight control for a helicopter uav. In *AIAA Guidance, Navigation, and Control Conference and Exhibit*, pages 1–10, 2005.
- [4] L Derafa, L Fridman, A Benallegue, and A Ouldali. Super twisting control algorithm for the four rotors helicopter attitude tracking problem. In *Variable Structure Systems (VSS), 2010 11th International Workshop on*, pages 62–67. IEEE, 2010.
- [5] S Bouabdallah and R Siegwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2247–2252. IEEE, 2005.
- [6] T Madani and A Benallegue. Backstepping sliding mode control applied to a miniature quadrotor flying robot. In *IEEE Industrial Electronics, IECON 2006-32nd Annual Conference on*, pages 700–705. IEEE, 2006.
- [7] T Madani and A Benallegue. Backstepping control with exact 2-sliding mode estimation for a quadrotor unmanned aerial vehicle. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 141–146. IEEE, 2007.
- [8] L Luque-Vega, B Castillo-Toledo, and AG Loukianov. On the quadrotor trajectory tracking problem via super twisting technique. In *World Automation Congress (WAC), 2012*, pages 1–6, June 2012.

- [9] A Fujimori, T Fujimoto, and G Bohacs. Distributed leader-follower navigation of mobile robots. In *Control and Automation, 2005. ICCA '05. International Conference on*, volume 2, pages 960–965 Vol. 2, June 2005.
- [10] C López-Limom, A Cervantes-Herrera, J Ruiz-León, and A Ramirez-Trevino. Formation and trajectory tracking of a class of nonlinear systems with super twisting control. In *Emerging Technologies & Factory Automation (ETFA), 2011 IEEE 16th Conference on*, pages 1–6. IEEE, 2011.
- [11] DA Mercado, R Castro, and R Lozano. Quadrotors flight formation control using a leader-follower approach. In *Control Conference (ECC), 2013 European*, pages 3858–3863, July 2013.
- [12] Y Zhou, X Dong, G Lu, and Y Zhong. Time-varying formation control for unmanned aerial vehicles with switching interaction topologies. In *Unmanned Aircraft Systems (ICUAS), 2014 International Conference on*, pages 1203–1209, May 2014.
- [13] L Luque-Vega, B Castillo-Toledo, and AG Loukianov. Robust block second order sliding mode control for a quadrotor. *Journal of the Franklin Institute*, 349(2):719–739, 2012.
- [14] Z Zuo. Trajectory tracking control design with command-filtered compensation for a quadrotor. *Control Theory & Applications, IET*, 4(11):2343–2355, 2010.
- [15] AG Lukyanov and VI Utkin. Methods of reducing equations for dynamic-systems to a regular form. *Automation and Remote Control*, 42(4):413–420, 1981.
- [16] AG Loukianov. Robust block decomposition sliding mode control design. *Mathematical Problems in Engineering*, 8(4-5):349–365, 2002.
- [17] V Utkin, J Guldner, and J Shi. *Sliding mode control in electro-mechanical systems*, volume 34. CRC press, 2009.
- [18] L Fridman and A Levant. Higher order sliding modes. *Sliding mode control in engineering*, 11:53–102, 2002.
- [19] JA Moreno and M Osorio. A lyapunov approach to second-order sliding mode controllers and observers. In *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, pages 2856–2861, Dec 2008.
- [20] W Ren and RW Beard. *Distributed consensus in multi-vehicle cooperative control*. Springer, 2008.
- [21] W Ren, RW Beard, and EM Atkins. Information consensus in multivehicle cooperative control. *IEEE Control systems magazine*, 2(27):71–82, 2007.

- [22] A Levant. Universal single-input-single-output (siso) sliding-mode controllers with finite-time convergence. *Automatic Control, IEEE Transactions on*, 46(9):1447–1451, 2001.
- [23] Santiaguillo-Salinas J and Aranda-Bricaire E. Leader-oriented formation flight control for multi-agent systems. *Memorias del XVI Congreso Latinoamericano de Control Automatico CLCA*, pages pp. 935–940., 2014.
- [24] T Arney. An efficient solution to autonomous path planning by approximate cell decomposition. 2007.
- [25] P Bhattacharya and ML Gavrilova. Voronoi diagram in optimal path planning. In *Voronoi Diagrams in Science and Engineering, 2007. ISVD'07. 4th International Symposium on*, pages 38–47. IEEE, 2007.
- [26] O Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research*, 5(1):90–98, 1986.
- [27] J Guldner, V Utkin, et al. Sliding mode control for an obstacle avoidance strategy based on an harmonic potential field. In *Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on*, pages 424–429. IEEE, 1993.



CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS AVANZADOS DEL I.P.N. UNIDAD GUADALAJARA

El Jurado designado por la Unidad Guadalajara del Centro de Investigación y de Estudios Avanzados del Instituto Politécnico Nacional aprobó la tesis

Sistema de navegación de UAV's en formación

del (la) C.

David FERNÁNDEZ FLORES

el día 25 de Septiembre de 2015.

Dr. Eduardo José Bayro Corrochano
Investigador CINVESTAV 3D
CINVESTAV Unidad Guadalajara

Dr. Antonio Ramírez Treviño
Investigador CINVESTAV 3A
CINVESTAV Unidad Guadalajara

Dr. Carlos Aurelio López Limón
Ingeniero Eléctrico
Intel Tecnología de México S.A. de
C.V



CINVESTAV - IPN
Biblioteca Central



SSIT0013504