



**CENTRO DE INVESTIGACIÓN Y DE ESTUDIOS
AVANZADOS DEL INSTITUTO POLITÉCNICO NACIONAL**

**UNIDAD ZACATENCO
DEPARTAMENTO DE INGENIERÍA ELÉCTRICA
SECCIÓN DE MECATRÓNICA**

**“Esquema de formación de robots móviles con evasión de
obstáculos estáticos y dinámicos”**

T E S I S

Que presenta

CRUZ MAURICIO ARTEAGA ESCAMILLA

Para obtener el grado de
DOCTOR EN CIENCIAS

EN LA ESPECIALIDAD DE
INGENIERÍA ELÉCTRICA

Directores de la Tesis:
**Dr. Rafael Castro Linares
Dr. Jaime Álvarez Gallegos**

Agradecimientos

A **Dios**, por acompañarme todos los días y guiarme a lo largo de mi vida, por ser mi fortaleza en los momentos de debilidad y por permitirme vivir.

A mis padres **Leticia** y **Alejandro**, en especial a mi madre, por apoyarme en todo momento, por los valores que me han inculcado, por sus consejos y por sus regaños que con justa razón me daban con el fin de formar un hombre de bien.

A mi esposa **Nayeli**, por todo su apoyo incondicional, por estar conmigo en los momentos de gran felicidad y de profunda tristeza. Gran parte de este logro se lo dedico a ella porque nunca dejó de confiar en mí.

A mi hija **Montserrat** y mi futuro bebé, por ser parte de mi vida y ser mi mayor estímulo para convertirme en una mejor persona y su ejemplo a seguir.

A mi hermana **Alejandra** y hermano **Antonio**, por ser parte importante de mi vida, por sus consejos, apoyos en los buenos y en los malos momentos. Gracias por todos estos años.

A mis asesores de tesis, el **Dr. Jaime Álvarez Gallegos** y el **Dr. Rafael Castro Linares**, por la oportunidad de desarrollar este trabajo, por el apoyo, tiempo, conocimiento compartido y sobre todo por darme la oportunidad de crecer profesional y académicamente.

A los Doctores de la Sección de Mecatrónica, por permitirme realizar mis estudios de Doctorado y transmitirme su conocimiento. Agradezco al Auxiliar M.en C. **Igor Morett Valenzulea** por el préstamo de equipo especializado en este proyecto y por todo el apoyo recibido.

Al **CINVESTAV** por todo su apoyo, al **CONACYT** por la beca otorgada para realizar mis estudios de Doctorado y por el apoyo dado por medio del proyecto de investigación CB-2015-01, 254329.

Resumen

Esta tesis está enfocada en estudiar el problema de control de formación con sincronización de robots móviles en presencia de obstáculos tanto estáticos como dinámicos. Se consideró un grupo de robots móviles terrestres y uno aéreo con cámara a bordo. Mediante el uso de la cámara monocular a bordo del robot aéreo, se estima la posición y orientación de los robots terrestres que componen la formación. De igual manera, se estima la posición y velocidad de los obstáculos dinámicos a evadir.

Primero, se presenta el modelo cinemático del robot móvil tipo unicycle y el modelo cinemático del robot móvil aéreo tipo cuádrirrotor, en el plano horizontal. Se describen los controladores propuestos, a nivel del robot. Estos controladores se diseñan tomando en cuenta la configuración y capacidades de cada tipo de robot. Además, se presenta un método de potenciales artificiales modificado, el cual toma en cuenta la posición y velocidad del obstáculo, y de la meta a alcanzar. También, se describe el modelo de proyección en perspectiva y el método de visión artificial usado para estimar la posición y orientación de los robots terrestres.

Se describe detalladamente el esquema de formación con sincronización de robots móviles propuesto en este trabajo. Se da un breve repaso al enfoque original de “*Cluster Space*” para resaltar la diferencia y contribución de esta tesis. Además, se presenta el análisis formal de estabilidad de los errores de seguimiento en la formación, para el caso de robots móviles terrestres. También, se presenta una explicación sobre la implementación del método de potenciales artificiales modificado en la formación.

Posteriormente, se describen brevemente todos los elementos que componen la plataforma experimental empleada. Se muestran los resultados experimentales considerando obstáculos estáticos y dinámicos en el esquema de formación usando tres robots móviles terrestres tipo unicycle.

Se describe el simulador 3D empleado para realizar las simulaciones realistas y verificar el desempeño del esquema de formación heterogénea, considerando obstáculos dinámicos; también, se estima la posición, orientación y velocidad de los robots terrestres. Finalmente, se dan algunas conclusiones e ideas de trabajo futuro.

Abstract

This thesis is focused on studying the problem of formation control with synchronization for mobile robots in the presence of obstacles, both dynamic and static. A group of ground mobile robots and one aerial robot with onboard camera was considered. By using the monocular camera onboard the aerial robot, the position and the orientation of the ground mobile robots that compose the formation, are estimated. In a similar way, the position and velocity of the dynamic obstacles to be avoided, are also estimated.

First, the kinematic model of unicycle type ground mobile robot and the kinematic model of the quadcopter type aerial mobile robot, in the horizontal plane, are introduced. The proposed controllers, at the robot level, are described; these controllers are designed taking into account the configuration and capabilities of each kind of robot. Furthermore, a modified artificial potential method which takes into account the position and velocity of the obstacle, and those of the goal to be reached, is introduced. The pinhole camera model and the computer vision method used to estimate the position and orientation of the ground mobile robots, are described.

The proposed formation control scheme, with synchronization, is described in detail. A brief review about the original “*Cluster Space*” approach is given in order to show the difference and the contribution of this work. Furthermore, the formal stability analysis of the trajectory tracking errors in the formation, for the case of ground mobile robots, is given. Also, an explanation about the implementation of the modified artificial potential method, when used in the formation, is presented.

Next, all the elements that compose the experimental platform used in this work, are briefly described. The experimental results considering static and dynamic obstacles in the formation scheme are shown using three unicycle type ground mobile robots.

The 3D simulator used to carry out realistic simulations and to verify the development of the heterogenous formation is described, considering dynamic obstacles avoidance; also, the position, orientation and velocity of the ground robots are estimated. Finally, some conclusions and future work ideas are given.

Índice general

1	Introducción	1
1.1	Generalidades	1
1.2	Planteamiento del problema	4
1.3	Objetivos	5
1.3.1	Objetivo general	5
1.3.2	Objetivos específicos	5
1.4	Contribuciones de la tesis	6
1.5	Organización de la tesis	6
2	Marco teórico	7
2.1	Modelo cinemático del robot móvil terrestre	7
2.1.1	Restricciones del modelo cinemático y velocidades deseadas	9
2.2	Controladores para los robots terrestres	9
2.2.1	Controlador por seguimiento práctico (usando el punto medio)	10
2.2.2	Controlador P con compensación (usando un punto externo)	11
2.3	Modelo cinemático del robot móvil aéreo	12
2.3.1	Controlador para el robot aéreo	13
2.4	Estrategia de estimación de estados basada en visión artificial	15
2.4.1	Modelo de proyección en perspectiva	15
2.4.2	Estimación de la posición y orientación relativa usando los marcadores arUco	17
2.5	Método de Potenciales Artificiales	20
2.5.1	Potencial atractivo y fuerza atractiva	22
2.5.2	Potencial repulsivo y fuerza repulsiva	23
2.5.3	Problema de mínimo local	26
3	Esquema de sincronización usando CS	29
3.1	Antecedentes	29
3.1.1	Trabajos relacionados	29
3.1.2	Enfoque de Cluster Space	31
3.1.3	Controlador utilizado a nivel de la formación	34
3.2	Esquema de sincronización desde la perspectiva de CS	34
3.2.1	Selección de las variables del clúster	35

3.2.2	Diseño del controlador a nivel de la formación	37
3.2.3	Enfoque de sincronización	38
3.2.4	Diseño del controlador a nivel de los robots	40
3.3	Evasión y navegación para el clúster	48
4	Plataforma experimental	49
4.1	Visión y sensado	49
4.2	Computadora central	50
4.3	Robots móviles TurtleBot3	51
4.4	ROS (Robot Operating System)	52
5	Resultados Experimentales (con robots terrestres)	53
5.1	Trayectoria deseada	53
5.2	Parámetros deseados para la formación	54
5.3	Seguimiento de trayectoria	54
5.4	Evasión de obstáculos y navegación	58
6	Resultados en Gazebo	63
6.1	Simulador Gazebo	63
6.2	Seguimiento de trayectoria (con robots terrestres)	65
6.3	Motivación para usar un robot aéreo con cámara	69
6.4	Clúster compuesto por un cuadricóptero con cámara	70
6.5	Seguimiento de trayectoria (para un clúster heterogéneo)	72
6.5.1	Simulaciones del esquema a altas velocidades	79
6.6	Evasión de obstáculos (con un clúster heterogéneo)	82
7	Conclusiones y trabajo futuro	87
7.1	Conclusiones	87
7.2	Trabajo futuro	88
	Referencias	89
A	Artículos publicados	97
A.1	Artículos de revista indizadas al JCR	97
A.2	Artículo de Congreso Internacional	97
A.3	Artículos de Congreso Nacional	97

Índice de figuras

2.1	Robot móvil terrestre tipo (2,0).	8
2.2	Robot móvil aéreo.	12
2.3	Robot móvil aéreo, simplificado a un robot móvil tipo omnidireccional en el plano horizontal.	14
2.4	Modelo de proyección en perspectiva.	16
2.5	Diferentes tipos de distorsión radial.	17
2.6	Marcos de coordenadas de la cámara y del patrón en blanco y negro.	18
2.7	Algunos IDs de los marcadores arUco del diccionario 4x4_50.	18
2.8	Sistema de coordenadas de cada marcador.	19
2.9	Pasos para detectar los marcadores arUco.	19
2.10	Estimación de la traslación y rotación entre el marco de referencia de la cámara y el marco de referencia del patrón.	21
2.11	Fuerzas de atracción.	23
2.12	Diferentes valores de k_δ con $\epsilon = 1$.	25
2.13	Fuerzas de repulsión.	26
3.1	Clúster compuesto por 3 robots móviles terrestres.	33
3.2	Diagrama de bloques del esquema propuesto. $\mathbf{C}_d = [X_{cd}, Y_{cd}, D_{xd}, D_{yd}, H_{xd}, H_{yd}]^T$, $\mathbf{C} = [X_c, Y_c, D_x, D_y, H_x, H_y]^T$, $\Delta\mathbf{R} = [\Delta x_i, \Delta y_i]^T$, con $i = 1, 2, 3$.	40
4.1	Cámaras Flex 13 y OptiHubs 2.	50
4.2	Entorno del software Motive.	51
4.3	Flujo de datos en la plataforma experimental.	51
4.4	Modelos del TurtleBot3.	52
5.1	Trayectorias en el plano.	55
5.2	Errores en los estados del clúster.	56
5.3	Comparación de las variables angulares.	56
5.4	Errores cartesianos (e_{xi}, e_{yi}) y errores de sincronización $(\Delta x_i, \Delta y_i)$.	57
5.5	Señales de control aplicadas a los robots.	57
5.6	Trayectorias en el plano.	59
5.7	Diferentes variables de interés.	59
5.8	Oscilaciones no deseadas que aparecen mientras se rodea al obstáculo.	60

5.9	Velocidades, fuerzas y distancias.	61
5.10	Señales de control de cada robot móvil.	61
6.1	Robot NAO de <i>Aldebaran Robotics</i>	64
6.2	Trayectorias en el plano (clúster terrestre).	66
6.3	Errores a nivel de la formación (clúster terrestre).	67
6.4	Variables angulares (clúster terrestre).	67
6.5	Errores de seguimiento y de sincronización (clúster terrestre).	68
6.6	Velocidades de control de los 3 robots (clúster terrestre).	68
6.7	Formación triangular incompleta.	70
6.8	Formación triangular con el robot 1 por delante.	71
6.9	Comparación entre las posiciones estimadas usando visión y simuladas obtenidas de Gazebo (formación heterogénea).	74
6.10	Resultado en simulación (usando el visualizador RViz).	75
6.11	Trayectorias en el plano horizontal (formación heterogénea).	76
6.12	Errores a nivel del clúster (formación heterogénea).	77
6.13	Errores a nivel los robots (formación heterogénea).	77
6.14	Comparación entre las orientaciones de los robots tipo diferencial usando visión (formación heterogénea).	78
6.15	Señales de control de 3 robots estimando la posición y orientación (for- mación heterogénea).	78
6.16	Comparación entre usar la postura dada por Gazebo (gráficas a la iz- quierda) y la postura estimada usando visión (gráficas a la derecha), con $T = 80$ [s].	80
6.17	Comparación entre las orientaciones de los robots terrestres obtenidas del simulador y las estimadas por la estrategia de visión, con $T = 80$ [s].	81
6.18	Entorno de simulación (prueba de evasión de 2 obstáculos).	83
6.19	Comparación de los posiciones globales (prueba de evasión de 2 obstáculos).	84
6.20	Orientación de cada robot terrestre (prueba de evasión de 2 obstáculos).	85
6.21	Fuerza total, distancias y velocidades de acercamiento (prueba de evasión de 2 obstáculos).	85
6.22	Señales de control de los 3 robots móviles (prueba de evasión de 2 obstáculos).	86

Capítulo 1

Introducción

1.1 Generalidades

La coordinación de movimiento y el control cooperativo han sido estudiados considerablemente en los últimos años debido a su gran potencial de aplicación en prácticas civiles y/o militares. Múltiples robots tienen mayor ventaja en comparación con un solo individuo, incluyendo robustez al fallo de elementos individuales, re-configurabilidad y realizar tareas más complicadas, imposibles para un solo individuo, como por ejemplo el monitoreo del entorno, exploración, búsqueda y rescate, escoltamiento, localización de un objetivo particular y transporte de objetos pesados. Un sistema de control cooperativo consiste de un grupo de individuos autónomos (agentes) con sensores, cámaras u otros dispositivos de comunicación inalámbrica. Así, un sistema de control cooperativo tiene cuatro elementos básicos: objetivo grupal, individuos, topología de la información y un algoritmo de control que gobierna el funcionamiento de los individuos [1]. Hay muchos ejemplos en la naturaleza donde se juntan múltiples trabajos individuales de manera coordinada para alcanzar los objetivos individuales y el objetivo del grupo. La formación de múltiples individuos ha sido estudiada extensamente en la literatura con aplicaciones en robots móviles terrestres, aéreos, acuáticos y satélites.

El control de formación se refiere al problema de controlar la posición relativa y orientación de los robots móviles en el grupo de acuerdo con una forma preestablecida para realizar una tarea dada [2]. Muchos enfoques han sido propuestos en la literatura para el control de formación de sistemas multi-robots y pueden ser divididos en tres notables grupos principales: estrategias líder-seguidor [3], basados en comportamiento y estructuras virtuales [4], [5]. Adicional a éstos, se han propuesto otros enfoques para el control de formación como potenciales artificiales [6], [7], teoría de grafos, espacio nulo [8], enfoques de sincronización y “*Cluster Space*”, por mencionar algunos. A continuación se da una breve descripción de los enfoques principales.

- **Líder-seguidor:** En este enfoque un vehículo es elegido como líder, mientras que los demás se designan como seguidores. El líder regula su postura para seguir alguna trayectoria deseada y los seguidores persiguen la posición y orientación de

su vecino más cercano considerando sus posiciones relativas preestablecidas. La ventaja de este enfoque es que es fácil de entender e implementar. La desventaja es que no hay retroalimentación de la postura de los seguidores hacia el líder y por lo tanto si el seguidor es perturbado se pierde la formación sin forma de corregirla.

- **Basado en comportamiento:** El comportamiento de cada individuo es descrito de forma preestablecida, el resultado final se deriva de la ponderación del comportamiento de cada individuo. Este enfoque es factible para grupos numerosos de individuos por ser una arquitectura descentralizada. La desventaja es que el modelo matemático que describe la dinámica del grupo es muy complejo ó no se puede obtener de forma explícita, además de la dificultad de garantizar matemáticamente la estabilidad del sistema.
- **Estructuras virtuales:** Todos los individuos tienen una relación geométrica rígida basada en puntos virtuales [9]. Con este enfoque la trayectoria deseada no se asigna de manera individual a cada móvil, en lugar de eso, se asigna al sistema completo como si fuera un único cuerpo y conociendo la dinámica de la estructura virtual se obtiene la ley de control para cada miembro [4]. La principal ventaja de este enfoque es la sencillez de determinar el comportamiento del grupo y que la formación puede ser mantenida bastante bien durante maniobras. La desventaja es que no es posible considerar formaciones variantes en el tiempo y además la prioridad de los individuos entre mantener la formación y seguir su trayectoria no puede ser cambiada [5].
- **Sincronización:** Se puede alcanzar una formación deseada para el grupo de robots a través de la sincronización del seguimiento individual de la trayectoria de cada móvil mientras mantienen una formación variante o invariante en el tiempo [10–12]. Para medir el desempeño de la formación, en [10], los autores presentan el concepto de error de sincronización de posición, el cual está definido como la diferencia de posición entre cada par de robots vecinos. Bajo este enfoque, no solo la convergencia a cero de los errores de posición sino también cómo estos errores convergen a cero es considerada al mismo tiempo.
- **Cluster Space:** Este enfoque fue propuesto por Christopher A. Kitts e Ignacio Mas en 2008 [13]. Esta técnica propone de manera simplificada especificaciones y monitoreo del movimiento de los robots móviles que conforman una entidad simple (clúster), de tamaño limitado, usando un enfoque centralizado y en un entorno local. Este clúster o formación tiene su propia dinámica. Los estados del clúster son función de los estados de todos los robots. Las señales de control se calculan a nivel de la formación. Para obtener las velocidades de control para los robots se emplean las matrices Jacobianas, de manera similar a como se estudian los robots manipuladores. Por lo tanto, se obtienen las señales de control para cada robot a partir de las señales a nivel del clúster [14]. Se asume que cada robot posee un controlador interno que garantiza la regulación de su velocidad.

Sin embargo, es posible obtener una configuración del clúster, tal que la matriz Jacobiana sea singular [15], [16].

En este trabajo, para resolver el problema de control de formación y sincronización de robots móviles se utiliza una versión modificada de dicho enfoque. Para entender mejor el aporte de esta tesis en cuanto al esquema de formación y sincronización, a continuación se describen brevemente los principales trabajos relacionados.

En [10] se presenta un enfoque de sincronización para resolver el problema de regulación de posición usando múltiples robots móviles, esto es, el movimiento de los robots es sincronizado desde una forma específica inicial a otra forma específica final. Con el controlador propuesto se garantiza convergencia asintótica a cero de los errores de posición y los errores de sincronización de cada robot, pero no se pide que la formación siga una trayectoria deseada.

En [5], los autores proponen una estrategia para resolver el problema de control de formación y sincronización de robots móviles usando el enfoque de estructura virtual con acoplamiento mutuo entre los robots. Este acoplamiento mutuo se impone directamente en la ley de control de los robots tipo diferencial debido a su restricción no holónoma. Con estos términos de acoplamiento mutuo es posible sincronizar los robots móviles dando mayor robustez a la formación ante perturbaciones externas. Se muestra que se debe elegir una correcta sintonización de las ganancias de acoplamiento mutuo y las ganancias del controlador para obtener el desempeño deseado en el seguimiento individual de la trayectoria de cada robot y la sincronización de la formación. Se presenta la prueba de estabilidad bajo la suposición de que todas las velocidades angulares de los robots no holónomos son iguales y diferentes de cero.

En [17], los autores presentan un controlador a nivel del clúster usando robots móviles tipo diferencial. Dicho controlador genera las señales de velocidad para cada robot móvil. En la prueba de estabilidad del sistema en lazo cerrado, se muestra que los errores de seguimiento del clúster están acotados. Sin embargo, los errores de seguimiento pueden reducirse incrementando las ganancias del controlador.

En [18], se estudian algunos de los enfoques de formación más comunes en la literatura desde la perspectiva de *Cluster Space*. En particular los enfoques de líder-seguidor, funciones potenciales y estructuras virtuales son abordados con el objetivo de mostrar la flexibilidad del esquema de Cluster Space. Se comparan las ventajas y desventajas de cada uno.

Por otra parte, la habilidad de los robots móviles de navegar libremente evadiendo obstáculos estáticos y dinámicos es necesaria en muchas aplicaciones reales como navegación en interiores [19], exploración [20], transporte automático en entornos dinámicos, entre otros. En esta situación se requiere, en línea, que el robot tome una decisión de hacia donde moverse y con qué velocidad. Muchos enfoques para abordar este problema han sido propuestos como: *Dynamic Window*, por jerarquías [21], *Velocity Obstacle* y sus diferentes versiones [22]- [25], *Collision Cone* y usando el método de potenciales artificiales [26–28].

No obstante, en los enfoques de formación, se asume que cada robot puede obtener de manera precisa su posición global mediante el uso de diferentes sensores. Para abordar este problema, algunos investigadores han utilizado sensores a bordo (e.g., sensores láser, cámaras y sensores tipo sonar, IMU o giroscopio). Comparado con los sensores tradicionales, los diferentes tipos de cámaras: monocular, estéreo, omnidireccionales y dispositivos kinect, pueden proveer bastante información, haciéndolos una opción muy popular en el control de formación usando única y exclusivamente mediciones relativas mediante sensores a bordo [29]. Sin embargo, las desventajas prácticas de incorporar sensores adicionales incluyen: aumento del costo, aumento de complejidad, decremento de fiabilidad e incremento de la carga computacional [30]. Tomando en cuenta las desventajas mencionadas, muchos autores han elegido estrategias servo-visuales usando una cámara monocular que dependen de técnicas analíticas para abordar la pérdida de información.

Las 2 principales estrategias servo-visuales (VS, por sus siglas en inglés) son: basada en posición (Position-based visual servoing, PBVS) y basada en imagen (Image-based visual servoing, IBVS). Los métodos de control usando PBVS [31–34] utilizan información del escenario tridimensional que se reconstruye a partir de la información de la imagen. Esto es, la cámara actúa como un “sensor cartesiano”, donde los algoritmos de estimación de la postura usan los datos de la cámara para generar una señal de error en el espacio cartesiano. Este error es entonces usado por la ley de control como retroalimentación. La principal ventaja de esta estrategia es que se puede controlar la trayectoria del robot, dada en el espacio cartesiano, usando técnicas de control bien conocidas para resolver el problema de seguimiento de trayectoria. Como una desventaja, es que ciertas trayectorias, definidas en el espacio cartesiano, pueden conducir a que el objetivo se localice fuera del campo de visión de la cámara (POV, por sus siglas en inglés) [35]. En los métodos que usan IBVS [36–40], los estados a controlar son imágenes características ó puntos característicos en una imagen de un objetivo, esto significa que los datos de la imagen son usados directamente en la ley de control. La mayor desventaja de estos métodos es que solamente se puede regular la postura de la cámara con respecto a la postura donde la imagen de referencia fue tomada.

1.2 Planteamiento del problema

En la literatura se presentan varias estrategias de control para resolver el problema de seguimiento de trayectoria, navegación evadiendo obstáculos dinámicos y convergencia a una meta, así como diferentes enfoques para resolver el problema de control de formación y sincronización de robots móviles. A continuación se presentan los aspectos considerados y los problemas a resolver en este trabajo:

1. Esquemas de formación y sincronización de robots móviles garantizando la convergencia asintótica de los errores del sistema.

2. Problema de navegación libre de colisiones utilizando un método de evasión de obstáculos estáticos y dinámicos.
3. Leyes de control para resolver el problema de seguimiento de trayectoria de cada robot móvil que componen la formación, utilizando el modelo cinemático.
4. Estrategia de estimación de la posición y orientación de los robots móviles que componen la formación utilizando una cámara a bordo mediante algoritmos de visión computacional.
5. Estrategia de estimación de la posición, orientación y velocidad de los obstáculos móviles utilizando la cámara a bordo de un robot móvil aéreo, empleando visión artificial.

1.3 Objetivos

1.3.1 Objetivo general

El objetivo general de este trabajo de tesis es proponer, implementar, evaluar y validar, un enfoque de formación y sincronización de robots móviles, con el fin de resolver el problema de navegación y convergencia a la meta, considerando obstáculos estáticos y dinámicos. También, solamente con sensores a bordo de los robots móviles (terrestres o aéreos), proponer, implementar, evaluar y validar una estrategia de estimación de las posiciones relativas y la orientación relativa entre los robots móviles mediante el uso de algoritmos de visión computacional.

1.3.2 Objetivos específicos

1. Diseñar, implementar y validar leyes de control para el seguimiento de la trayectoria de cada robot móvil utilizando el modelo cinemático.
2. Proponer, implementar y validar un esquema de formación y sincronización de robots móviles a partir de los trabajos en la literatura.
3. Demostrar formalmente la estabilidad en lazo cerrado del esquema y de las leyes de control propuestas.
4. Implementar y validar un método de evasión considerando obstáculos tanto estáticos como dinámicos.
5. Desarrollar una plataforma experimental para probar las leyes de control y el esquema de formación propuesto.
6. Implementar una estrategia usando visión computacional para estimar la posición y orientación de los robots móviles que componen la formación mediante el uso de una cámara monocular a bordo de un robot aéreo.

1.4 Contribuciones de la tesis

Las principales contribuciones de este trabajo se presentan a continuación:

1. Se propone un enfoque de sincronización basado en el esquema de “Cluster Space” para resolver el problema de control de formación.
2. Se demuestra formalmente la convergencia de los errores de seguimiento de cada robot móvil que compone la formación.
3. Se evalúa, numérica y experimentalmente, el esquema de sincronización y las leyes de control propuestas.
4. Se implementa una estrategia de visión artificial para estimar la posición y orientación de los robots móviles mediante el uso de una cámara monocular a bordo del robot aéreo, sin requerir de algún otro tipo de sensor a bordo como IMU, giroscopio, LIDAR (radar) o dispositivo kinect. Asumiendo que sólo éste cuenta con un sistema de posicionamiento global (GPS).

1.5 Organización de la tesis

Este documento se organiza como sigue, en el Capítulo 2 se muestran las bases teóricas para el desarrollo del presente trabajo. En el Capítulo 3 se describen detalladamente los aspectos y las características del esquema de formación y sincronización con evasión de obstáculos propuesto. Además, se presenta el análisis de la estabilidad del sistema en lazo cerrado. En el Capítulo 4 se describen las características de los componentes que integran la plataforma experimental utilizada. En el Capítulo 5 se presentan las pruebas realizadas en el laboratorio utilizando única y exclusivamente una formación compuesta por 3 robots móviles terrestres tipo unicycle sin cámara a bordo ni visión artificial. En el Capítulo 6 se describe el simulador utilizado para validar la estrategia de estimación de posición utilizando una cámara a bordo del robot móvil tipo cuádrirrotor. También, se presentan los resultados de simulación para la formación heterogénea usando visión artificial. Finalmente, en el Capítulo 7 se presentan las conclusiones y el trabajo futuro.

Capítulo 2

Marco teórico

En este capítulo se presentan el modelo cinemático del robot móvil tipo unicycle, su restricción no holónoma y el modelo cinemático para un robot móvil tipo omnidireccional en el plano horizontal, el cual es utilizado para modelar el robot aéreo tipo cuadricóptero. También se describen los controladores utilizados a nivel de los robots. Asimismo, se presenta el método de potenciales artificiales el cual toma en cuenta la posición y velocidad del obstáculo y la meta. Para resolver el problema de mínimo local que se presenta con este método, se propone una nueva fuerza con ganancia variable perpendicular a la fuerza repulsiva. Finalmente, se describe el modelo de proyección en perspectiva y el método de visión artificial usado para estimar la posición y orientación de los robots mediante una cámara monocular.

2.1 Modelo cinemático del robot móvil terrestre

El tipo de robot móvil terrestre utilizado en este trabajo es un tipo (2,0), tipo diferencial o unicycle, por tal motivo a continuación se describe el modelo cinemático para este tipo de robot. El modelo cinemático de postura en tiempo continuo del i -ésimo robot tipo (2,0), en un grupo de n robots, está definido como

$$\dot{x}_i(t) = V_i(t) \cos q_i(t), \quad (2.1.1)$$

$$\dot{y}_i(t) = V_i(t) \sin q_i(t), \quad (2.1.2)$$

$$\dot{q}_i(t) = W_i(t), \quad (2.1.3)$$

con $i = 1, 2, \dots, n$, donde $P_i = (x_i(t), y_i(t))$ es el punto medio del eje de las ruedas del i -ésimo robot que compone la formación, con respecto del marco de referencia global $\{G\} = [X_G, Y_G, Z_G]^T$; y $q_i(t)$ es el ángulo de orientación con respecto al eje X_G , véase la Fig. 2.1. $V_i(t)$, $W_i(t)$ son la velocidad lineal y la velocidad rotacional del robot, respectivamente, las cuales son las entradas de control. De esta manera, las velocidades lineales \dot{x}_i y \dot{y}_i son la proyección de la velocidad traslacional a lo largo de los ejes X_G y Y_G , respectivamente. Por otro lado, la velocidad angular del robot es igual a la razón

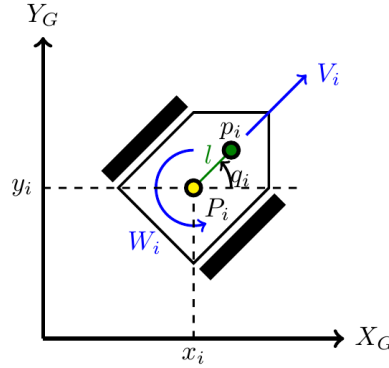


Figura 2.1: Robot móvil terrestre tipo (2,0).

de cambio del ángulo q_i con respecto al tiempo. Por medio de las señales V_i y W_i se puede representar la totalidad de movimientos posibles a efectuar por el robot móvil en un espacio de trabajo compuesto por un plano coordenado formado por los ejes X_G y Y_G .

Desde un punto de vista práctico, los cambios de dirección y la tracción del robot se obtienen mediante la diferencia existente entre las velocidades angulares de sus ruedas motrices. Para obtener dicha relación se tienen que tomar las siguientes consideraciones. En primera instancia, se asume que las ruedas del móvil son lo suficientemente rígidas para no sufrir ninguna deformación cuando el vehículo se desplaza. Además, se asume que no existe ningún tipo de deslizamiento ocasionado por el movimiento de las ruedas del robot. Finalmente, dado que se emplea el modelo cinemático de postura para deducir la relación entre las velocidades rotacional y traslacional y la velocidad angular de cada rueda, se considera que el espacio de trabajo del robot es perfectamente plano y que existe únicamente un punto de contacto entre cada rueda y el plano de trabajo. Entonces, las velocidades traslacional y rotacional expresadas en función de las velocidades angulares de cada rueda están dadas por

$$\begin{bmatrix} w_{r_i}(t) \\ w_{l_i}(t) \end{bmatrix} = \frac{1}{R_w} \begin{bmatrix} 1 & b \\ 1 & -b \end{bmatrix} \begin{bmatrix} V_i(t) \\ W_i(t) \end{bmatrix}, \quad (2.1.4)$$

donde R_w es el radio de cada rueda, $2b$ es la distancia entre las ruedas. $w_{r_i}(t)$ y $w_{l_i}(t)$ son las velocidades angulares de la rueda derecha e izquierda, respectivamente. Es importante notar que la matriz de transformación es no singular cuando $R_w > 0$ y $b > 0$, por lo tanto existe una relación uno a uno y sobre entre $[V_i(t), W_i(t)]^T$ y $[w_{r_i}(t), w_{l_i}(t)]^T$, pudiéndose alcanzar cualquier valor de velocidad lineal y angular, eligiendo adecuadamente las velocidades en las ruedas.

2.1.1 Restricciones del modelo cinemático y velocidades deseadas

La utilización de modelos cinemáticos implica algunas limitaciones en su aplicación práctica debido a que sólo considera las dinámicas de las velocidades asociadas al sistema mecánico. En este trabajo se considera el modelo cinemático del vehículo, bajo la hipótesis de que opera a bajas velocidades. El robot móvil utilizado es no holónimo, ya que el modelo cinemático que lo describe tiene una restricción no holónoma¹, la cual se expresa implícitamente en (2.1.1) y (2.1.2).

Por lo tanto, es posible llevar al robot considerado, el tipo (2,0) desde una posición inicial hacia cualquier otra posición final moviendo sus ruedas sin deslizamiento, sin embargo, el camino recorrido entre ambas posiciones no puede ser cualquiera, debido a que el vehículo no puede desplazarse hacia cualquier dirección en un instante de tiempo cualquiera. Así como el sistema de referencia deseado debe cumplir también con una dinámica de la forma

$$\dot{x}_{id} = V_{id} \cos(q_{id}), \quad \dot{y}_{id} = V_{id} \sin(q_{id}), \quad \dot{q}_{id} = W_{id}, \quad (2.1.5)$$

donde el subíndice d indica los valores deseados para cada variable. De las ecuaciones anteriores se tiene que la restricción no holónoma está dada por

$$\dot{x}_{id} \sin(q_{id}) - \dot{y}_{id} \cos(q_{id}) = 0. \quad (2.1.6)$$

Por lo tanto, la orientación deseada para el robot móvil tipo unicycle debe satisfacer

$$q_{id} = \tan^{-1} \left(\frac{\dot{y}_{id}}{\dot{x}_{id}} \right), \quad (2.1.7)$$

y las velocidades deseadas en función de los estados deseados tienen la forma

$$V_{id} = \sqrt{(\dot{x}_{id})^2 + (\dot{y}_{id})^2}, \quad W_{id} = \frac{\dot{x}_{id} \ddot{y}_{id} - \ddot{x}_{id} \dot{y}_{id}}{(\dot{x}_{id})^2 + (\dot{y}_{id})^2}, \quad (2.1.8)$$

con $(\dot{x}_{id})^2 + (\dot{y}_{id})^2 \neq 0 \quad \forall t \geq 0$.

2.2 Controladores para los robots terrestres

En esta sección se describen los controladores implementados en cada uno de los robots móviles que componen la formación para resolver el problema de seguimiento de trayectoria y sincronización. Éstos son llamados controladores internos debido a que se tienen controladores configurados en cascada. En el capítulo siguiente se describe el controlador principal o externo que se realiza a nivel de la formación.

Generalmente, para abordar el problema de seguimiento de trayectoria para un robot móvil tipo unicycle considerando el modelo cinemático se toma como salida:

¹Una restricción definida por la ecuación $J(q)\dot{q} = 0$, donde $q(t)$ es continua y diferenciable y $J(q)$ es continua y diferenciable no es integrable cuando no existe una función $F(q)$ tal que $J(q) = \frac{\partial F(q)}{\partial q}$, [41].

1. el punto medio del eje de las ruedas o
2. un punto externo del eje de las ruedas.

Observación 2.1 *Los controladores presentados en este trabajo han sido ampliamente utilizados en la literatura, [42-44], y en cada uno se toma una de las salidas mencionadas anteriormente. También, otro objetivo de utilizar 2 diferentes controladores es para demostrar que con este esquema de formación es posible reemplazar el controlador interno por algún otro diferente haciendo algunas adaptaciones.*

2.2.1 Controlador por seguimiento práctico (usando el punto medio)

Considere el modelo cinemático para el robot tipo (2,0) descrito en (2.1.1)-(2.1.3), con la salida el punto medio del eje de las ruedas P_i , esto es $h_{xi} = x_i$, $h_{yi} = y_i$. El subíndice i hace referencia al i -ésimo robot en la formación. V_i, W_i son las entradas de control, velocidad lineal y velocidad angular respectivamente. Este controlador logra resolver el problema de seguimiento de trayectoria para robots móviles tipo (2,0) haciendo que el ángulo de orientación también converja a su valor deseado, siempre que la trayectoria deseada cumpla con la restricción no holónoma descrita en (2.1.6). Este controlador está basado en [42] y [43].

Derivando una vez cada salida y la orientación angular, e igualando a una señal virtual se tiene

$$\dot{h}_{xi} = \dot{x}_i = \nu_{xi}, \quad \dot{h}_{yi} = \dot{y}_i = \nu_{yi}, \quad W_i = \nu_{qi}. \quad (2.2.1)$$

A partir del modelo cinemático del robot tipo unicycle, multiplicando los términos \dot{x}_i, \dot{y}_i por $\cos(q_i), \sin(q_i)$, respectivamente, y sumándolos se obtiene la velocidad lineal dada de la forma

$$V_i = \nu_{xi} \cos(q_i) + \nu_{yi} \sin(q_i), \quad (2.2.2)$$

donde las señales virtuales se elijen como

$$\nu_{xi} = \dot{x}_{id} - k_{xi}(x_i - x_{id}), \quad (2.2.3)$$

$$\nu_{yi} = \dot{y}_{id} - k_{yi}(y_i - y_{id}), \quad (2.2.4)$$

$$\nu_{qi} = \dot{q}_{id} - k_{qi}(q_i - q_{id}), \quad (2.2.5)$$

donde el subíndice d indica el valor deseado para cada estado. k_{xi}, k_{yi}, k_{qi} son ganancias positivas de diseño.

Los errores de seguimiento cartesianos y de orientación de cada robot móvil, en el marco de referencia global, están dados por

$$e_{xi} = x_i - x_{id}, \quad (2.2.6)$$

$$e_{yi} = y_i - y_{id}, \quad (2.2.7)$$

$$e_{qi} = q_i - q_{id}. \quad (2.2.8)$$

De (2.1.3) y (2.2.5) se tiene en lazo cerrado

$$\dot{q}_i = \dot{q}_{id} - k_{qi}e_{qi} \Rightarrow \dot{e}_{qi} + k_{qi}e_{qi} = 0. \quad (2.2.9)$$

Por lo tanto, la señal q_i , con $i = 1, \dots, n$, converge exponencialmente a su valor deseado. La convergencia de los errores e_{xi} y e_{yi} se muestra posteriormente.

2.2.2 Controlador P con compensación (usando un punto externo)

Considere nuevamente el modelo cinemático para el robot tipo (2,0) descrito en (2.1.1)-(2.1.3), pero en este caso, tome en cuenta la salida del sistema un punto por fuera del eje de las ruedas recorrido una distancia $l > 0$, $p_i = (x'_i, y'_i)$, véase la Fig. 2.1. El punto externo está dado por

$$p_i = (x_i + l \cos q_i, y_i + l \sin q_i). \quad (2.2.10)$$

Este controlador logra resolver el problema de seguimiento de trayectoria para robots móviles tipo (2,0) tomando en cuenta que el ángulo de orientación es otro estado el cual se mantiene acotado, [44].

Derivando una vez cada salida ($h_{xi} = x'_i$, $h_{yi} = y'_i$), e igualando a una señal auxiliar se tiene

$$\begin{bmatrix} \dot{h}_{xi} \\ \dot{h}_{yi} \end{bmatrix} = \begin{bmatrix} \nu_{xi} \\ \nu_{yi} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos(q_i) & -l \sin(q_i) \\ \sin(q_i) & l \cos(q_i) \end{bmatrix}}_{\mathbf{A}_i(q_i)} \begin{bmatrix} V_i \\ W_i \end{bmatrix}, \quad (2.2.11)$$

donde $\mathbf{A}_i(q_i)$ es la matriz de desacoplamiento de cada robot. Dado que $\det(\mathbf{A}_i) = l$, la matriz de desacoplamiento es no singular $\forall l \neq 0$. Por tanto, la ley de control está dada por

$$[V_i, W_i]^T = \mathbf{A}_i^{-1}(q_i)[\nu_{xi}, \nu_{yi}]^T, \quad (2.2.12)$$

donde las señales auxiliares se elijen como

$$\nu_{xi} = \dot{x}'_{id} - \kappa_{xi}e'_{xi}, \quad (2.2.13)$$

$$\nu_{yi} = \dot{y}'_{id} - \kappa_{yi}e'_{yi}, \quad (2.2.14)$$

con $i = 1, \dots, n$. $e'_{xi} = x'_i - x'_{id}$, $e'_{yi} = y'_i - y'_{id}$ son los errores de seguimiento cartesiano considerando el punto p_i . x'_{id} , y'_{id} , \dot{x}'_{id} , \dot{y}'_{id} son las posiciones y velocidades deseadas, respectivamente. κ_{xi} , κ_{yi} son ganancias positivas.

2.3 Modelo cinemático del robot móvil aéreo

El robot móvil aéreo utilizado en este trabajo es un cuadrirrotor. En la literatura se pueden encontrar múltiples formas de nombrar a este vehículo aéreo no tripulado, estas son: UAV (unmanned aerial vehicle), cuatrirrotor. A partir de este momento se empleará cualquiera de estos nombres indistintamente.

Un cuadrirrotor es un vehículo aéreo tipo helicóptero propulsado por cuatro rotores. Cada rotor se compone de un motor eléctrico tipo brushless y una hélice. Cuando el rotor está girando, se genera una fuerza de empuje perpendicular al plano de rotación de la hélice. Dicha fuerza de empuje es proporcional al cuadrado de la velocidad angular de la hélice.

Para obtener la posición y orientación del cuadrirrotor en el espacio tridimensional, considérese el marco de coordenadas inercial $\{G\} = [X_G, Y_G, Z_G]^T$ y un marco de coordenadas adjunto al cuadrirrotor $\{B\} = [X_B, Y_B, Z_B]^T$, como se muestra en la Fig. 2.2. La orientación en el espacio tridimensional de un cuadrirrotor se representa por medio de los tres ángulos de Euler: alabeo (ϕ), cabeceo (θ), guiñada (ψ), con respecto al marco B . El ángulo de alabeo (ϕ) representa la rotación del cuadrirrotor respecto al eje X_B , el ángulo de cabeceo (θ) representa la rotación respecto al eje Y_B y el ángulo de guiñada (ψ) representa la rotación respecto al eje Z_B . Debido a que solamente se utiliza un cuadrirrotor en la formación, la posición del centro de gravedad del cuadrirrotor con respecto al marco $\{G\}$, está dado por $P = (x, y, z)$.

Para lograr el vuelo estacionario (hover) del cuadrirrotor es necesario que la suma de las fuerzas de empuje de los 4 rotores sea igual al peso del vehículo. Por otro lado, un rotor al girar genera un momento reactivo debido a las fuerzas aplicadas sobre la hélice, este momento reactivo es de sentido contrario al giro de la hélice, debido a este fenómeno, dos hélices giran en sentido horario y las otras dos giran en sentido antihorario, con el fin de contrarrestar los momentos reactivos que el cuerpo experimenta al

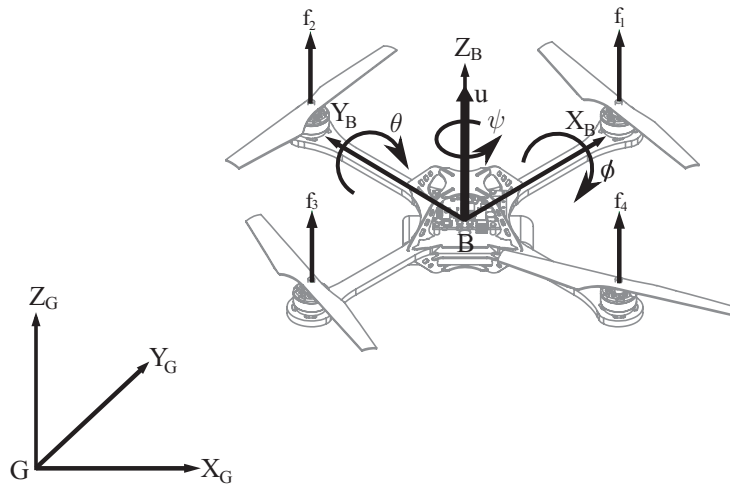


Figura 2.2: Robot móvil aéreo.

hacer girar los rotores; así cada par de motores opuestos giran en el mismo sentido. Al variar las velocidades angulares de sus rotores se pueden modificar los ángulos de Euler, lo cual produce movimiento traslacional en el cuadirrotor. Existen 2 tipos de configuraciones de los rotores en los cuadirrotores, configuración en “+” y “×”. La configuración utilizada en la figura 2.2, es una en “+”. La diferencia entre estas configuraciones radica en que la forma “+” sólo utiliza un motor para generar el par de entrada, mientras que la forma “×” emplea 2 motores simultáneamente.

Observación 2.2 *Debido a que el esquema de formación y sincronización propuesto en este trabajo está basado en el esquema de “Cluster Space”, el cual toma en cuenta los modelos cinemáticos de los robots móviles que componen la formación, se toma en cuenta únicamente el modelo cinemático del cuadirrotor. No obstante, es importante mencionar que existen muchos cuadirrotores comerciales o de uso para investigación que incorporan diferentes controladores internos, comúnmente PID’s, tal que se garantiza que la aeronave pueda ser controlada mediante diferentes entradas de control.*

Para obtener el modelo cinemático del cuadirrotor se hacen las siguientes suposiciones:

- **Suposición 2.1** *El cuadirrotor posee controladores internos tal que, para provocar un movimiento de traslación, se garantiza que las entradas del sistema son velocidades traslacionales, en los ejes X_B , Y_B y Z_B .*
- **Suposición 2.2** *El cuadirrotor posee un controlador interno tal que, para provocar un movimiento de rotación sobre su eje Z_B , se garantiza que la entrada de control es una velocidad angular.*

Bajo las suposiciones anteriores, el modelo cinemático del cuadirrotor está dado por

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \underbrace{\begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix}}_{\mathbf{R}_z(\psi)} \begin{bmatrix} v_x \\ v_y \end{bmatrix}, \quad (2.3.1)$$

$$\dot{z} = v_z, \quad (2.3.2)$$

$$\dot{\psi} = w_z, \quad (2.3.3)$$

donde las velocidades v_x , v_y , v_z y w_z , con respecto del marco de coordenadas $\{B\}$, son las entradas de control. Debido a que solamente se pretende mantener el robot aéreo volando a una cierta altura fija, las ecuaciones (2.3.1)-(2.3.3) corresponden al modelo cinemático de un robot móvil terrestre tipo (3,0) u omnidireccional, (Fig. 2.3).

2.3.1 Controlador para el robot aéreo

Considere el modelo cinemático del cuadirrotor descrito por (2.3.1)-(2.3.3), con el vector de salida del sistema $\mathbf{h} = [x, y, z, \psi]^T$. Debido a que se estudia un sistema lineal,

el controlador P con compensación implementado en el robot móvil aéreo logra resolver el problema de seguimiento de trayectoria haciendo que todos los estados (x, y, z, ψ) converjan exponencialmente a su valor deseado (x_d, y_d, z_d, ψ_d) , para el caso sin perturbaciones.

Derivando una vez cada salida e igualando a una señal virtual se tiene

$$\dot{h}_x = \dot{x} = U_x, \quad \dot{h}_y = \dot{y} = U_y, \quad \dot{h}_z = \dot{z} = U_z, \quad \dot{h}_\psi = U_\psi. \quad (2.3.4)$$

La ley de control está dada por

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \mathbf{R}_z^{-1}(\psi) \begin{bmatrix} U_x \\ U_y \end{bmatrix}, \quad \begin{bmatrix} v_z \\ w_z \end{bmatrix} = \begin{bmatrix} U_z \\ U_w \end{bmatrix}, \quad (2.3.5)$$

con

$$\begin{bmatrix} U_x \\ U_y \\ U_z \\ U_w \end{bmatrix} = \begin{bmatrix} \dot{x}_d - \kappa_x e_x \\ \dot{y}_d - \kappa_y e_y \\ \dot{z}_d - \kappa_z e_z \\ \dot{\psi}_d - \kappa_\psi e_\psi \end{bmatrix}, \quad (2.3.6)$$

con $\kappa_x, \kappa_y, \kappa_z, \kappa_\psi > 0$, donde los errores de seguimiento, en el marco de referencia global, están dados por

$$e_x = x - x_d, \quad e_y = y - y_d, \quad e_z = z - z_d, \quad e_\psi = \psi - \psi_d. \quad (2.3.7)$$

Sustituyendo la ley de control (2.3.5)-(2.3.6) en el modelo cinemático (2.3.1)-(2.3.3), se obtiene

$$\dot{\lambda} = \dot{\lambda}_d - \kappa_\lambda e_\lambda \Rightarrow \dot{e}_\lambda + \kappa_\lambda e_\lambda = 0, \quad \text{con } \lambda = x, y, z, \psi. \quad (2.3.8)$$

Por lo tanto, los estados (x, y, z, ψ) convergen exponencialmente a su valor deseado.

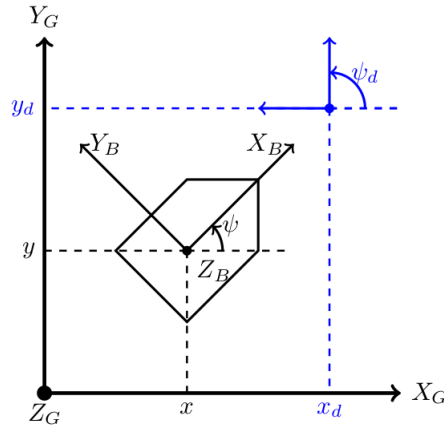


Figura 2.3: Robot móvil aéreo, simplificado a un robot móvil tipo omnidireccional en el plano horizontal.

2.4 Estrategia de estimación de estados basada en visión artificial

2.4.1 Modelo de proyección en perspectiva

Para explicar el modelo de proyección en perspectiva, considere un punto $P_m \in \mathbb{R}^3$. El vector de posición de ese punto, expresado con respecto al marco de referencia global $\{G\} = [X_G, Y_G, Z_G]^T$, está dado por ${}^G\mathbf{P}_m = [X_g, Y_g, Z_g]^T$. El vector de posición de ese mismo punto, expresado con respecto al marco de referencia de una cámara monocular $\{C\} = [X_C, Y_C, Z_C]^T$, está dado por ${}^C\mathbf{P}_m = [X_c, Y_c, Z_c]^T$. La transformación que relaciona el cambio de coordenadas está dada como [45], [46]

$${}^C\mathbf{P}_m = {}^C_G \mathcal{R} {}^G\mathbf{P}_m + {}^C_G \mathbf{t}, \quad (2.4.1)$$

donde ${}^C_G \mathcal{R} \in \mathbb{R}^{3 \times 3}$ es la matriz de rotación que alinea el marco $\{G\}$ con $\{C\}$, ${}^C_G \mathbf{t}$ es el vector de traslación entre los marcos de referencia, con respecto al marco $\{C\}$.

La ecuación (2.4.1) se puede reescribir como $\hat{\mathbf{P}}_c = \hat{\mathbf{H}} \hat{\mathbf{P}}_g$, donde $\hat{\mathbf{P}}_c = [X_c, Y_c, Z_c, 1]^T$ y $\hat{\mathbf{P}}_g = [X_g, Y_g, Z_g, 1]^T$ son los vectores de coordenadas homogéneas. $\hat{\mathbf{H}} \in \mathbb{R}^{4 \times 4}$ es la matriz de transformación homogénea dada por

$$\hat{\mathbf{H}} = \begin{bmatrix} {}^C_G \mathcal{R} & {}^C_G \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \quad (2.4.2)$$

La transformación que mapea del vector $\hat{\mathbf{P}}_g$ al vector ${}^C\mathbf{P}_m$ está dada por

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} {}^C_G \mathcal{R} & {}^C_G \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \hat{\mathbf{P}}_g = [\mathcal{R}|\mathbf{t}] \begin{bmatrix} X_g \\ Y_g \\ Z_g \\ 1 \end{bmatrix}, \quad (2.4.3)$$

donde a la matriz $[\mathcal{R}|\mathbf{t}] \in \mathbb{R}^{3 \times 4}$ se le llama matriz de los parámetros extrínsecos de la cámara, la cual contiene la rotación y la traslación entre los marcos de referencia de la cámara y del marco global.

Usando el modelo de proyección en perspectiva de una cámara monocular, el punto ${}^C\mathbf{P}_m$ se proyecta en un punto en el plano de la imagen $p_m = (u, v)$, a través del centro óptico de la cámara O_c , véase la Fig. 2.4.

La transformación que describe el modelo de proyección en perspectiva de una cámara libre de distorsión está dada por

$$\hat{\mathbf{P}}_m = \mathcal{K} {}^C\mathbf{P}_m = \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_u f & \gamma & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}, \quad (2.4.4)$$

donde \mathcal{K} es la matriz de calibración, la cual está compuesta por los parámetros intrínsecos de la cámara $(f, k_u, k_v, u_0, v_0, \gamma)$. f es la distancia focal de la cámara (en metros),

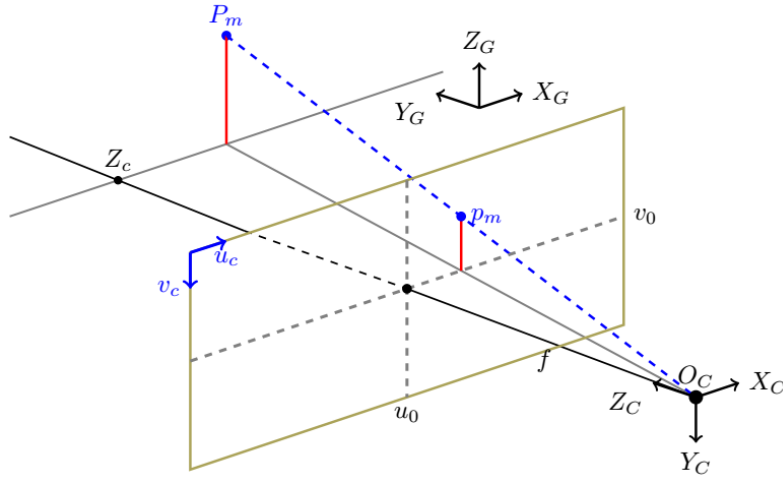


Figura 2.4: Modelo de proyección en perspectiva.

k_u y k_v son el número de píxeles por unidad de distancia en coordenadas de la imagen (en pixel/m), el punto (u_0, v_0) es la coordenada en el plano de la imagen del punto principal, es decir el punto de intersección del plano de la imagen y el eje óptico Z_C (en píxeles), γ es el factor de torcedura que en la mayoría de los casos se considera cero.

Es importante mencionar que, a partir del modelo de proyección en perspectiva de la cámara y utilizando los parámetros intrínsecos de la misma, se pueden obtener las coordenadas del punto p_m a partir del punto ${}^C\mathbf{P}_m$ dadas por

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} k_u f X_c / Z_c + u_0 \\ k_v f Y_c / Z_c + v_0 \end{bmatrix} \quad (2.4.5)$$

Note de la ecuación anterior, la profundidad Z_c , del punto ${}^C\mathbf{P}$, está implícita en las coordenadas de la imagen (u, v) .

Finalmente, el modelo completo de proyección en perspectiva que describe la relación entre un punto 3D, en coordenadas homogéneas $\hat{\mathbf{P}}_g$, expresado con respecto al marco global, y su proyección en el plano de la imagen, en coordenadas homogéneas $\hat{\mathbf{P}}_m$, está dada por $\hat{\mathbf{P}}_m = \mathcal{K}[\mathcal{R}|\mathbf{t}]\hat{\mathbf{P}}_g$, más precisamente

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} k_u f & \gamma & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} {}^C_G \mathcal{R} & {}^C_G \mathbf{t} \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X_g \\ Y_g \\ Z_g \\ 1 \end{bmatrix} \quad (2.4.6)$$

La calibración de la cámara es el proceso de obtener los parámetros fundamentales de una cámara. En una cámara ideal, el punto ${}^C\mathbf{P}_m$ se proyecta en el punto en el plano de la imagen $p_m = (u, v)$. Sin embargo, las lentes de las cámaras normalmente distorsionan la escena haciendo que los puntos alejados del centro estén aún más lejos. Por lo tanto, las franjas verticales cerca de los bordes de la imagen aparecen ligeramente dobladas.

En consecuencia, si se desea conocer la proyección de un píxel, se deben considerar los componentes de distorsión. Hay dos tipos de distorsiones (radial y tangencial) y éstas están representadas por los parámetros k_1, k_2, k_3, p_1, p_2 .

Los parámetros de distorsión son los coeficientes radiales k_1, k_2 y k_3 ; p_1 y p_2 son los coeficientes de distorsión tangencial. La Figura 2.5 muestra dos tipos comunes de distorsión radial: distorsión de barril (que disminuye monótonamente) y distorsión de acerico (que aumenta monótonamente). La distorsión radial es siempre monótona para lentes reales, y si el estimador produce un resultado no monótono, esto debe considerarse un error de calibración.

La distorsión radial ocurre en el proceso de convertir el sistema de coordenadas de la cámara al sistema de coordenadas físico de la imagen. La distorsión tangencial se produce en el proceso de fabricación de la cámara porque el plano del elemento fotosensible no es paralelo al objetivo, la causa es que la lente no está completamente paralela al plano de la imagen.

2.4.2 Estimación de la posición y orientación relativa usando los marcadores arUco

El módulo ArUco se basa en la biblioteca ArUco, una biblioteca popular para la detección de marcadores cuadrados desarrollada por Rafael Muñoz y Sergio Garrido [47]. La biblioteca está escrita en C++ y requiere OpenCV. ArUco es una biblioteca de código abierto para detectar marcadores cuadrados en imágenes. Además, si la cámara está calibrada, se puede estimar la pose de la cámara con respecto a los marcadores, véase la Fig. 2.6.

Hay varios tipos de marcadores, cada uno de ellos perteneciente a un diccionario. Cada biblioteca ha propuesto su propio conjunto de marcadores. Entonces, se tiene ArToolKit+, Chilitags, AprilTags y, por supuesto, el diccionario ArUco. El diseño de un diccionario es importante ya que la idea es que sus marcadores sean lo más diferentes posible para evitar confusiones.

ArUco usa la clase *Marker*, que representa un marcador observado en la imagen.



Figura 2.5: Diferentes tipos de distorsión radial.

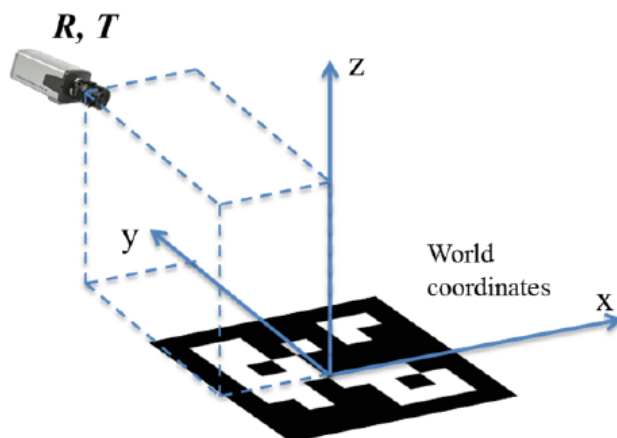


Figura 2.6: Marcos de coordenadas de la cámara y del patrón en blanco y negro.

Cada marcador es un vector de 4 puntos (que representan las esquinas de la imagen), una identificación única, su tamaño (en metros) y la traslación y rotación que relaciona el centro del marcador y la ubicación de la cámara. Está preparada para detectar marcadores de cualquiera de los diccionarios permitidos. De forma predeterminada, *MarkerDetector* buscará cuadrados y luego analizará el código binario que contiene.

Los marcadores están compuestos por un borde negro externo y una región interna que codifica un patrón binario. El patrón binario es único e identifica cada marcador. Dependiendo del diccionario, hay marcadores con más o menos bits. Por ejemplo, un tamaño de marcador de 4x4 está compuesto por 16 bits. La Fig. 2.7 muestra los primeros números de los marcadores arUco del diccionario 4x4. Cuantos más bits, más palabras hay en el diccionario y menor es la posibilidad de confusión. Sin embargo, más bits significa que se requiere más resolución para una detección correcta. Los marcadores se



Figura 2.7: Algunos IDs de los marcadores arUco del diccionario 4x4_50.

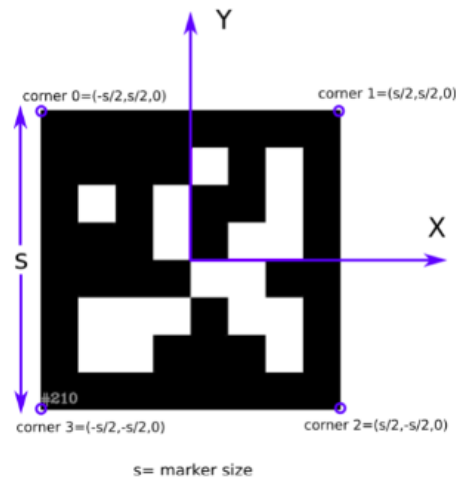


Figura 2.8: Sistema de coordenadas de cada marcador.

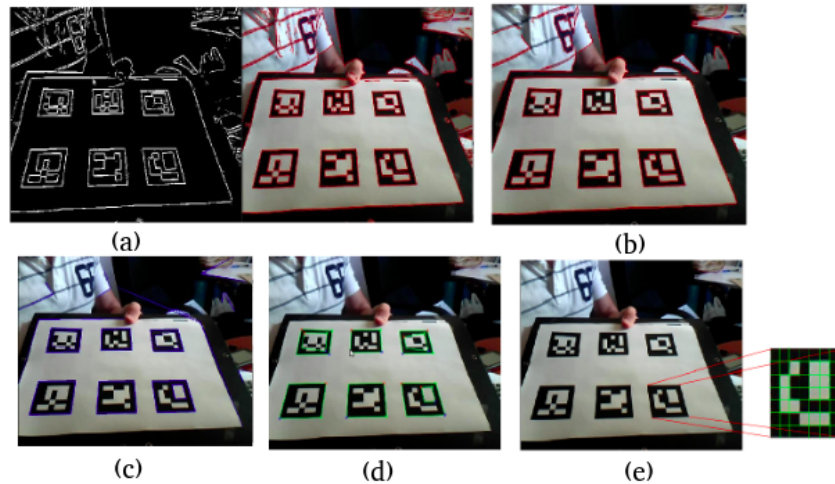


Figura 2.9: Pasos para detectar los marcadores arUco.

pueden utilizar como puntos de referencia en 3D para la estimación de la pose de la cámara. Se denota el tamaño del marcador (s) una vez impreso en una hoja de papel. La Fig. 2.8 muestra el sistema de coordenadas adscrito a cada marcador.

Para detectar los marcadores, una vez capturada la imagen, se siguen los siguientes pasos:

1. Se aplica un umbral adaptativo a la imagen en escala de grises, para obtener bordes, Fig. 2.9-(a).
2. Se encuentran los contornos. Después de eso, no solo se detectan los marcadores

reales, sino también una gran cantidad de bordes no deseados. El resto del proceso tiene como objetivo filtrar las fronteras no deseadas.

3. Se eliminan los bordes con una pequeña cantidad de puntos, Fig. 2.9-(b).
4. Se aproxima a un polígono cada contorno encontrado y se mantienen los contornos cóncavos con exactamente 4 esquinas (es decir, rectángulos), Fig. 2.9-(c).
5. Se ordenan las esquinas en sentido antihorario.
6. Se eliminan los rectángulos demasiado cercanos. Esto es necesario porque el umbral adaptativo normalmente detecta la parte interna y externa del borde del marcador. En esta etapa, se mantiene la frontera más externa, Fig. 2.9-(d).
7. Se identifican los marcadores:
 - Se retira la perspectiva de proyección para obtener una vista frontal del área del rectángulo mediante una homografía, Fig. 2.9-(e).
 - Se aplican los algoritmos de Otsu. Estos algoritmos asumen una distribución bimodal y encuentran el umbral que maximiza la varianza extra clase manteniendo una baja varianza intra clase.
 - Se identifica el código interno. Si es un marcador, entonces tiene un código interno. El marcador está dividido en una cuadrícula de $n \times n$, de las cuales las celdas internas de $(n-2) \times (n-2)$ contienen información de identificación. El resto corresponde al borde negro externo. En este paso, primero se verifica que esté presente el borde negro externo. Luego, se leen las celdas internas y se verifica si proporcionan un código válido (podría ser necesario rotar el código para obtener el válido).
 - Para los marcadores válidos, se redefinen las esquinas mediante la interpolación de subpíxeles.
8. Finalmente, utilizando los parámetros intrínsecos de la cámara, se obtienen los parámetros extrínsecos de la misma, véase la Fig. 2.10.

Es importante decir que el error en la estimación usando esta estrategia depende principalmente de la resolución de la imagen y la precisión en la calibración de la cámara.

2.5 Método de Potenciales Artificiales

En esta sección se describe el método utilizado para resolver el problema de regulación de posición y evasión de obstáculos estáticos y dinámicos. Este algoritmo ha sido implementado en este trabajo debido a su simplicidad, a que se consideran obstáculos estáticos y dinámicos, y a que se trabaja con coordenadas cartesianas.

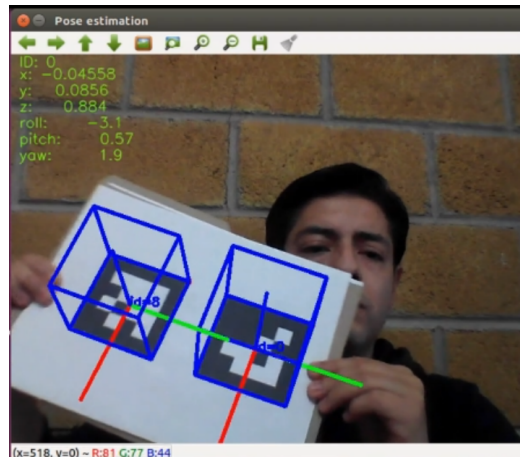


Figura 2.10: Estimación de la traslación y rotación entre el marco de referencia de la cámara y el marco de referencia del patrón.

Este método consiste en la creación de una función potencial artificial en el cual la meta constituye un polo atractivo y los obstáculos representan polos repulsivos. El móvil navega siguiendo el negativo del gradiente de la función potencial hasta alcanzar el mínimo global (meta). Sin embargo, en este método, cuando las fuerzas de atracción y repulsión tienen la misma magnitud y dirección contraria, el vehículo queda atrapado en un mínimo local de la función de potencial, deteniendo el móvil sin haber alcanzado la meta. Para resolver este problema, es posible crear una fuerza adicional de repulsión para hacer que el robot se mueva y salga del mínimo local.

Como se propone en [26], la función de potencial artificial toma en cuenta las posiciones del centro de la formación ó clúster, de la meta y de los obstáculos, y además las velocidades de éstos ya que la posibilidad de colisión depende de la posición/velocidad relativa entre el centro del clúster y el obstáculo.

A continuación considere las siguientes suposiciones en cuanto al método de potenciales artificiales:

- **Suposición 2.3** *La velocidad de los obstáculos y la meta es siempre menor que la velocidad de cada robot móvil que forman el clúster.*
- **Suposición 2.4** *Por simplicidad, la forma de los obstáculos es circular con radio conocido, r .*
- **Suposición 2.5** *Para garantizar que el clúster puede alcanzar a la meta, la posición de la meta no está dentro de la zona de influencia de algún obstáculo.*
- **Suposición 2.6** *Para garantizar que el clúster puede alcanzar la meta, cualquier obstáculo no se puede dirigir hacia ésta. En caso contrario se desplazará el mínimo global.*

2.5.1 Potencial atractivo y fuerza atractiva

En esta subsección se llama al sistema en cuestión *robot*, aunque en el siguiente capítulo dicho sistema será el *clúster*. El potencial atractivo es una función de la distancia y velocidad relativa entre la meta y el robot, definida de la forma [26]

$$U_{att}(p, v) = \frac{1}{2}\alpha_p \|p_g(t) - p(t)\|^{\alpha_m} + \frac{1}{2}\alpha_v \|v_g(t) - v(t)\|^{\alpha_n}, \quad (2.5.1)$$

donde $p(t)$ y $p_g(t)$ denotan la posición del robot y de la meta en el tiempo, respectivamente; con $p = [x \ y]^T$ y $p_g = [x_g \ y_g]^T$ en el marco de referencia global; $v(t)$ y $v_g(t)$ denotan la velocidad del robot y de la meta en el tiempo, respectivamente; $\|p_g(t) - p(t)\|$ es la distancia euclidiana entre el robot y la meta en el tiempo t ; $\|v_g(t) - v(t)\|$ es la magnitud de la velocidad relativa entre el robot y la meta en el tiempo t , con $v = [v_x \ v_y]^T$ y $v_g = [v_{g_x} \ v_{g_y}]^T$. α_p y α_v son factores de escalamiento positivos. α_m , y α_n son constantes positivas que definen la manera de que el robot alcance a la meta. El robot puede alcanzar la meta suavemente ó bruscamente, dependiendo de dichas constantes. En este caso el potencial atractivo está definido como

$$U_{att}(p, v) = \frac{1}{2}\alpha_p \|p_g(t) - p(t)\|^2 + \frac{1}{2}\alpha_v \|v_g(t) - v(t)\|^2. \quad (2.5.2)$$

Ya que $d_g = \|p_g(t) - p(t)\|$ y $\|v_g(t) - v(t)\|$ están elevadas al cuadrado, presentan una forma parabólica cuya fuerza atractiva converge linealmente conforme el robot se aproxima a la posición y velocidad de la meta, es decir, cuando el robot está cerca de la meta, éste se mueve lentamente hacia ésta y si la meta es dinámica, el robot converge a la velocidad de la meta. La fuerza atractiva está definida como el gradiente negativo del potencial atractivo en términos de la posición y la velocidad del robot cuando $p \neq p_g$ y $v \neq v_g$ de la forma [26]

$$F_{att}(p, v) = -\frac{\nabla U_{att}(p, v)}{\partial p} - \frac{\nabla U_{att}(p, v)}{\partial v} = F_{attp}(p) + F_{attv}(v), \quad (2.5.3)$$

con

$$F_{attp}(p) = \alpha_p \|p_g(t) - p(t)\| \hat{p}_{RG} = [f_{x_{attp}} \ f_{y_{attp}}]^T, \quad (2.5.4)$$

$$F_{attv}(v) = \alpha_v \|v_g(t) - v(t)\| \hat{v}_{RG} = [f_{x_{attv}} \ f_{y_{attv}}]^T, \quad (2.5.5)$$

donde \hat{p}_{RG} representa el vector unitario apuntando en la dirección de la posición del robot hacia la meta y \hat{v}_{RG} representa el vector unitario de la velocidad relativa apuntando en la dirección del robot hacia la meta. De manera simplificada la fuerza atractiva está definida como

$$F_{att}(p, v) = \begin{bmatrix} f_{x_{attp}} \\ f_{y_{attp}} \end{bmatrix} + \begin{bmatrix} f_{x_{attv}} \\ f_{y_{attv}} \end{bmatrix} = \alpha_p \begin{bmatrix} (x_g - x) \\ (y_g - y) \end{bmatrix} + \alpha_v \begin{bmatrix} (v_{g_x} - v_x) \\ (v_{g_y} - v_y) \end{bmatrix}. \quad (2.5.6)$$

La componente de la fuerza atractiva $F_{attp}(p)$ jala al robot a la posición de la meta y reduce la distancia entre éstos, mientras que la componente $F_{attv}(v)$ trata de que el robot se mueva a la misma velocidad que la meta, véase la Fig. 2.11.

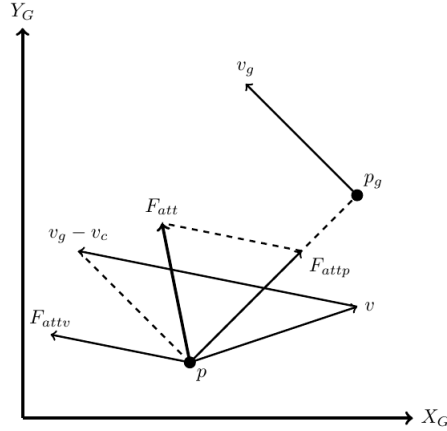


Figura 2.11: Fuerzas de atracción.

2.5.2 Potencial repulsivo y fuerza repulsiva

Antes de definir el potencial repulsivo y la fuerza repulsiva, considere la posición y velocidad de un obstáculo cualquiera dados por $p_{obs}(t) = [x_{obs} \ y_{obs}]^T$ y $v_{obs}(t) = [v_{obs_x} \ v_{obs_y}]^T$, respectivamente. La distancia euclidiana entre el robot y el obstáculo está dada como

$$d_{obs} = \|p_{obs}(t) - p(t)\|. \quad (2.5.7)$$

La velocidad relativa entre el robot y el obstáculo está dada por

$$v_{RO}(t) = [v(t) - v_{obs}(t)]^T. \quad (2.5.8)$$

La proyección de la velocidad relativa entre el robot y el obstáculo en la dirección del robot hacia el obstáculo está dada por

$$v_{RO}(t) = [v(t) - v_{obs}(t)]^T \cdot \hat{p}_{RO} = [v(t) - v_{obs}(t)]^T \cdot \frac{(p_{obs}(t) - p(t))}{\|p_{obs}(t) - p(t)\|}, \quad (2.5.9)$$

donde \hat{p}_{RO} es el vector unitario que apunta en la dirección del robot hacia el obstáculo y el operador “ \cdot ” indica el producto punto entre los dos vectores. Es importante notar que $v_{RO}(t) \in \mathbb{R}$. Si $v_{RO}(t) \leq 0$ implica que el robot se está alejando del obstáculo, y no es necesaria una acción de evasión. Si $v_{RO}(t) > 0$ implica que el robot se está acercando al obstáculo y por lo tanto es necesaria una acción de evasión. Asumiendo que el robot se acerca al obstáculo, la distancia mínima para detener al robot está dada de la forma

$$d_m(v_{RO}) = \frac{v_{RO}^2(t)}{2a_{max}}, \quad (2.5.10)$$

donde a_{max} es la máxima desaceleración del robot. Con el fin de evitar que se desplace el mínimo de la función de potencial atractivo, el potencial repulsivo debe tener una

región de influencia que delimite al obstáculo, la cual está dada por ρ_0 . El potencial repulsivo generado por el obstáculo $U_{rep}(p, v_{RO})$ está dado de la forma

$$U_{rep}(p, v_{RO}) = \begin{cases} \text{Indefinido,} & \text{Si } v_{RO} > 0 \text{ AND } d_{obs}(p, p_{obs}) < d_m(v_{RO}) + r \\ h_{rep}(p, p_{obs}, v_{RO}), & \text{Si } 0 < d_{obs}(p, p_{obs}) - r - d_m(v_{RO}) < \rho_0 \text{ OR } v_{RO} > 0 \\ 0, & \text{Caso contrario} \end{cases} \quad (2.5.11)$$

donde $h_{rep}(p, p_{obs}, v_{RO}) = \eta \left(\frac{1}{d_{obs}(p, p_{obs}) - r - d_m(v_{RO})} - \frac{1}{\rho_0} \right)$. r es el radio del obstáculo y η es un factor de escala positivo.

Observación 2.3 *A partir de los resultados en simulación numérica y experimentales, si se utiliza AND en $h_{rep}(p, p_{obs}, v_{RO})$ como en [26], se requiere aplicar a los robots velocidades lineales y angulares mayores y discontinuas. Así mismo, se obtienen errores de seguimiento más grandes. Utilizar OR indica a los robots móviles realizar una acción de evasión si éstos se encuentran dentro de la zona de influencia del obstáculo ó si se están acercando al obstáculo y, bajo la Suposición 2.6, la formación puede converger a la meta.*

De (2.5.11), el potencial repulsivo es cero si el robot está fuera de la zona de influencia del obstáculo ó sí se está alejando de éste, por lo tanto no es necesaria una acción de evasión. Por otra parte, el potencial repulsivo es indefinido o no está definido cuando $d_{obs}(p, p_{obs}) < d_m(v_{RO}) + r$ y $v_{RO} > 0$, es decir, que la distancia entre el robot y el obstáculo es menor que la requerida para frenar totalmente y además que el robot se está acercando al obstáculo, esto implica que la colisión entre el robot y el obstáculo es inevitable.

La fuerza repulsiva está definida como el gradiente negativo del potencial repulsivo en términos de la posición y la velocidad relativa del robot de la forma

$$F_{rep1}(p, v_{RO}) = -\frac{\nabla U_{rep}(p, v_{RO})}{\partial p} - \frac{\nabla U_{rep}(p, v_{RO})}{\partial v_{RO}} = F_{repp}(p) + F_{repv}(v_{RO}), \quad (2.5.12)$$

con

$$F_{repp}(p) = \frac{-\eta}{(d_{obs}(p, p_{obs}) - r - d_m(v_{RO}))^2} \hat{p}_{RO}, \quad (2.5.13)$$

y

$$F_{repv}(v_{RO}) = \frac{-\eta v_{RO}}{a_{max} [d_{obs}(p, p_{obs}) - r - d_m(v_{RO})]^2} \hat{p}_{RO}. \quad (2.5.14)$$

Para resolver el problema de mínimo local es necesario agregar otra fuerza de repulsión llamada F_{repn} . Dicha fuerza de repulsión es inversamente proporcional a la distancia entre el robot y el obstáculo, y apunta en la dirección perpendicular al vector unitario \hat{p}_{RO} . Además, de manera similar a [48], se propone una ganancia $k(\delta)$ que tome en

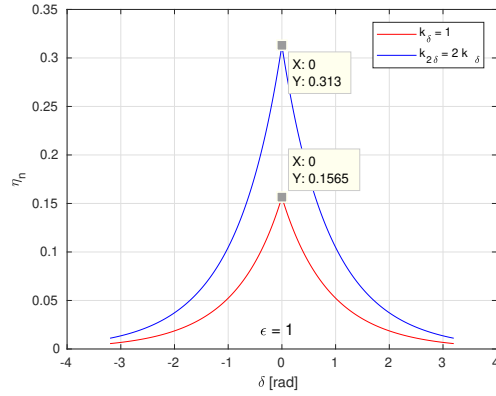


Figura 2.12: Diferentes valores de k_δ con $\epsilon = 1$.

cuenta el ángulo de acercamiento δ , el cual es el ángulo formado entre la distancia que une al robot y el obstáculo con la orientación del robot, véase la Fig. 2.13.

De esta manera, para resolver el problema de mínimo local, no solamente se considera la distancia entre el obstáculo y el robot, sino que también se considera el ángulo con el que se acercan para que se pueda omitir una acción de evasión cuando no sea necesaria. La ganancia η_n está dada de la forma

$$\eta_n = -\frac{k_\delta}{1 - e^{|\delta| + \epsilon}}, \quad (2.5.15)$$

donde k_δ es una ganancia positiva. ϵ es un parámetro estrictamente positivo que evita una indeterminación en η_n . También, en la Fig. 2.12 se presenta la gráfica de la ganancia η_n utilizando el mismo valor para ϵ . Asignar un valor fijo a ϵ permite sintonizar más fácilmente el valor para k_δ . En [48] se describen dos matrices de transformación para asignar la dirección de la fuerza F_{repp} debido a que el robot móvil tiene dos posibles opciones para evitar una colisión con el obstáculo, girar a la derecha o a la izquierda. En dicho trabajo no se especifica de manera analítica en la ecuación para definir F_{repp} la matriz a utilizar para que el robot tome una decisión entre rotar a la derecha o a la izquierda.

La fuerza de repulsión perpendicular a F_{repp} está dada de la forma

$$F_{repp}(p, p_{obs}) = \frac{\eta_n(\delta)}{d_{obs}(p, p_{obs})} \hat{p}_{RO\perp}. \quad (2.5.16)$$

La fuerza repulsiva total es la suma de las fuerzas F_{rep1} y F_{repp} , esto es, de forma simplificada

$$F_{rep}(p, p_{obs}, v_{RO}) = \frac{-\eta(1 + v_{RO}/a_{max})}{[d_{obs}(p, p_{obs}) - r - d_m(v_{RO})]^2} \hat{p}_{RO} + \frac{\eta_n(\delta)}{d_{obs}(p, p_{obs})} \hat{p}_{RO\perp}. \quad (2.5.17)$$

En la Fig. 2.13 se muestra la descomposición de los vectores utilizados y la fuerza repulsiva total. Notar que la componente F_{rep1} apunta en el sentido opuesto al vector

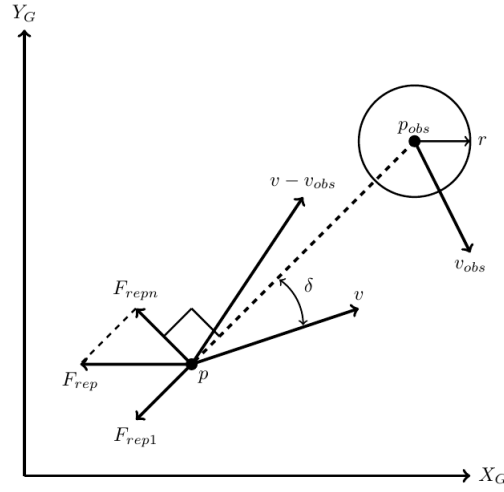


Figura 2.13: Fuerzas de repulsión.

$v_{RO}\hat{p}_{RO}$. Esto implica que ésta fuerza trata de alejar al robot del obstáculo, mientras que la componente F_{repn} está apuntando en la dirección del vector $\hat{p}_{RO\perp}$ y por lo tanto ésta trata de dirigir al robot del obstáculo a una orientación perpendicular.

La fuerza total resultante es la suma de la fuerza de atracción y de repulsión descrita por

$$F(p, p_{obs}, v_{RO}) = F_{att}(p, v) + F_{rep}(p, p_{obs}, v) = [F_x \ F_y]^T. \quad (2.5.18)$$

Para el caso de un escenario con más de un obstáculo, el potencial atractivo está dado por (2.5.6), mientras que por cada obstáculo existirá una fuerza repulsiva, definida en (2.5.17) con las componentes dadas por (2.5.12) y (2.5.16). El número de fuerzas se incrementará proporcionalmente al número de obstáculos.

La orientación a la que apunta la fuerza resultante para evadir al obstáculo y alcanzar la meta está dado por

$$\theta_f = \tan^{-1}(F_y/F_x). \quad (2.5.19)$$

En este trabajo se propone la velocidad lineal deseada para la navegación V_d de la forma

$$V_d = K_v d_g, \quad (2.5.20)$$

donde K_v es una ganancia positiva. De esta manera, el robot converge a la meta de forma suave y sin sobre impulso. Utilizar $V_d = d_{obs}V_{op}/R_o$ ó $V_d = K_v d_g - k_o/d_{obs}$ funciona de manera aceptable pero puede resultar en que el clúster se detenga cuando se localice cerca del obstáculo.

2.5.3 Problema de mínimo local

Bajo la *Suposición 2.5*, no es posible que el robot, la meta y el obstáculo se localicen alineados con la meta en medio y dentro de la zona de influencia del obstáculo. Bajo la

Suposición 2.6, no es posible que estos 3 individuos se localicen alineados y el obstáculo se dirija hacia la meta. En este esquema, el problema de mínimo local sucede cuando el robot, el obstáculo y la meta se localizan alineados, con el obstáculo en medio y acercándose a la meta, impidiendo que el robot alcance dicho objetivo. Al ser un entorno dinámico la solución más simple es esperar a que se muevan de esa configuración. Pero sí el robot, el obstáculo y la meta se mueven con la misma velocidad, la misma dirección y el mismo sentido, dicha situación se mantendrá indefinida.

Para resolver este problema, en la sección anterior al igual que en [48], se propuso la fuerza repulsiva F_{repn} , la cual actúa en la dirección perpendicular al vector que apunta del robot hacia el obstáculo y con ganancia variable. Esta fuerza resuelve la situación mencionada anteriormente pero, de la misma manera, se tendrá un mínimo local en alguna otra configuración aunque sea por un corto periodo de tiempo debido a los múltiples obstáculos obteniéndose una fuerza total igual a cero. Para resolver este problema, y sólo en este caso, se considera $F_{repn} = 0$.

Capítulo 3

Esquema de sincronización usando CS

En este capítulo se describe detalladamente el esquema de formación y sincronización de robot móviles propuesto en este trabajo. Primero, se da un breve repaso a los trabajos relacionados y al enfoque original de “*Cluster Space*” para resaltar la diferencia y contribución de este trabajo. También, se describen los controladores a nivel del clúster ó formación, y a nivel de los robots (controlador interno) utilizados en este esquema. Después, se presenta el análisis de la estabilidad de la formación de móviles. Finalmente, se explica cómo se implementa el método de potenciales artificiales para el clúster.

3.1 Antecedentes

3.1.1 Trabajos relacionados

En [10], se define el término de error de sincronización y se presenta un enfoque de sincronización para el seguimiento de la trayectoria de múltiples robots móviles mientras se mantiene la formación variante en el tiempo. El controlador propuesto garantiza la convergencia asintótica a cero tanto de la posición como de los errores de sincronización de cada robot. Solo se aborda el problema de la regulación, es decir, el movimiento de los robots móviles se sincroniza desde una forma inicial específica a otra forma final específica.

En [5], los autores proponen una estrategia para resolver el problema de sincronización de robots móviles y de control de formación mediante el uso de un enfoque de estructura virtual con acoplamiento mutuo entre los robots. Los *términos de acoplamiento mutuo* se agregan directamente en la estrategia de control de los robots móviles unicyclo, debido a la restricción no holonómica. Con esos términos de acoplamiento mutuo, es posible resolver el problema de la sincronización mutua entre los elementos dotando de mayor robustez a la formación ante perturbaciones. Además, se muestra que el ajuste de las ganancias de control de acoplamiento proporciona una manera para

mediar entre el seguimiento individual de las trayectorias deseadas de los robots y el objetivo de lograr la formación. Se muestra la prueba de estabilidad para la dinámica del error de formación de dos robots tipo (2,0), bajo el supuesto de que las velocidades de rotación deseadas de ambos robots móviles son idénticas.

Similar a [5], los autores proponen en [49], un enfoque de sincronización, usando múltiples robots móviles tipo unicycle, para seguir trayectorias de referencia individuales mientras se mantiene una formación variante en el tiempo. Los movimientos de todos los robots móviles se sincronizan gracias a los términos de acoplamiento en sus leyes de control. Las leyes de control garantizan la convergencia asintótica a cero de los errores de sincronización y la estabilidad asintótica de la dinámica del error de seguimiento. Motivado por [10], el error de sincronización de posición se define como la diferencia entre dos errores de seguimiento. Las velocidades lineal y angular de cada robot móvil pueden ser todas diferentes, mientras que las velocidades angulares pueden incluso ser discontinuas.

En [17], los autores presentan un controlador a nivel de la formación (clúster) para robots móviles no holonómicos. El controlador produce comandos a nivel del clúster que se traducen en movimientos válidos a nivel de cada robot. Con el controlador propuesto a nivel del clúster, se demuestra la estabilidad en el sentido de Lyapunov de la arquitectura de control en lazo cerrado, lo que indica que las trayectorias se pueden seguir con errores acotados. Además, el error de seguimiento se puede reducir aumentando los valores de ganancia.

En [18], algunos de los métodos de control de formación más comunes, que se encuentran en la literatura, se estudian desde la perspectiva del enfoque de *Cluster Space*. En particular, los enfoques de líder-seguidor, funciones potenciales y estructuras virtuales se analizan con la intención de abordar la generalidad y flexibilidad del enfoque de control de la formación de *Cluster Space*. Se comparan y contrastan diferentes implementaciones.

Este trabajo de tesis presenta las contribuciones que ya están publicadas en [50]. Primero, se presenta un esquema de sincronización desde la perspectiva del enfoque de *Cluster Space*. Además, se presenta el diseño de una ley de control interna, a nivel de robots. Con este controlador interno, se aborda un enfoque de sincronización para resolver tanto el problema del control de la formación como el problema de seguimiento de la trayectoria de los robots móviles tipo unicycle. Con esto, el movimiento de cada robot se compone de dos partes: una es conducir al robot a lo largo de su trayectoria individual deseada para resolver el problema de seguimiento de trayectoria y la otra es sincronizar el movimiento de cada robot con el resto de robots para resolver el problema de sincronización. Hasta donde sabemos, no se había propuesto un enfoque de sincronización desde la perspectiva de *Cluster Space*.

Segundo, se presenta el análisis de estabilidad basado en la teoría de Lyapunov, considerando el caso general, esto es, que tanto las orientaciones deseadas como las velocidades de rotación deseadas de todos los robots móviles pueden ser todas diferentes entre sí. A diferencia del enfoque de *Cluster Space*, la estrategia de control propuesta garantiza que tanto los errores de seguimiento de los robots como los errores de sincro-

nización convergen asintóticamente al origen. Al lograr esto, los errores de seguimiento del clúster también convergen asintóticamente a cero. Además, el controlador interno propuesto garantiza que los errores de orientación de los robots tienden exponencialmente a cero. Hasta donde sabemos, no se había demostrado formalmente estabilidad asintótica de todos los errores.

3.1.2 Enfoque de Cluster Space

Se define a un clúster como una entidad compuesta por varios robots móviles la cuál puede ser representada como segmentos de líneas rectas que une los centros de los robots. Esta línea (2 robots) o figura cerrada (más de 3 robots) tiene orientación, forma definida y un tamaño particular conocido [13].

Para implementar el enfoque de *Cluster Space* con n robots móviles, se eligen apropiadamente las variables de la formación (clúster) que representen la posición, orientación y forma de ésta. Sea $m_m = m_t + m_r$, donde m_m es el número de grados de libertad (GDL) de cada robot, m_t es el número de GDL traslacionales y m_r es el número de GDL rotacionales. Entonces, se necesitan nm_m parámetros independientes, con respecto al marco de referencia global, para describir la postura (posición y orientación) de todos los robots en la formación [51]. Para una formación compuesta por 3 robots móviles terrestres, se asume que $n = 3$, $m_m = 3$, $m_t = 2$, $m_r = 1$. Las variables que describen la postura de los robots se llaman variables en el espacio de los robots, y las variables que describen la postura y forma del clúster se llaman variables en el espacio del clúster. Ésta es la razón del nombre del enfoque utilizado.

De acuerdo con este enfoque, la selección de las variables en el espacio de la formación es en función a la aplicación, al diseño del sistema, o al criterio subjetivo del usuario. Abusando de la libre selección de estas variables, similar a [52, 53], en este trabajo, las orientaciones de los n robots, están dadas con respecto al eje global X_G , independientemente del espacio empleado. Por lo tanto, es posible tomar en cuenta 3 vectores separadamente. Uno de ellos está dado por las nm_r variables que describen la orientación de los robots móviles, y los otros describen la postura de los robots y de la formación (clúster). Entonces, la orientación de los robots está dada por $\mathbf{q} = [q_1, q_2, q_3]^T$. El vector de orientaciones deseadas está dado por $\mathbf{q}_d = [q_{1d}, q_{2d}, q_{3d}]^T$.

El vector de postura de los robots, con respecto del marco global, está dado por

$${}^G\mathbf{R} = [x_1, y_1, x_2, y_2, x_3, y_3]^T, \quad (3.1.1)$$

donde ${}^G\mathbf{R} \in \mathbb{R}^{nm_t}$; (x_i, y_i) representa la posición del i -ésimo robot móvil.

Antes de presentar la selección propuesta para las variables en el espacio de la formación, se presenta la selección común para estas variables. En la mayoría de trabajos, para un clúster de 3 robots móviles terrestres, el vector de postura de la formación está dado por

$$\mathbf{C}_s = [X_c, Y_c, Q_c, D_2, D_3, \alpha_c]^T, \quad (3.1.2)$$

donde (X_c, Y_c) es el centroide del triángulo que se forma por la posición de los 3 robots $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$ y $P_3 = (x_3, y_3)$. D_2 es la distancia euclidiana desde el punto

P_1 al punto P_2 . Por simplicidad en la notación, considere cualquier distancia euclidiana, $\|\Lambda\|$, simplemente como Λ . D_3 es la distancia euclidiana desde el punto P_1 al punto P_3 , (véase la fig. 3.1). m es el punto medio del segmento de línea recta que conecta los puntos P_2 con P_3 . Q_c describe la orientación del clúster, la cual apunta desde P_1 hacia m , y está dada con respecto del eje global X_G . α_c es el ángulo con vértice en el punto P_1 .

La relación entre las variables en el espacio de la formación y el espacio de los robots está descrita por la cinemática de posición directa e inversa, las cuales se pueden representar como $\mathbf{C}_s = KIN({}^G\mathbf{R})$ y ${}^G\mathbf{R} = INV(\mathbf{C}_s)$, respectivamente. La cinemática de posición directa para el sistema compuesto por 3 robots está dada por

$$\begin{bmatrix} X_c \\ Y_c \\ Q_c \\ D_2 \\ D_3 \\ \alpha_c \end{bmatrix} = \begin{bmatrix} (x_1 + x_2 + x_3)/3 \\ (y_1 + y_2 + y_3)/3 \\ ATAN2[(y_2 + y_3 - 2y_1)/(x_2 + x_3 - 2x_1)] \\ \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2} \\ \sqrt{(x_1 - x_3)^2 + (y_1 - y_3)^2} \\ ACOS[(D_2^2 + D_3^2 - D_1^2)/(2D_2D_3)] \end{bmatrix} \quad (3.1.3)$$

donde ATAN2 es una función computacional que calcula la tangente inversa, con respecto al eje horizontal, en un rango de $[-\pi, \pi]$. D_1 es la distancia euclidiana desde P_2 a P_3 , esto es, $D_1 = \sqrt{(x_2 - x_3)^2 + (y_2 - y_3)^2}$.

De la ley de cosenos, D_1 está dado, en función de las variables del clúster, como

$$D_1 = \sqrt{D_2^2 + D_3^2 - 2D_2D_3\cos\alpha_c}. \quad (3.1.4)$$

Del teorema de Apolonio, la siguiente igualdad se satisface:

$$D_2^2 + D_3^2 = \frac{1}{2}D_1^2 + 2H^2, \quad (3.1.5)$$

donde H es la distancia euclidiana desde P_1 a m . Sustituyendo (3.1.4) en (3.1.5), H está dado por

$$H = \frac{1}{2}\sqrt{D_2^2 + D_3^2 + 2D_2D_3\cos\alpha_c}. \quad (3.1.6)$$

De la ley de cosenos, α_2 está dada por

$$\alpha_2 = ACOS\left(\frac{D_2^2 + H^2 - (1/4)D_1^2}{2D_2H}\right). \quad (3.1.7)$$

Por lo tanto, la cinemática inversa de posición está dada por

$$\begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix} = \begin{bmatrix} X_c - (2/3)H \cos(Q_c) \\ Y_c - (2/3)H \sin(Q_c) \\ X_c - (2/3)H \cos(Q_c) + D_2 \cos(Q_c + \alpha_2) \\ Y_c - (2/3)H \sin(Q_c) + D_2 \sin(Q_c + \alpha_2) \\ X_c - (2/3)H \cos(Q_c) + D_3 \cos(Q_c + \alpha_2 - \alpha_c) \\ Y_c - (2/3)H \sin(Q_c) + D_3 \sin(Q_c + \alpha_2 - \alpha_c) \end{bmatrix} \quad (3.1.8)$$

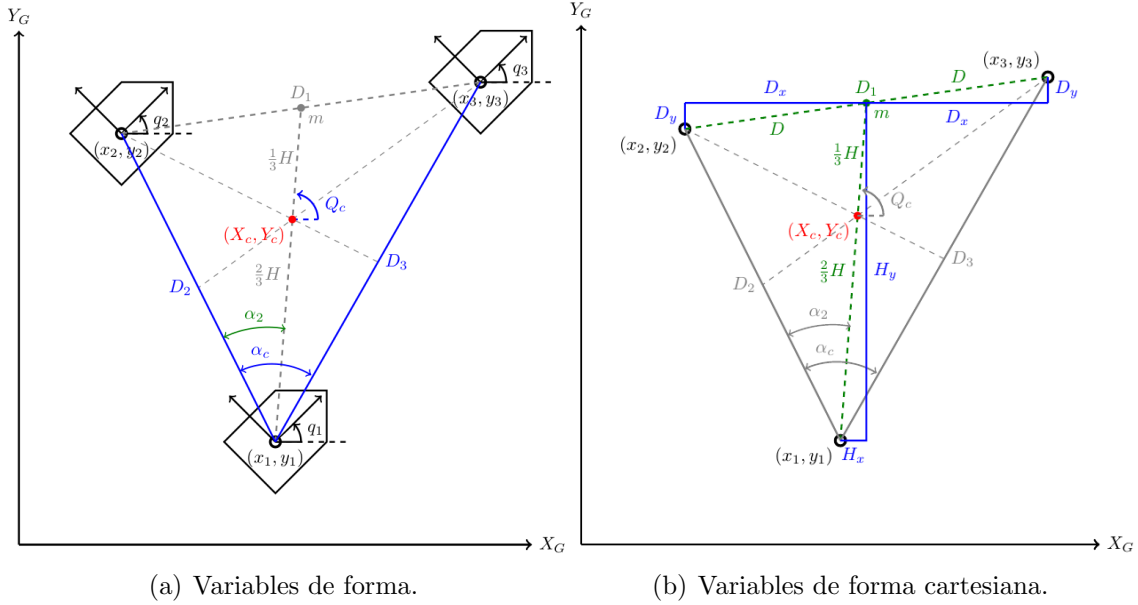


Figura 3.1: Clúster compuesto por 3 robots móviles terrestres.

La cinemática directa de velocidad que mapea las velocidades de los robots a las velocidades del clúster, se puede expresar como $\dot{\mathbf{C}}_s = J({}^G\mathbf{R})G\dot{\mathbf{R}}$, donde la matriz Jacobiana ($J({}^G\mathbf{R})$) se obtiene calculando las derivadas parciales de las variables de la formación (\mathbf{C}_s), con respecto de las variables en el espacio de los robots. Recíprocamente, la cinemática inversa de velocidad que mapea las velocidades de la formación a las de los robots, se puede expresar como ${}^G\dot{\mathbf{R}} = J^{-1}(\mathbf{C}_s)\dot{\mathbf{C}}_s$, donde la matriz Jacobiana inversa ($J^{-1}(\mathbf{C}_s)$) se obtiene calculando las derivadas parciales de las variables de los robots (\mathbf{R}), con respecto de las variables de la formación. Las expresiones algebraicas completas para $J({}^G\mathbf{R})$ y $J^{-1}(\mathbf{C}_s)$, para el caso con $n = 2$, se muestran en [16].

Es importante notar que, dependiendo de la selección de las variables de la formación, pueden ocurrir singularidades utilizando el enfoque de *Cluster Space*. Las singularidades se presentan cuando la matriz, ya sea $J({}^G\mathbf{R})$ o $J^{-1}(\mathbf{C}_s)$, se vuelve singular. Similar a [54], en la especificación particular presentada anteriormente, se obtiene una configuración singular cuando $D_2 = 0$, $D_3 = 0$ y $\alpha_c = 0, \pi$. Las 2 primeras son imposibles en una aplicación real, pero singularidades en el ángulo α_c podrían presentarse. Con ambas configuraciones se obtiene un línea recta en lugar de una forma triangular. Para evitar estas singularidades, una solución es mantener la forma del clúster lejos de dicha configuración. Si esto no es posible, otra solución es seleccionar otras variables para el clúster tal que las singularidades se presenten lejos de la configuración requerida.

3.1.3 Controlador utilizado a nivel de la formación

Por defecto, en el enfoque de *Cluster Space* se utiliza un controlador P a nivel de la formación. Este controlador, en el espacio del clúster, compara tanto la posición como la velocidad de la formación con valores deseados y genera las salidas de velocidad para la formación, las cuales son transformadas a velocidades individuales para cada robot móvil a través de la matriz Jacobiana inversa. La postura de los robots se convierte en la postura en el espacio del clúster mediante la cinemática directa de posición y la matriz Jacobiana, y retroalimentada al controlador. La restricción no holónoma reduce las variables de la formación a 6 GDL. Para garantizar que la restricción no holónoma se satisface (2.1.6), se implementa un controlador interno, a nivel de cada robot. El controlador a nivel de la formación no regula directamente la orientación de los robots $q_i(t)$ [17, 54].

También, se considera el modelo cinemático de los robots móviles bajo la suposición de que los robots móviles usan su propio controlador embebido que regula y garantiza las velocidades de control asignadas (lineal y angular).

En [17], se prueba la estabilidad del sistema en lazo cerrado en el sentido de Lyapunov, donde se indica que se pueden seguir las trayectoria deseadas para la formación obteniendo errores de seguimiento acotados por una banda. Además, dichos errores pueden ser reducidos incrementando los valores de las ganancias utilizadas por el controlador, a nivel del clúster. Con esto en mente, se puede concluir que tanto los errores de seguimiento a nivel de la formación como los errores de seguimiento a nivel de los robots están acotados. Por lo tanto, mediante el uso solamente del controlador a nivel del clúster, no se garantiza convergencia asintótica a cero de los errores de seguimiento en el espacio de la formación ni en el espacio de los robots.

Problema a resolver: *Dada una trayectoria deseada para el centroide del clúster, se requiere diseñar un esquema de sincronización para resolver el problema de seguimiento de trayectoria usando robots móviles tipo (2,0), desde la perspectiva de Cluster Space, tal que el movimiento de los robots debe estar sincronizado durante todo el seguimiento de trayectoria. Además, se requiere diseñar una ley de control que garantice convergencia asintótica al origen de todos los errores, errores de seguimiento a nivel de la formación (clúster), a nivel de los robots, y los errores de orientación de los robots.*

3.2 Esquema de sincronización desde la perspectiva de CS

En esta Sección, se presenta en detalle el esquema de sincronización de robots móviles tipo unicycle, desde la perspectiva de *Cluster Space* (CS), propuesto. Similar a [53], se implementan dos controladores en cascada, uno es el controlador clásico a nivel del clúster, y el otro se implementa a nivel de los robots, no solo para satisfacer la res-

tricción no holónoma, como en [17, 54], sino también para garantizar el movimiento sincronizado de los robots y la convergencia asintótica a su trayectoria individual.

3.2.1 Selección de las variables del clúster

Considere los siguientes vectores: $\mathbf{D} = [D_x, D_y]^T = \frac{1}{2}(\mathbf{P}_3 - \mathbf{P}_2)$ y $\mathbf{H} = [H_x, H_y]^T = \mathbf{D} - \mathbf{P}_1$, donde $\mathbf{P}_i = [x_i, y_i]^T$, con $i = 1, 2, 3$, denota el vector de posición del i -ésimo robot móvil. Por lo tanto, los vectores \mathbf{D} y \mathbf{H} están dados, respectivamente, como

$$\mathbf{D} = \frac{1}{2} \begin{bmatrix} x_3 - x_2 \\ y_3 - y_2 \end{bmatrix}, \quad \mathbf{H} = \begin{bmatrix} (x_3 - x_2)/2 - x_1 \\ (y_3 - y_2)/2 - y_1 \end{bmatrix}. \quad (3.2.1)$$

En lugar de usar las variables comunes $\mathbf{C}_k = [Q_c, D_2, D_3, \alpha_c]^T$ que describen la forma y orientación del clúster, en este trabajo, las variables seleccionadas son las componentes de los vectores \mathbf{D} y \mathbf{H} , esto es, $\mathbf{C}_c = [D_x, D_y, H_x, H_y]^T$. Estas 4 variables también describen, de manera implícita, la forma y orientación del clúster, (véase la Fig. 3.1). Por lo tanto, el vector de la postura de la formación, $\mathbf{C} \in \mathbb{R}^{nm_t}$, se elige como

$$\mathbf{C} = [X_c, Y_c, D_x, D_y, H_x, H_y]^T. \quad (3.2.2)$$

Con esta selección, tanto la cinemática directa como inversa de posición no están dadas por funciones variantes en el tiempo $\mathbf{C} = \text{KIN}({}^G\mathbf{R})$ y ${}^G\mathbf{R} = \text{INV}(\mathbf{C})$, respectivamente. Estas relaciones se pueden definir mediante el uso de matrices invariantes en el tiempo. La cinemática directa de posición del sistema compuesto por 3 robots móviles está dada como $\mathbf{C} = \mathbf{K}{}^G\mathbf{R}$, esto es

$$\begin{bmatrix} X_c \\ Y_c \\ D_x \\ D_y \\ H_x \\ H_y \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} \\ -1 & 0 & \frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -1 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix}, \quad (3.2.3)$$

donde $\mathbf{K} \in \mathbb{R}^{nm_t \times nm_t}$. Note que, dado que la matriz \mathbf{K} es de rango completo, la cinemática inversa de posición está dada como ${}^G\mathbf{R} = \mathbf{K}^{-1}\mathbf{C}$, donde \mathbf{K}^{-1} es la matriz inversa de \mathbf{K} , esto es

$$\begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & -2/3 & 0 \\ 0 & 1 & 0 & 0 & 0 & -2/3 \\ 1 & 0 & -1 & 0 & 1/3 & 0 \\ 0 & 1 & 0 & -1 & 0 & 1/3 \\ 1 & 0 & 1 & 0 & 1/3 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1/3 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ D_x \\ D_y \\ H_x \\ H_y \end{bmatrix}. \quad (3.2.4)$$

Observación 3.1: De la Fig. 3.1, es fácil ver que la posición de los robots está definida por el corrimiento del centroide del triángulo a lo largo de los ejes X_G y Y_G , los cuales dependen de las variables en el espacio de la formación. Por lo tanto, es simple deducir la cinemática directa e inversa para un clúster compuesto por 3 robots móviles aéreos, añadiendo la traslación a lo largo del eje Z_G de cada robot. Así mismo, la cinemática de posición para una formación compuesta por 2 robots móviles terrestres se puede obtener removiendo el robot 1 y sus variables correspondientes H_x, H_y .

Derivando la cinemática de posición directa e inversa, se obtienen las correspondientes cinemática de velocidades, esto es

$$\dot{\mathbf{C}} = \mathbf{K}^G \dot{\mathbf{R}}, \quad {}^G \dot{\mathbf{R}} = \mathbf{K}^{-1} \dot{\mathbf{C}}, \quad (3.2.5)$$

respectivamente. Por lo tanto, usando $\mathbf{C}_c = [D_x, D_y, H_x, H_y]^T$, se evitan las singularidades debido a que no es necesario calcular las matrices Jacobianas. Note que la matriz \mathbf{K} es constante. Así mismo, es posible considerar una formación colineal, esto significa que se requiere formar una línea recta con los 3 robots móviles en lugar de un triángulo. Asignar $H_x = H_y = 0$ implica que el robot 1 es el centroide del clúster. Aunque es posible asignar $D_x = D_y = 0$, esto implica que $D_1 = 0$, lo que es imposible en una aplicación real.

Para resolver el problema de seguimiento de trayectoria, considere el vector de postura deseado para la formación dado por

$$\mathbf{C}_d = [X_{cd}, Y_{cd}, D_{xd}, D_{yd}, H_{xd}, H_{yd}]^T. \quad (3.2.6)$$

En contraste con el enfoque original de *Cluster Space*, con el esquema de sincronización propuesto, solamente se requiere la cinemática de velocidad inversa en el controlador a nivel de los robots. Similar a [53] y [55], la cinemática de velocidad inversa se requiere para calcular las velocidades deseadas para cada uno de los robots ($\dot{x}_{id}(t), \dot{y}_{id}(t)$), con $i = 1, 2, 3$. La orientación deseada para cada robot $q_{id}(t)$, la cual satisface la restricción no holónoma, se obtiene usando (2.1.7). Las correspondientes velocidades rotacionales $\dot{q}_{id}(t)$, con $i = 1, \dots, n$, se pueden calcular usando (2.1.8).

Por lo tanto, las velocidades deseadas para los robots están dadas por ${}^G \dot{\mathbf{R}}_d = \mathbf{K}^{-1} \dot{\mathbf{C}}_d$, esto es

$${}^G \dot{\mathbf{R}}_d = \begin{bmatrix} \dot{x}_{1d} \\ \dot{y}_{1d} \\ \dot{x}_{2d} \\ \dot{y}_{2d} \\ \dot{x}_{3d} \\ \dot{y}_{3d} \end{bmatrix} = \begin{bmatrix} \dot{X}_{cd} - \frac{2}{3} \dot{H}_{xd} \\ \dot{Y}_{cd} - \frac{2}{3} \dot{H}_{yd} \\ \dot{X}_{cd} + \frac{1}{3} \dot{H}_{xd} - \dot{D}_{xd} \\ \dot{Y}_{cd} + \frac{1}{3} \dot{H}_{yd} - \dot{D}_{yd} \\ \dot{X}_{cd} + \frac{1}{3} \dot{H}_{xd} + \dot{D}_{xd} \\ \dot{Y}_{cd} + \frac{1}{3} \dot{H}_{yd} + \dot{D}_{yd} \end{bmatrix}, \quad (3.2.7)$$

donde $\dot{\mathbf{C}}_d = [\dot{X}_{cd}, \dot{Y}_{cd}, \dot{D}_{xd}, \dot{D}_{yd}, \dot{H}_{xd}, \dot{H}_{yd}]^T$ es la derivada temporal de la trayectoria deseada para el clúster.

Aunque se requiere una formación invariante en el tiempo, las variables del vector $\mathbf{C}_c = [D_x, D_y, H_x, H_y]^T$ son variantes en el tiempo porque están expresadas como las proyecciones sobre los ejes globales. Las derivadas temporales de estas variables serán diferentes dependiendo de los requerimientos de la trayectoria deseada para el clúster e incluso sí es que una formación variante en el tiempo se quiere considerar. A continuación se presentan las derivadas temporales deseadas para la formación tomando en cuenta el caso cuando se requiere un clúster en forma triangular invariante en el tiempo, asumiendo que la orientación de éste apunta tangente a su trayectoria deseada.

En este caso, las variables H_x y H_y representan la orientation del clúster. Por lo tanto, H_{xd} , H_{yd} están dadas, respectivamente, como

$$H_{xd}(t) = H_d \cos(Q_{cd}(t)), \quad H_{yd}(t) = H_d \sin(Q_{cd}(t)), \quad (3.2.8)$$

donde $Q_{cd}(t)$ es la orientación deseada de la formación según la trayectoria asignada. La distancia euclidiana H_d se puede obtener usando (3.1.6). Entonces, las variables D_{xd} y D_{yd} son

$$\begin{aligned} D_{xd} &= H_{xd} - D_{2d} \cos(Q_{cd} + \alpha_{2d}), \\ D_{yd} &= H_{yd} - D_{2d} \sin(Q_{cd} + \alpha_{2d}), \end{aligned} \quad (3.2.9)$$

donde α_{2d} está expresada en (3.1.7). Derivando (3.2.8) y (3.2.9), con respecto del tiempo, se tiene

$$\begin{aligned} \dot{H}_{xd} &= -\dot{Q}_{cd} H_d \sin(Q_{cd}), \quad \dot{H}_{yd} = \dot{Q}_{cd} H_d \cos(Q_{cd}), \\ \dot{D}_{xd} &= \dot{Q}_{cd} D_{2d} \sin(Q_{cd} + \alpha_{2d}) + \dot{H}_{xd}, \\ \dot{D}_{yd} &= -\dot{Q}_{cd} D_{2d} \cos(Q_{cd} + \alpha_{2d}) + \dot{H}_{yd}. \end{aligned} \quad (3.2.10)$$

Observación 3.2: Aunque las variables $(Q_c, D_2, D_3, \alpha_c)$ no se consideran variables de posición, orientación o forma del clúster, para calcular las variables deseadas de la formación $(D_{xd}, D_{yd}, H_{xd}, H_{yd})$, por simplicidad, primeramente se elijen las variables deseadas que describen la forma $(Q_{cd}, D_{2d}, D_{3d}, \alpha_{cd})$. Después de eso, usando (3.1.6), (3.2.8), (3.2.9) y (3.1.7), se calculan las correspondientes variables deseadas. Las variables de la formación seleccionadas se pueden ver como el “esqueleto” con forma de “T”, (véase la Fig. 3.1).

3.2.2 Diseño del controlador a nivel de la formación

En este trabajo, se implementa un controlador P a nivel de la formación. Por lo tanto, el controlador genera el siguiente vector de errores ponderados de la formación

$$\begin{aligned} \Delta \mathbf{E}_c &= \mathbf{K}_c \mathbf{E}_c = \mathbf{K}_c (\mathbf{C} - \mathbf{C}_d) = \\ &[\Delta E_{Xc}, \Delta E_{Yc}, \Delta E_{Dx}, \Delta E_{Dy}, \Delta E_{Hx}, \Delta E_{Hy}]^T, \end{aligned} \quad (3.2.11)$$

donde \mathbf{E}_c es el vector de errores del clúster; $\mathbf{K}_c \in \mathbb{R}^{nm_t \times nm_t}$ es una matriz diagonal compuesta por las ganancias positivas: $K_{Xc}, K_{Yc}, K_{Dx}, K_{Dy}, K_{Hx}, K_{Hy}$.

En el enfoque original, se aplica la cinemática de velocidad inversa (${}^G\dot{\mathbf{R}} = J^{-1}(\mathbf{C})\dot{\mathbf{C}}$), y usando la salida del controlador a nivel del clúster ($\dot{\mathbf{C}}_{\text{cmd}}$), se obtienen las velocidades de los robots ($\dot{\mathbf{R}}_{\text{cmd}}$).

De manera similar, en el esquema propuesto, la cinemática inversa de posición (3.2.4) se utiliza para obtener la salida del controlador a nivel del clúster, $\Delta \mathbf{R} = \mathbf{K}^{-1} \Delta \mathbf{E}_c$, esto es

$$\Delta \mathbf{R} = \begin{bmatrix} \Delta x_1 \\ \Delta y_1 \\ \Delta x_2 \\ \Delta y_2 \\ \Delta x_3 \\ \Delta y_3 \end{bmatrix} = \begin{bmatrix} \Delta E_{Xc} - (2/3)\Delta E_{Hx} \\ \Delta E_{Yc} - (2/3)\Delta E_{Hy} \\ \Delta E_{Xc} + (1/3)\Delta E_{Hx} - \Delta E_{Dx} \\ \Delta E_{Yc} + (1/3)\Delta E_{Hy} - \Delta E_{Dy} \\ \Delta E_{Xc} + (1/3)\Delta E_{Hx} + \Delta E_{Dx} \\ \Delta E_{Yc} + (1/3)\Delta E_{Hy} + \Delta E_{Dy} \end{bmatrix}. \quad (3.2.12)$$

Note que, la salida del controlador a nivel de la formación está compuesta por los errores de sincronización de los robots ($\Delta \mathbf{R}$), como se muestra a continuación.

3.2.3 Enfoque de sincronización

Considere los errores de seguimiento de los robots, con respecto al marco de referencia global $\{G\}$, e_{xi}, e_{yi} , dados en (2.2.6) y (2.2.7), con $i = 1, \dots, n$.

La relación entre los errores de seguimiento de los robots y la entrada del controlador a nivel de los robots se puede obtener mediante el uso de (3.2.11), (3.2.12) y (3.2.3). El error Δx_2 , se presenta como

$$\begin{aligned} \Delta x_2 &= K_{Xc}(X_c - X_{cd}) + \frac{K_{Hx}}{3}(H_x - H_{xd}) - K_{Dx}(D_x - D_{xd}) \\ &= \frac{K_{Xc}}{3}(e_{x1} + e_{x2} + e_{x3}) + \frac{K_{Hx}}{(3)(2)}(e_{x2} + e_{x3} - 2e_{x1}) \\ &\quad - \frac{K_{Dx}}{2}(e_{x3} - e_{x2}) \\ &= \frac{1}{3}(K_{Xc} + \frac{1}{2}K_{Hx} + \frac{3}{2}K_{Dx})e_{x2} + \frac{1}{3}(K_{Xc} - K_{Hx})e_{x1} \\ &\quad + \frac{1}{3}(K_{Xc} + \frac{1}{2}K_{Hx} - \frac{3}{2}K_{Dx})e_{x3}. \end{aligned} \quad (3.2.13)$$

Sin pérdida de generalidad, considere las siguientes ganancias:

$$\begin{aligned} K_C &= K_{Xc} = K_{Yc}, K_D = K_{Dx} = K_{Dy}, \\ K_H &= K_{Hx} = K_{Hy}, \\ \beta &= K_C + \frac{1}{2}K_H + \frac{3}{2}K_D, \quad \bar{\beta} = K_C + \frac{1}{2}K_H - \frac{3}{2}K_D, \\ \mu &= K_C + 2K_H, \quad \bar{\mu} = K_C - K_H, \end{aligned} \quad (3.2.14)$$

donde $K_C, K_D, K_H, \beta, \mu$ son también ganancias positivas; $\bar{\mu}, \bar{\beta} \in \mathbb{R}$. Más información sobre estas ganancias se presenta posteriormente. Sustituyendo (3.2.14) en (3.2.13) se tiene

$$\Delta x_2 = \frac{1}{3} (\bar{\mu} e_{x1} + \beta e_{x2} + \bar{\beta} e_{x3}). \quad (3.2.15)$$

El resto de errores de sincronización (3.2.12) se pueden obtener de manera similar. Por lo tanto, los errores de sincronización de los robots, como función de sus errores de seguimiento, están dados por

$$\begin{aligned} \Delta \lambda_1 &= \frac{1}{n} (\mu e_{\lambda 1} + \bar{\mu} e_{\lambda 2} + \bar{\mu} e_{\lambda 3}), \\ \Delta \lambda_2 &= \frac{1}{n} (\beta e_{\lambda 2} + \bar{\mu} e_{\lambda 1} + \bar{\beta} e_{\lambda 3}), \\ \Delta \lambda_3 &= \frac{1}{n} (\beta e_{\lambda 3} + \bar{\mu} e_{\lambda 1} + \bar{\beta} e_{\lambda 2}), \end{aligned} \quad (3.2.16)$$

donde $n = 3$, $\lambda = x, y$. Por lo tanto, la relación entre los errores de sincronización y los errores de seguimiento de los robots se pueden reducir a la siguiente expresión

$$\begin{aligned} [\Delta x_1, \Delta x_2, \Delta x_3]^T &= (1/n) \mathbf{S}_{\mathbf{G}} [e_{x1}, e_{x2}, e_{x3}]^T, \\ [\Delta y_1, \Delta y_2, \Delta y_3]^T &= (1/n) \mathbf{S}_{\mathbf{G}} [e_{y1}, e_{y2}, e_{y3}]^T, \end{aligned} \quad (3.2.17)$$

donde la matriz de sincronización está dada por

$$\mathbf{S}_{\mathbf{G}} = \begin{pmatrix} \mu & \bar{\mu} & \bar{\mu} \\ \bar{\mu} & \beta & \bar{\beta} \\ \bar{\mu} & \bar{\beta} & \beta \end{pmatrix}. \quad (3.2.18)$$

Por lo tanto, la salida del controlador a nivel del clúster (3.2.12) y los errores de sincronización (3.2.17) son expresiones equivalentes. La matriz de sincronización para $n = 2$ está dada por las 2 últimas columnas y filas de (3.2.18).

Note que, si se requiere un movimiento sincronizado entre los robots móviles, es necesario asignar un valor diferente de cero a las ganancias $\bar{\mu}, \bar{\beta}$. Por otro lado, si no se requiere un movimiento sincronizado, entonces se tiene que los errores de sincronización son proporcionales a los errores de seguimiento, esto es $\Delta x_i \propto e_{xi}$, $\Delta y_i \propto e_{yi}$, con $i = 1, 2, 3$.

Para garantizar que $\mathbf{S}_{\mathbf{G}}$ es no singular, con $\bar{\mu} \neq 0$ y $\bar{\beta} \neq 0$, por simplicidad y sin pérdida de generalidad, considere las siguientes asignaciones: $\mu = \beta \in \mathbb{R}^+$ y $\bar{\mu} = \bar{\beta} \in \mathbb{R} - \{0\}$ implicando, de (3.2.14), que $K_D = K_H$ y $K_C \neq K_H$. Por tal motivo, la matriz $\mathbf{S}_{\mathbf{G}}$ es no singular cuando la siguiente condición se satisface

$$\bar{\beta} \in (-\beta/2, 0) \cup (0, \beta). \quad (3.2.19)$$

A diferencia de [10], note que en este trabajo, los errores de sincronización $\Delta x_i, \Delta y_i$, con $i = 1, \dots, n$, están definidos como una suma ponderada de todos los errores de seguimiento e_{xi}, e_{yi} , respectivamente. Según el valor de las ganancias en (3.2.14)

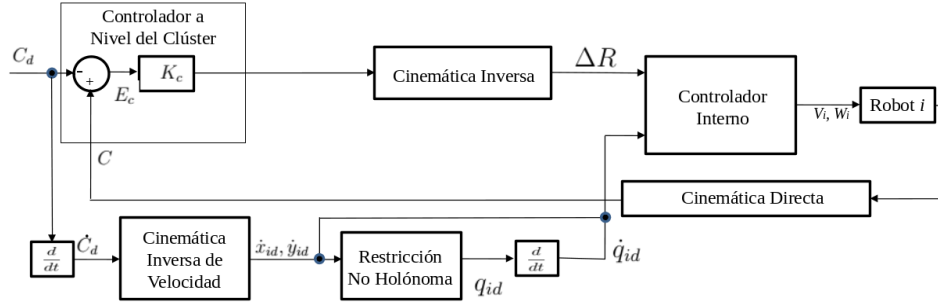


Figura 3.2: Diagrama de bloques del esquema propuesto. $\mathbf{C}_d = [X_{cd}, Y_{cd}, D_{xd}, D_{yd}, H_{xd}, H_{yd}]^T$, $\mathbf{C} = [X_c, Y_c, D_x, D_y, H_x, H_y]^T$, $\Delta \mathbf{R} = [\Delta x_i, \Delta y_i]^T$, con $i = 1, 2, 3$.

y (3.2.19), las ganancias $\bar{\mu}, \bar{\beta}$ pueden ser positivas o negativas, entonces, el concepto de error de sincronización es el presentado por Sun et al. en [10]. Además, no solo se necesitan los errores de seguimiento de los robots vecinos, sino que también son necesarios el resto de errores de seguimiento.

Un control de posición sin sincronización puede tener éxito en el problema de seguimiento de trayectoria, pero en general, no se garantiza lograr el control de formación durante todo el movimiento [10].

3.2.4 Diseño del controlador a nivel de los robots

Como se mencionó, a partir de los errores de sincronización (3.2.17), se obtienen las entradas de control para cada robot mediante el uso de otra ley de control. Similar a [53] y [55], en este trabajo, se implementa la ley de control para los robots móviles tipo diferencial considerando el punto medio del eje de las ruedas, descrita en la Sección 2.2.1. Si la orientación deseada para cada robot móvil tipo (2,0) está dada por (2.1.7), entonces se resuelve el problema de seguimiento de trayectoria, con convergencia a cero de los errores de seguimiento y de orientación, para todos los robots móviles, como se muestra a continuación. El diagrama de bloques del esquema propuesto se muestra en la Fig. 3.2.

Observación 3.3: *Note que el controlador implementado en este trabajo, a nivel de los robots, no es el único que se puede utilizar. Haciendo las adaptaciones correspondientes, se puede implementar un controlador diferente. En [56], se implementa este enfoque de sincronización utilizando, en el espacio de los robots, el controlador descrito en la Sección 2.2.2. Con ese controlador, no se requiere que la orientación de los robots móviles alcancen su valor deseado.*

A partir de la ley de control para un robot móvil tipo (2,0) descrita en 2.2.1, realizando la adaptación de cambiar e_{xi}, e_{yi} por $\Delta x_i, \Delta y_i$, para utilizarla en el esquema

propuesto, considere las señales virtuales ν_{xi} , ν_{yi} , dadas de la forma

$$\nu_{xi} = \dot{x}_{id} - k_{xi}\Delta x_i, \quad \nu_{yi} = \dot{y}_{id} - k_{yi}\Delta y_i, \quad \text{con } i = 1, 2, 3. \quad (3.2.20)$$

En este trabajo, como en [58], los errores de sincronización (3.2.17) están dados de manera similar a los términos de acoplamiento, presentados en [57], donde se usan el resto de errores de seguimiento y no solo el de los 2 vecinos más cercanos. Es fácil probar que la matriz $\mathbf{S}_{\mathbf{G}}$ proporciona una equivalencia entre los errores de sincronización y los términos de acoplamiento mutuo. Con estos términos, se puede resolver el problema de sincronización entre los robots proveiendo más robustez a la formación ante perturbaciones.

Por otra parte, la velocidad rotacional no se modifica y está dada por (2.2.5), utilizando el error en la orientación angular e_{qi} . Note que, en contraste con los errores de seguimiento (2.2.6) y (2.2.7), para las orientaciones de los robots móviles, no se definen errores de sincronización. Como ya se demostró en (2.2.9), la variable q_i , con $i = 1, \dots, n$, converge exponencialmente a su valor deseado.

A continuación considere las siguientes suposiciones en cuanto al esquema de sincronización:

- **Suposición 3.1** *Las entradas de control, $(V_i(t), W_i(t))$, están acotadas $\forall t \geq 0$, esto es $0 \leq V_i(t) \leq V_{max}$ y $|W_i(t)| \leq W_{max}$, con $i = 1, \dots, n$, donde $V_{max}, W_{max} \in \mathbb{R}^+$.*
- **Suposición 3.2** *La velocidad rotacional deseada para el i -ésimo robot móvil, está acotada y es diferente de cero $\forall t \geq 0$, esto es $\dot{q}_{id}(t) = \omega_{id}(t) \neq 0$ y $|\omega_{id}(t)| \leq \omega_{max}$, con $i = 1, \dots, n$, donde $\omega_{max} \in \mathbb{R}^+$.*

Sustituyendo (2.1.1), (2.1.2), (2.2.2) y (3.2.20) en las derivadas temporales de (2.2.6) y (2.2.7), se obtiene la dinámica de los errores de seguimiento, dada por

$$\begin{aligned} \dot{e}_{xi} = & V_i \cos(q_i) - \dot{x}_{id} = [\dot{x}_{id} - k_{xi}\Delta x_i] \cos^2(q_i) \\ & + [\dot{y}_{id} - k_{yi}\Delta y_i] \sin(q_i) \cos(q_i) - \dot{x}_{id}, \end{aligned} \quad (3.2.21)$$

$$\begin{aligned} \dot{e}_{yi} = & V_i \sin(q_i) - \dot{y}_{id} = [\dot{x}_{id} - k_{xi}\Delta x_i] \cos(q_i) \sin(q_i) \\ & + [\dot{y}_{id} - k_{yi}\Delta y_i] \sin^2(q_i) - \dot{y}_{id}, \end{aligned} \quad (3.2.22)$$

con $i = 1, \dots, n$. Considere la siguiente notación $C_i = \cos(q_i)$, $S_i = \sin(q_i)$. (3.2.21) y (3.2.22) pueden reescribirse, respectivamente, como

$$\begin{aligned} \dot{e}_{xi} = & -k_{xi}\Delta x_i C_i^2 - k_{yi}\Delta y_i S_i C_i + \dot{x}_{id}(C_i^2 - 1) + \dot{y}_{id}(S_i C_i) \\ = & -k_{xi}\Delta x_i C_i^2 - k_{yi}\Delta y_i S_i C_i + [\dot{y}_{id}(S_i C_i) - \dot{x}_{id} S_i^2] \\ = & -k_{xi}\Delta x_i C_i^2 - k_{yi}\Delta y_i S_i C_i + r_x(\dot{x}_{id}, \dot{y}_{id}, S_i, C_i), \end{aligned} \quad (3.2.23)$$

$$\begin{aligned} \dot{e}_{yi} = & -k_{yi}\Delta y_i S_i^2 - k_{xi}\Delta x_i C_i S_i + \dot{x}_{id}(C_i S_i) + \dot{y}_{id}(S_i^2 - 1) \\ = & -k_{yi}\Delta y_i S_i^2 - k_{xi}\Delta x_i S_i C_i + [\dot{x}_{id}(C_i S_i) - \dot{y}_{id} C_i^2] \\ = & -k_{yi}\Delta y_i S_i^2 - k_{xi}\Delta x_i S_i C_i + r_y(\dot{x}_{id}, \dot{y}_{id}, S_i, C_i). \end{aligned} \quad (3.2.24)$$

Sin pérdida de generalidad, tome en cuenta la ganancia $k = k_{xi} = k_{yi}$. Entonces, la dinámica de los errores de seguimiento y la dinámica de los errores de orientación, en lazo cerrado, están dadas por

$$\dot{e}_{xi} = -k\Delta x_i C_i^2 - k\Delta y_i S_i C_i + r_x(\dot{x}_{id}, \dot{y}_{id}, S_i, C_i), \quad (3.2.25)$$

$$\dot{e}_{yi} = -k\Delta y_i S_i^2 - k\Delta x_i S_i C_i + r_y(\dot{x}_{id}, \dot{y}_{id}, S_i, C_i), \quad (3.2.26)$$

$$\dot{e}_{qi} = -k_{qi} e_{qi}. \quad (3.2.27)$$

El siguiente lema establece las condiciones necesarias y suficientes bajo las cuales la dinámica de los errores de seguimiento y de orientación, para una formación compuesta por 3 robots móviles tipo diferencial, son asintóticamente estables, considerando el caso particular cuando tanto la orientación deseada como su derivada temporal correspondiente son iguales para todos los robots.

Lema 1: Considere tres robots móviles tipo diferencial (2.1.1)-(2.1.3). Suponga que la trayectoria, orientación y forma para el clúster deseado (3.2.6) son variantes en el tiempo de tal forma que la trayectoria deseada y orientación deseada para cada robot móvil $(x_{id}(t), y_{id}(t), q_{id}(t))$ satisfacen la restricción no holónoma (2.1.7) $\forall t \geq 0$. Suponga que tanto la orientación deseada como la velocidad rotacional deseada $(q_d(t), \omega_d(t))$, con $i = 1, 2, 3$, son iguales para todos los robots móviles $\forall t \geq 0$. Si se utiliza el controlador a nivel de la formación (3.2.11) y el controlador a nivel de los robots (2.2.2)-(2.2.1), donde

- $(V_{id}(t), W_{id}(t))$ están impuestas por (2.1.8);
- las *Suposiciones 3.1* y *3.2* se toman en cuenta;
- $k, k_{qi} > 0$, con $i = 1, 2, 3$;
- $K_C, K_D, K_H > 0$, $K_D = K_H$, $K_C \neq K_H$, $\mu = \beta \in \mathbb{R}^+$, $\bar{\mu} = \bar{\beta}$, donde $\bar{\beta}$ satisface (3.2.19);

entonces, la dinámica de los errores de seguimiento (3.2.25)-(3.2.26) converge asintóticamente a cero. Además, tanto el vector de errores del clúster (3.2.11) como los errores de sincronización (3.2.17) también convergen asintóticamente a cero. Así mismo, los errores de orientación de los robots (2.2.8) son exponencialmente estables.

Demostración: Como se muestra en (2.2.9), con $k_{qi} > 0$, el error de orientación para cada robot móvil tipo diferencial converge exponencialmente a cero. Esto implica que $q_i \rightarrow q_{id}$, $C_i \rightarrow \cos(q_{id})$, $S_i \rightarrow \sin(q_{id})$, con $i = 1, 2, 3$. Después de un transitorio, los términos $r_x(\dot{x}_{id}, \dot{y}_{id}, S_i, C_i)$ y $r_y(\dot{x}_{id}, \dot{y}_{id}, S_i, C_i)$ dados en (3.2.23) y (3.2.24), respectivamente, satisfacen la restricción no holónoma (2.1.6), esto es

$$r_x \rightarrow \dot{y}_{id}[\sin(q_{id}) \cos(q_{id})] - \dot{x}_{id} \sin^2(q_{id}) = 0,$$

$$r_y \rightarrow \dot{x}_{id}[\sin(q_{id}) \cos(q_{id})] - \dot{y}_{id} \cos^2(q_{id}) = 0.$$

Por lo tanto, los términos r_x, r_y son señales desvanecientes del sistema (3.2.25)-(3.2.27) dado que su convergencia a cero sólo depende de los errores de orientación e_{qi} , con $i = 1, \dots, n$.

Debido a que la convergencia de los errores de orientación es muy rápida (convergencia exponencial) e independiente de los errores de seguimiento, y bajo la condición de que la orientación deseada para todos los robots móviles es la misma, esto es $q_{id} = q_d$, y $\omega_{id} = \omega_d$, el sistema (3.2.25)-(3.2.26) se puede reescribir como el siguiente subsistema

$$\dot{e}_{xi} = -k\Delta x_i C^2 - k\Delta y_i SC, \quad (3.2.28)$$

$$\dot{e}_{yi} = -k\Delta y_i S^2 - k\Delta x_i CS, \quad (3.2.29)$$

donde $C = \cos(q_d)$, $S = \sin(q_d)$, con $i = 1, 2, 3$. Aunque (3.2.28)-(3.2.29) es un sistema lineal variante en el tiempo, en este caso particular, es posible usar un cambio de coordenadas globalmente definido para obtener un sistema equivalente. Las nuevas variables de estado están dadas por

$$e_{li} = \Delta x_i C + \Delta y_i S, \quad (3.2.30)$$

$$e_{pi} = -\Delta x_i S + \Delta y_i C, \quad (3.2.31)$$

con $i = 1, \dots, n$. Geométricamente, e_{li} y e_{pi} son los errores de seguimiento sobre los ejes X_i y Y_i , del marco de referencia del i -ésimo robot, respectivamente.

Por lo tanto, la dinámica de dichos errores de seguimiento está dada por

$$\begin{aligned} \dot{e}_{l1} &= e_{p1}\omega_d - \frac{k}{n}\mu e_{l1} - \frac{k}{n}\bar{\mu}(e_{l2} + e_{l3}), \\ \dot{e}_{l2} &= e_{p2}\omega_d - \frac{k}{n}\beta e_{l2} - \frac{k}{n}\bar{\mu}e_{l1} - \frac{k}{n}\bar{\beta}e_{l3}, \\ \dot{e}_{l3} &= e_{p3}\omega_d - \frac{k}{n}\beta e_{l3} - \frac{k}{n}\bar{\mu}e_{l1} - \frac{k}{n}\bar{\beta}e_{l2}, \\ \dot{e}_{pi} &= -e_{li}\omega_d, \end{aligned} \quad (3.2.32)$$

con $i = 1, 2, 3$. Tomando en cuenta la función candidata de Lyapunov dada por

$$V(e_{li}, e_{pi}) = \frac{1}{2} \sum_{i=1}^n (e_{li}^2 + e_{pi}^2), \quad (3.2.33)$$

con $n = 3$, es fácil ver que la derivada con respecto al tiempo de $V(e_{li}, e_{pi})$ está dada por

$$\dot{V}(e_{li}, e_{pi}) = [e_{l1} \ e_{l2} \ e_{l3}] \left(-\frac{k}{n} \right) \mathbf{S}_G [e_{l1} \ e_{l2} \ e_{l3}]^T, \quad (3.2.34)$$

donde \mathbf{S}_G se expresa en (3.2.18). Debido a que S_G es una matriz definida positiva, según (3.2.19), $\dot{V}(e_{li}, e_{pi})$ es definida negativa. Sin embargo, dado que los términos e_{pi} no se

presentan en \dot{V} , garantizar que \dot{V} es definida negativa implica que el sistema (3.2.32) es solamente estable.

Note que $\ddot{V}(e_l, e_p)$ está acotada, dado que la *Suposición 3.2* se satisface; por tanto $\dot{V}(e_l, e_p)$ es uniformemente continua. Considerando el lema de Barbalat [59], se tiene que $\dot{V}_3(e_l, e_p) \rightarrow 0$ cuando $t \rightarrow \infty$ y consecuentemente $e_{li} \rightarrow 0$ con $i = 1, 2, 3$. De las primeras tres ecuaciones de (3.2.32) y bajo la suposición de que $\omega_d \neq 0$, está claro que $e_{pi} \rightarrow 0$, por lo tanto los errores de seguimiento de todos los robots convergen a cero. Entonces, el origen del sistema (3.2.25)-(3.2.27) es asintóticamente estable.

Finalmente, dado que la matriz S_G es no singular, garantizar que e_{xi}, e_{yi} converjan a cero implica que $\Delta x_i, \Delta y_i$ converjan a cero también. De manera similar, dado que la matriz \mathbf{K} es no singular (3.2.3), también el vector de errores del clúster (3.2.11) converge asintóticamente a cero. ■

El siguiente lema establece las condiciones necesarias y suficientes bajo las cuales la dinámica de los errores de seguimiento y de orientación, para una formación compuesta por 3 robots móviles tipo diferencial, son asintóticamente estables, considerando que tanto la orientación deseada como su derivada temporal correspondiente son diferentes para todos los robots. En otras palabras, se resuelve el caso general del problema de control de formación variante en el tiempo.

Lema 2: Considere tres robots móviles tipo diferencial (2.1.1)-(2.1.3). Suponga que la trayectoria, orientación y forma para el clúster deseado (3.2.6) son variantes en el tiempo de tal forma que la trayectoria deseada y orientación deseada para cada robot móvil $(x_{id}(t), y_{id}(t), q_{id}(t))$ satisfacen la restricción no holónoma (2.1.7) $\forall t \geq 0$. Suponga que tanto la orientación deseada como la velocidad rotacional deseada $(q_d(t), \omega_d(t))$, con $i = 1, 2, 3$, pueden ser distintas para todos los robots móviles $\forall t \geq 0$. Si se utiliza el controlador a nivel de la formación (3.2.11) y el controlador a nivel de los robots (2.2.2)-(2.2.1), donde

- $(V_{id}(t), W_{id}(t))$ están impuestas por (2.1.8);
- las *Suposiciones 3.1* y *3.2* se toman en cuenta;
- $k, k_{qi} > 0$, con $i = 1, 2, 3$;
- $K_C, K_D, K_H > 0$, $K_D = K_H$, $K_C \neq K_H$, $\mu = \beta \in \mathbb{R}^+$, $\bar{\mu} = \bar{\beta}$, donde $\bar{\beta}$ satisface $-(L-1)\beta < \bar{\beta} < \frac{L-1}{2}\beta$, con $1 < L \leq 3$;

entonces, la dinámica de los errores de seguimiento (3.2.25)-(3.2.26) converge asintóticamente a cero. Además, tanto el vector de errores del clúster (3.2.11) como los errores de sincronización (3.2.17) también convergen asintóticamente a cero. Así mismo, los errores de orientación de los robots (2.2.8) son exponencialmente estables.

Demostración: Aplicando una metodología similar al *Lema 1*, bajo la condición de que todas las orientaciones deseadas de los robots son diferentes entre ellas, después de

un breve transitorio, el sistema (3.2.25)-(3.2.27) se puede reducir al subsistema lineal variante en el tiempo dado por

$$\dot{e}_{xi}(t) = -kg_{ci}(t)\Delta x_i - k\Delta y_i g_i(t), \quad (3.2.35)$$

$$\dot{e}_{yi}(t) = -kg_{si}(t)\Delta y_i - k\Delta x_i g_i(t), \quad (3.2.36)$$

donde

$$\begin{aligned} g_{ci}(t) &= \cos^2[q_{id}(t)], & g_{si}(t) &= \sin^2[q_{id}(t)], \\ g_i(t) &= \sin[q_{id}(t)] \cos[q_{id}(t)] = \frac{1}{2} \sin[2q_{id}(t)], \end{aligned} \quad (3.2.37)$$

con $i = 1, 2, 3$. Las funciones previas satisfacen

$$0 \leq g_{ci}(t) \leq 1, \quad 0 \leq g_{si}(t) \leq 1, \quad |g_i(t)| \leq 1/2, \quad \forall t \geq 0. \quad (3.2.38)$$

Las funciones $g_i(t)$ son continuamente diferenciables y de la *Suposición 3.2*, sus derivadas temporales satisfacen

$$-\omega_{max} \leq \dot{g}_i(t) = \omega_{id}(t) \cos[2q_{id}(t)] \leq \omega_{max}, \quad \forall t \geq 0, \quad (3.2.39)$$

con $i = 1, 2, 3$. Considere el vector de errores de seguimiento de los robots dado como $e = [e_{x1}, e_{y1}, e_{x2}, e_{y2}, e_{x3}, e_{y3}]^T$, y la función candidata de Lyapunov dada por

$$V(t, e) = \sum_{i=1}^n ([1 + g_i(t)] (e_{xi}^2 + e_{yi}^2)). \quad (3.2.40)$$

Es fácil ver que

$$\frac{1}{2} \sum_{i=1}^n (e_{xi}^2 + e_{yi}^2) \leq V(t, e) \leq \frac{3}{2} \sum_{i=1}^n (e_{xi}^2 + e_{yi}^2), \quad \forall e \in \mathbb{R}^{2n}. \quad (3.2.41)$$

Por lo tanto, $V(t, e)$ es definida positiva, decreciente, y radialmente no acotada [59]. La derivada de $V(t, e)$ a lo largo de las trayectorias del sistema está dada por

$$\begin{aligned} \dot{V}(t, e) &= \sum_{i=1}^n (2[1 + g_i(t)]e_{xi}\dot{e}_{xi} \\ &\quad + 2[1 + g_i(t)]e_{yi}\dot{e}_{yi} + \dot{g}_i(t)[e_{xi}^2 + e_{yi}^2]). \end{aligned} \quad (3.2.42)$$

Para garantizar que $\dot{V}(t, e)$ es definida negativa, se debe satisfacer la siguiente desigualdad

$$\dot{V}(t, e) \leq -W_3(e),$$

donde $W_3(e)$ es una función definida positiva invariante en el tiempo.

Sustituyendo (3.2.35) y (3.2.36) en (3.2.42), y usando los límites inferiores de las desigualdades (3.2.38) y (3.2.39), se tiene

$$\begin{aligned} -W_3(e) &= \left(\frac{k}{2}\right) \sum_{i=1}^n (e_{xi}\Delta y_i + e_{yi}\Delta x_i - \omega_{max}(e_{xi}^2 + e_{yi}^2)) \\ &= -e^T Q_3 e, \end{aligned} \quad (3.2.43)$$

donde $Q_3 =$

$$\begin{pmatrix} \omega_{max} & \frac{-k\bar{\mu}}{2n} & 0 & \frac{-k\bar{\mu}}{2n} & 0 & \frac{-k\bar{\mu}}{2n} \\ \frac{-k\bar{\mu}}{2n} & \omega_{max} & \frac{-k\bar{\mu}}{2n} & 0 & \frac{-k\bar{\mu}}{2n} & 0 \\ 0 & \frac{-k\bar{\mu}}{2n} & \omega_{max} & \frac{-k\bar{\beta}}{2n} & 0 & \frac{-k\bar{\beta}}{2n} \\ \frac{-k\bar{\mu}}{2n} & 0 & \frac{-k\bar{\beta}}{2n} & \omega_{max} & \frac{-k\bar{\beta}}{2n} & 0 \\ 0 & \frac{-k\bar{\mu}}{2n} & 0 & \frac{-k\bar{\beta}}{2n} & \omega_{max} & \frac{-k\bar{\beta}}{2n} \\ \frac{-k\bar{\mu}}{2n} & 0 & \frac{-k\bar{\beta}}{2n} & 0 & \frac{-k\bar{\beta}}{2n} & \omega_{max} \end{pmatrix}, \quad (3.2.44)$$

con $n = 3$. Note que, la matriz requerida Q_n para $n = 2$ es la que se compone por las 4 columnas y filas de (3.2.44).

Para obtener una función definida positiva invariante en el tiempo ($W_3(e)$), la matriz Q_3 debe ser definida positiva. Por lo tanto, es necesario analizar las propiedades de Q_3 para presentar el intervalo correspondiente de las ganancias que componen Q_3 .

Nuevamente, si $\bar{\mu} = \bar{\beta} = 0$ no se requiere una sincronización entre los robots móviles, obteniendo 3 sistemas desacoplados. Debido a que ω_{max} es un parámetro que depende de la trayectoria deseada, es fácil asignar un valor suficientemente grande a ω_{max} para garantizar que Q_3 sea definida positiva. De hecho, el parámetro ω_{max} provee la *velocidad de convergencia* del sistema [59].

Del segundo menor principal de la matriz Q_3 , para garantizar que su determinante sea positivo, se debe satisfacer la siguiente desigualdad

$$\omega_{max} > k\bar{\mu}/(2n), \quad \text{con } n = 3. \quad (3.2.45)$$

Por lo tanto, para omitir el parámetro ω_{max} , tome en cuenta la siguiente asignación

$$\omega_{max} = Lk\bar{\beta}/(2n), \quad (3.2.46)$$

con $\mu = \beta$ (3.2.19), donde $L > 1$ es un factor de escalamiento, el cual no puede ser muy grande debido a que solo se requiere satisfacer la desigualdad en (3.2.45).

Note que Q_3 es una matriz simétrica compuesta por las ganancias $k, \mu, \beta \in \mathbb{R}^+$, $\bar{\mu} = \bar{\beta} \neq 0$, entonces los valores propios de Q_3 son todos valores reales. Tomando en cuenta (3.2.19) y (3.2.46), los valores propios de Q_3 están dados por

$$\begin{aligned} \lambda_1 = \lambda_2 &= [\bar{\beta}k + (L-1)\beta k]/(2n), \\ \lambda_3 = \lambda_4 &= [-\bar{\beta}k + (L+1)\beta k]/(2n), \\ \lambda_5 &= [2\bar{\beta}k + (L+1)\beta k]/(2n), \\ \lambda_6 &= -[2\bar{\beta}k + (1-L)\beta k]/(2n). \end{aligned} \quad (3.2.47)$$

La solución que satisface que todos los valores propios de Q_3 son positivos, tomando $\bar{\beta} > 0$, es

$$\bar{\beta} < \frac{L-1}{2}\beta. \quad (3.2.48)$$

De otra forma, tomando $\bar{\beta} < 0$, hay 2 intervalos solución que satisfacen que todos los valores propios de Q_3 son positivos, dado que el intervalo menor depende del intervalo L , éstos son

$$-(L-1)\beta < \bar{\beta}, \text{ con } L \leq 3; \quad -\frac{L+1}{2}\beta < \bar{\beta}, \text{ con } L \geq 3. \quad (3.2.49)$$

Como ya se mencionó, se requiere que L sea grande para satisfacer (3.2.45), por lo tanto, se desprecia el intervalo para $L \geq 3$.

Para considerar valores positivos y negativos de la ganancia $\bar{\beta}$ y para garantizar que la matriz Q_3 sea definida positiva, es necesario tomar la unión del intervalo (3.2.48) con (3.2.49) considerando $L \leq 3$, esto es

$$\bar{\beta} \in (-[L-1]\beta, 0) \cup (0, [L-1]\beta/2), \text{ donde } 1 < L \leq 3. \quad (3.2.50)$$

De manera similar para $\bar{\mu}$. La matriz \mathbf{S}_G puede ser definida positiva de acuerdo con la desigualdad expresada en (3.2.19). Por lo tanto, tomando $L = 2$ en la solución para el caso variante en el tiempo (3.2.50), el intervalo solución está dado por

$$\bar{\beta} \in (-\beta, 0) \cup (0, \beta/2). \quad (3.2.51)$$

Entonces, considerar $L = 2$ es una asignación apropiada.

Realizando un procedimiento similar para el sistema variante en el tiempo, con $n = 2$, el intervalo de solución está dado por

$$-(L-1)\beta < \bar{\beta} < (L-1)\beta, \text{ con } L > 1.$$

El intervalo solución para el sistema invariante en el tiempo, con $n = 2$, está dado por

$$-\beta < \bar{\beta} < \beta.$$

Nuevamente, considerar $L = 2$ es una asignación apropiada.

Con este resultado se puede concluir que existe una solución, eligiendo apropiadamente los valores para las ganancias $k, \mu, \beta, \bar{\mu}, \bar{\beta}$, y el parámetro ω_{max} , que hace que Q_3 sea una matriz definida positiva. Por lo tanto, el origen del sistema (3.2.25)-(3.2.27) es asintóticamente estable.

Finalmente, dado que la matriz \mathbf{S}_G es no singular, también los errores de sincronización (3.2.17) convergen asintóticamente a cero. Así mismo, debido a que la matriz \mathbf{K} es no singular (3.2.3), los errores de la formación (3.2.11) también convergen asintóticamente a cero. ■

Observación 3.4: Sin tomar en cuenta la topología del flujo de información entre los robots móviles, una posible manera de definir un clúster compuesto por más de 3 robots móviles es implementando un super clúster, esto es, un clúster compuesto por varios clústers que están formados por 2 y 3 robots (formaciones anidadas utilizando las variables de forma presentadas en este trabajo). Asimismo, implementando el esquema de sincronización propuesto en un super clúster, todos los errores de seguimiento, de sincronización y de orientación de los robots también son asintóticamente estables para el caso general, como se demuestra en el Lema 2.

3.3 Evasión y navegación para el clúster

A partir del método de potenciales artificiales se obtiene la posición y orientación deseada para el clúster $[X_{cd}, Y_{cd}, Q_{cd}]^T$, las cuales se deben seguir para resolver el problema de navegación evadiendo obstáculos y converger a la meta. Como se describe en la Sección 2.5, el método de potenciales artificiales requiere en todo momento las posiciones y velocidades del clúster, la meta y los obstáculos, para generar una fuerza total $F(p, p_{obs}, V_{RO})$ y un ángulo de orientación deseado θ_f . En este trabajo se considera al clúster como una entidad simple, por lo tanto, se emplea el esquema descrito utilizando la posición y velocidad del centroide del clúster, y no considerando cada robot de manera individual para después obtener un promedio de las fuerzas totales. Al igual que en (2.5.20), la velocidad lineal deseada para la navegación de la formación está dada por

$$V_d = K_v d_g.$$

Las velocidades cartesianas deseadas para el clúster están dadas de la siguiente manera

$$\dot{X}_{cd} = V_d \cos(Q_{cd}), \quad \dot{Y}_{cd} = V_d \sin(Q_{cd}), \quad (3.3.1)$$

donde Q_{cd} es la orientación deseada del clúster, la cual se describe más adelante. Las posiciones cartesianas deseadas para la formación se obtienen mediante un bloque integrador de las señales anteriores.

Similar a [26], la velocidad angular deseada para el clúster tiene la forma de un controlador PI, es decir

$$\dot{Q}_{cd} = -K_s E_{Q_c} - K_{s0} \int_0^t E_{Q_c}(\lambda) d\lambda, \quad (3.3.2)$$

donde K_s y K_{s0} son ganancias positivas. $E_{Q_c} = (Q_c - \theta_f)$ es el error en la orientación del clúster. La orientación del clúster está descrita por (3.1.3). Para obtener la orientación deseada del clúster Q_{cd} , se aplica un bloque integrador a la señal \dot{Q}_{cd} . Utilizar esta integral para obtener la orientación deseada de la formación en lugar de considerar $Q_{cd} = \theta_f$, permite que dicha orientación no sea discontinua al entrar o salir de la zona de influencia de algún obstáculo o al detectar un acercamiento con éste. Además, debido a la integral del error en la orientación, es posible demostrar que Q_{cd} tiende a θ_f en un tiempo finito eligiendo apropiadamente la ganancia K_{s0} . Utilizar K_{s0} grande implica en un recorrido muy curvo no deseable.

Debido a las restricciones no holonómicas de los robots móviles tipo diferencial considerados, las orientaciones deseadas para cada uno de éstos están descritas en (2.1.7), es por eso que se utiliza (3.2.7) para determinar las velocidades cartesianas deseadas de cada robot y finalmente obtener las orientaciones deseadas correspondientes. Es muy importante notar que cuando se quiere que el clúster se mueva apuntando en la dirección de su velocidad traslacional (como si fuera un unicycle más grande), solo las orientaciones deseadas para los robots móviles 2 y 3 son iguales ya que están ubicados de forma colineal. Las orientaciones deseadas de clúster y del robot 1 están desfasadas en el tiempo.

Capítulo 4

Plataforma experimental

En este capítulo se describen brevemente todos y cada uno de los componentes de la plataforma experimental utilizada en este trabajo. Se presentan las características del sistema de visión y sensado, de la computadora central que realiza las leyes de control y se muestran los robots móviles tipo unicycle de la marca *ROBOTIS* que se utilizaron. Finalmente, se da una breve descripción de ROS (Robot Operating System).

4.1 Visión y sensado

La detección de la posición y orientación de los robots son parte fundamental para el desarrollo de los experimentos y cálculos matemáticos. Por tal motivo estos datos deben de ser obtenidos de la manera más precisa posible. Para lograr este objetivo, durante el experimento, se cuenta con el sistema de visión y sensado especializado en la obtención de captura de movimiento vía señales infrarrojas auxiliares conocido como *OptiTrack*, sistema desarrollado por la compañía *NaturalPoint* [60] que determina la posición y orientación de objetos en un espacio tridimensional limitado por el rango de trabajo de la constelación de cámaras que constituyen el sistema a un máximo de 120 cuadros por segundo. El sistema *OptiTrack* implementado está conformado por las siguientes partes:

- 8 cámaras Flex 13.
- 2 OptiHubs 2.
- Marcadores reflejantes
- Equipo de calibración OptiWand y escuadra de volumen.
- Llave USB de licencia.
- Software Motive.

Una de las grandes ventajas que ofrecen las cámaras Flex 13, Fig. 4.1, es su preprocesamiento interno al contar con un microprocesador que no solo ayuda al proceso de captura de imágenes, sino que además realiza cálculos de visión artificial entregando al software “Motive” datos de la posición de un conjunto de marcadores reflejantes filtrados, lo que se refleja significativamente en la capacidad de procesamiento aligerando los cálculos en la máquina de control.



Figura 4.1: Cámaras Flex 13 y OptiHubs 2.

El sistema de visión utiliza una serie de marcadores reflejantes de señales infrarrojas configurados en patrones geométricos irregulares con la finalidad de distinguir a los objetos rígidos a sensor. En el caso de utilizar múltiples objetos es necesario que cada objeto tenga un patrón único para que el sistema de visión los pueda distinguir. Cada patrón es sensado para obtener el centroide y la orientación de cada objeto.

“Motive” es un programa que sirve como interfaz de control para el sistema *OptiTrack*, (véase la Fig. 4.2), con ayuda de un servidor VRPN (siglas en inglés Virtual Reality Peripheral Network), y utilizando el puerto local 3883 es posible enviar las coordenadas de postura de los robots móviles hacia la computadora central, por medio de un router, donde se realiza el cálculo de las señales de control que son enviadas a todos y cada uno de los robots móviles para obtener la evolución del sistema, retroalimentando nuevamente los estados del sistema cerrando el lazo de control. La Fig. 4.3 muestra el diagrama a bloques del flujo de datos en la plataforma experimental.

4.2 Computadora central

La unidad de procesamiento es una computadora con el sistema operativo Linux con la distribución Ubuntu 20.04 y la versión de ROS Noetic. Esta computadora fue asignada como *Master*, por lo que todos los dispositivos se conectan a ella por medio de un router. En esta unidad se realizan todos los procesos de las leyes de control desde recibir la posiciones de todos los robots móviles que Motive envía y posteriormente enviando las velocidades correspondientes a todos y cada uno de los vehículos a través de la comunicación inalámbrica. Es importante notar que la plataforma experimental cuenta con un sistema de localización absoluta y una computadora central, por lo que se tiene un esquema centralizado que funciona correctamente.

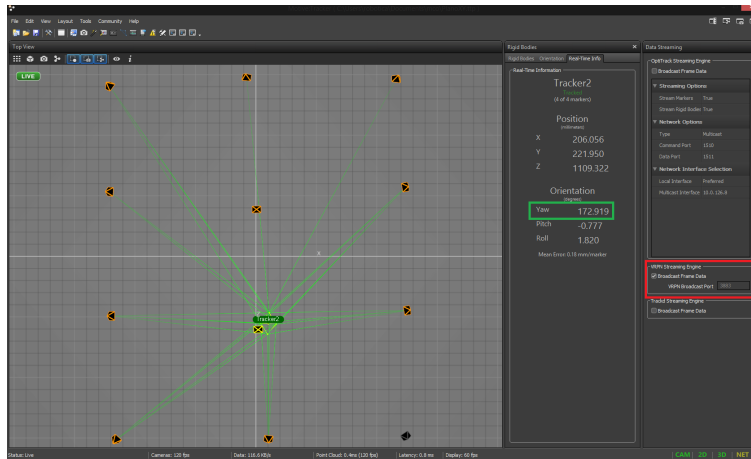


Figura 4.2: Entorno del software Motive.

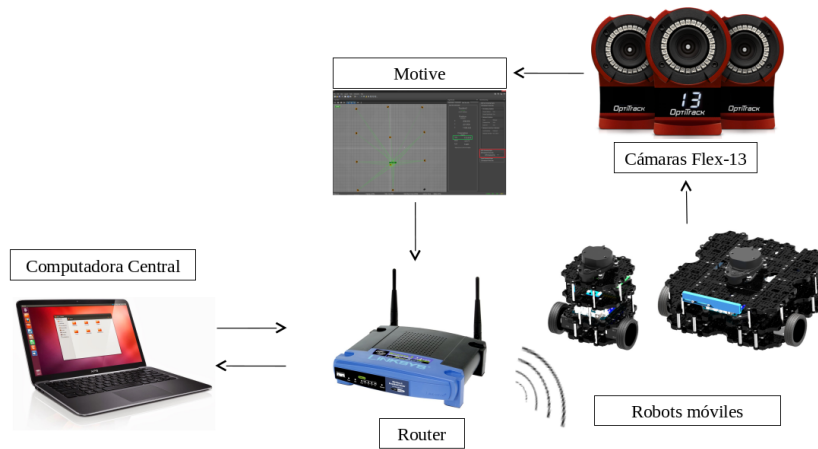


Figura 4.3: Flujo de datos en la plataforma experimental.

4.3 Robots móviles TurtleBot3

Los robots móviles tipo (2,0) utilizados para la ejecución de las pruebas experimentales son los robots Turtlebot3, los cuales son desarrollados por *Open Robotics* en colaboración con *ROBOTIS* y diferentes socios con la finalidad de diseñar un robot que cuente con software, hardware y contenido basados en código abierto. Su uso está principalmente enfocado para la educación, investigación y creación de prototipos. Los Turtlebot3 son la tercera y más reciente versión de la familia de robot Turtlebot, los cuales están estandarizados para la plataforma ROS. Una de las características más importantes de este robot es su diseño modular, que permite que este pueda ser personalizado reconstruyendo las partes mecánicas o añadiendo diferentes sensores.

El Turtlebot3 se encuentra disponible en 2 modelos, Turtlebot3 Burger y Turtlebot3 Waffle, los cuales se muestran en la Fig. 4.4. Ambos modelos están compuestos por

2 servomotores DYNAMIXEL, una batería de polímero de litio, una tarjeta OpenCR 1.0 que sirve como etapa de potencia, un sensor de distancia láser 360° (LIDAR), una computadora Raspberry Pi 3 modelo B, la cual es la que se encarga de conectarse al router vía wi-fi para recibir las señales de control calculadas por la computadora central. Para mayor información sobre las especificaciones de los TurtleBot3 consulte [61].

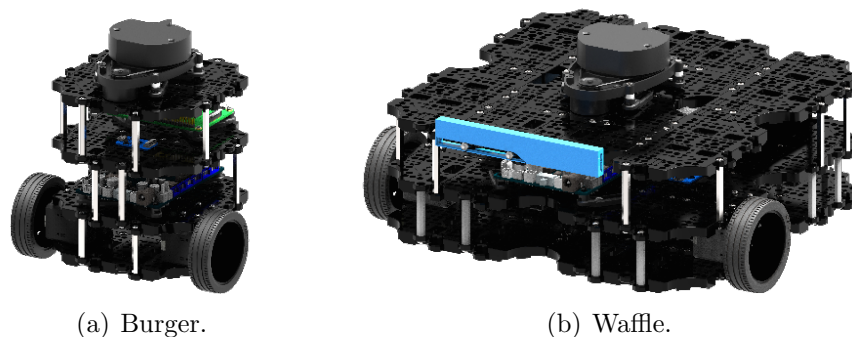


Figura 4.4: Modelos del TurtleBot3.

4.4 ROS (Robot Operating System)

El meta-sistema operativo ROS tiene como objetivo gestionar todos los dispositivos que forman la plataforma experimental [62]. Fue inicialmente desarrollado por el Laboratorio de Inteligencia Artificial de Stanford en el 2007, aunque a partir del 2008, Willow Garage, un instituto de investigación robótica se encarga del desarrollo del sistema operativo bajo una licencia de código abierto. ROS es un entorno de trabajo que funciona como un meta-sistema operativo para el desarrollo de aplicaciones robóticas. Proporciona la gran mayoría de servicios que ofrece un sistema operativo, como son abstracción del hardware, uso de librerías, paso de mensajes y gestión de paquetes.

Su principal ventaja es que está basado en la computación distribuida, la cual utiliza distintos dispositivos organizados y unidos entre sí normalmente por una red de alta velocidad para cumplir objetivos en común, comportándose como un mismo sistema. ROS permite considerar a los robots móviles, computadora central y sistema de localización *OptiTrack* como un solo sistema facilitando la comunicación entre ellos debido a su funcionamiento como sistema distribuido. Los robots móviles considerados en esta plataforma cuentan con ROS, mientras que los datos del sistema de localización son enviados a la unidad de procesamiento a través de ROS. Otra ventaja importante de este sistema operativo es la simplicidad del modelo de comunicación y la gestión de mensajes del sistema, lo que ha hecho que ROS sea un estándar para muchas plataformas robóticas. El funcionamiento de ROS se puede definir como un sistema de arquitectura de grafos, donde cada elemento del software se conecta a un nodo maestro que permite que los elementos puedan comunicarse entre sí para cumplir un objetivo en común.

Capítulo 5

Resultados Experimentales (con robots terrestres)

En este capítulo se presentan los resultados experimentales obtenidos en el laboratorio para el esquema de evasión de obstáculos estáticos y dinámicos para una formación compuesta por 3 robots móviles terrestres tipo (2,0) modelo burger. Primero, se describe la trayectoria deseada que se utilizó en las pruebas de seguimiento para evaluar los controladores a nivel del clúster y a nivel de los robots. También, se muestra la prueba de evasión de obstáculos dinámicos y navegación utilizando el controlador P con compensación y el sistema de visión *OptiTrack* para determinar la posición y orientación de los robots móviles. Se describe detalladamente en qué consistieron las diferentes pruebas y se muestran las gráficas de los resultados.

5.1 Trayectoria deseada

Debido a que una lemniscata requiere mayor dificultad al tener curvas de diferentes radios, la trayectoria deseada elegida para las pruebas fue una lemniscata descrita de la forma

$$X_{cd} = X_0 + a \sin(2\pi\tau/T), \quad Y_{cd} = Y_0 + b \sin(4\pi\tau/T), \quad (5.1.1)$$

$$Q_{cd} = ATAN2(\dot{Y}_{cd}/\dot{X}_{cd}), \quad (5.1.2)$$

donde (X_{cd}, Y_{cd}) es la posición deseada para el centroide del clúster. $a = 1.7$ [m] y $b = 0.7$ [m] son las medidas de los semi-ejes X y Y, respectivamente. $(X_0, Y_0) = (0.2, -0.4)$ es la posición del centro de la lemniscata en metros, T es el periodo de la trayectoria y $\tau = t + \nu T$, con $\nu \in [0, 1)$, es un corrimiento temporal para poder iniciar la prueba en cualquier punto de la trayectoria. Con Q_{cd} variable se quiere que el clúster rote para que cada robot siga una trayectoria concéntrica a la del centroide del clúster cumpliéndose la restricción no holónoma de los robots móviles tipo (2,0).

La derivada temporal de la orientación deseada del clúster está descrita por

$$\dot{Q}_{cd} = \frac{\ddot{Y}_{cd}/\dot{X}_{cd} - \ddot{X}_{cd}\dot{Y}_{cd}/\dot{X}_{cd}^2}{1 + (\dot{Y}_{cd}/\dot{X}_{cd})^2}. \quad (5.1.3)$$

El periodo de la lemniscata fue de 180 [s] y la duración de ambas pruebas fue de 220 [s].

Previamente se realizaron simulaciones numéricas de cada controlador para validar el esquema propuesto. Por brevedad se omiten los resultados en MatLab-Simulink y en el simulador Gazebo bajo el sistema operativo Linux (Ubuntu 20.04).

5.2 Parámetros deseados para la formación

En esta prueba se resuelve el problema de seguimiento de trayectoria. Los colores negro, azul y rojo son los colores de las trayectorias deseadas para los robots 1, 2 y 3, respectivamente. Los colores verde, cian y magenta son las trayectorias descritas por los móviles 1, 2 y 3, respectivamente. La distancia deseada de separación entre los robots es de 40 [cm] y el ángulo de abertura deseado de 60° con sus derivadas temporales iguales a cero. De esta manera se pide formar un triángulo equilátero. En resumen se tiene para las variables de forma

$$D_{2d} = D_{3d} = 0.4 \text{ [m]}, \quad \alpha_{cd} = \pi/3 \text{ [rad]}, \quad \dot{D}_{2d} = \dot{D}_{3d} = \dot{\alpha}_{cd} = 0.$$

De esta manera, de (3.1.4), (3.1.6) y (3.1.7), se tiene que $D_d = 0.5D_{1d} = 0.2 \text{ [m]}$, $H_d = 0.34641 \text{ [m]}$ y $\alpha_{2d} = \pi/6 \text{ [rad]}$, respectivamente. Las variables de forma cartesiana deseadas están dadas por

$$H_{xd} = H_d \cos(Q_{cd}), \quad H_{yd} = H_d \sin(Q_{cd}), \quad (5.2.1)$$

$$D_{xd} = -D_{2d} \cos(Q_{cd} + \alpha_{2d}) + H_{xd}, \quad D_{yd} = -D_{2d} \sin(Q_{cd} + \alpha_{2d}) + H_{yd}. \quad (5.2.2)$$

Por seguridad, se aplican saturaciones a las señales de control de los robots, inferiores a los límites indicados en las especificaciones del fabricante, dadas por

$$V_{max} = 0.2 \text{ [m/s]}, \quad W_{max} = 1.5 \text{ [rad/s]},$$

donde V_{max} y W_{max} son las velocidades lineales y angular máximas de cada robot móvil.

5.3 Seguimiento de trayectoria

Por brevedad, solamente se presenta el experimento del seguimiento de trayectoria usando el controlador descrito en la Sección 2.2.1. En el análisis matemático de los errores de seguimiento cartesianos utilizando ese controlador se prueba que dichos errores convergen a cero cuando $t \rightarrow \infty$. Experimentalmente se espera obtener errores de seguimiento acotados por una banda.

Para obtener los valores de los ganancias requeridas por el controlador interno, primero, se realizaron pruebas de seguimiento de trayectoria individual para cada robot. Las ganancias para todos los robots fueron sintonizadas de manera heurística:

$$k_{xi}, k_{yi} = 0.07, \quad k_{qi} = 0.1,$$

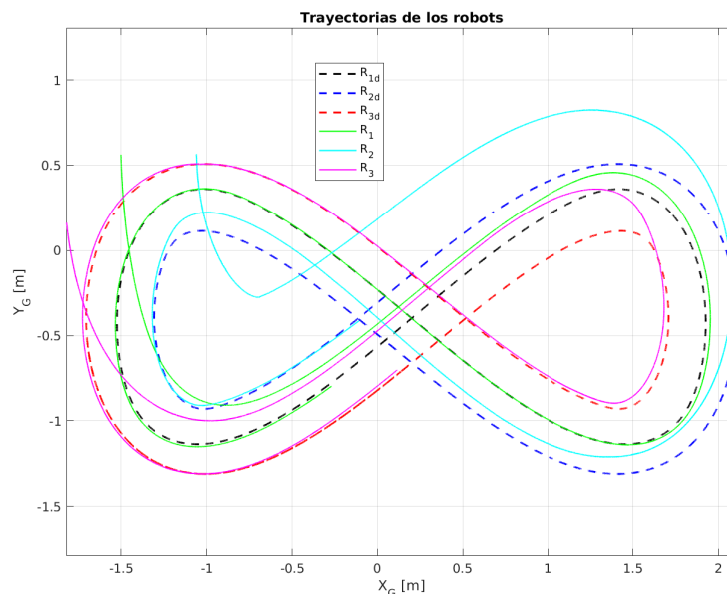


Figura 5.1: Trayectorias en el plano.

con $i = 1, 2, 3$. Posteriormente, se realizaron las pruebas de seguimiento de trayectoria para la formación. Las ganancias del esquema de formación fueron sintonizadas de manera heurística:

$$K_C = K_{X_c} = K_{Y_c} = 1, \quad K_D = K_{D_x} = K_{D_y} = 0.5K_C, \quad K_H = K_{H_x} = K_{H_y} = 0.5K_C.$$

De (3.2.14) se obtienen las ganancias de sincronización con los valores

$$\beta = 2K_C = 2, \quad \bar{\beta} = 0.5K_C = 0.5, \quad \mu = 2K_C = 2, \quad \bar{\mu} = 0.5K_C = 0.5,$$

las cuales satisfacen la condición (3.2.19), implicando que se requiere un movimiento sincronizado entre los robots móviles.

En la Fig. 5.1, se muestran las trayectorias deseadas y seguidas por los robots móviles. Debido a que se inició la prueba lejos de la trayectoria deseada, los robots logran converger a la formación y trayectoria deseada con errores aceptables a los 110 segundos. Como se puede apreciar, al inicio de la prueba, la trayectoria descrita por el robot 2 (en cian), parece desviarse y después converger a la deseada. Notar que dicho comportamiento se obtuvo de manera similar en la simulación en Matlab-Simulink. Esto se debe a diferentes factores: al controlador interno implementado y a la condición inicial de posición y orientación de los robots. Es importante mencionar que en esta prueba se quiere evaluar que, independientemente de la condición inicial de posición y orientación de alguno de los robots móviles, con el esquema de formación propuesto es posible resolver el problema de seguimiento de trayectoria.

En la Fig. 5.2, se muestran los errores en los estados de la formación. A partir de



Figura 5.2: Errores en los estados del clúster.

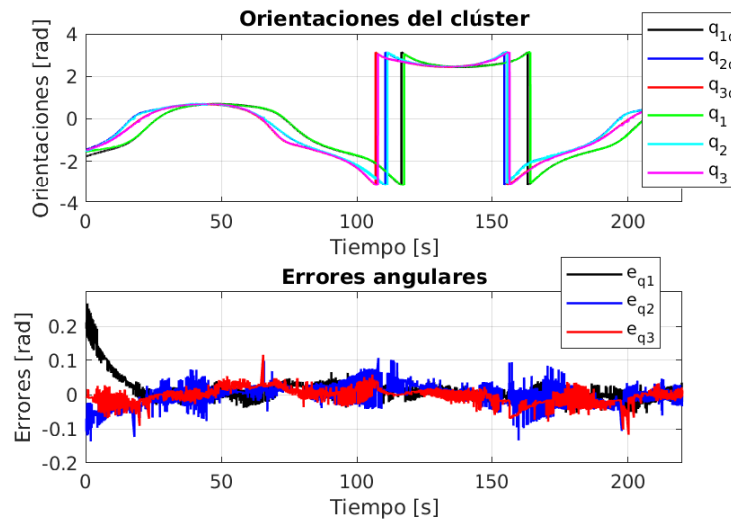


Figura 5.3: Comparación de las variables angulares.

los 175 segundos, los errores en todos los estados se mantienen en una banda acotada de 0.03 [m].

En la gráfica superior de la Fig. 5.3 se muestra la orientación deseada y la orientación real para cada uno de los 3 robots móviles. Debido a la trayectoria deseada (lemniscata), se presentan 2 discontinuidades, en el ángulo de orientación, durante el experimento. En la gráfica inferior se presentan los errores en las orientaciones de los robots. Una vez que se reduce el error de condición inicial, los errores en las orientaciones se mantienen en una banda de 0.06 [rad] con algunos picos.

En la Fig. 5.4 se muestra una comparación entre los errores cartesianos y los errores de sincronización. Estos últimos se parecen en forma a los errores de seguimiento cartesiano. A partir de los 190 [s] ambos errores se mantienen dentro de una banda de

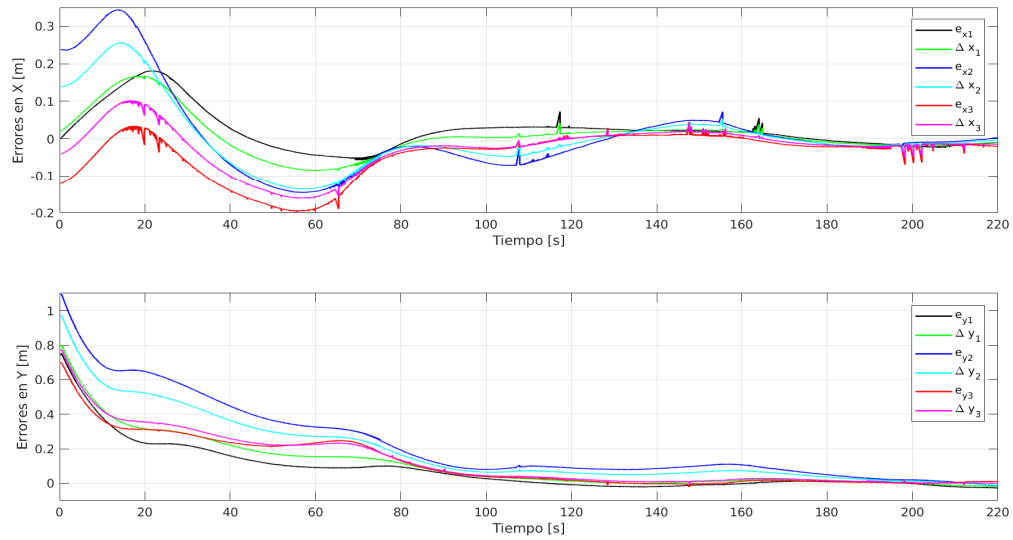


Figura 5.4: Errores cartesianos (e_{xi}, e_{yi}) y errores de sincronización ($\Delta x_i, \Delta y_i$).

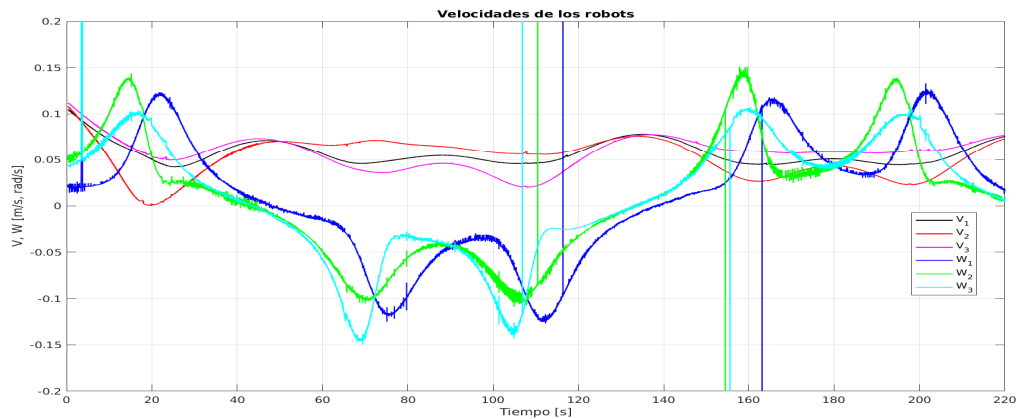


Figura 5.5: Señales de control aplicadas a los robots.

0.03 [m].

En la Fig. 5.5 se muestran las velocidades lineal y angular aplicadas a cada uno de los 3 robots móviles. Se tienen algunas saturaciones de corta duración cuando los robots cruzan la discontinuidad en la orientación de la trayectoria deseada. También, de acuerdo con la trayectoria deseada y las posiciones iniciales, la velocidad lineal de los robots móviles está dada entre 0 y 0.13 [m/s].

Discusión de los resultados

Los errores de seguimiento cartesiano y los errores en la orientación de cada uno de los robots móviles son aceptables para ser una prueba real ya que se mantienen dentro de una banda reducida. También, las señales de control asignadas por el controlador

son suaves. Note que con el esquema propuesto, se requieren realizar 2 tareas, el seguimiento de trayectoria y el control de formación y, consecuentemente, los errores a nivel del clúster también se reducen asintóticamente. Por lo tanto, los robots móviles convergen a la formación deseada mientras convergen a su trayectoria deseada de manera sincronizada ya que se toma en cuenta el error de posición de los demás robots.

5.4 Evasión de obstáculos y navegación

Aunque con el controlador por seguimiento práctico y el controlador P con compensación se puede resolver el problema de evasión de obstáculos estáticos y dinámicos, para realizar las pruebas en la plataforma experimental, solamente se implementó el controlador P con compensación [44], debido a que es posible reducir los errores de seguimiento mucho más rápidamente que utilizando el controlador por seguimiento práctico.

Por brevedad, en este trabajo se presenta solo la prueba más representativa de este esquema de evasión, la cual es cuando la formación, el obstáculo y la meta se localizan colinealmente. La duración de esta prueba es de 48 segundos. Se aplicaron saturaciones a las señales de control de los robots dadas como

$$V_{max} = 0.2 \text{ [m/s]}, \quad V_{dmax} = V_{max}/2, \quad W_{max} = 1.5 \text{ [rad/s]},$$

donde V_{dmax} , V_{max} y W_{max} son las velocidades lineales máximas de los robots, del clúster y la velocidad angular máxima para todos, respectivamente. Los parámetros deseados para el clúster fueron elegidos como:

$$D_{2d} = D_{3d} = 0.4 \text{ [m]}, \quad \alpha_{cd} = \pi/3 \text{ [rad]}, \quad \dot{D}_{2d} = \dot{D}_{3d} = \dot{\alpha}_{cd} = 0,$$

$$D_d = 0.2 \text{ [m]}, \quad H_d = 0.34641 \text{ [m]}, \quad \alpha_{2d} = \pi/6 \text{ [rad]}.$$

Las ganancias del controlador y de sincronización fueron:

$$k_{xi} = k_{yi} = 0.07, \quad W_x, W_y = 0.01, \quad K_C = K_{Xc} = K_{Yc} = 1, \quad K_D = K_{Dx} = K_{Dy} = 2K_C,$$

$$K_H = K_{Hx} = K_{Hy} = 2K_C, \quad \beta = 5, \quad \bar{\beta} = -1, \quad \mu = 5, \quad \bar{\mu} = -1,$$

con $i = 1, 2, 3$ y $l = 0.1$ [m]. Las ganancias del controlador fueron obtenidas de manera heurística. Los valores de las ganancias del método de potenciales artificiales fueron:

$$K_s = 0.2, \quad k_{s0} = 0.01, \quad K_v = 0.2, \quad \alpha_p = \alpha_v = 1.5, \quad \eta = 1.8, \quad k_\delta = 1.8, \quad \epsilon = 0.1,$$

$$r = 0.4[\text{m}], \quad \rho_o = 1.5[\text{m}], \quad a_{max} = 0.1[\text{m/s}^2].$$

Por simplicidad, la velocidad de la meta fue 0 y su posición fue $(1.5, -0.5)$ [m]. La velocidad y dirección del obstáculo móvil fue $V_{obs} = 0.05$ [m/s], $q_{obs} = -\pi$ [rad], con posición inicial $p_{obs}(0) = (0.5, -0.5)$ [m]. Se trató de colocar a los robots móviles de manera colineal al obstáculo y a la meta para evaluar el caso más común de mínimo local pero debido a que se tiene una prueba real no es posible colocar a los robots en

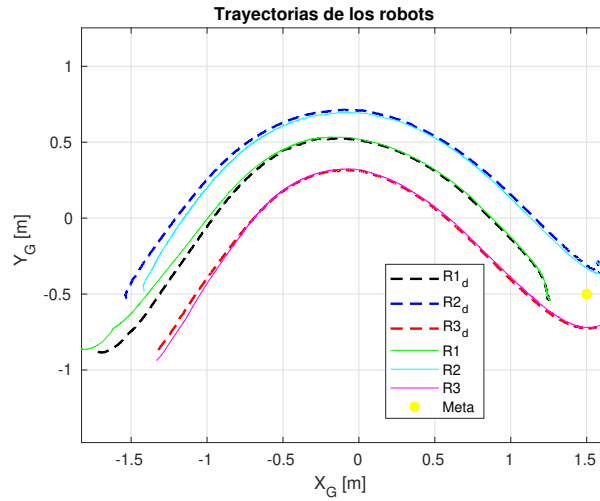
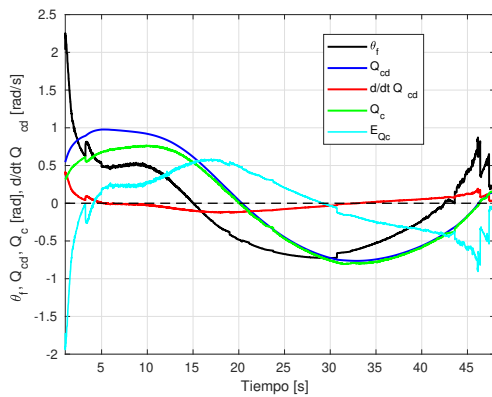
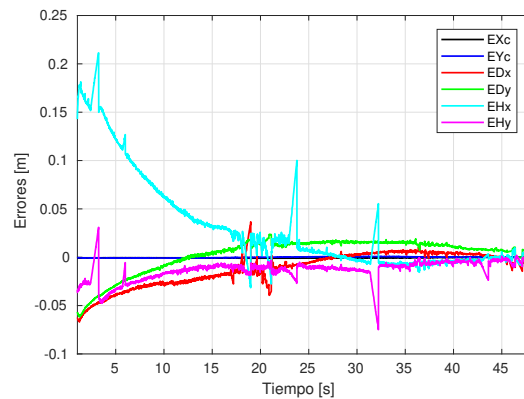


Figura 5.6: Trayectorias en el plano.



(a) Variables angulares.



(b) Errores en los estados de la formación.

Figura 5.7: Diferentes variables de interés.

una posición exacta. Por tal motivo, en esta prueba se pide que los robots alcancen sus posiciones deseadas mientras evaden al obstáculo.

En la Fig. 5.6 se muestran las trayectorias descritas por los robots y las trayectorias deseadas para cada uno. Se omite la trayectoria descrita por el obstáculo.

En la gráfica izquierda de la Fig. 5.7 se muestran las variables angulares relacionadas con la orientación del clúster. Nótese que la orientación deseada para el clúster Q_{cd} es suave. La orientación de la formación Q_c tiende a θ_f . En la gráfica derecha de la Fig. 5.7 se muestran los errores en los estados de la formación, los cuales están acotados, con algunos picos, en una banda de 0.015 [m] a partir de los 25 segundos.

En la gráfica superior de la Fig. 5.8 se muestra la velocidad deseada para el clúster

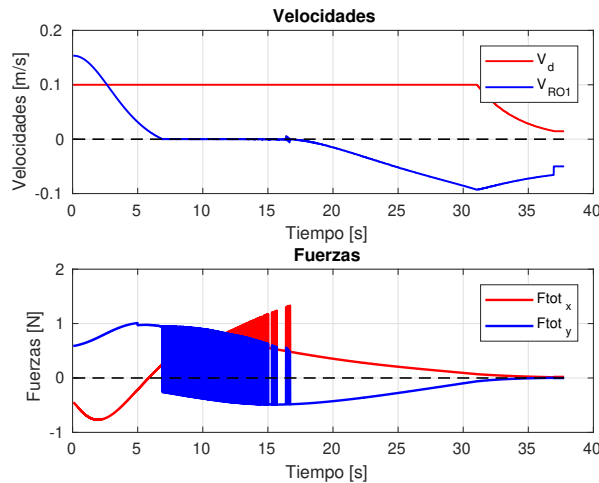


Figura 5.8: Oscilaciones no deseadas que aparecen mientras se rodea al obstáculo.

V_d y la velocidad de acercamiento con el obstáculo dinámico V_{RO1} , usando la condición AND en $h_{rep}(p, p_{obs}, v_{RO})$, de (2.5.11). V_d se mantiene saturada en 0.1 [m/s] y a medida que los robots se aproximan a la meta, V_d se reduce suavemente.

De la *Observación 2.3*, con la modificación de la condición lógica AND por OR en el potencial repulsivo (2.5.11), se logran eliminar las oscilaciones de alta frecuencia que aparecen mientras el clúster está rodeando al obstáculo, véase la Fig. 5.8. Esto sucede porque usando AND se realiza una acción de evasión cuando el centro del clúster está dentro de la zona de influencia y se aproxima al obstáculo. Justo en el momento que sale ó $V_{RO1} \leq 0$, no se pide una acción de evasión y debido a la posición muy particular de los robots y de la meta, éstos intentan acercarse al objetivo, haciendo que nuevamente se cumplan las condiciones de una evasión. Por lo tanto, este fenómeno ocurre siempre que se rodea al obstáculo. En la Fig. 5.8 se puede apreciar que dichas oscilaciones ocurren mientras que V_{RO1} está cruzando por cero.

En la gráfica superior de la Fig. 5.9 se muestra la velocidad deseada para el clúster V_d y la velocidad de acercamiento con el obstáculo dinámico V_{RO1} , usando la condición OR en $h_{rep}(p, p_{obs}, v_{RO})$, de (2.5.11). En la gráfica inferior de la Fig. 5.9 se presentan las componentes cartesianas de la fuerza total, la distancia del centroide del clúster al obstáculo y a la meta. Al finalizar la prueba se obtuvo un error entre el centroide del clúster y la meta de 1 [cm]. Como se puede apreciar, la mínima distancia entre el centroide del clúster y el obstáculo es de 90.8 [cm].

En la Fig. 5.10 se muestran las velocidades lineales y angulares de los 3 robots móviles. Es importante notar que al inicio de la prueba se tienen altas velocidades angulares en todos los robots. Probablemente debido a que se tienen errores iniciales en los estados de la formación grandes y el controlador muy agresivo bajo esta situación. Otra posible razón es porque el ángulo de acercamiento inicial es muy próximo a cero

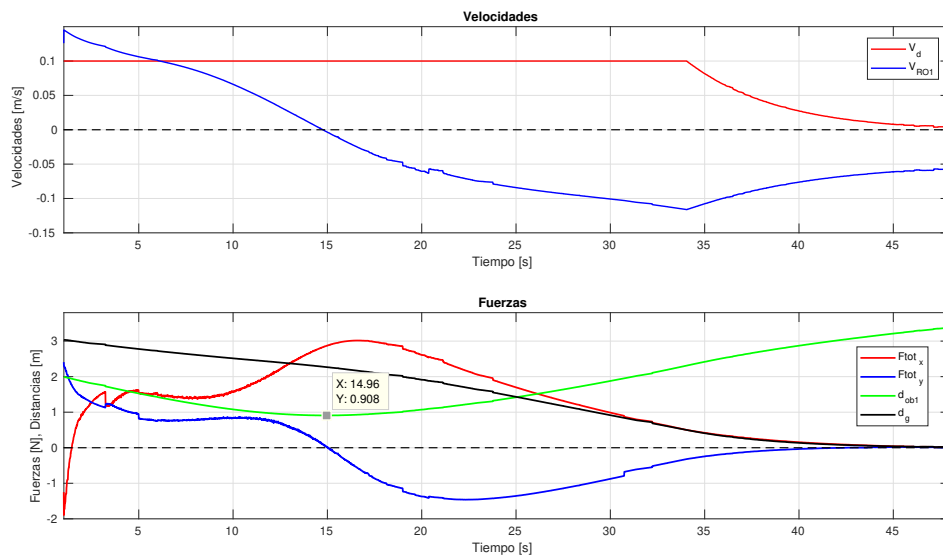


Figura 5.9: Velocidades, fuerzas y distancias.

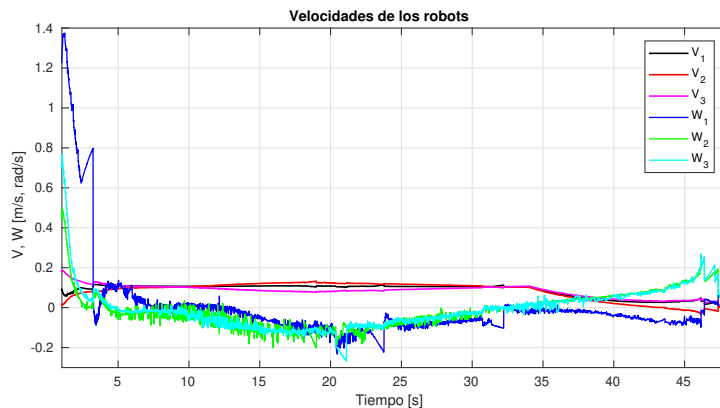


Figura 5.10: Señales de control de cada robot móvil.

resultando en una fuerza de repulsión muy grande.

Discusión de los resultados

De las gráficas mostradas, se puede concluir que el desempeño de la formación es relativamente bueno para ser una prueba real. Como ya se mencionó, al inicio de la prueba las señales de velocidad angular de los robots son muy grandes. Una posible solución a este problema puede ser que la posición inicial de los robots móviles sea más alejada del obstáculo y de esta manera se puedan corregir los errores en la formación antes de empezar a evadir al obstáculo con el fin de evitar realizar ambas tareas simultáneamente. Debido al reducido espacio que se tiene en el laboratorio, no fue posible realizar una

prueba con la suficiente separación inicial entre la formación y el obstáculo móvil.

Para obtener todas las ganancias del esquema propuesto, y como se explicó anteriormente, primero se deben sintonizar las ganancias del controlador interno para asegurar que la trayectoria libre de colisiones se puede seguir de manera aceptable. Posteriormente, se deben sintonizar las ganancias de convergencia a la meta sin considerar obstáculos para que, finalmente, se obtengan las ganancias de repulsión de un sólo obstáculo. Por simplicidad, se pueden suponer las mismas ganancias de repulsión para todos los obstáculos.

Capítulo 6

Resultados en Gazebo

En este capítulo se describe el simulador 3D empleado para realizar las pruebas de seguimiento de trayectoria y evasión de obstáculos dinámicos para una formación compuesta por un robot móvil aéreo tipo cuadrirrotor y 2 robots móviles terrestres tipo diferencial. Para mostrar la eficacia del simulador 3D, se muestra una comparación entre la prueba experimental descrita en el capítulo anterior y su simulación correspondiente utilizando Gazebo. También, se presenta la justificación de utilizar un cuadrirrotor con una cámara monocular a bordo para estimar la posición y orientación de los robots terrestres mediante estrategias de visión computacional. Finalmente, se muestran las pruebas en simulación del esquema de formación heterogénea con evasión de obstáculos dinámicos donde también se estima la posición, orientación y velocidad de los mismos, utilizando la cámara a bordo.

6.1 Simulador Gazebo

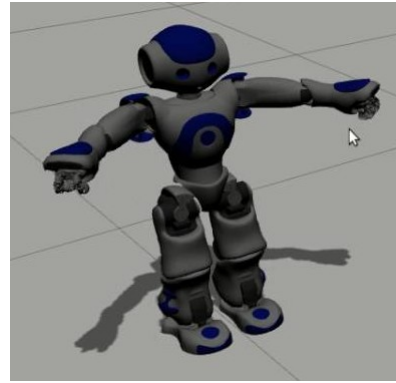
Gazebo es un simulador 3D de código abierto, el cual permite simular eficientemente la cinemática y la dinámica de los cuerpos rígidos (robots). Éste permite utilizar múltiples motores de física, motor de renderizado avanzado, soporte para complementos personalizados, programación en la nube y una gran variedad de repositorios con bastantes robots comerciales. Además, se puede simular un robot personalizando características dinámicas como inercias, fricciones, contacto entre elementos, etc.

También permite simular una gran cantidad de sensores como cámaras, dispositivos kinect, LIDAR, IMU, sonares, barómetros, GPS, etc. Además de simular entornos complejos tanto interiores como exteriores y muy realistas (luces, sombras y texturas). En la Fig. 6.1 se muestra uno de los muchos robots comerciales que se pueden utilizar en este simulador, el robot humanoide NAO.

Es importante mencionar que además de simular la cinemática y dinámica de los robots, con este simulador por defecto se agrega ruido gaussiano en las medidas de odometría y de velocidades. Esto con el fin de dar más realismo y una mejor aproximación a las señales que se obtendrían en una prueba real en el laboratorio.



(a) NAO real.



(b) NAO simulado.

Figura 6.1: Robot NAO de *Aldebaran Robotics*.

A continuación se enlistan algunas de las ventajas de utilizar el simulador 3D Gazebo:

- **Ahorro de tiempo:** Permite el rápido prototipado.
- **Ahorro de dinero:** No es necesario tener los robots ni los sensores reales para hacer pruebas.
- **Reducción de software:** Es posible simular muchos de los robots y sensores comúnmente utilizados en robótica.
- **Reutilización de código:** Si se utiliza Gazebo y ROS, se puede usar el mismo código de la simulación en un robot real.

El desarrollo del simulador Gazebo comenzó en 2002 en la Universidad del Sur de California. Los creadores originales fueron el Dr. Andrew Howard y su alumno Nate Koenig. El concepto de un simulador de alta fidelidad surgió de la necesidad de simular robots en entornos exteriores bajo diversas condiciones. Como simulador complementario de Stage, se eligió el nombre Gazebo como la estructura más cercana a un escenario al aire libre. El nombre se ha quedado a pesar del hecho de que la mayoría de los usuarios de Gazebo simulan ambientes interiores.

A lo largo de los años, Nate continuó desarrollando Gazebo mientras completaba su doctorado. En 2009, John Hsu, un ingeniero de investigación senior en Willow, integró ROS y el PR2 en Gazebo, que desde entonces se ha convertido en una de las principales herramientas utilizadas en la comunidad ROS. Unos años más tarde, en 2011, Willow Garage comenzó a brindar apoyo financiero para el desarrollo de Gazebo. En 2012, *Open Source Robotics Foundation* (OSRF) se separó de Willow Garage y se convirtió en la administradora del proyecto Gazebo. Después de un importante esfuerzo de desarrollo por parte de un equipo de personas talentosas, OSRF usó Gazebo para ejecutar el Virtual Robotics Challenge, un componente del DARPA Robotics Challenge, en julio

de 2013. OSRF continúa el desarrollo de Gazebo con el apoyo de una comunidad diversa y activa. Estén atentos a los desarrollos más interesantes relacionados con la simulación de robots.

Más información sobre el simulador Gazebo se puede consultar: <http://gazebo.org/projects>.

6.2 Seguimiento de trayectoria (con robots terrestres)

Para corroborar y verificar que los resultados en simulación en Gazebo son muy próximos a los que se obtienen en el laboratorio, se realizó una simulación de la prueba de laboratorio mostrada en el capítulo anterior.

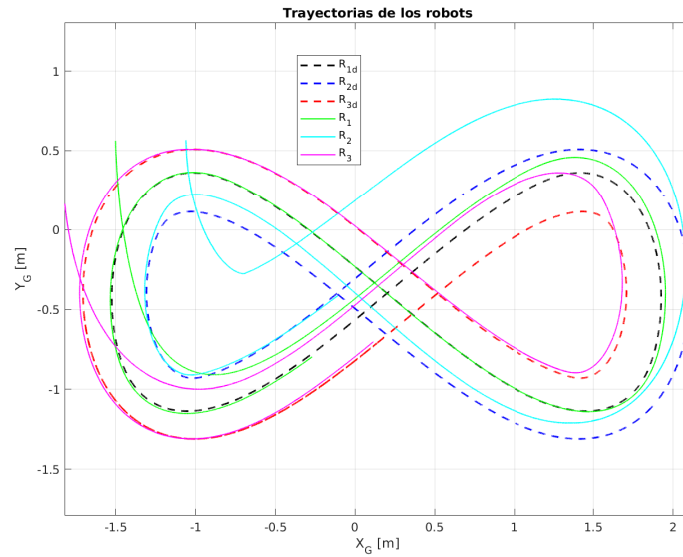
Utilizando el simulador Gazebo es posible obtener la posición y orientación exacta de todos los robots simulados como si se tuviera el sistema de cámaras *OptiTrack* o mediante la odometría de los robots usando sus encoders. En las simulaciones mostradas en este capítulo, siempre se utiliza la medición de Gazebo como si se hubieran realizado las pruebas en el laboratorio. A continuación se muestra una comparación de los resultados.

Considere exactamente las mismas condiciones que las descritas en el experimento: la trayectoria deseada, los parámetros deseados para la formación, las ganancias del controlador interno de todos los robots, de la formación y de sincronización, así como las condiciones iniciales de posición y orientación, con una precisión de milímetros. La única diferencia a tomar en cuenta es la frecuencia de ejecución del nodo del controlador, en el experimento fue de 100 [Hz] y en la simulación de 50 [Hz]. En principio, debido a la velocidad de los robots utilizados, no debería de ser un factor relevante para la comparación.

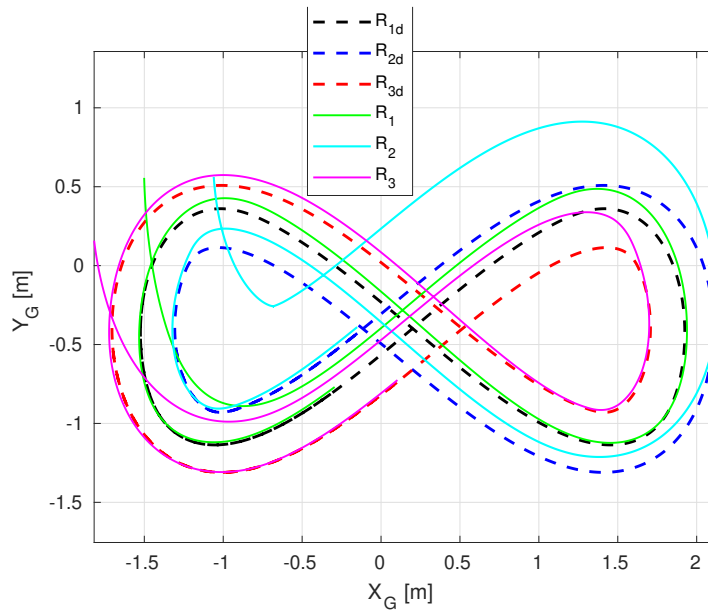
En la Fig. 6.2 se muestran las trayectorias descritas por los 3 robots móviles tipo diferencial. Como se puede apreciar, cada trayectoria individual es muy aproximada entre ambas gráficas, incluso para la trayectoria descrita por el robot 2 (en color cian). Dicho comportamiento también se presenta en la simulación en Matlab-Simulink y puede deberse a las posiciones iniciales y al controlador interno implementado.

En la Fig. 6.3 se presenta la comparación de los errores a nivel del clúster. A excepción del error E_{Y_c} , la cual presenta un incremento en $t \in [150, 170]$, cada señal de error es muy próxima en magnitud y forma. Después de 200 segundos, para ambas gráficas, todos los errores están acotados por una banda de 3 [cm].

En la Fig. 6.4 se muestra la comparación de las orientaciones de los robots diferenciales y los errores angulares, correspondientes. El comportamiento de las variables angulares mostradas es similar entre ellas. En cuanto a los errores angulares de la Fig. (b), se presentan algunos picos debido a la discontinuidad provocada al cruzar de $-\pi$ a π [rad], y viceversa. También, se observa que las señales en la gráfica (a) presentan ruido en la medición, como es de esperarse para un experimento. Exceptuando algunos picos,



(a) Experimento.



(b) Simulación en Gazebo.

Figura 6.2: Trayectorias en el plano (clúster terrestre).

todos los errores de orientación angular, mostrados en ambas gráficas, están acotados por una banda de 0.07 [rad].

En la Fig. 6.5 se muestra la comparación de los errores de seguimiento y de sincronización. Los errores son similares en magnitud y forma. Después de los 190 [s], todos los errores en ambas gráficas están acotados por una banda de 0.03 [cm].

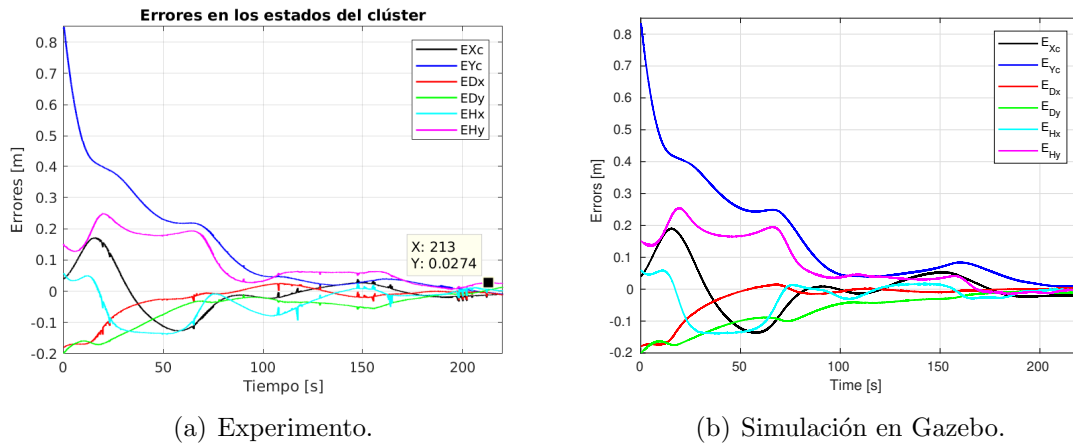


Figura 6.3: Errores a nivel de la formación (clúster terrestre).

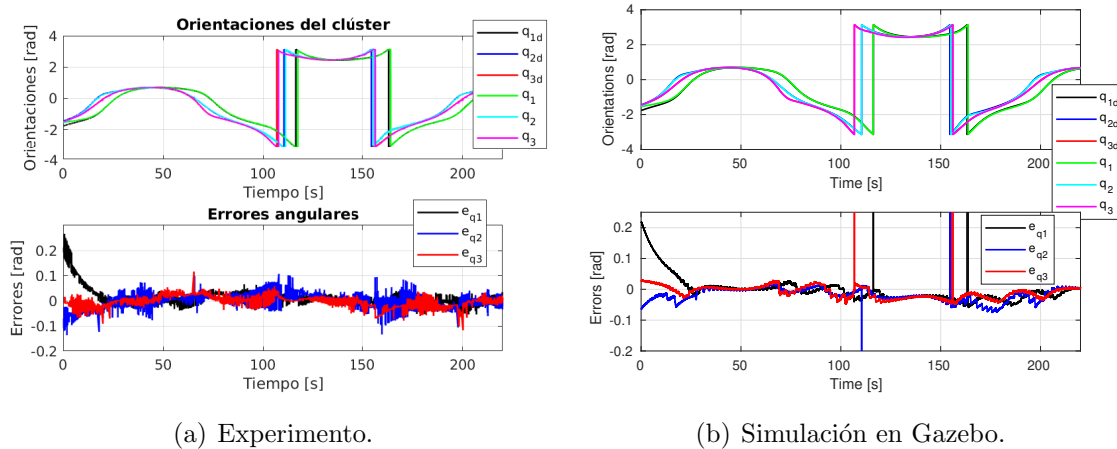
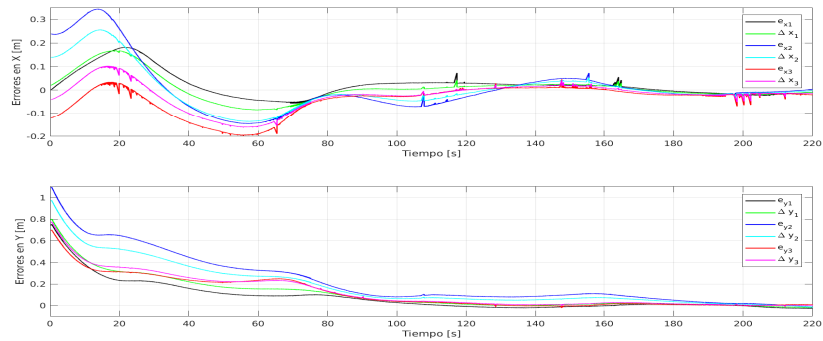
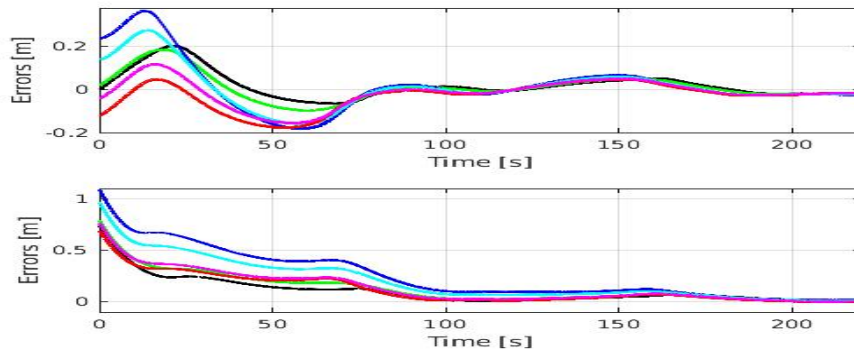


Figura 6.4: Variables angulares (clúster terrestre).

En la Fig. 6.6 se muestra la comparación de las señales de control calculadas por el controlador interno. Las velocidades de control se parecen entre ellas. En cuanto a las velocidades lineales, no hay duda de que en ambas gráficas se muestra el mismo comportamiento. Contrariamente, para las velocidades angulares, en la simulación en Gazebo se tiene ruido que no se presenta en el experimento. A diferencia de todas las gráficas anteriores, solo en esta comparación, las señales en simulación tienen más ruido que las señales del experimento.

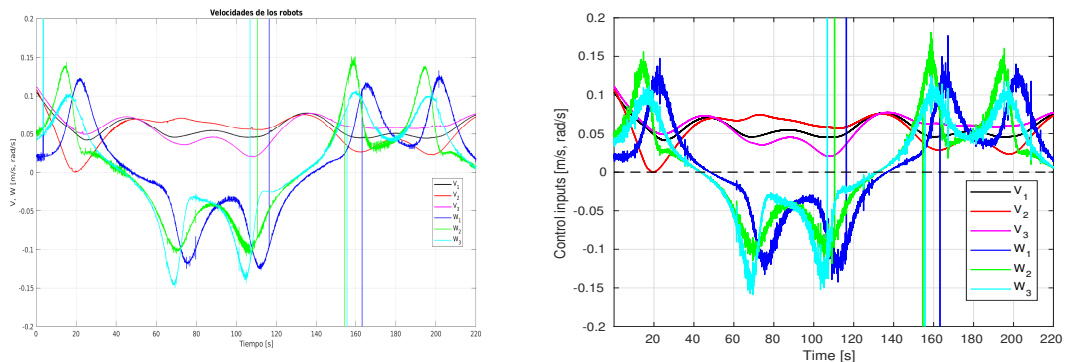


(a) Experimento.



(b) Simulación en Gazebo.

Figura 6.5: Errores de seguimiento y de sincronización (clúster terrestre).



(a) Experimento.

(b) Simulación en Gazebo.

Figura 6.6: Velocidades de control de los 3 robots (clúster terrestre).

Discusión de los resultados

De las gráficas mostradas, se puede concluir que utilizando el simulador Gazebo es posible obtener un comportamiento muy similar al esperado tanto matemáticamente como experimentalmente. Como ventaja adicional, es posible simular la medición e integración de diferentes sensores a bordo.

6.3 Motivación para usar un robot aéreo con cámara

Como se muestra en los experimentos de laboratorio, es posible implementar un esquema de formación, sincronización y evasión de obstáculos dinámicos, empleando robots móviles terrestres, siempre que se conozca de manera precisa la posición, orientación y velocidad de los mismos.

Para tomar en cuenta la suposición de que, en todo momento, se tiene la posición y la orientación de todos y cada uno de los robots móviles que componen la formación, y además, la velocidad de los obstáculos, es necesario un sistema de posicionamiento global como el sistema de cámaras *OptiTrack*.

Para reducir la necesidad de utilizar dicho sistema de posicionamiento global en todos y cada uno de los robots móviles, se requieren implementar sensores a bordo.

Una posible alternativa es la publicada en [63]. En ese trabajo se estudia una formación de robots tipo diferencial, líder-seguidor variante en el tiempo, bajo restricciones de comunicación y visibilidad. Se propone un nuevo método para eliminar el efecto de desviación entre la trayectoria descrita por el robot líder y la seguida por el otro robot, considerando una formación invariante en el tiempo como un sistema tractor-tráiler, para caminos con radios de curvatura desconocidos pero constantes, tomando en cuenta las restricciones de visibilidad. Para una formación variante en el tiempo, con radio de curvatura del camino del robot líder variable, el método propuesto reduce significativamente el efecto de desviación entre los caminos. Únicamente se requiere la posición y la orientación relativa entre los robots móviles, calculada mediante un algoritmo de visión computacional usando la cámara monocular a bordo del robot seguidor. Entonces, tanto la posición/orientación absoluta como la velocidad absoluta del robot líder no son requeridas. Además, para evitar usar comunicación explícita entre los robots, se implementa un observador de estado extendido [64,65] para estimar la velocidad lineal y angular del robot líder. Con esto, las tareas deseadas se ejecutan y logran de manera descentralizada.

Sin embargo, dicha alternativa puede ser implementada con éxito solamente en formaciones líder-seguidor o tipo convoy, en donde cada robot móvil sigue únicamente a su robot líder, el cual adelante del robot, debido al reducido campo de visión de la cámara monocular a bordo del robot móvil tipo diferencial.

Es importante mencionar que se intentó implementar la alternativa descrita en [63], para estimar la posición, orientación y velocidad relativa entre los robots terrestres en un formación triangular como la descrita en el capítulo anterior, pero se presentaron

los 2 inconvenientes siguientes:

1. No es posible abordar una formación triangular.
2. No se puede estimar la posición y velocidad de los obstáculos con la suficiente anticipación, debido al reducido campo de visión de la cámara.

Como se puede ver en la Fig. 6.7, incluso en línea recta, no es posible abarcar por completo a ambos robots que se mueven en frente del robot número 1 (robot en color rojo). Además, un obstáculo que se aproxima a alguno de los robots, solo puede ser visto siempre que éste se aproxime de frente y no de manera lateral. Debido a estos 2 problemas principales, se propone sustituir un robot móvil tipo unicycle por un robot móvil aéreo con cámara a bordo. De esta manera el campo de visión se incrementa considerablemente.

Por otro lado, para estimar la posición y la orientación de los robots terrestres, se utiliza la estrategia de visión descrita en la Sección 2.4.2. De esta manera se puede obtener la orientación relativa entre los robots y el marco de referencia del cuádrirrotor cuando se tiene una rotación sobre el eje óptico de la cámara (eje Z_G), lo que no es posible utilizando el algoritmo de reconstrucción descrito en la contribución [63].

6.4 Clúster compuesto por un cuádrirrotor con cámara

Para aprovechar al máximo el amplio campo de visión de la cámara monocular a bordo del robot móvil tipo cuádrirrotor, se tiene que modificar la cinemática directa e inversa de posición (3.2.3) y (3.2.4), respectivamente, del sistema compuesto por 3 robots móviles, de manera que el robot 1 en lugar de moverse atrás de los otros dos robots, se mueva por delante, como se muestra en la fig. 6.8. De esta manera, la porción

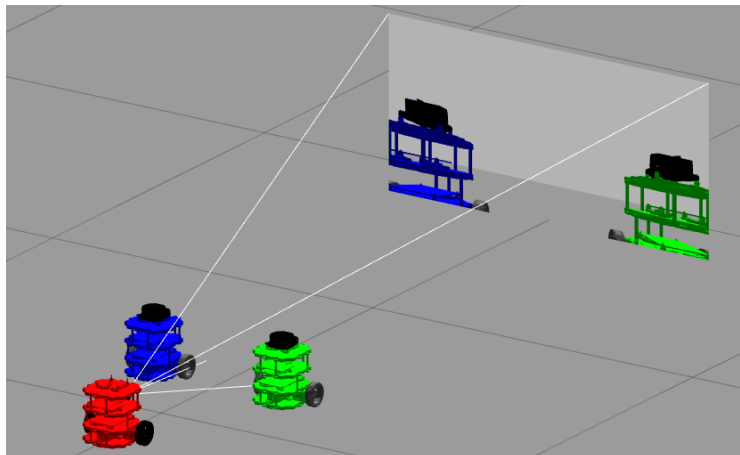


Figura 6.7: Formación triangular incompleta.

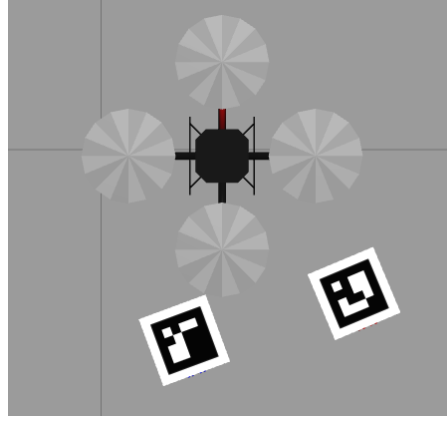


Figura 6.8: Formación triangular con el robot 1 por delante.

trasera del campo de visión se utiliza para detectar a los robots terrestres que componen la formación y la porción delantera se utiliza para detectar con mayor anticipación posibles obstáculos dinámicos que se acercan a la formación.

La formación triangular descrita en el Capítulo 3, donde los robots 2 y 3 se mueven por delante del robot 1, se propuso para abordar el problema de transporte de objetos sin ningún tipo de sujeción, utilizando robots móviles terrestres, mediante el método llamado “caging” o *enjaulamiento* [66], en el cual los robots móviles rodean al objeto a desplazar de manera que no se puede salir durante el recorrido.

La cinemática directa de posición del sistema compuesto por 2 robots móviles terrestres y uno aéreo está dada por

$$\begin{bmatrix} X_c \\ Y_c \\ D_x \\ D_y \\ H_x \\ H_y \end{bmatrix} = \begin{bmatrix} \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 \\ 0 & \frac{1}{3} & 0 & \frac{1}{3} & 0 & \frac{1}{3} \\ 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & -\frac{1}{2} & 0 & \frac{1}{2} \\ 1 & 0 & -\frac{1}{2} & 0 & -\frac{1}{2} & 0 \\ 0 & 1 & 0 & -\frac{1}{2} & 0 & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix}, \quad (6.4.1)$$

donde $(x_1, y_1) = (x, y)$ es la posición del centroide del cuadrirrotor, (x_2, y_2) y (x_3, y_3) son las posiciones de los puntos externos del eje de las ruedas de los robots móviles tipo unicyclo, (p_2) y (p_3) , respectivamente. Con esta selección, el controlador interno implementado en los robots tipo diferencial es el descrito en la Sección 2.2.2. Esto es porque con dicho controlador, los errores de seguimiento convergen más rápidamente a cero que utilizando el controlador interno descrito en la Sección 2.2.1.

La cinemática inversa de posición está dada como

$$\begin{bmatrix} x_1 \\ y_1 \\ x_2 \\ y_2 \\ x_3 \\ y_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 2/3 & 0 \\ 0 & 1 & 0 & 0 & 0 & 2/3 \\ 1 & 0 & -1 & 0 & -1/3 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1/3 \\ 1 & 0 & 1 & 0 & -1/3 & 0 \\ 0 & 1 & 0 & 1 & 0 & -1/3 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ D_x \\ D_y \\ H_x \\ H_y \end{bmatrix}. \quad (6.4.2)$$

De la *Observación 3.1*, se debería incluir un estado adicional para describir la altura del robot aéreo. Sin embargo, en este trabajo el robot móvil tipo cuadrirrotor se utiliza como si fuera otro robot móvil terrestre tipo omnidireccional, donde se considera el centroide del mismo para describir su posición en el plano cartesiano y las entradas de control v_z , w_z , sólo se utilizan para mantener en su valor deseado los estados de altura (z) y orientación (ψ) del cuadrirrotor, respectivamente, los cuales no influyen en el esquema de sincronización.

Por lo tanto, la cinemática inversa de posición (6.4.2) se utiliza para obtener la salida del controlador a nivel del clúster, esto es

$$\Delta \mathbf{R} = \begin{bmatrix} \Delta x_1 \\ \Delta y_1 \\ \Delta x_2 \\ \Delta y_2 \\ \Delta x_3 \\ \Delta y_3 \end{bmatrix} = \begin{bmatrix} \Delta E_{X_c} + (2/3)\Delta E_{H_x} \\ \Delta E_{Y_c} + (2/3)\Delta E_{H_y} \\ \Delta E_{X_c} - (1/3)\Delta E_{H_x} - \Delta E_{D_x} \\ \Delta E_{Y_c} - (1/3)\Delta E_{H_y} - \Delta E_{D_y} \\ \Delta E_{X_c} - (1/3)\Delta E_{H_x} + \Delta E_{D_x} \\ \Delta E_{Y_c} - (1/3)\Delta E_{H_y} + \Delta E_{D_y} \end{bmatrix}, \quad (6.4.3)$$

y para obtener las velocidades deseadas de los robots, esto es

$${}^G \dot{\mathbf{R}}_d = \begin{bmatrix} \dot{x}_{1d} \\ \dot{y}_{1d} \\ \dot{x}_{2d} \\ \dot{y}_{2d} \\ \dot{x}_{3d} \\ \dot{y}_{3d} \end{bmatrix} = \begin{bmatrix} \dot{X}_{cd} + \frac{2}{3}\dot{H}_{xd} \\ \dot{Y}_{cd} + \frac{2}{3}\dot{H}_{yd} \\ \dot{X}_{cd} - \frac{1}{3}\dot{H}_{xd} - \dot{D}_{xd} \\ \dot{Y}_{cd} - \frac{1}{3}\dot{H}_{yd} - \dot{D}_{yd} \\ \dot{X}_{cd} - \frac{1}{3}\dot{H}_{xd} + \dot{D}_{xd} \\ \dot{Y}_{cd} - \frac{1}{3}\dot{H}_{yd} + \dot{D}_{yd} \end{bmatrix}, \quad (6.4.4)$$

6.5 Seguimiento de trayectoria (para un clúster heterogéneo)

Para evaluar y verificar el desempeño de la estrategia de visión implementada para estimar las posiciones y orientaciones relativas de los robots terrestres con respecto del robot aéreo, primero se realizaron pruebas de seguimiento de trayectoria, antes de realizar las simulaciones de evasión de obstáculos dinámicos. Tomando en cuenta las

capacidades físicas de los robots tipo diferencial TurtleBot3, se configuró una trayectoria deseada que pueda ser seguida por los robots terrestres. En las simulaciones, se utilizaron modelos modificados de los TurtleBot3 burger con un marcador aruco diferente para estimar su posición y orientación sin confundirlos. Es importante mencionar que sólo el robot aéreo conoce su posición global usando un sensor de posicionamiento global, la posición y orientación de los robot terrestres se estima usando los datos de la cámara a bordo del robot aéreo y la postura del robot aéreo.

Trayectoria deseada

La trayectoria deseada es la que se describe en (5.1.1)-(5.1.2), con $a = 2.5$ [m], $b = 1.1$ [m], $(X_0, Y_0) = (0.2, -0.5)$ [m], $\tau = t$. La derivada temporal de la orientación deseada del clúster está descrita por (5.1.3). El periodo de la lemniscata fue 160 [s] y la duración de la simulación fue 180 [s].

Parámetros deseados para la formación

Los colores negro, azul y rojo son los colores de las trayectorias deseadas para los robots 1, 2 y 3, respectivamente. Los colores verde, cian y magenta son las trayectorias descritas por los móviles 1, 2 y 3, respectivamente. La distancia entre el punto medio del eje de las ruedas y el punto externo, l , es 0.05 [m]. La distancia deseada de separación entre los robots es 50 [cm] y el ángulo de abertura deseado de 60° con sus derivadas temporales iguales a cero. De esta manera se pide formar un triángulo equilátero. La altura deseada de vuelo para el cuadricóptero es 2 [m]. Dicha altura se elige así debido a que, a partir de diferentes pruebas, es una altura con la cual se pueden detectar a los robots terrestres y seguir la trayectoria establecida. También, ya que el robot aéreo es considerado un robot móvil tipo omnidireccional en el plano horizontal, su orientación es libre y no depende de la trayectoria deseada. Por simplicidad se elige igual a cero.

De esta manera, las variables D_d , H_d y α_{2d} se obtienen de (3.1.4), (3.1.6) y (3.1.7), respectivamente. Las variables de forma cartesiana deseadas están dadas por (5.2.1) y (5.2.2).

Se aplican saturadores a las señales de control de los robots, indicados en las especificaciones del fabricante, dadas por

$$V_{max} = 0.22 \text{ [m/s]}, \quad W_{max} = 2.84 \text{ [rad/s]}, \quad (6.5.1)$$

donde V_{max} y W_{max} son las velocidades lineales y angular máximas de cada robot móvil. Debido a que las velocidades máximas del robot aéreo están muy lejanas a las de los robots terrestres, para el cuadricóptero, se tiene un vuelo cuasi-estático.

Para simular el tiempo en calcular las entradas de control a partir de la estimación de la posición y orientación mediante la estrategia de visión artificial, la frecuencia de ejecución del nodo fue de 20 [Hz].

Para obtener los valores de los ganancias requeridas por el controlador interno, primero, se realizaron simulaciones de seguimiento de trayectoria individual para cada

robot. Para el robot aéreo se implementó el controlador descrito en la Sección 2.3. Las ganancias para todos los robots fueron:

$$\kappa_x = \kappa_y = \kappa_z = \kappa_\psi = \kappa_{xi} = \kappa_{yi} = 0.2,$$

con $i = 2, 3$. Posteriormente, se realizaron las simulaciones de seguimiento de trayectoria para la formación. Las ganancias del esquema de formación fueron:

$$K_C = K_{X_c} = K_{Y_c} = 1, \quad K_D = K_{D_x} = K_{D_y} = 2K_C, \quad K_H = K_{H_x} = K_{H_y} = 2K_C,$$

esto con el fin de dar más peso a las variables de separación entre los robots. Además, con estos valores, de (3.2.14), se obtienen las ganancias de sincronización

$$\beta = \mu = 3, \quad \bar{\beta} = \bar{\mu} = -1,$$

las cuales también satisfacen la condición (3.2.19), implicando que se requiere un movimiento sincronizado entre los robots móviles.

En la Fig. 6.9 se muestra una comparación entre las trayectorias descritas por los robots terrestres cuya posición es obtenida del simulador Gazebo, y las trayectorias estimadas por la estrategia de visión artificial, asumiendo que sólo el robot aéreo conoce su posición y orientación mediante el sistema *OptiTrack* o GPS. A esa altura (2 [m]),

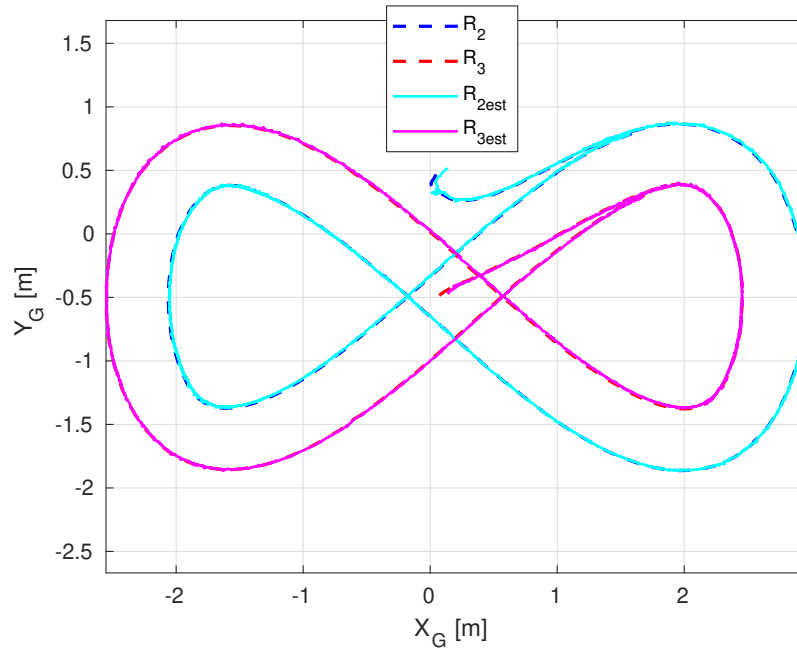


Figura 6.9: Comparación entre las posiciones estimadas usando visión y simuladas obtenidas de Gazebo (formación heterogénea).

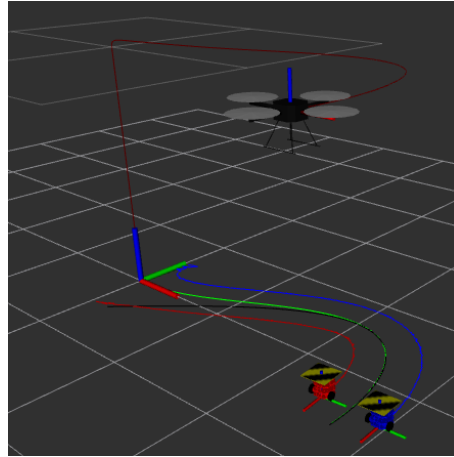


Figura 6.10: Resultado en simulación (usando el visualizador RViz).

los errores en la estimación de las posiciones globales están acotados por una banda de 1.8 [cm].

En la Fig 6.10 se muestra el resultado de la simulación utilizando el visualizador de ROS llamado RViz. La posición del centro de gravedad del cuadrirrotor se proyecta en el plano horizontal al igual que los puntos externos del eje de las ruedas de los robots tipo diferencial.

En la Fig. 6.11, se muestran las trayectorias deseadas y seguidas por los robots móviles que componen el clúster, un cuadrirrotor y 2 robots tipo diferencial. Debido a que se inició cerca de la trayectoria deseada, para evitar sobre impulsos grandes y con esto perder de vista a los robots terrestres, los robots logran converger a la formación y trayectoria deseada con errores aceptables a los 25 segundos.

En la Fig. 6.12, se muestran los errores en los estados de la formación. A partir de los 25 segundos, los errores en todos los estados se mantienen en una banda acotada de 0.05 [m].

En la Fig. 6.13 se muestra una comparación entre los errores cartesianos y los errores de sincronización. Nuevamente, estos últimos se parecen en forma a los errores de seguimiento cartesiano. A partir de los 30 [s] ambos errores se mantienen, con algunos picos, dentro de una banda de 0.07 [m].

En el experimento que se presenta en el Capítulo 5, para una trayectoria lemniscata con diferentes parámetros, se obtienen errores del clúster, de seguimiento y de sincronización, acotados por una banda de 0.03 [m]. No obstante, los resultados que se obtienen para la trayectoria descrita en el experimento, son muy similares a los mostrados en esta simulación.

En la Fig. 6.14 se muestra la orientación simulada y la orientación estimada para cada uno de los 2 robots móviles terrestres. Aunque la estimación de la orientación angular de los robots terrestres es muy próxima a la simulada, se presenta ruido en dicha estimación. Para la prueba de seguimiento de trayectoria, este ruido parece no tener ningún tipo de efecto negativo que impida o modifique el desempeño del seguimiento

de la trayectoria deseada para la formación heterogénea. Sin embargo, este ruido sí repercute en el desempeño de la evasión de los obstáculos dinámicos, como se verá en la sección siguiente.

En la Fig. 6.15 se muestran las velocidades lineal y angular aplicadas a cada uno de los 3 robots móviles. De acuerdo con la trayectoria deseada y las posiciones iniciales, la velocidad lineal de todos los robots móviles está dada entre 0 y 0.16 [m/s]. Como se puede ver, todas las velocidades lineales presentan ligeras oscilaciones. Sin embargo, en cuanto a las velocidades angulares de los robots terrestres, las oscilaciones presentes son muy grandes y de alta frecuencia. Cabe aclarar que no se implementó ningún tipo de filtro en las velocidades angulares. W_2 y W_3 mostradas en la Fig. 6.15 son las que se asignaron a los robots terrestres calculadas por el controlador interno. No obstante, al ser los robots móviles sistemas mecánicos, éstos funcionan como un filtro ante dichas velocidades angulares, es decir, se obtiene un comportamiento ligeramente oscilante.

Discusión de los resultados

De los resultados mostrados en esta simulación se puede concluir que es posible resolver los problemas de control de formación y seguimiento de trayectoria para una formación heterogénea donde se utiliza visión artificial para estimar la posición y orientación de los robots móviles terrestres.

A diferencia de otros esquemas de formación, como lo es líder-seguidor, con el esquema propuesto, se toma en cuenta la posición de todos los robots, de esta manera se tiene mayor robustez ante perturbaciones externas durante todo el recorrido. Esta ventaja se utiliza para reducir la posibilidad de que el cuadricóptero pueda perder de

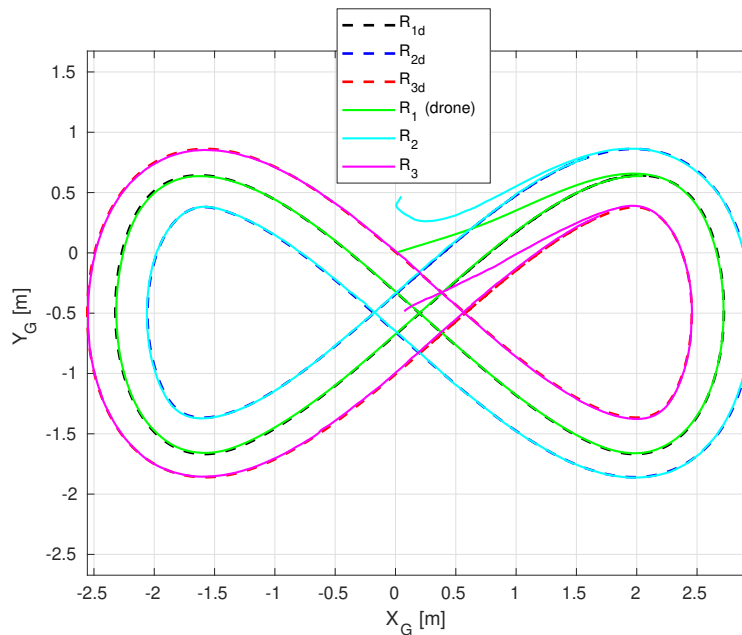


Figura 6.11: Trayectorias en el plano horizontal (formación heterogénea).

6.5. SEGUIMIENTO DE TRAYECTORIA (PARA UN CLÚSTER HETEROGÉNEO)77

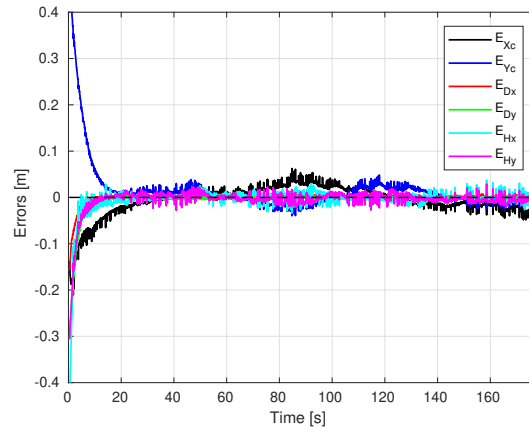


Figura 6.12: Errores a nivel del clúster (formación heterogénea).

vista a alguno de los robots terrestres, ya que todos los robots compensan el error de seguimiento de algún otro.

Como ya se mencionó, usando el simulador Gazebo, se pueden simular entornos muy realistas con varios robots móviles interactuando entre ellos. Además, debido a que también se simula ruido en las mediciones de todos los estados, las gráficas presentadas son similares a las obtenidas en experimentos de laboratorio.

Por otro lado, es importante recordar que en el experimento de laboratorio, mostrado en el Capítulo 5, se implementó el controlador interno que considera el punto medio del eje de las ruedas de los robots, y los errores de seguimiento de todos los robots terrestres

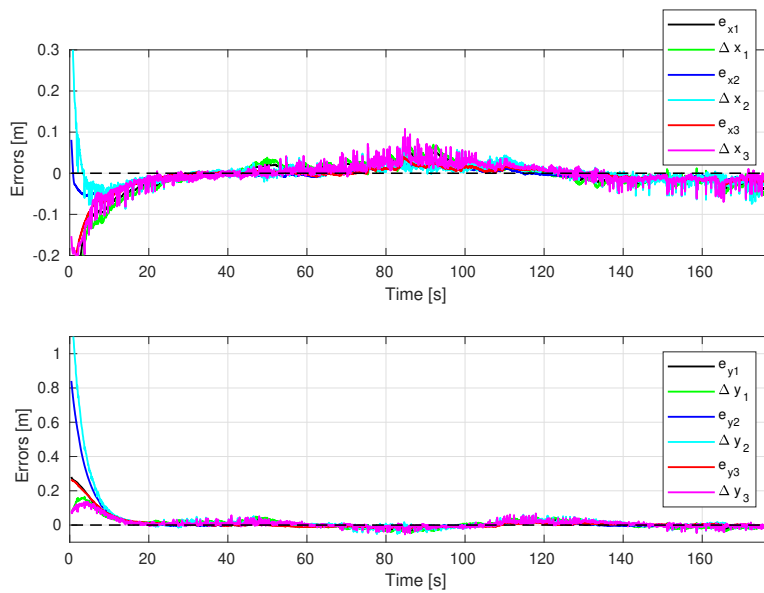


Figura 6.13: Errores a nivel de los robots (formación heterogénea).

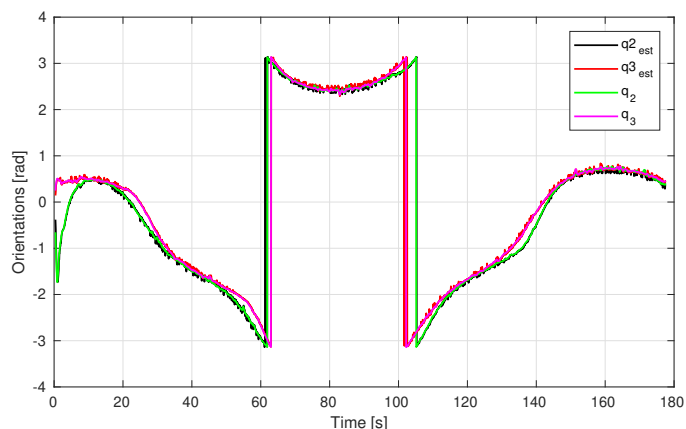


Figura 6.14: Comparación entre las orientaciones de los robots tipo diferencial usando visión (formación heterogénea).

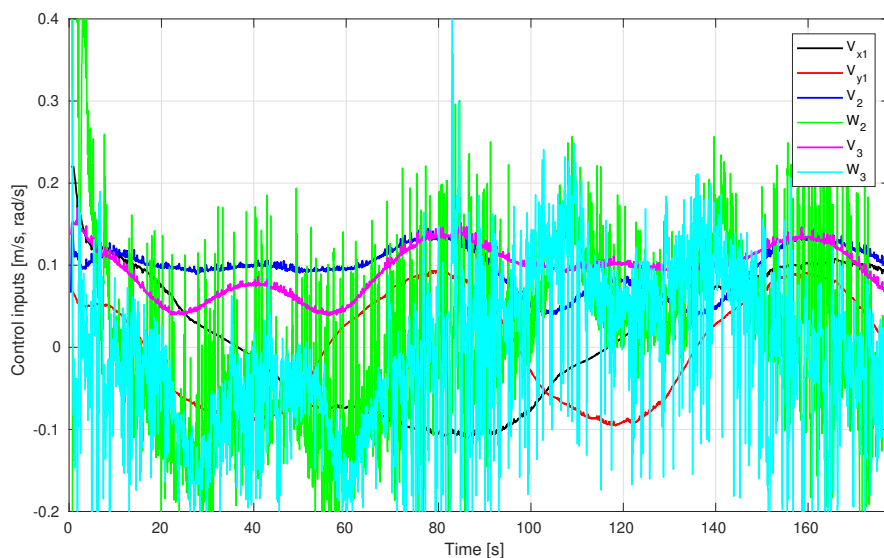


Figura 6.15: Señales de control de 3 robots estimando la posición y orientación (formación heterogénea).

obtenidos están acotados por una banda de 3 [cm]. En la simulación mostrada en este capítulo, se implementó el controlador interno que considera un punto externo del eje de las ruedas. Por lo tanto, no es posible hacer una comparación directa entre ambos resultados.

De acuerdo con los parámetros seleccionados para la trayectoria deseada en esta simulación, la velocidad lineal máxima de todos los robots móviles, incluyendo el cuadrirrotor, es de 16 [cm/s], la cual se asigna al inicio de la prueba cuando se tienen los

errores de seguimiento y de sincronización mayores. Nuevamente, debido a las máximas velocidades que puede alcanzar el robot aéreo, a la rápida convergencia en sus ángulos de orientación, para éste, prácticamente se tiene un vuelo estacionario. A medida que se requiere seguir una trayectoria más rápida con cambios bruscos en sus radios de curvatura, implicando cambios bruscos en las aceleraciones lineal y angular, se espera obtener un deterioro de la estimación de la posición y orientación de los robots terrestres. Para validar esta idea, y comprobar el límite de velocidad, al menos en simulación utilizando el esquema de formación heterogéneo propuesto, a continuación se presentan los resultados de simulaciones a velocidades más grandes.

6.5.1 Simulaciones del esquema a altas velocidades

Considerando las capacidades físicas, en cuanto a velocidad de la mayoría de los robots móviles terrestres utilizados en investigación, por lo general se tienen experimentos a velocidades de 0 a 15 [cm/s], para robots de un tamaño similar a los utilizados en este trabajo. De acuerdo con las velocidades máximas de los robots móviles tipo burger (6.5.1), los parámetros de la trayectoria deseada y de la formación utilizados en las simulaciones a altas velocidades fueron seleccionados tales que se sobre pasaran dichos límites, teniendo velocidades lineales de 0 a 30 [cm/s]. Al ser una simulación, es posible configurar velocidades superiores a las establecidas por el fabricante. Esto con el fin de mostrar el deterioro en el desempeño obtenido por la formación heterogénea usando visión artificial y un cuadirrotor comercial.

La trayectoria deseada es nuevamente la que se describe en (5.1.1)-(5.1.2), con $a = 2.5$ [m], $b = 1.1$ [m], $(X_0, Y_0) = (0.2, -0.5)$ [m], $\tau = t$. La derivada temporal de la orientación deseada del clúster está descrita por (5.1.3). Para generar una trayectoria más rápida, el periodo de la lemniscata no fue de 160 [s], sino de 80 [s]. Este valor se obtuvo al disminuirlo gradualmente hasta el punto en el que no es posible seguir la trayectoria deseada.

Las ganancias de seguimiento individual de cada robot y de sincronización no se modificaron. Los parámetros deseados para la formación se resumen a continuación:

$$D_2 = D_3 = 0.5 \text{ [m]}, \quad \alpha_2 = \pi/3 \text{ [rad]}, \quad Z_d = 2 \text{ [m]}, \quad \psi_d = 0.$$

No se aplican saturaciones a las señales de control de los robots y la frecuencia de ejecución del nodo es nuevamente de 20 [Hz].

Se realizaron 2 simulaciones (con duraciones aproximadas), una considerando la posición y orientación de los robots terrestres obtenidas del simulador Gazebo, como si se tuviera el sistema *OptiTrack*, y la otra considerando la posición y orientación estimada mediante la estrategia de visión. Por lo tanto, en todas las comparaciones, la figura izquierda corresponde a cuando se usa la postura obtenida de Gazebo, mientras que la derecha corresponde a cuando se usa la postura estimada.

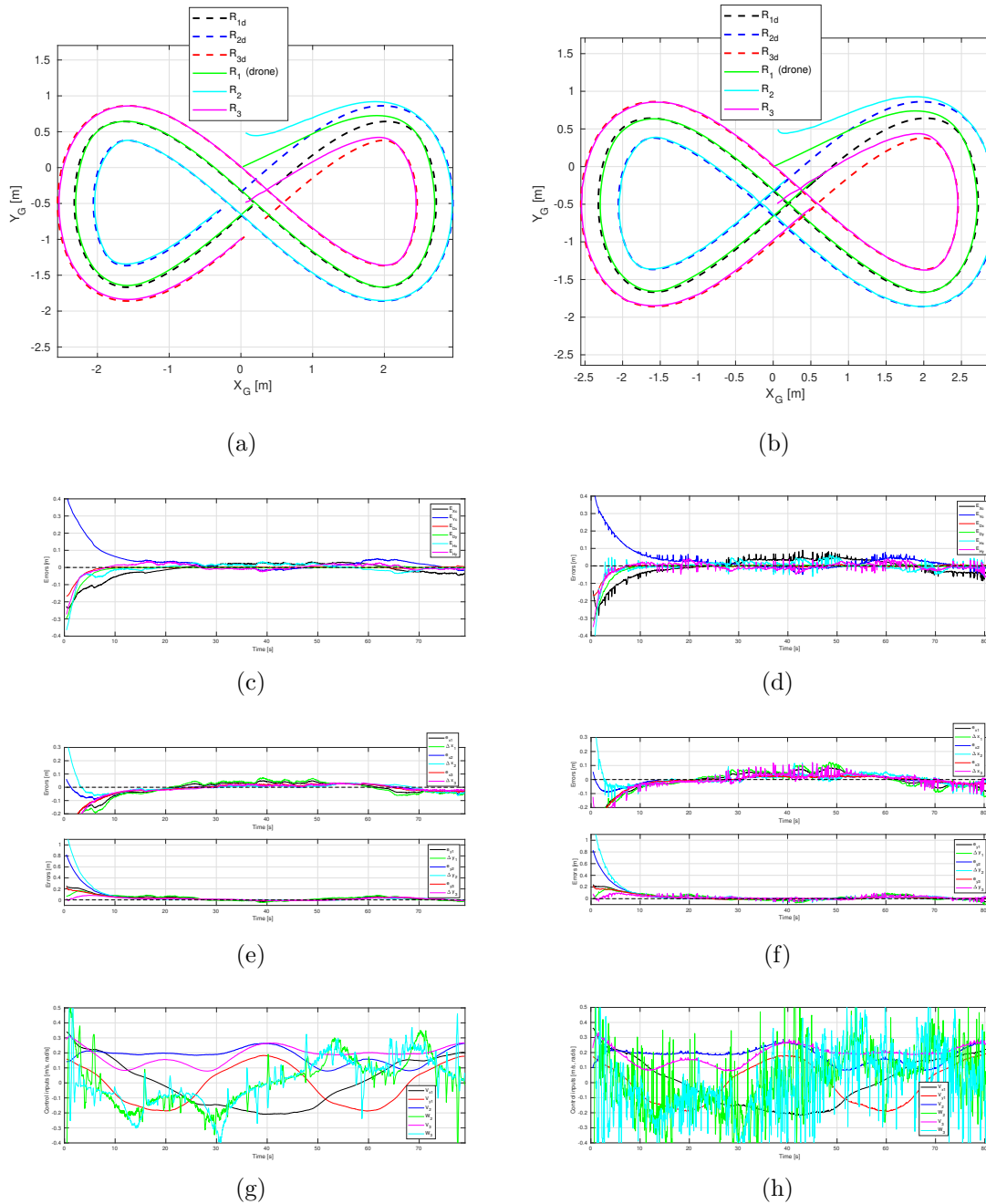


Figura 6.16: Comparación entre usar la postura dada por Gazebo (gráficas a la izquierda) y la postura estimada usando visión (gráficas a la derecha), con $T = 80$ [s].

En las gráficas (a-b) de la Fig. 6.16, se muestran las trayectorias descritas por los tres robots móviles para la prueba de seguimiento considerando el periodo de la misma igual a 80 [s]. A simple vista no parece existir una diferencia significativa entre las trayectorias.

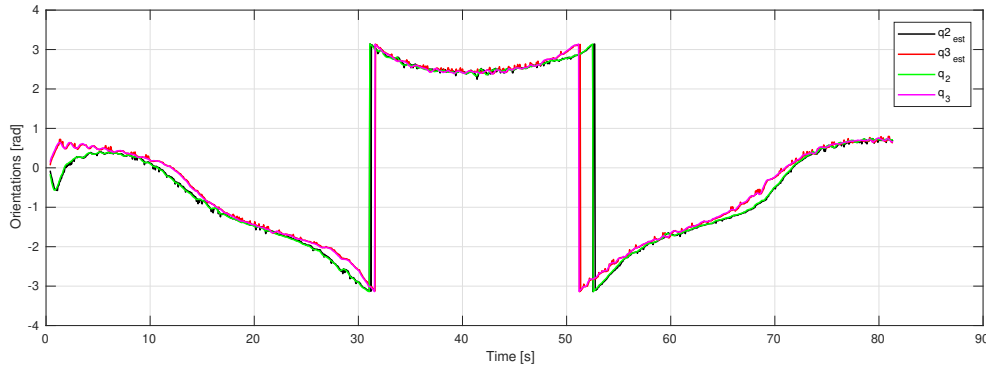


Figura 6.17: Comparación entre las orientaciones de los robots terrestres obtenidas del simulador y las estimadas por la estrategia de visión, con $T = 80$ [s].

En las gráficas (c-d) se presenta la comparación entre los errores a nivel del clúster. Para la gráfica (c), después de alcanzar el estado estacionario, todos los errores están acotados por una banda de 5 [cm], incluyendo el intervalo de tiempo entre 57 y 67 segundos. En cuanto a la gráfica (d), todos los errores están acotados por una banda de 8 [cm], la cual se alcanza desde los 30 segundos.

En las gráficas (e-f) de la Fig. 6.16, se presentan los errores de seguimiento y de sincronización de los 3 robots móviles. Para la gráfica (e), después de alcanzar el estado estacionario alrededor de los 12 segundos, todos los errores están acotados por una banda de 8 [cm]. Para la gráfica (d), todos los errores están acotados por una banda de 11 [cm], la cual se alcanza desde los 10 segundos.

En las gráficas (g-h) se presentan las velocidades asignadas a todos los robots móviles calculadas por el mismo controlador interno. Para ambas gráficas, note que solamente las velocidades traslacionales del robot aéreo (v_{x1}, v_{y1}), son positivas y negativas para la trayectoria elegida. Además, son menores en magnitud que las velocidades lineales de los robots terrestres. Las velocidades traslacionales de los robots 2 y 3, están dadas entre 5 [cm/s] y 35 [cm/s] debido a las condiciones iniciales.

En cuanto a las velocidades angulares de los robots terrestres, está claro que utilizando la postura dada por el simulador, mostradas en la gráfica (g), se presentan algunas oscilaciones y sobre impulsos durante toda la prueba. Las velocidades angulares de los robots terrestres obtenidas utilizando la postura estimada mediante visión artificial, mostradas en la gráfica (h), presentan grandes oscilaciones de alta frecuencia y sobre impulsos grandes durante toda la prueba. Este deterioro puede deberse principalmente al ruido que se tiene en la estimación de la orientación de los móviles, como se muestra en la Fig. 6.17. Aunque el error en la estimación de las orientaciones no es significativamente grande, aproximadamente de 0.125 [rad] en el estado estacionario, se presenta oscilaciones no deseadas y éstas posiblemente pueden provocar que se amplifique el ruido al momento de calcular las velocidades angulares de control para los robots terrestres no holonómicos.

6.6 Evasión de obstáculos (con un clúster heterogéneo)

A continuación se presentan los resultados de la simulación de la evasión de obstáculos dinámicos para la formación heterogénea, compuesta por 2 robots móviles terrestres tipo diferencial y un cuadirrotor con cámara. Por brevedad, se omiten los resultados de la prueba de evasión de un obstáculo, similar a la presentada en la Sección 5.4, y se muestra la prueba considerando 2 robots móviles tipo diferencial como los obstáculos a evadir.

Parámetros deseados

A partir de diferentes pruebas de evasión, se propuso modificar la distancia de separación entre los robots de 0.5 a 1 [m]. También, para juntar a los robots terrestres, se modificó el ángulo α_{cd} de $\pi/3$ a $\pi/6$ [rad]. Asimismo, para aumentar el espacio que puede ser visto por la cámara del robot aéreo, y de esta manera detectar con la suficiente anticipación a los obstáculos, se propuso cambiar la altura de vuelo de 2 a 3 [m]. Tomando en cuenta estas modificaciones fue posible llevar a cabo la prueba de evasión de 2 obstáculos dinámicos en un ambiente denso, esto es evadir obstáculos con muy poco espacio y con poca anticipación, con respecto al tamaño de la formación.

A partir de la altura deseada de vuelo y del ángulo de visión de la cámara simulada, se determinó que la zona de influencia de los obstáculos a tomar en cuenta es $\rho_0 = 1.8$ [m]. Con esta selección, se aplica una acción de evasión en cuanto se detectan los robots móviles. Las ganancias para el esquema de evasión de obstáculos son $\alpha_p = 1.5$, $\alpha_v = \alpha_p/4$. Esto con el fin de dar mayor ponderación a la posición relativa entre el clúster y los obstáculos, debido a la mala estimación de la velocidad de los mismos.

Es importante remarcar que la velocidad de los obstáculos se obtuvo usando una derivada sucia a partir de las posiciones estimadas mediante la estrategia de visión.

Las ganancias de los controladores internos y de sincronización son las mismas que las utilizadas en la simulación anterior.

Postura inicial, velocidad de los obstáculos y posición de la meta

La posición y orientación de los obstáculos en la simulación es la siguiente, en metros y radianes respectivamente:

$$[p_{obs1}, q_{obs1}] = [3, 1, -2.4], \quad [p_{obs2}, q_{obs2}] = [3, -1, 2.7], \quad p_g = [3, 0].$$

Por simplicidad, los 3 robots móviles inician colocados sobre el eje Y_G , esto es:

$$[P_1, q_1] = [0, 0.5, 0], \quad [P_2, q_1] = [0, -0.5, 0], \quad [P, q] = [0, 0, 0],$$

de esta forma el centroide del clúster inicia cerca del origen, con una orientación igual a 0 [rad]. Las velocidades asignadas para los obstáculos, los cuales son también robots tipo unicycle, son:

$$V_{obs1} = 0.1, \quad W_{obs1} = 0, \quad V_{obs2} = 0.1, \quad W_{obs2} = 0.02,$$

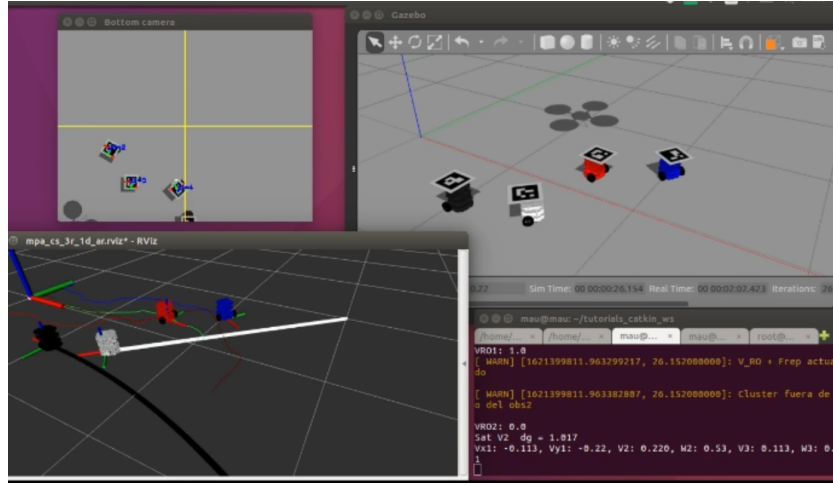


Figura 6.18: Entorno de simulación (prueba de evasión de 2 obstáculos).

en [m/s] y [rad/s], respectivamente. Con esta asignación, el obstáculo número 1 se mueve en línea recta acercándose a la formación. El obstáculo 2 se mueve describiendo una trayectoria circular con radio $R_2 = V_{obs2}/W_{obs2} = 5$ [m]. Por simplicidad, se considera la meta estática, $v_g = 0$.

Los valores de los ganancias del método de potenciales artificiales fueron:

$$K_s = 0.2, k_{s0} = 0.01, K_v = 0.2, \alpha_p = 1.5, \alpha_v = \alpha_p/4, \eta = 1.8, k_\delta = 1.8, \epsilon = 0.1,$$

$$r = 0.4[\text{m}], \rho_o = 1.8[\text{m}], a_{max} = 0.1[\text{m/s}^2].$$

Nuevamente, estas condiciones se eligieron para poner a prueba el esquema propuesto bajo una situación complicada, es decir, considerar una distancia de anticipación reducida, un espacio muy reducido en comparación con el tamaño deseado para la formación, donde los obstáculos se mueven relativamente rápido.

En la fig. 6.18 se muestra el entorno de la simulación, utilizando el simulador Gazebo y el meta-sistema operativo ROS para controlar a todos los robots. En la fig. 6.19 se muestran las trayectorias descritas por el clúster para la simulación de evasión de 2 obstáculos dinámicos (se omiten sus trayectorias), donde se estima la posición y velocidad de los mismos mediante la cámara a bordo del robot móvil aéreo.

Las 3 tareas que se piden llevar a cabo son: converger a la formación deseada, evadir a los 2 obstáculos y alcanzar la meta. Como se puede apreciar, se presentan oscilaciones no deseadas en las posiciones de los 3 robots móviles (líneas en negro, azul y rojo). Sin embargo, las trayectorias descritas por los 3 robots móviles son las mostradas en verde, cian y magenta. Estas oscilaciones se obtienen porque cuando se está evadiendo a los obstáculos, el robot aéreo presenta cambios bruscos en los ángulos de orientación alabeo y cabeceo, ϕ y θ , respectivamente.

Bajo estas condiciones, cuando se está en un ambiente denso, los errores de estimación, de seguimiento y de formación son mayores que cuando solamente se requiere

seguir una trayectoria deseada, como se mostró en la sección anterior. Aunque en este caso, donde se requiere forzosamente evitar colisionar con los obstáculos dinámicos, no es tan importante obtener errores de seguimiento tan pequeños, porque no se quiere seguir una trayectoria específica.

En la Fig. 6.20 se presenta una comparación entre la orientación de los robots terrestres. Aunque dichos valores estimados son muy próximos a los valores reales, en esta simulación bajo las condiciones descritas anteriormente, se logra apreciar un deterioro en comparación con la estimación obtenidos en la fig. 6.14, donde sólo se sigue una trayectoria deseada.

En la Fig. 6.21 se muestran las componentes de la fuerza total calculada por el método de potenciales, la distancia entre el centroide del clúster y la meta, las distancias entre el centroide de la formación y los 2 obstáculos y sus velocidades relativas. La distancia de separación entre el centroide del clúster y el obstáculo número 2 (d_{ob2}), alrededor de los 10 segundos y en el intervalo 16 a 19 segundos, presenta discontinuidades debido a que en esos instantes el cuadrirrotor pierde de vista al obstáculo. Esto puede ser causado porque el robot móvil rota sobre sus ejes X_B y Y_B , causando que el obstáculo se salga de su zona de visibilidad. Esta discontinuidad se ve reflejada en una peor estimación de la velocidad de acercamiento correspondiente V_{RO2} .

Como se puede notar, utilizando una derivada sucia para estimar las velocidades de los obstáculos, a partir de la estimación por visión, no se logra obtener un resultado confiable.

En la Fig. 6.22 se presentan las velocidades de control de los 3 robots móviles para la prueba de evasión de 2 obstáculos estimando la posición y velocidad de éstos mediante visión. A diferencia de la simulación donde sólo se requiere seguir una trayectoria deseada, en este caso no solo las velocidades angulares de los robots terrestres presentan

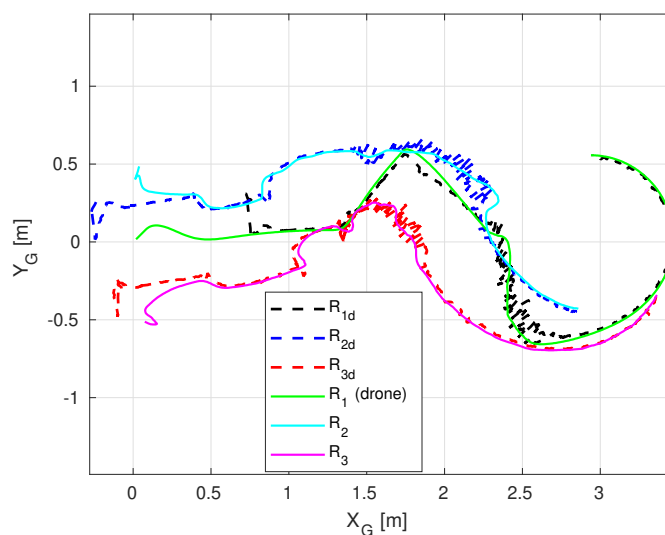


Figura 6.19: Comparación de los posiciones globales (prueba de evasión de 2 obstáculos).

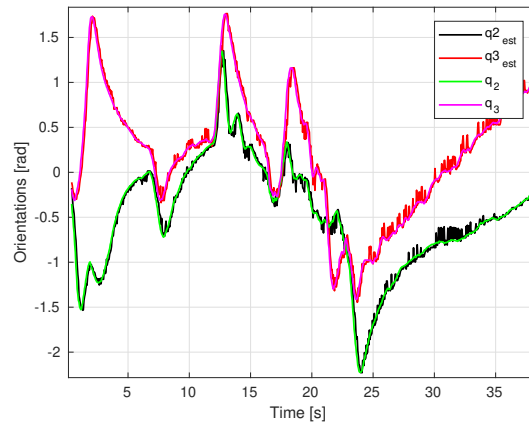


Figura 6.20: Orientación de cada robot terrestre (prueba de evasión de 2 obstáculos).

oscilaciones de alta frecuencia sino también sus velocidades lineales. Este deterioro se debe principalmente a las oscilaciones que presenta el cuadrirrotor cuando el clúster está dentro de la zona de influencia de algún obstáculo.

En contraste, usando la postura dada por el simulador Gazebo y tomando las velocidades conocidas de los obstáculos, el resultado de la prueba de evasión es muy similar al obtenido en MATLAB.

Incluso utilizando la estimación de la posición (mediante visión) y tomando la posición/velocidad exacta de los obstáculos, el resultado obtenido es muy similar al obtenido en MATLAB, con cierto deterioro en las señales de control (oscilaciones de alta frecuencia).

Discusión de los resultados

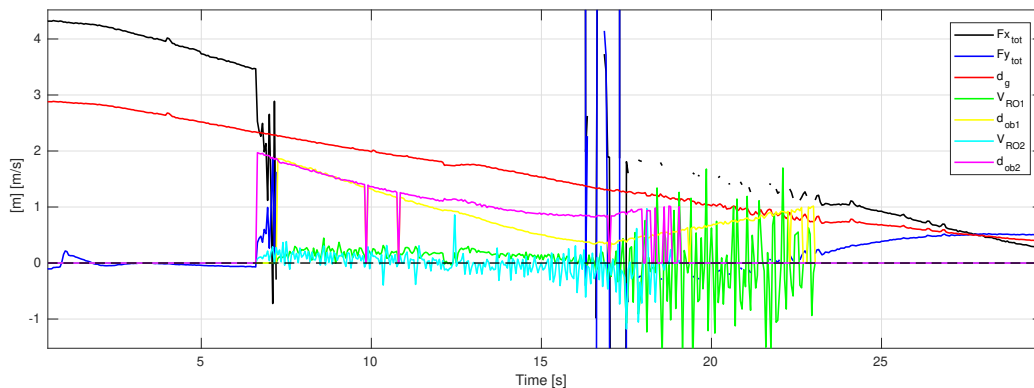


Figura 6.21: Fuerza total, distancias y velocidades de acercamiento (prueba de evasión de 2 obstáculos).

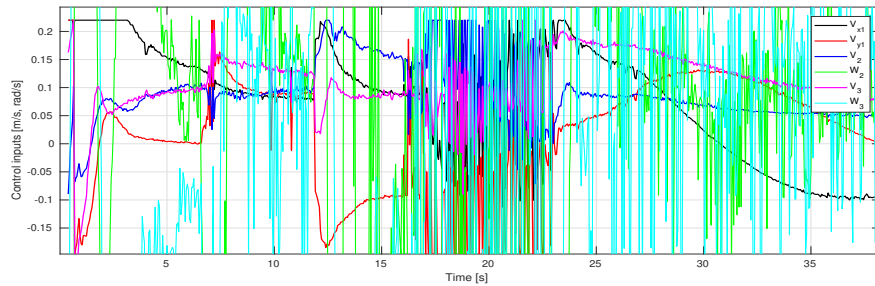


Figura 6.22: Señales de control de los 3 robots móviles (prueba de evasión de 2 obstáculos).

Es importante recordar que el esquema de evasión de obstáculos con cámara a bordo se evaluó en un entorno muy denso, es decir, la distancia de separación deseada entre los robots 1 y 2 es $D_2 = D_3 = 1$ [m], con la meta a 3 [m], aproximadamente 3 veces el tamaño de la formación. También la zona de influencia de cualquier obstáculo $\rho_0 = 1.8$ [m], casi el doble que el tamaño de la formación. Estos parámetros se escogieron así con el fin de evaluar el desempeño que se obtendría en el laboratorio de la Sección de Mecatrónica. Por lo tanto, los resultados mostrados se pueden mejorar considerablemente en un espacio más amplio y utilizando una mejor aproximación de la velocidad de los obstáculos, estimada mediante alguna otra estrategia.

Capítulo 7

Conclusiones y trabajo futuro

7.1 Conclusiones

En este trabajo se implementa el esquema de formación para un grupo compuesto por tres robots móviles, para resolver los problemas de control de formación, sincronización, seguimiento de trayectoria y evasión de obstáculos desde la perspectiva del enfoque de *Cluster Space*.

Con el fin de probar la versatilidad que ofrece este esquema de formación y sincronización, resolviendo el problema de seguimiento de trayectoria individual de cada robot móvil tipo diferencial, se implementaron 2 controladores a nivel de los robots, uno controlando el punto medio del eje de las ruedas y otro controlando un punto externo. Con esto se prueba que es posible utilizar algún otro controlador interno tomando en cuenta ligeras adaptaciones.

También, se presenta el análisis de estabilidad, basado en la teoría de Lyapunov, de la dinámica de los errores de seguimiento de cada robot móvil tomando en cuenta el punto medio del eje de las ruedas de cada robot tipo unicycle, considerando el caso general, esto es, que tanto las orientaciones deseadas como las velocidades de rotación deseadas de todos los robots móviles pueden ser todas diferentes entre sí. A diferencia del enfoque de *Cluster Space*, la estrategia de control propuesta garantiza que tanto los errores de seguimiento de los robots como los errores de sincronización convergen asintóticamente al origen. Al lograr esto, los errores de seguimiento del clúster convergen asintóticamente a cero también. Además, el controlador interno propuesto garantiza que los errores de orientación de los robots tienden exponencialmente a cero.

Se logra implementar el método de potenciales artificiales modificado en un clúster compuesto de 3 robots móviles terrestres para resolver el problema de navegación y evasión de obstáculos estáticos y dinámicos teniendo como un objetivo importante mantener la formación unida, esto es, mantener los errores de forma cartesiana acotados por una banda pequeña.

Para resolver el problema de mínimo local, común en el método de potenciales artificiales, se propone una fuerza de repulsión perpendicular y una función que describe la ganancia de ésta.

Aunque se obtiene un incremento en la banda que delimita los errores de seguimiento y de sincronización a nivel de los robots móviles, así como un deterioro en las señales de control asignadas a éstos, usando una cámara monocular a bordo del robot móvil tipo cuadrirrotor y la estrategia de visión propuesta, se estima la posición y orientación de los robots de la formación para resolver los problemas de control de formación, sincronización y seguimiento de trayectoria de manera satisfactoria en la simulación utilizando Gazebo.

También, mediante la estrategia de visión utilizando la cámara, se logra estimar la posición y orientación de los obstáculos dinámicos de manera confiable siempre que no se este realizando una acción de evasión. Cuando se intenta calcular la velocidad de los obstáculos mientras se evaden a éstos, utilizando una derivada sucia (aproximación de Euler) se obtiene una estimación bastante imprecisa.

Se resuelve el problema de evasión de obstáculos dinámicos usando una cámara a bordo del robot móvil aéreo tipo cuadrirrotor incluso con una mala aproximación de las velocidades. Los resultados mostrados demuestran la efectividad del esquema propuesto en un ambiente denso.

7.2 Trabajo futuro

A continuación se en listan algunas extensiones de este trabajo de tesis doctoral:

1. Proponer una variación de la fuerza de repulsión perpendicular para que ésta tenga una dirección contraria al acercamiento entre el clúster y el obstáculo, ó si es posible, que se pueda determinar la mejor toma de decisión (dejarlo pasar, esperar, pasar por arriba o por abajo).
2. Proponer una velocidad de navegación que proporcione un mejor desempeño. Si es necesario que tome en cuenta la distancia entre el clúster y el obstáculo, el clúster y la meta o el ángulo de acercamiento, ya que utilizar una velocidad constante de navegación produce resultados aceptables.
3. Para mejorar el desempeño de evasión en ambientes densos, considerar los ángulos de orientación ϕ y θ , no solo ψ , en la estimación de la postura relativa, así como sus respectivas derivadas temporales $(\dot{\phi}, \dot{\theta}, \dot{\psi})$, con respecto al marco de referencia $\{B\}$.
4. Implementar un observador o estimador para calcular las velocidades de los obstáculos dinámicos, tomando en cuenta que se tiene una limitada zona de visión de la cámara monocular a bordo de robot aéreo.

Referencias

- [1] He Bai, Murat Arcak, John Wen. Cooperative Control Design: A Systematic, Passivity-Based Approach, *Communications and Control Engineering*. Springer, 2011.
- [2] Amit Ailon and Ilan Zohar. (2012). Control Strategies for Driving a Group of Nonholonomic Kinematic Mobile Robots in Formation Along a Time-Parameterized Path. *IEEE/ASME Transactions On Mechatronics*, Vol. 17, No. 2, April 2012.
- [3] Travis Dierks and S. Jagannathan. (2007). Control of Nonholonomic Mobile Robot Formations: Backstepping Kinematics into Dynamics. *16th IEEE International Conference on Control Applications*, Part of IEEE Multi-conference on Systems and Control Singapore, 1-3 October 2007.
- [4] Jawhar Ghommam, Maarouf Saad and Faical Mnif. (2008). Formation path following control of unicycle-type mobile robots. *2008 IEEE International Conference on Robotics and Automation*, Pasadena, CA, USA, May 19-23, 2008.
- [5] Thijs H.A. van den Broek, Nathan van de Wouw and Henk Nijmeijer. (2009). Formation Control of Unicycle Mobile Robots: a Virtual Structure Approach. *48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, Shanghai, P.R. China, December 16-18, 2009.
- [6] Hamed Rezaee and Farzaneh Abdollahi. (2011). Mobile Robots Cooperative Control and Obstacle Avoidance Using Potential Field. *2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM2011)*, Budapest, Hungary, July 3-7, 2011.
- [7] K. D. Do. (2007). Formation tracking control of unicycle-type mobile robots. *2007 IEEE International Conference on Robotics and Automation*, Roma, Italy, 10-14 April 2007.
- [8] Gianluca Antonelli, Filippo Arrichiello, Stefano Chiaverini. The null-space-based behavioral control for autonomous robotic systems. *Intelligent Service Robotics* (2008) 1:27-39. DOI 10.1007/s11370-007-0002-3.

- [9] M. Anthony Lewis and Kar-Han Tan. (1997). High Precision Formation Control of Mobile Robots Using Virtual Structures. *Autonomous Robots* 4, 387-403, 1997 Kluwer Academic Publishers. Printed The Netherlands.
- [10] Dong Sun, Can Wang, Wen Shang, and Gang Feng. *A Synchronization Approach to Trajectory Tracking of Multiple Mobile Robots While Maintaining Time-Varying Formations*, IEEE Transactions On Robotics, Vol. 25, No. 5, October 2009.
- [11] D. Kostić, S. Adinandra, J. Caarls, N. van de Wouw and H. Nijmeijer. (2010). *Saturated Control of Time-Varying Formations and Trajectory Tracking for Unicycle Multi-agent Systems*, 49th IEEE Conference on Decision and Control. December 15-17, 2010.
- [12] Zhe Liu, Weidong Chen, Junguo Lu, and Hesheng Wang. (2016). Synchronous Trajectory Tracking for Mobile Robot Network without velocity measurements between coupling robots. *Proceedings of The 2016 IEEE International Conference on Real-time Computing and Robotics*, June 6-9, 2016, Angkor Wat, Cambodia.
- [13] Ignacio Mas, Ognjen Petrovic and Christopher Kitts. (2008). Cluster Space Specification and Control of a 3-Robot Mobile System, *2008 IEEE International Conference on Robotics and Automation* Pasadena, CA, USA, May 19-23, 2008.
- [14] Mohammad Tariq Nasir and Sami El-Ferik. Adaptive sliding-mode cluster space control of a non-holonomic multi-robot system with applications. *IET Control Theory Appl.*, 2017, Vol. 11 Iss. 8, pp. 1264-1273.
- [15] I. Mas, J. Acain, O. Petrovic, C. Kitts. Error characterization in the vicinity of singularities in multi-robot cluster space control, *IEEE International Conference on Robotics and Biomimetics, 2008 (ROBIO 2008)* (2008) 1911-1917.
- [16] C.A. Kitts, K. Stanhouse, P. Chindaphorn. Cluster space collision avoidance for mobile two-robot systems, *IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009 (IROS 2009)* (2009) 1941-1948.
- [17] Ignacio Mas, and Christopher Kitts. Obstacle Avoidance Policies for Cluster Space Control of Nonholonomic Multirobot Systems, *IEEE/ASME Transactions On Mechatronics*, Vol. 17, No. 6, December 2012.
- [18] Ignacio Mas and Christopher Kitts. Centralized and Decentralized Multi-Robot Control Methods using the Cluster Space Control Framework. *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pp. 115-122.
- [19] Zhiqiang Yu, Gao Meng, Huaping Liu, Xiaoyan Deng, Jianhua Liu, Qiurui Wu and Yuewei Liu. (2008). Dynamic Obstacle Avoidance in Polar Coordinates for Mobile Robot Based on Laser Radar. *2008 IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*.

- [20] Ramdane Hedjar and Messaoud Bounkhel. (2014). Real-time Obstacle Avoidance for a Swarm of Autonomous Mobile Robots. *International Journal of Advanced Robotic Systems*. DOI: 10.5772/58478
- [21] K. Fujimura and H. Samet. (1989). A Hierarchical Strategy for Path Planning Among Moving Obstacles. *IEEE Transactions On Robotics and Automation*, Vol. 5, No. 1, February 1989.
- [22] David Wilkie, Jur van den Berg and Dinesh Manocha. (2009). Generalized Velocity Obstacles. *The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 11-15, 2009 St. Louis, USA.
- [23] F. Large, C. Laugier and Z. Shiller. (2005). Navigation Among Moving Obstacles Using the NLVO: Principles and Applications to Intelligent Vehicles. *2005 Springer Science + Business Media, Inc. Printed Netherlands*.
- [24] Jur van den Berg, Jamie Snape, Stephen J. Guy and Dinesh Manocha. (2011). Reciprocal Collision Avoidance with Acceleration-Velocity Obstacles. *2011 IEEE International Conference on Robotics and Automation, Shanghai International Conference Center*, May 9-13, 2011, Shanghai, China.
- [25] Jur van den Berg, Ming Lin and Dinesh Manocha. (2008). Reciprocal Velocity Obstacles for Real-Time Multi-Agent Navigation. *2008 IEEE International Conference on Robotics and Automation*. Pasadena, CA, USA, May 19-23, 2008.
- [26] S.S. GE and Y.J. CUI. (2002). *Dynamic Motion Planning for Mobile Robots Using Potential Field Method*, *Autonomous Robots* 13, 207–222, 2002.
- [27] Luis Valbuena and Herbert G. Tanner. (2012). Hybrid Potential Field Based Control of Differential Drive Mobile Robots. *J Intell Robot Syst* 68, 307-322 (2012). <https://doi.org/10.1007/s10846-012-9685-6>
- [28] Liu Chuan-Ling, Tang Jing and Yang Jing-Yu, . (2011). Path Planning of Mobile Robot Using New Potential Field Method in Dynamic Environments. *2011 Seventh International Conference on Natural Computation*.
- [29] R. Tron, J. Thomas, G. Loianno, K. Daniilidis, and V. Kumar. A distributed optimization framework for localization and formation control: applications to vision-based measurements, *IEEE Control Syst. Mag.*, vol. 36, no. 4, pp. 22-44, 2016.
- [30] Jian Chen, Darren M. Dawson, Warren E. Dixon, and Aman Behal. Adaptive Homography-Based Visual Servo Tracking for a Fixed Camera Configuration With a Camera-in-Hand Extension. *IEEE Transactions On Control Systems Technology*, Vol. 13, No. 5, September 2005.

- [31] N. R. Gans, A. P. Dani and W. E. Dixon. Visual Servoing to an Arbitrary Pose with Respect to an Object Given a Single Known Length. 2008 American Control Conference Westin Seattle Hotel, Seattle, Washington, USA, June 11-13, 2008.
- [32] A. P. Dani, N. Gans, and W. E. Dixon. Position-Based Visual Servo Control of Leader-Follower Formation Using Image-Based Relative Pose and Relative Velocity Estimation. 2009 American Control Conference Hyatt Regency Riverfront, St. Louis, MO, USA June 10-12, 2009.
- [33] Ricardo Carelli, Carlos M. Soria and Beatriz Morales. Vision-Based Tracking Control for Mobile Robots. ICAR 2005. Proceedings., 12th International Conference on Advanced Robotics, 2005.
- [34] Hasan Poonawala, Aykut C Satici, Nicholas Gans and Mark W Spong. Formation Control of Wheeled Robots with Vision-Based Position Measurement. 2012 American Control Conference, Fairmont Queen Elizabeth, Montréal, Canada, June 27-June 29, 2012.
- [35] Jie Lin, Zhiqiang Miao, Hang Zhong, Weixing Peng, Yaonan Wang, and Rafael Fierro. Adaptive Image-Based Leader-Follower Formation Control of Mobile Robots With Visibility Constraints, IEEE Transactions On Industrial Electronics, 2020. DOI 10.1109/TIE.2020.2994861.
- [36] Xinwu Liang, Yun-Hui Liu, Hesheng Wang, Weidong Chen, Kexin Xing, and Tao Liu. Leader-Following Formation Tracking Control of Mobile Robots Without Direct Position Measurements. IEEE Transactions On Automatic Control, Vol. 61, NO. 12, December 2016.
- [37] Luyang Li, Yun-Hui Liu, Tianjiao Jiang, Kai Wang, and Mu Fang. Adaptive Trajectory Tracking of Nonholonomic Mobile Robots Using Vision-Based Position and Velocity Estimation. IEEE Transactions On Cybernetics, Vol. 48, No. 2, February 2018.
- [38] Kai Wang, Yunhui Liu, and Luyang Li. Visual Servoing Trajectory Tracking of Nonholonomic Mobile Robots Without Direct Position Measurement. IEEE Transactions On Robotics, Vol. 30, No. 4, August 2014.
- [39] Gian Luca Mariottini, Domenico Prattichizzo, and Giuseppe Oriolo. Image-based Visual Servoing for Nonholonomic Mobile Robots with Central Catadioptric Camera. Proceedings of the 2006 IEEE International Conference on Robotics and Automation. Orlando, Florida - May 2006.
- [40] Xinwu Liang, Hesheng Wang, Yun-Hui Liu, Weidong Chen, and Zhongliang Jing. Image-Based Position Control of Mobile Robots With a Completely Unknown Fixed Camera, IEEE Transactions on Automatic Control, vol. 63, no. 9, Sept. 2018. DOI 10.1109/TAC.2018.2793458.

- [41] C. Canudas, B. Siciliano, G. Bastin, B. Brogliato, G. Campion, B. D'AndreaNovel, A. De Luca, W. Khalil, R. Lozano, R. Ortega, C. Samson, and P. Tomei. *Theory of Robot Control*. Springer-Verlag, London, 1996.
- [42] C. Samson, "Velocity and torque feedback control of a nonholonomic cart," in *Advanced robot control*. Springer, 1991, pp. 125-151.
- [43] C. Canudas de Wit, G. Bastin, and B. Siciliano, Eds., *Theory of Robot Control*, 1st ed. Berlin, Heidelberg: Springer-Verlag, 1996.
- [44] H. M. Becerra and C. Sagues. Pose-Estimation-Based Visual Servoing for Differential-Drive Robots using the 1D Trifocal Tensor. The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, October 11-15, 2009 St. Louis, USA.
- [45] https://docs.opencv.org/4.x/d9/d0c/group__calib3d.html, último acceso en Mayo de 2022.
- [46] X. Liang, H. Wang, W. Chen, D. Guo, and T. Liu. "Adaptive image- based trajectory tracking control of wheeled mobile robots with an uncalibrated fixed camera," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2266-2282, 2015.
- [47] S. Garrido-Jurado, R. Muñoz-Salinas, F. J. Madrid-Cuevas, and M. J. Marín-Jiménez. 2014. "Automatic generation and detection of highly reliable fiducial markers under occlusion". *Pattern Recogn.* 47, 6 (June 2014), 2280-2292. DOI=10.1016/j.patcog.2014.01.005
- [48] Tharindu Weerakoon, Kazuo Ishii and Amir Ali Forough Nassiraei. (2014). Dead-lock Free Mobile Robot Navigation Using Modified Artificial Potential Field. *SCIS and ISIS 2014*, Kitakyushu, Japan, December 3-6, 2014.
- [49] D. Kostić, S. Adinandra, J. Caarls, N. van de Wouw, and H. Nijmeijer. Saturated Control of Time-Varying Formations and Trajectory Tracking for Unicycle Multi-agent Systems. 49th IEEE Conference on Decision and Control December 15-17, 2010.
- [50] C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Synchronization Approach to Formation Control of Mobile Robots from the Cluster Space Perspective. *Journal of Intelligent & Robotic Systems* 103, 56 (2021) DOI: 10.1007/s10846-021-01495-y
- [51] Christopher A. Kitts, and Ignacio Mas. Cluster Space Specification and Control of Mobile Multirobot Systems. *IEEE/ASME Transactions On Mechatronics*, Vol. 14, No. 2, April 2009.

- [52] V. T. L. Rampinelli, A. S. Brandao, M. Sarcinelli-Filho, F. N. Martinsy and R. Carelli. (2010). Embedding Obstacle Avoidance in the Control of a Flexible MultiRobot Formation, 2010 IEEE International Symposium on Industrial Electronics, pp. 1846-1851.
- [53] C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Formation Control of Unicycle Mobile Robots using Cluster Space Approach in Dynamic Environments. *Proceedings of the 44th Annual Conference of the IEEE Industrial Electronics Society 2018*. Washington D.C., USA, October 21-23, 2018, pp. 2528-2533.
- [54] Ignacio Mas, Ognjen Petrovic and Christopher Kitts. Cluster Space Specification and Control of a 3-Robot Mobile System, 2008 IEEE International Conference on Robotics and Automation Pasadena, CA, USA, May 19-23, 2008.
- [55] C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Formation Control to Unicycle Mobile Robots using the Cluster Space Approach with Internal Controller (in spanish). *Memorias del XX Congreso Mexicano de Robótica 2018*. September 12-14, 2018, Ensenada, Baja California, México.
- [56] C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Formation Control of 3 Unicycle Mobile Robots in Dynamic Environments. *Memorias del XXI Congreso Mexicano de Robótica 2019*. November 13 -15, 2019, Manzanillo, Colima, México.
- [57] T.H.A. van den Broek, N. van de Wouw, H. Nijmeijer, *Formation Control of Unicycle Mobile Robots: a Virtual Structure Approach*, IEEE Conf. Dec. Control, pp. 3264-3269, Shanghai, China, 2009.
- [58] Anna Sadowska, Thijs van den Broek, Henri Huijberts, Nathan van de Wouw, Dragan Kostić and Henk Nijmeijer. A virtual structure approach to formation control of unicycle mobile robots using mutual coupling. *International Journal of Control* Vol. 84, No. 11, November 2011, 1886-1902.
- [59] H. K. Khalil, *Nonlinear Systems*, ser. Pearson Education. Prentice Hall, 2002.
- [60] NaturalPoint, OptiTrack, <http://www.naturalpoint.com/optitrack/>, último acceso en Diciembre de 2021.
- [61] <http://emanual.robotis.com/docs/en/platform/turtlebot3/overview/>, último acceso en Diciembre de 2021.
- [62] YoonSeok Pyo, HanCheol Cho, RyuWoon Jung, and TaeHoon Lim, "ROS Robot Programming". *A Handbook Written by TurtleBot3 Developers*, ROBOTIS Co.,Ltd. Dec 22, 2017.

- [63] C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. (2021). “Leader-follower formation with reduction of the off-tracking and velocity estimation under visibility constraints”, *International Journal of Advanced Robotic Systems*. DOI: 10.1177/172988142111057691
- [64] Qing Zheng, Linda Q. Gao, and Zhiqiang Gao. On Validation of Extended State Observer Through Analysis and Experimentation, *Journal of Dynamic Systems, Measurement, and Control*, March 2012, Vol. 134, pp. 024505.
- [65] Carlos Aguilar-Ibanez, Hebertt Sira-Ramirez and José Angel Acosta. Stability of active disturbance rejection control for uncertain systems: A Lyapunov perspective, *Int. J. Robust Nonlinear Control*, 2017, DOI: 10.1002/rnc.3812.
- [66] W. Wan, B. Shi, Z. Wang and R. Fukui, “Multirobot Object Transport via Robust Caging,” in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 1, pp. 270-280, Jan. 2020, doi: 10.1109/TSMC.2017.2733552.

Apéndice A

Artículos publicados

A continuación se presenta un lista de los artículos publicados durante el periodo Agosto de 2017 a Octubre de 2021.

A.1 Artículos de revista indizadas al JCR

C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. (2021). “Leader-follower formation with reduction of the off-tracking and velocity estimation under visibility constraints”, *International Journal of Advanced Robotic Systems*. DOI: 10.1177/172988142111057691

C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Synchronization Approach to Formation Control of Mobile Robots from the Cluster Space Perspective. *Journal of Intelligent & Robotic Systems* 103, 56 (2021) DOI: 10.1007/s10846-021-01495-y

A.2 Artículo de Congreso Internacional

C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Formation Control of Unicycle Mobile Robots using Cluster Space Approach in Dynamic Environments. *Proceedings of the 44th Annual Conference of the IEEE Industrial Electronics Society 2018*. Washington D.C., USA, October 21-23, 2018, pp. 2528-2533.

A.3 Artículos de Congreso Nacional

C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Formation Control of 3 Unicycle Mobile Robots in Dynamic Environments. *Memorias del XXI Congreso Mexicano de Robótica 2019*. November 13 -15, 2019, Manzanillo, Co-

lima, México.

C. Mauricio Arteaga-Escamilla, Rafael Castro-Linares and Jaime Álvarez-Gallegos. Formation Control to Unicycle Mobile Robots using the Cluster Space Approach with Internal Controller (in spanish). *Memorias del XX Congreso Mexicano de Robótica 2018*. September 12-14, 2018, Ensenada, Baja California, México.