



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

UNIDAD ZACATENCO

Laboratorio UMI-LAFMIA 3175 CNRS

**Seguimiento de trayectoria con evasión de obstáculos de un
vehículo submarino no tripulado mediante un sistema de
guía, navegación y control**

T E S I S

Que presenta

M. en C. Adrián Manzanilla Magallanes

Para obtener el grado de

Doctor en Ciencias

en

Sistemas Autónomos de Navegación Aérea y Submarina

Director de Tesis:

Dr. Sergio Rosario Salazar Cruz

Ciudad de México

Dedicatoria

A mi esposa:

Jessica Jennifer Ayala Ojeda

Por estar conmigo en situaciones difíciles incluso cuando el estudio y el trabajo ocuparon la mayor parte de mi tiempo. Este proyecto no fue fácil, pero estuviste motivándome para alcanzar esta meta. Por ello, eres el pilar principal para la culminación de este proyecto, te lo agradezco; eres mi inspiración y mi motivación.

A mis hijas:

Camila Manzanilla Ayala y Regina Manzanilla Ayala

Su afecto, cariño y amor son los detonantes de mi vida, son una fuente de motivación e inspiración para superarme cada día más, son unas princesas que han iluminado mi vida y hacen mi camino más claro. Dedico este trabajo a esas dos personas que a su corta edad me han dado lecciones de vida.

A mi padre:

José Jesús Manzanilla Valencia

Me has apoyado incondicionalmente, siempre con palabras de motivación para impulsarme en los momentos más difíciles. Te agradezco porque hoy puedo ver alcanzada mi meta y puedo decir que este es un logro de ambos.

A mis hermanos:

Jesús y Evelía

Les agradezco porque con su amor y apoyo he logrado cumplir mis metas y sueños. Son un respaldo incondicional, a quienes les deseo el mejor de los éxitos y una vida llena de logros.

Agradecimientos

A través de estas líneas quiero expresar mi agradecimiento a todas las personas que han colaborado en la realización de este trabajo de investigación.

En primer lugar, al Consejo Nacional de Ciencia y Tecnología (**CONACyT**) le agradezco por otorgarme una beca para lograr este grado académico, así como los múltiples apoyos para estancias y proyectos.

Al **Dr. Sergio Salazar Cruz** expreso mi profundo agradecimiento por las aportaciones y la acertada orientación para guiarme en la realización de este trabajo de tesis.

Al **Dr. Rogelio Lozano Leal**, por sus aportaciones e interés en el desarrollo y conclusión de este trabajo.

Al **Dr. Filiberto Muñoz Palacios**, por su apoyo para el desarrollo y conclusión de este proyecto.

Al **Dr. Eduardo Steed Espinoza Quesada** y al **Dr. Jorge Said Cervantes Rojas** les agradezco su tiempo y dedicación para sugerir los cambios adecuados para mejorar el manuscrito final y fungir como miembros del jurado evaluador.

A **mis amigos** que me han mostrado mil veces en su propio ejemplo lo que significa ser una gran persona.

A **Dios**, por haberme dado la sabiduría y la fortaleza para que fuera posible alcanzar esta meta.

Notaciones

AHRS	Sistema de referencia de orientación y rumbo.
AUV's	Vehículos submarinos autónomos.
BIA	Bakstepping integral adaptativo.
CAD	Dibujo asistido por computadora.
C_{β}	Centro de flotabilidad.
C_G	Centro de gravedad.
DVL	Registro de velocidad Doppler.
EKF	Filtro extendido de Kalman.
GDL	Grados de libertad.
GNC	Guía, navegación y control.
GPS	Sistema de posicionamiento global.
GUI	Interfaz gráfica de usuario.
I-AUV's	Vehículos submarinos autónomos de intervención.
IMU	Unidad de medición inercial.
NOAA	Oficina nacional de administración oceánica y atmosférica.
PD + G	Proporcional derivativo con compensación de fuerzas restitutivas.
ROV's	Vehículos remotamente operados.
ROS	Sistema operativo robótico.
SNAME	Sociedad de arquitectos navales e ingenieros marinos.
UUV's	Vehículos submarinos no tripulados.

Resumen

Este trabajo presenta un sistema de guía, navegación y control para el seguimiento de trayectoria con evasión de obstáculos de un vehículo submarino no tripulado. Este objetivo relaciona el desarrollo de métodos de localización, el diseño de esquemas de control y algoritmos de planificación de trayectorias con evasión de obstáculos.

Los métodos de estimación de pose están basados en dispositivos acústicos y visuales. El primer procedimiento utiliza análisis de intervalos para estimar la posición del robot mediante la fusión de datos de un sonar acústico y un sensor inercial. El segundo método está basado en el uso de un algoritmo de odometría visual y de un filtro de Kalman extendido. Un tercer método es implementado utilizando un sistema de posicionamiento acústico de línea de base corta mediante hidrófonos.

Por otra parte, para la generación de trayectoria se propone el uso de puntos de referencia. Para lograr el seguimiento de la trayectoria deseada se diseñan dos controladores robustos ante perturbaciones del entorno y a las incertidumbres de los parámetros hidrodinámicos del vehículo. Un controlador de modo deslizante integral de segundo orden con un efecto de super-twisting, garantiza compensar las perturbaciones y reducir el efecto de conmutación en la entrada de control. Por otro lado, un controlador backstepping integral con efecto adaptativo se utiliza para estimar las perturbaciones desconocidas. El diseño de ambos controladores se basa en el modelo dinámico de un vehículo submarino y para garantizar la estabilidad a lazo cerrado del sistema el análisis de estabilidad es realizado mediante el método de Lyapunov.

La detección de obstáculos se lleva a cabo mediante un algoritmo que contiene un filtro de intensidad para una nube de puntos proporcionada por mediciones obtenidas por un sonar mecánico y un método de agrupamiento basado en algoritmos K-means. La evasión de obstáculos se logra mediante la implementación del algoritmo de campo de potencial tangencial, los resultados muestran que el vehículo puede seguir una trayectoria compuesta por varios puntos de referencia incluso si existen múltiples obstáculos, además la técnica utilizada evita el problema de mínimo local presentado en los algoritmos de campos de potencial artificial.

Abstract

This work presents a guidance, navigation and control system for trajectory tracking with obstacle avoidance of an unmanned underwater vehicle. This objective relates the development of location methods, the design of control schemes and trajectory planning algorithms with obstacle avoidance.

The pose estimation methods are based on acoustic and visual devices. The first procedure uses interval analysis to estimate the robot's position by fusion data from an acoustic sonar and inertial sensor. The second method is based on the use of a visual odometry algorithm and extended Kalman filter. A third method is implemented using hydrophones in a short baseline acoustic positioning system.

On the other hand, for the generation of trajectory the use of reference points is proposed. To achieve a reference trajectory tracking, two robust controllers are designed against environmental disturbances and the uncertainties of the hydrodynamic parameters of the vehicle. A integral second-order sliding mode controller with a super-twisting effect, guarantees to compensate for disturbances and reduce the chattering effect at the control input. Moreover, a backstepping controller with an adaptive effect is used to estimate unknown disturbances. The design of both controllers is based on the dynamic model of an underwater vehicle. To guarantee the closed-loop stability of the vehicle, the stability analysis is performed by the Lyapunov method.

Obstacle detection is carried out by an algorithm that contains an intensity filter for a point cloud provided by measurements obtained by a mechanical sonar and a clustering method based on K-means algorithms. Obstacle avoidance is achieved by implementing the tangential potential field algorithm, the results show that the vehicle can track a trajectory composed by several reference points even have exist multiple obstacles, also the technique used avoids the local minimum problem presented in artificial potential fields algorithms.

Índice general

Dedicatoria	III
Agradecimientos	V
Notaciones	VII
Resumen	IX
Abstract	XI
Índice de tablas	XVI
Índice de figuras	XVII
1. Introducción	1
1.1. Antecedentes	2
1.2. Planteamiento del problema	4
1.3. Propuesta de solución	5
1.4. Justificación	7
1.5. Hipótesis del trabajo	8
1.6. Objetivos	8
1.6.1. Objetivo general	8
1.6.2. Objetivos específicos	8
1.7. Alcance de la solución	9
1.8. Publicaciones resultado de la tesis	9
1.8.1. Revistas	9
1.8.2. Capítulo de libro	10
1.8.3. Congresos internacionales	10
1.9. Estructura del documento	11
2. Modelado de vehículos submarinos	12
2.1. Plataforma experimental “LIRMIA III”	13
2.1.1. Modelo cinemático	13
2.1.2. Modelo dinámico	16
2.2. Plataforma experimental “BlueROV 2”	24
2.2.1. Modelo cinemático	24
2.2.2. Modelo dinámico	25

2.3. Conclusión del capítulo	29
3. Diseño del sistema de control	30
3.1. Control proporcional derivativo con compensación de fuerzas restitutivas .	32
3.1.1. Estrategia de control	32
3.2. Control de modo de deslizantes integral con efecto de super-twisting	34
3.2.1. Estrategia de control	35
3.3. Bakstepping integral adaptativo	42
3.3.1. Estrategia de control	42
3.4. Conclusión del capítulo	45
4. Integración del sistema de guía y navegación	47
4.1. Sistema de navegación	48
4.1.1. Antecedentes de localización	48
4.1.2. Localización acústica	50
4.1.3. Localización mediante visión	53
4.2. Sistema de guía	55
4.2.1. Generación de trayectoria 2D por puntos de referencia	55
4.2.2. Generación de trayectoria 3D	56
4.2.3. Detección y evasión de obstáculos	57
4.2.3.1. Detección de obstáculos	57
4.2.3.2. Técnicas de evasión de obstáculos	64
4.3. Conclusión del capítulo	71
5. Plataformas experimentales	72
5.1. LIRMIA III	72
5.1.1. Sistema de propulsión	73
5.1.2. Arquitectura electrónica	74
5.1.3. Estación terrestre	74
5.2. BlueROV 2	76
5.2.1. Arquitectura electrónica	77
5.2.2. Sistema de propulsión	77
5.2.3. Estación terrestre	78
5.3. Conclusión del capítulo	80
6. Resultados	81
6.1. Simulaciones numéricas	81
6.1.1. Sistema de control	81
6.1.1.1. Control de modos deslizante integral de segundo orden .	81
6.1.1.2. Bakstepping integral con efecto adaptativo	85
6.1.1.3. Comparativa del PD+G, BIA y SISMC	90
6.1.2. Sistema de guía	92
6.1.2.1. Detección y evasión de obstáculos	92
6.2. Experimentos en tiempo real	99
6.2.1. Sistema de navegación	99
6.2.1.1. Sistema de posicionamiento acústico	99
6.2.2. Sistema de control	106
6.2.2.1. Control de modos deslizante integral de segundo orden .	106
6.2.2.2. Bakstepping integral con efecto adaptativo caso escalar .	108

6.2.2.3. Prueba experimental comparativa de PD+G, BIA, SISMC.	109
6.2.3. Sistema de guía	112
6.2.3.1. Detección de dos obstáculos estáticos	112
6.3. Conclusión del capítulo	114
7. Conclusiones y trabajo futuro	115
7.1. Conclusión general	115
7.2. Conclusiones específicas	115
7.2.1. Sistema de guía	115
7.2.2. Sistema de navegación	116
7.2.3. Sistema de control	116
7.3. Trabajo futuro	117
A. Apéndice	119
Bibliografía	121

Índice de tablas

2.1. Notación para vehículos marinos.	12
5.1. Parámetros físicos y dinámicos del LIMRIA III.	73
5.2. Parámetros físicos y dinámicos del BlueROV 2.	76
6.1. Datos de simulación del controlador SISMC	82
6.2. Criterio de error cuadrático medio para SISMC e ISMC.	84
6.3. Criterio de error cuadrático medio para BIA.	88
6.4. Criterio de error cuadrático medio para prueba comparativa.	92
6.5. Criterio de error cuadrático medio, PTAM.	103
6.6. Criterio de error cuadrático medio, BIA implementado en LIRMIA III.	109
6.7. Criterio de error cuadrático medio para pruebas experimentales comparativas.	110

Índice de figuras

1.1.	Vehículos submarinos en tareas de exploración, inspección y arqueología marina. . .	1
1.2.	Torpedo Whitehead, foto adaptada de [1].	2
1.3.	Vehículo autónomo experimental EAVE, foto adaptada de [2].	3
1.4.	En la imagen se pueden observar UUV's realizando aplicaciones complejas como: recolección de imágenes de un naufragio para realizar una reconstrucción 3D, fo- tografía de la fauna marina y mapeo del lecho marino mediante el uso de sensores acústicos.	4
2.1.	Marcos de referencia del LIRMIA III.	13
2.2.	Fuerzas y momentos en LIRMIA III.	22
2.3.	Marcos de referencia BlueROV 2.	24
2.4.	Fuerzas actuando en BlueROV 2, los propulsores 1 a 4 se utilizan para generar los movimientos de: avance, retroceso, el desplazamiento lateral y rotación en guiñada. Mientras que, los propulsores 5 y 6 se usan para emerger y sumergir el vehículo. . .	27
4.1.	Interacción de los bloques en un sistema GNC.	47
4.2.	Sistema acústico Water Linked, el transceptor se localiza en el UUV y 4 transponder son ubicados cerca de la superficie. La computadora de posicionamiento utiliza al- goritmos para estimar la posición del localizador en función de las señales recibidas de los transponder.	50
4.3.	La GUI de Water Linked GPS nos permite visualizar la posición global y acústica del vehículo, además de configurar la distancia de cada una de las balizas receptoras. Observe, que el nodo de ROS publica en un tópico la estimación de la posición. . .	51
4.4.	Estimación de la posición mediante análisis de intervalos de un vehículo submarino en una alberca utilizando las mediciones del sensor inercial y sonar acústico. . . .	52
4.5.	Etapas del algoritmo de mapeo y seguimiento paralelo (PTAM).	53
4.6.	Predicción de la posición con respecto al mapa generado por el algoritmo de PTAM.	53
4.7.	Principales nodos de ROS utilizados en la implementación PTAM+EKF.	54
4.8.	Trayectoria mediante seguimiento de puntos de referencia.	56
4.9.	Generación de trayectoria 3D, las flechas indican la dirección y el sentido de como se llega a cada punto de referencia en un intervalo de tiempo específico.	57
4.10.	Rango de operación de la ecosonda.	58
4.11.	Escenario de prueba experimental, lado derecho se visualiza la pecera sin obstáculo y del lado izquierdo con el objeto dentro de la pecera.	58
4.12.	Resultados experimentales con la ecosonda mediante ROS.	59
4.13.	Escaneo en una región, la línea azul representa el haz acústico medido.	59

4.14. Escaneo de 360° en una alberca circular.	60
4.15. Mediciones de cada haz acústico y su respectiva intensidad.	60
4.16. Diagramas de dispersión con un umbral de 100, 150, 180 y 190 respectivamente.	61
4.17. Algoritmo K-means, pasos para implementación.	63
4.18. Agrupamiento de datos K-means utilizando los datos filtrados del sonar.	64
4.19. Algoritmo de campo potencial, en la imagen izquierda note que el robot evita la colisión con el objeto debido a la fuerza de repulsión. Sin embargo, como se ve en la imagen derecha en un entorno con múltiples obstáculos se puede presentar la problemática de mínimo local dejando parado al robot debido a que cada obstáculo genera una fuerza de repulsión.	66
4.20. Evasión de obstáculos mediante algoritmos de BUG, imágenes adaptadas de [3].	69
4.21. Simulación de algoritmo de errores BUG tangencial.	70
5.1. LIRMIA III.	72
5.2. CAD LIRMIA III, sistema de propulsión mediante 4 propulsores Seabotix BT150.	73
5.3. Arquitectura electrónica del LIRMIA III.	74
5.4. Estación en tierra LIRMIA III.	75
5.5. UvxCtrl, interfaz de usuario.	75
5.6. BlueROV 2.	76
5.7. Arquitectura electrónica del BlueROV 2.	77
5.8. Esquemático del sistema de propulsión del BlueROV 2.	78
5.9. Estación terrestre del BlueROV 2.	78
5.10. Diagrama de bloques sobre la interacción de los nodos del BlueROV y del Water Linked.	79
6.1. Vista 3D del seguimiento de trayectoria.	83
6.2. Comparaciones de seguimiento de trayectoria en posición y orientación. Trayectoria deseada (línea roja discontinua), trayectoria con SISMC (línea azul sólida) y trayectoria con controlador ISMC (línea naranja sólida).	83
6.3. Entradas de control SISMC con atenuación del efecto de conmutación, mientras que las superficies deslizantes y variables auxiliares que convergen en una región cercana a cero.	84
6.4. Las entradas de control ISMC tienen efecto de conmutación, en tanto que las superficies de deslizamiento y las variables auxiliares oscilan en una región muy cercana a cero.. . . .	85
6.5. Evolución de la posición y guiñada, problema de regulación.	86
6.6. Entradas de control.	86
6.7. Estimación de la perturbación δ_i con $i : x, y, z, \psi$	87
6.8. Seguimiento de una trayectoria elipsoidal.	87
6.9. Entradas de control.	88
6.10. Estimación de perturbaciones, para apreciar de mejor forma la estimación solo se muestran los valores de un intervalo de tiempo de la prueba.	88
6.11. Comportamiento del controlador durante el seguimiento de puntos de referencia.	89
6.12. Seguimiento de trayectoria mediante puntos de referencia, resultados de simulación con perturbaciones externas.	90
6.13. Comportamiento en posición (x, y) usando SISMC, BIA y PD+G.	91
6.14. Comportamiento en profundidad (z) y guiñada (ψ) usando SISMC, BIA y PD+G.	91
6.15. Errores de seguimiento a lo largo de la trayectoria.	91

6.16. Entradas de control, note que el SISMC tiene efecto de conmutación lo que podría generar desgaste mecánico en los propulsores.	92
6.17. Diagramas de dispersión de los datos filtrados.	92
6.18. Agrupación de datos con su centroide utilizando K-means en Python.	93
6.19. Técnica de agrupamiento, puntos rojos definen al clúster 1 mientras que los puntos azules son el clúster 2, los centroides están definidos en círculos amarillos, los puntos naranja y morado definen la ubicación del obstáculo en un entorno acuático.	94
6.20. Posición del objeto durante la prueba, y medición de la longitud del objeto.	94
6.21. Simulación de algoritmo BUG1 para evasión de obstáculos, cuando encuentra un obstáculo lo rodea hasta ubicar la pendiente de la línea recta de la distancia más corta al objetivo.	95
6.22. Simulación del algoritmo de BUG Tangencial en Matlab, los círculos azules definen la trayectoria del robot, los asteriscos verdes denotan la detección del obstáculo mediante el sensor acústico, la superficie roja muestra el obstáculo y el círculo magenta el punto objetivo.	96
6.23. Simulación del algoritmo de evasión de obstáculos por campo potencial tangencial.	97
6.24. Evasión de múltiples obstáculos estáticos.	98
6.25. Seguimiento de camino mediante puntos de referencia con evasión de múltiples obstáculos, donde los rombos verdes son los puntos de referencia a lo largo de la trayectoria y el rombo amarillo es la ubicación inicial del vehículo.	98
6.26. Pruebas experimental, la línea azul representa la localización acústica y la línea roja la posición deseada.	100
6.27. Diseño del filtro pasa bajas de segundo orden mediante herramienta computacional.	100
6.28. Seguimiento de trayectoria utilizando la localización acústica del sistema de posicionamiento Water Linked.	101
6.29. A) BlueROV 2 utilizando localización basada en visión. B) Se utiliza una cámara frontal en el vehículo para obtener los puntos característicos para la localización. C) Mapa y localización generado por el algoritmo PTAM.	102
6.30. Seguimiento de trayectoria autónomo, regulación de profundidad y orientación.	103
6.31. Pruebas para comprobar las limitaciones del algoritmo de visión.	104
6.32. Prueba experimental con distintas perturbaciones externas.	104
6.33. Seguimiento de trayectoria mediante odometría visual, comparación de estimación por PTAM y EKF.	105
6.34. Seguimiento de trayectorias utilizando localización basada en visión: orientación.	105
6.35. Regulación de guiñada (ψ) con SISMC.	106
6.36. Evolución de la superficie deslizante y la entrada de control.	107
6.37. Regulación de profundidad, guiñada y del movimiento traslacional x	107
6.38. Respuesta del control en θ , ψ , z y evolución de las entradas de control.	109
6.39. Respuesta del control PD+G al seguir un perfil sinusoidal en z , ψ	110
6.40. Evolución de las entradas de control PD+G.	111
6.41. Respuesta del control BIA al seguir un perfil sinusoidal en z , ψ	111
6.42. Evolución de las entradas de control BIA.	111
6.43. Respuesta del control SISMC al seguir un perfil sinusoidal en z , ψ	112
6.44. Evolución de las entradas de control SISMC.	112
6.45. Escenario de prueba experimental con dos obstáculos estáticos.	113
6.46. Detección de múltiples objetos mediante el filtro de umbral de intensidad y la técnica de K-means.	113

Introducción

Alrededor de tres cuartas partes de la superficie terrestre está cubierta por agua, de las cuales más del 90 % es agua salada que se distribuye entre los océanos que tienen una gran cantidad de recursos vivos y minerales. En la actualidad, aproximadamente el 20 % de los ambientes acuáticos han sido explorados en base a los datos recopilados por la Oficina Nacional de Administración Oceánica y Atmosférica (NOAA, por sus siglas en inglés), [4]. A lo largo de la historia se han diseñado distintas herramientas para la exploración de entornos acuáticos, por ejemplo, el uso de trajes de buceo con mezclas ternarias han permitido al ser humano hacer inmersiones hasta 150 *m* de profundidad. Sin embargo, la innovación tecnológica de distintos equipos de investigación ha revolucionado el acceso a los océanos mediante vehículos submarinos no tripulados (UUV's, por sus siglas en inglés). Estos vehículos pueden operar en áreas de mayor profundidad y de mayor riesgo, en los últimos años, se han desarrollado un número creciente de UUV's que realizan una amplia gama de aplicaciones que incluyen exploración del lecho marino, inspección de estructuras marinas, monitoreo oceanográfico, cartografía marina, muestreo geológico y arqueología en aguas profundas [5, 6]. Véase, Figura 1.1.

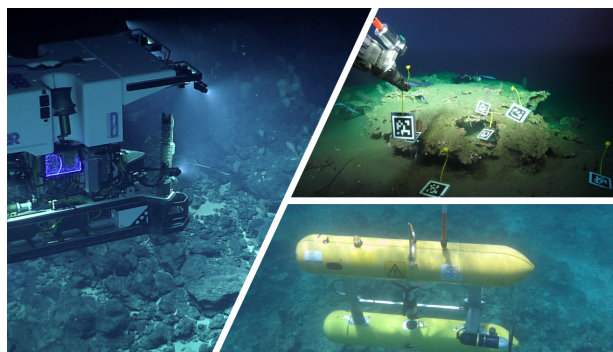


Figura 1.1: Vehículos submarinos en tareas de exploración, inspección y arqueología marina.

Los UUV's por su modo de operación se dividen en vehículos remotamente operados (ROV's, por sus siglas en inglés) y en vehículos submarinos autónomos (AUV's, por sus siglas en inglés). Sin embargo, los AUV's tienen ventajas porque son plataformas totalmente autónomas con cierta inteligencia que le permite realizar misiones bajo el agua sin intervención humana mediante toma de decisiones [7, 8].

1.1. Antecedentes

Vehículos submarinos autónomos

En la actualidad, se conoce a un AUV como un robot que puede sumergirse a distintas profundidades para viajar por largas distancias sin cables conectados y entradas de comando de los operadores de forma independiente, es decir, son plataformas móviles que tienen actuadores, sensores y son controlados a sí mismos mientras realizan una misión predefinida [9]. Sin embargo, el desarrollo de AUV's comenzó en 1866 cuando Robert Whitehead diseñó para la Armada Austro-Húngara un vehículo tipo torpedo que utilizaba un motor por aire comprimido que le permitió navegar hasta 6.5 nudos en una distancia de 200 yardas [10]. Véase, Figura 1.2.

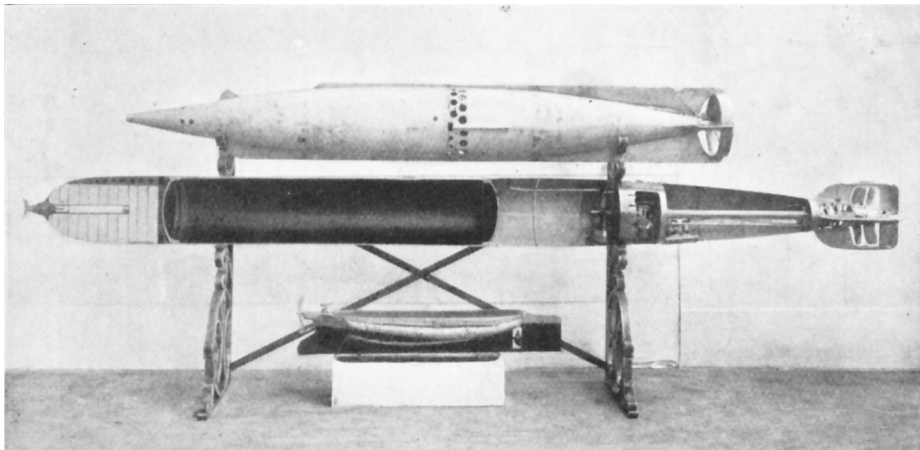


Figura 1.2: Torpedo Whitehead, foto adaptada de [1].

Años más adelante, Stan Murphy, Bob Francois y Terry Ewart del Laboratorio de Física Aplicada de la Universidad de Washington desarrollan un AUV a finales de 1950 con la finalidad de obtener datos oceanográficos a lo largo de trayectorias bajo el hielo y para estudiar la dispersión de estelas submarinas. Su trabajo condujo al desarrollo y operación de vehículos submarinos de investigación autopropulsados (SPURV, por sus siglas en inglés) a principios

de los años 60's y hasta mediados de los 70's [11, 12]. En la siguiente década, la tecnología había avanzado lo suficiente para permitir que estos vehículos operen eficientemente a mayor profundidad que el límite alcanzado por un buzo. Esto motivó a gobiernos de distintos países a desarrollar AUV's para proyectos de investigación con distintas aplicaciones produciendo avances significativos, un ejemplo de ello, es el desarrollo del sistema avanzado de búsqueda no tripulado (EAVE, por sus siglas en inglés) por parte de la Marina de los Estados Unidos en respuesta al hundimiento de embarcaciones y la pérdida de la bomba de Palomares. El vehículo fue lanzado en 1983, contaba con un sistema acústico de comunicación con un alcance de hasta 6 *km* que transmitía imágenes al buque [13], ver Figura 1.3.



Figura 1.3: Vehículo autónomo experimental EAVE, foto adaptada de [2].

En los 90's, el crecimiento en el desarrollo de AUV's pasó de prototipos a una generación de vehículos submarinos capaz de realizar misiones predefinidas. Según el directorio de vehículos submarinos de Busby a principios de la década había seis AUV's operativos y otros quince vehículos adicionales que fueron considerados prototipos [14]. Muestra de ello, es el desarrollo de seis AUV's Odyssey por parte del laboratorio Sea Grant de Massachusetts en el año de 1994, estos vehículos podrían desplazarse a 1.5 *m/s* con una autonomía de 6 horas y tenían el objetivo de hacer una red autónoma de muestreo bajo el hielo a una profundidad de 1.4 *km* [15]. A finales de esta década, se desarrolla el vehículo REMUS en EE.UU para proporcionar a los científicos la capacidad de monitorear los océanos, el vehículo puede operar hasta por 20 horas a 1.5 *m/s* hasta una profundidad de 100 *m*. Actualmente, existen más de 50 vehículos REMUS en 20 diferentes configuraciones operadas de forma independiente por nueve universidades, por la Marina de los EE. UU. y un laboratorio británico [16].

El cambio de siglo marcó el comienzo de las primeras empresas comerciales de AUV's, entre las más importantes se encuentran Kongsberg Maritime, Bluefin, International Submarine Engineering y Hafmynd. En la actualidad, los avances tecnológicos han mejorado la eficiencia en los AUV's y han permitido realizar aplicaciones con mayor grado de complejidad, por ejemplo, navegación y reconstrucción 3D de naufragios [17], construcción de un modelo de terreno digital del fondo marino a partir de mapas del entorno obtenidos por un enjambre de AUV's [18, 19], inspección visual de contenedores de agua en centrales nucleares [20] y proyectos para reconocer fauna marina [21, 22]. Véase, Figura 1.4.

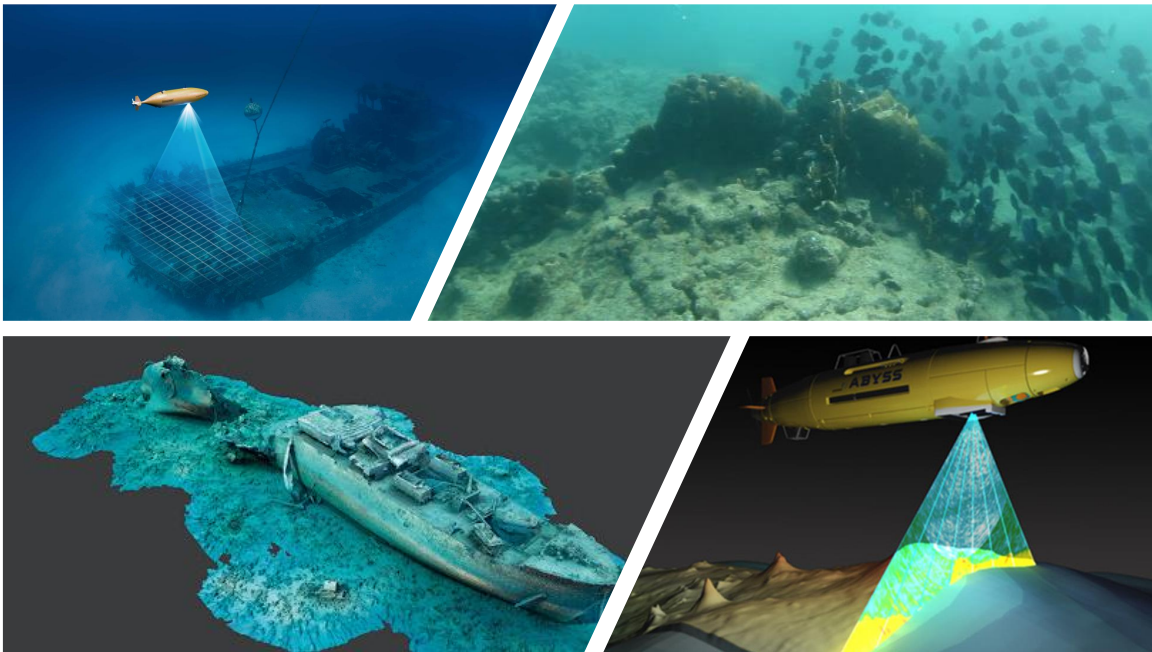


Figura 1.4: En la imagen se pueden observar UUV's realizando aplicaciones complejas como: recolección de imágenes de un naufragio para realizar una reconstrucción 3D, fotografía de la fauna marina y mapeo del lecho marino mediante el uso de sensores acústicos.

Entonces, los AUV's son una herramienta potencial para el área de investigación submarina donde desempeñan un gran papel en los futuros proyectos de estudios subacuáticos realizados por instituciones de investigación en aplicaciones científicas, comerciales y militares.

1.2. Planteamiento del problema

Un AUV debe de tener la capacidad de realizar misiones y ejecutar tareas detalladas por su cuenta. Por ello, es necesario que el robot cuente con un sistema de control de movimiento capaz de reaccionar y tomar decisiones basándose en observaciones de su entorno, aún cuando éste sea completamente desconocido. Para alcanzar este objetivo, el robot requiere de

tres subsistemas independientes para controlar el movimiento: Guía, Navegación y Control (GNC, por sus siglas en inglés). Sin embargo, para que un AUV realice una tarea autónoma se presentan distintos retos, por ejemplo, contar con un sistema de localización preciso para garantizar una navegación fiable contemplando que en entornos acuáticos existe rápida atenuación de la señal del GPS y de las señales de radiofrecuencia. Por otra parte, considerando que los AUV navegan a través del océano en entornos desconocidos con distinto relieve, pueden existir diversos obstáculos que pueden impedir que el vehículo finalice su misión. Por consiguiente, el desafío es que al navegar el vehículo sea capaz de realizar el seguimiento de trayectoria con detección y evasión de objetos. Por ello, es importante tener una estrategia de control que permita al vehículo seguir la trayectoria de referencia. No obstante, para controlar un AUV se tiene que considerar el comportamiento dinámico no lineal del vehículo, las incertidumbres en coeficientes hidrodinámicos y las perturbaciones desconocidas causadas por el oleaje y corrientes marinas.

1.3. Propuesta de solución

Las soluciones que se han dado para resolver los desafíos en la localización de un vehículo submarino de acuerdo al estado del arte, en [23] se presenta el método de navegación por estima el cual utiliza la posición inicial, así como la medición de la velocidad y su orientación del vehículo. Su costo computacional es bajo, no obstante, ocasiona deriva en la estimación de la pose a lo largo del tiempo lo que disminuye la precisión. Por ello, algunos autores proponen el uso de dispositivos acústicos en distintas configuraciones para estimar la posición en tres dimensiones de un robot submarino. Sin embargo, existen deficiencias como: retardos debido al reducido ancho de banda y a la baja velocidad de transmisión de datos, además de que existe pérdida de datos porque la velocidad del sonido fluctúa por el cambio de temperatura y salinidad del agua [24, 25]. Otros métodos, propuestos en [26, 27] utilizan algoritmos de visión para el procesamiento de los datos de la cámara del robot utilizando técnicas de mapeo y localización simultáneos (SLAM, por sus siglas en inglés). Aunque para implementar este tipo de técnicas se requiere de escenas visualmente estáticas sin cambios en la iluminación.

Por otro lado, para el problema de planificación de trayectoria con evasión de obstáculos, en [28] proponen algoritmos basados en muestreo probabilístico porque son adecuados para

una búsqueda rápida de trayectoria fuera de línea, sin embargo, la trayectoria de búsqueda no es óptima; tiene gran costo computacional y dependen del entorno. Por lo cual, en [29] proponen el uso de algoritmos basados en la técnica de campo de potencial artificial para la búsqueda de trayectorias locales con evasión de obstáculos. Sus resultados demuestran que se puede implementar en entornos dinámicos, con un bajo costo computacional. No obstante, presenta el problema de mínimo local. Por ello, otros autores en [30] desarrollan algoritmos inteligentes porque realizan la búsqueda de la trayectoria mediante la información del entorno, sin embargo, presentan alto costo computacional, poca estabilidad y se pueden presentar problemas de optimización.

Particularmente, acerca de la problemática de control se han propuesto diferentes enfoques, por ejemplo, controladores clásicos como: PID [31], PD jerárquico [32], etc. Permiten que el vehículo se estabilice en condiciones ideales, sin embargo, no compensan las incertidumbres del modelo dinámico debido a los parámetros hidrodinámicos ni las perturbaciones del entorno a causa de las corrientes, olas y mareas. Por esta razón, se han desarrollado controladores no lineales e inteligentes, como: backstepping [33, 34], que puede compensar las incertidumbres y su respuesta es suave, no obstante, no garantiza un buen rendimiento debido a que tiene error de estado estable, además de que se necesita la información sobre todos los estados del sistema; modos deslizantes [35], es robusto a las incertidumbres del modelo y a las perturbaciones pero tiene el efecto de conmutación que puede afectar a los propulsores del robot; control adaptativo y redes neuronales [36, 37], que en algunos casos requiere entrenamiento fuera de línea y su respuesta adaptativa puede ser lenta; por ello, otros investigadores en [38] proponen el desarrollo de un control adaptativo basado en una red neuronal dinámica mediante un modelo matemático de un AUV reducido y sin conocimiento de los parámetros del sistema, donde un sistema de control neuronal dinámico permite estimar y compensar las dinámicas no modeladas y perturbaciones externas, además de la estimación de las ganancias que calculan la señal de control.

Del estudio del estado del arte, para solucionar la problemática de localización de un AUV debido a que no se tiene un dispositivo de registro de velocidad Doppler (DVL, por sus siglas en inglés), se propone la estimación de la posición mediante el uso de análisis de intervalos con la fusión de datos de un sonar acústico y un sensor inercial. Además, debido a que solamente contamos con un sonar se propone un método de bajo costo computacional para

localizar un AUV utilizando un algoritmo de visión y un filtro de Kalman extendido (EKF, por sus siglas en inglés). El algoritmo de mapeo y seguimiento paralelo (PTAM, por sus siglas en inglés), es implementado porque separa los procesos de seguimiento y mapeo, lo que permite realizar un procesamiento rápido en la estimación de la posición. Aunque ambos métodos de localización estiman la posición del vehículo con una precisión de centímetros, están limitados. En el caso del PTAM+EKF su operación se restringe por el rango de visión de la cámara monocular y por las condiciones del entorno acuático (turbidez e iluminación). Por otro lado, utilizando la estimación mediante análisis de intervalos se asume un conocimiento de las características del entorno y cuando el vehículo se encuentra en movimiento la localización dinámica se realiza por medio del método de navegación por estima. Por ello, se optó por utilizar un sistema de posicionamiento acústico de línea de base corta.

Por otra parte, para planificar el camino se elige la generación de trayectorias por puntos de referencia con detección de obstáculos en línea mediante el uso de un sonar acústico y el método de evasión de obstáculos por medio del algoritmo de campo potencial tangencial para evitar caer en un mínimo local. Para alcanzar el valor de referencia deseado, se propone el uso de técnicas de control combinadas basadas en modos deslizantes integral con un efecto de super-twisting, para asegurar la compensación de las perturbaciones y reducir el efecto de conmutación en las entradas de control, así como el diseño de un control bakstepping integral con un efecto adaptativo para estimar las perturbaciones y compensar las incertidumbres. Esto dotará al usuario de un vehículo submarino no tripulado con la capacidad de realizar operaciones subacuáticas de manera autónoma como el seguimiento de trayectoria con detección y evasión de obstáculos.

1.4. Justificación

Actualmente, en México existen muy pocos laboratorios e instituciones que trabajen en el estudio de vehículos submarinos no tripulados. Aun cuando se tiene la necesidad de la innovación y desarrollo tecnológico en UUV's con el fin de realizar misiones autónomas para inspeccionar y explorar los recursos naturales que se encuentran disponibles en nuestro país. En el laboratorio UMI-LAFMIA se han desarrollado UUV's, pero carecen de un sistema GNC que les permita realizar tareas autónomas. Por ello, problemas como la localización de un vehículo submarino, la planificación de trayectoria con evasión de obstáculos en línea

y el diseño de una estrategia de control robusta ante las incertidumbres de los parámetros hidrodinámicos y las perturbaciones externas debido a la corriente marina son investigados. Bajo esta perspectiva, este trabajo de investigación presenta un sistema de guía, navegación y control que resuelve la problemática presentada en un UUV al realizar una tarea autónoma.

1.5. Hipótesis del trabajo

El desarrollo adecuado de cada uno de los módulos que componen un sistema de guía, navegación y control para un vehículo submarino no tripulado, permitirá realizar el seguimiento de trayectorias con evasión de obstáculos.

1.6. Objetivos

1.6.1. Objetivo general

Desarrollar un sistema de guía, navegación y control para que un UUV realice una tarea de seguimiento de trayectoria con evasión de obstáculos mediante algoritmos de localización, control y planificación de trayectoria.

1.6.2. Objetivos específicos

1. Desarrollar un sistema de navegación para localizar en un espacio 3D un UUV mediante el uso de dispositivos acústicos o visuales.
2. Diseñar un sistema de control para el seguimiento de trayectorias de un UUV por medio de estrategias de control robustas basadas en técnicas de modos deslizantes y backstepping.
3. Desarrollar un algoritmo de detección de obstáculos para un UUV utilizando dispositivos acústicos.
4. Diseñar un sistema de guía para que un UUV realice la planificación de trayectoria basados en la información de la posición del obstáculo y la localización del vehículo.

5. Validar los algoritmos propuestos mediante herramientas computacionales y pruebas experimentales, para comprobar el funcionamiento del sistema de guía, navegación y control.

1.7. Alcance de la solución

- Este trabajo se valida experimentalmente con dos vehículos, debido a que las plataformas experimentales tienen distintos dispositivos para su localización en un entorno acuático.
- Los escenarios de prueba son entornos controlados de dimensiones reducidas.
- Solo se propone la detección y evasión de obstáculos en dos dimensiones $x - y$, donde los objetos permanecen estáticos.
- La verificación del funcionamiento del algoritmo de planificación de trayectoria con evasión de obstáculos es por medio de simulaciones numéricas.

1.8. Publicaciones resultado de la tesis

El trabajo de investigación que se presenta en esta memoria de tesis ha generado publicaciones en artículos de revista, un capítulo de libro y congresos internacionales. A continuación, se enlistan la compilación de artículos derivados de este trabajo.

1.8.1. Revistas

1. **A. Manzanilla**, S. Reyes, M. Garcia, D. Mercado and R. Lozano. Autonomous Navigation for Unmanned Underwater Vehicles: Real-Time Experiments using Computer Vision. IEEE Robotics and Automation Letters (RA-L) and ICRA19 presentation, vol. 4, no. 2, pp. 1351-1356, April 2019. DOI: 10.1109/LRA.2019.2895272.
2. M. García, **A. Manzanilla**, A. Zamora, F. Muñoz, S. Salazar and R. Lozano. Adaptive Non-Singular Terminal Sliding Mode Control for an Unmanned Underwater Vehicle: Real-Time Experiments. International Journal of Control, Automation and Systems, vol. 18, pp. 615–628, March 2020. DOI: 10.1007/s12555-019-0674-4.

3. **A. Manzanilla**, E. Ibarra, S. Salazar, A. Zamora, R. lozano and F. Muñoz. Super-Twisting Integral Sliding Mode Control for Trajectory Tracking of an Unmanned Underwater Vehicle. Ocean Engineering, OE-D-20-02908, 2021. **En revisión**

1.8.2. Capítulo de libro

1. **A. Manzanilla**, M. Garcia, R. Lozano and S. Salazar. Development and control of an AUV. In Marine Robotics and Applications. Springer, Cham, pp. 87-100, 2018. DOI: 10.1007/978-3-319-70724-2_6.

1.8.3. Congresos internacionales

1. **A. Manzanilla**, P. Castillo and R. Lozano. Nonlinear algorithm with adaptive properties to stabilize an underwater vehicle: real-time experiments. IFAC-PapersOnLine, pp. 6857-6862, 2017, vol. 50, no 1. DOI: 10.1016/j.ifacol.2017.08.1207.
2. J. Guerrero, E. Antonio, **A. Manzanilla**, J. Torres and R. Lozano. Autonomous Underwater Vehicle Robust Path Tracking: Auto-Adjustable Gain High Order Sliding Mode Controller. IFAC-PapersOnLine, pp. 161-166, 2018, vol. 51, no 13. DOI: 10.1016/j.ifacol.2018.07.272.
3. A.E. Zamora, **A. Manzanilla**, M. Garcia, R.Lozano, S. Salazar and F. Muñoz. Depth control of an underwater vehicle using robust PD controller: real-time experiments. In 2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV), pp. 1-6, 2018. DOI: 10.1109/AUV.2018.8729783.
4. A.E. Zamora, M. Garc a, **A. Manzanilla**, R. Lozano, S. Salazar and F. Mu oz. Nonsingular Fast Terminal Sliding Mode Control for an Autonomous Underwater Vehicle. In 2019 16th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), pp. 1-6, 2019, September. DOI: 10.1109/ICEEE.2019.8884505.

1.9. Estructura del documento

Este documento está organizado en siete capítulos, a continuación se hace una breve descripción de cada uno de ellos:

- **Capítulo 1:** Describe el planteamiento del problema de este trabajo de tesis, así como la propuesta de solución. Además se plantea la hipótesis y el objetivo para que un UUV realice el seguimiento de trayectoria con evasión de obstáculos.
- **Capítulo 2:** Presenta el modelo matemático de dos vehículos submarinos de 4 GDL; en ambos casos se analizan las propiedades del vehículo, la cinemática, la dinámica y las consideraciones hechas para simplificar el modelo.
- **Capítulo 3:** Expone un control PD+G y el diseño de algoritmos de control utilizando técnicas de backstepping y modos deslizantes.
- **Capítulo 4:** Describe los sistemas independientes de guía y navegación. Debido a ello, se detallan los métodos de localización usados para estimar la posición del vehículo submarino mediante técnicas basadas en algoritmos de visión, análisis de intervalos y posicionamiento acústico de línea de base corta. Adicionalmente, se proponen los algoritmos de generación de trayectorias por puntos de referencia, así como, los algoritmos de detección de obstáculos utilizando los datos adquiridos por el sonar acústico y las técnicas de evasión de obstáculos basadas en campos potenciales.
- **Capítulo 5:** Presenta las plataformas experimentales.
- **Capítulo 6:** Expone los resultados de los algoritmos propuestos mediante pruebas de simulación numérica y experimentos en tiempo real.
- **Capítulo 7:** Presenta las conclusiones y el trabajo futuro.

Modelado de vehículos submarinos

Este capítulo, presenta el modelo matemático del vehículo submarino utilizado para el diseño de los controladores. De igual forma, se analizan y denotan las ecuaciones que describen el comportamiento del robot. En la literatura, existen diferentes enfoques para obtener el modelo matemático de los vehículos submarinos, por ejemplo, los modelos propuestos por: Graver [39] para vehículos tipo glider donde contempla factores como la masa agregada, mientras Bailey [40] propone un modelo matemático unificado que considera la teoría de navegación y Fossen [41] que propone un modelo vectorial que considera las propiedades de las matrices. La representación propuesta por Fossen, se adopta en este trabajo para describir el modelo no lineal de UUV's y se utiliza la notación descrita por la SNAME [42] para denotar fuerzas, momentos, velocidad y posición. Véase, Tabla 2.1.

	Fuerzas y Momentos	Velocidad lineal y angular en el marco del cuerpo	Posición lineal y angular en el marco inercial
Mov. en dirección del eje x (surge)	X	u	x
Mov. en dirección del eje y (sway)	Y	v	y
Mov. en dirección del eje z (heave)	Z	w	z
Rotación alrededor del eje x (roll)	K	p	ϕ
Rotación alrededor del eje y (pitch)	M	q	θ
Rotación alrededor del eje z (yaw)	N	r	ψ

Tabla 2.1: Notación para vehículos marinos.

Observe que el movimiento de un vehículo marino involucra 6 GDL, estos determinan la posición y orientación del vehículo en el espacio.

2.1. Plataforma experimental “LIRMIA III”

El análisis del modelo matemático del vehículo submarino LIRMIA III, es presentado en dos secciones: cinemática, referente a como se mueve el vehículo y dinámica que describe el análisis de las fuerzas que originan el movimiento.

2.1.1. Modelo cinemático

La cinemática es el estudio del movimiento en el espacio sin considerar las fuerzas que lo producen, debido a ello, este estudio se basa principalmente en la consideración de dos movimientos: traslación y rotación. Para ello, es importante considerar en este análisis dos marcos de referencia: el sistema fijo al cuerpo del vehículo (B) en el cuál se expresan la velocidad lineal y angular. Mientras que el sistema de coordenadas fijo en la tierra (I), describe la posición y orientación del vehículo. Véase, Figura. 2.1.

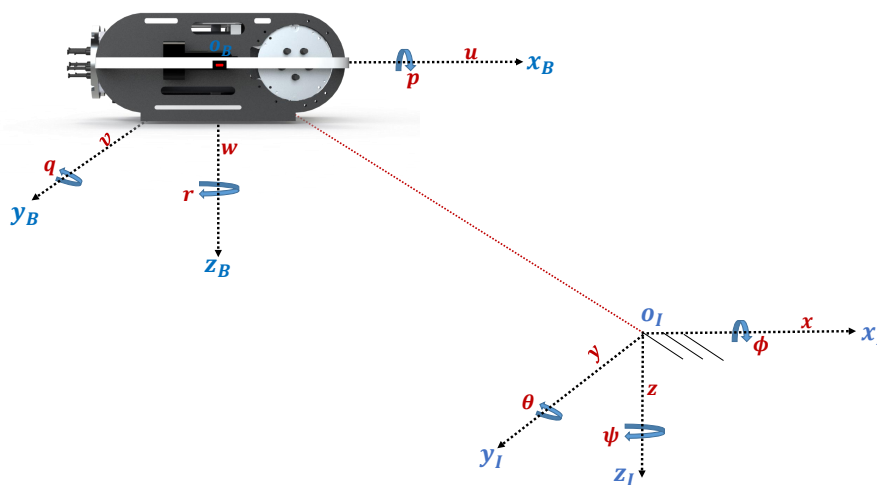


Figura 2.1: Marcos de referencia del LIRMIA III.

Considerando que se simplificará la representación del modelo dinámico del vehículo, se asumen las suposiciones que se describen a continuación:

Suposición. *El origen del sistema del marco fijo al cuerpo (O_B) está ubicado en el mismo punto que el centro de gravedad (C_G) del vehículo.*

Suposición. *Se establece que los ejes (x_B, y_B, z_B) son paralelos a los ejes principales de inercia.*

Suposición. *El vehículo submarino se desplaza a bajas velocidades, es decir ($< 2m/s$).*

Observe, que el sistema de coordenadas fijo a la tierra es considerado un marco inercial y su movimiento es descrito por las ecuaciones de Newton Euler. Donde los vectores que describen el movimiento del submarino se denotan como: $\nu = [\nu_1^T, \nu_2^T]^T \in \mathbb{R}^6$ con $\nu_1 = [u, v, w]^T$ que define las velocidades lineales y $\nu_2 = [p, q, r]^T$ que representan las velocidades angulares del vehículo en el marco del cuerpo, en tanto que, $\eta = [\eta_1^T, \eta_2^T]^T \in \mathbb{R}^6$ con $\eta_1 = [x, y, z]^T$ que denota la posición y $\eta_2 = [\phi, \theta, \psi]^T$ que define la orientación del vehículo en el marco inercial.

Transformación entre marcos de referencia

Las principales matrices de rotación en aplicaciones de navegación y control en vehículos submarinos, se pueden consultar en [43, 44] y corresponden a $R_{\phi,x}$, $R_{\theta,y}$ y $R_{\psi,z} \in \mathbb{R}^{3 \times 3}$. Estas matrices son utilizadas para llevar a cabo transformaciones de velocidad angular y lineal, se denotan de la siguiente manera:

$$R_{\phi,x} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \phi & -\sin \phi \\ 0 & \sin \phi & \cos \phi \end{bmatrix}, R_{\theta,y} = \begin{bmatrix} \cos \theta & 0 & \sin \theta \\ 0 & 1 & 0 \\ -\sin \theta & 0 & \cos \theta \end{bmatrix}, R_{\psi,z} = \begin{bmatrix} \cos \psi & -\sin \psi & 0 \\ \sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2.1)$$

La transformación rotacional entre el marco fijo en la tierra (I) y el marco fijo al cuerpo (B) se puede representar matemáticamente por la secuencia de rotación siguiente:

$$J_1(\eta_2) := R_{\psi,z} R_{\theta,y} R_{\phi,x},$$

$$J_1(\eta_2) = \begin{bmatrix} \cos \theta \cos \psi & \cos \psi \sin \theta \sin \phi - \cos \phi \sin \psi & \cos \phi \cos \psi \sin \theta + \sin \phi \sin \psi \\ \cos \theta \sin \psi & \cos \phi \cos \psi + \sin \theta \sin \phi \sin \psi & \cos \phi \sin \theta \sin \psi - \cos \psi \sin \phi \\ -\sin \theta & \cos \theta \sin \phi & \cos \theta \cos \phi \end{bmatrix}, \quad (2.2)$$

mediante la cual se obtiene la transformación de la velocidad lineal entre el marco de referencia del cuerpo (B) y el marco inercial (I). Entonces,

$$\dot{\eta}_1 = J_1(\eta_2) \nu_1, \quad (2.3)$$

debido a que la matriz de rotación dada en (2.2) es ortogonal, se tiene que

$$\nu_1 = J_1^{-1}(\eta_2)\dot{\eta}_1 = J_1^T(\eta_2)\dot{\eta}_1. \quad (2.4)$$

Ahora, para la transformación de velocidad angular entre el marco coordenado fijo en la tierra (I) y el marco coordenado fijo en el cuerpo (B), se tiene que:

$$\dot{\eta}_2 = J_2(\eta_2)\nu_2, \quad (2.5)$$

donde $J_2(\eta_2) \in \mathbb{R}^{3 \times 3}$ y esta definida como:

$$J_2(\eta_2) = \begin{bmatrix} 1 & 0 & -\sin \theta \\ 0 & \cos \phi & \cos \theta \sin \phi \\ 0 & -\sin \phi & \cos \theta \cos \phi \end{bmatrix}, \quad (2.6)$$

en este caso en particular $J_2(\eta_2)$ no es ortogonal, es decir, la inversa de la matriz $J_2(\eta_2)$ no es igual a su transpuesta, entonces se tiene que:

$$\nu_2 = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_{\phi,x}^T \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + R_{\phi,x}^T R_{\theta,y}^T \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} = J_2^{-1}(\eta_2)\dot{\eta}_2,$$

desarrollando la ecuación anterior tenemos que,

$$J_2^{-1}(\eta_2) = \begin{bmatrix} 1 & \sin \phi \tan \theta & \cos \phi \tan \theta \\ 0 & \cos \phi & -\sin \phi \\ 0 & -\sin \phi / \cos \theta & \cos \phi / \cos \theta \end{bmatrix}, \quad (2.7)$$

Entonces, la ecuación que describe la cinemática de un vehículo submarino de 6 GDL es:

$$\dot{\eta} = J(\eta)\nu, \quad (2.8)$$

donde

$$J(\eta) = \begin{bmatrix} J_1(\eta_2) & 0_{3 \times 3} \\ 0_{3 \times 3} & J_2(\eta_2) \end{bmatrix}, \quad (2.9)$$

para más detalles ver [44], capítulo 2.

2.1.2. Modelo dinámico

La representación vectorial propuesta por Fossen en [41], propone las ecuaciones para describir el análisis de las fuerzas que originan el movimiento del vehículo submarino con respecto al marco fijo al cuerpo (B). Se pueden definir como:

$$\begin{aligned} M\dot{\nu} + C(\nu)\nu + D(\nu)\nu + g(\eta) &= \tau + w \\ \dot{\eta} &= J(\eta)\nu, \end{aligned} \quad (2.10)$$

donde M representa la matriz de inercia, $C(\nu)$ describe los de términos Coriolis y fuerza centrípeta, $D(\nu)$ describe la matriz de amortiguamiento hidrodinámico, $g(\eta)$ es el vector de la fuerza gravitacional y empuje hidrostático, $\tau = [\tau_1, \tau_2]^T = [(X, Y, Z), (K, M, N)]^T$ representa el vector de fuerzas y momentos, w denota el vector de perturbación y $J(\eta)$ es la transformación cinemática entre el marco del cuerpo y el marco inercial.

Matriz de inercia

Esta matriz considera la masa del vehículo y los parámetros del tensor de inercia, está conformada por la matriz de inercia del cuerpo rígido M_{RB} y la matriz de inercia de masa agregada M_A . Por lo tanto, la matriz de inercia se define como:

$$M = M_{RB} + M_A, \quad (2.11)$$

Es importante mencionar que la matriz M es simétrica, definida positiva y satisface:

$$M = M^T > 0; \quad \dot{M} = (0)_{6 \times 6}. \quad (2.12)$$

Para más detalles, ver [44]. De la ecuación 2.11, se tiene que la matriz de inercia del cuerpo rígido M_{RB} se puede representar de manera general como:

$$M_{RB} = \begin{bmatrix} mI_{3 \times 3} & -mS(r_G) \\ mS(r_G) & I_I \end{bmatrix}, \quad (2.13)$$

donde m es la masa del vehículo, r_G denota el vector de la posición del centro de gravedad con respecto al marco fijo al cuerpo (B), I_I los parámetros debidos al tensor de inercia,

S representa una matriz antisimétrica e $I_{3 \times 3}$ define una matriz identidad. Expandiendo la ecuación (2.13) se tiene:

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m & 0 & -mz_g & 0 & mx_g \\ 0 & 0 & m & my_g & -mx_g & 0 \\ 0 & -mz_g & my_g & I_x & 0 & 0 \\ mz_g & 0 & -mx_g & 0 & I_y & 0 \\ -my_g & mx_g & 0 & 0 & 0 & I_z \end{bmatrix}, \quad (2.14)$$

donde I_x, I_y, I_z son los momentos de inercia y x_g, y_g, z_g son elementos del vector r_g . Por otro lado, analizando la forma del vehículo y considerando que se desplaza a bajas velocidades, la matriz M_A puede ser definida como:

$$M_A = - \begin{bmatrix} X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{\dot{v}} & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{\dot{w}} & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{\dot{p}} & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{\dot{q}} & 0 \\ 0 & 0 & 0 & 0 & 0 & N_{\dot{r}} \end{bmatrix}, \quad (2.15)$$

entonces, podemos reescribir la ecuación (2.11) de la siguiente manera,

$$M = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & mz_g & -my_g \\ 0 & m - Y_{\dot{v}} & 0 & -mz_g & 0 & mx_g \\ 0 & 0 & m - Z_{\dot{w}} & my_g & -mx_g & 0 \\ 0 & -mz_g & my_g & I_x - K_{\dot{p}} & 0 & 0 \\ mz_g & 0 & -mx_g & 0 & I_y - M_{\dot{q}} & 0 \\ -my_g & mx_g & 0 & 0 & 0 & I_z - N_{\dot{r}} \end{bmatrix}, \quad (2.16)$$

donde $X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, K_{\dot{p}}, M_{\dot{q}}, N_{\dot{r}}$ representan los coeficientes hidrodinámicos de masa añadida y pueden ser calculados con ayuda de herramientas computacionales como SolidWorks o SEAWAY.

Matriz de Coriolis y de fuerzas centrípeta

La matriz de Coriolis, es generada por la fuerza de inercia que actúa perpendicular a la dirección del movimiento de un cuerpo. Esta matriz esta formada por la matriz de Coriolis de cuerpo rígido $C_{RB}(\nu)$ y por la matriz de Coriolis de masa agregada $C_A(\nu)$, entonces se puede definir como:

$$C(\nu) = C_A(\nu) + C_{RB}(\nu), \quad (2.17)$$

es importante mencionar que se puede obtener esta matriz mediante las ecuaciones de Kirchhoff's, además para un cuerpo rígido en movimiento en un fluido ideal puede ser parametrizada como una matriz antisimétrica que satisface la siguiente propiedad:

$$C(\nu) = -C^T(\nu) \quad \forall \nu \in \mathbb{R}^6. \quad (2.18)$$

Para obtener la matriz $C(\nu)$, se utilizan las características del vehículo previamente utilizadas para calcular M . Entonces, $C_{RB}(\nu)$ esta definida como:

$$C_{RB}(\nu) = \begin{bmatrix} 0 & 0 & 0 & 0 & mw & -mv \\ 0 & 0 & 0 & -mw & 0 & mu \\ 0 & 0 & 0 & mv & -mu & 0 \\ 0 & -mw & mv & 0 & I_z r & -I_y q \\ mw & 0 & -mu & -I_z r & 0 & I_x p \\ -mv & mu & 0 & I_y q & -I_x p & 0 \end{bmatrix}, \quad (2.19)$$

mientras que la matriz $C_A(\nu)$ esta denotada por,

$$C_A(\nu) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & -Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}}w & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}}p \\ -Y_{\dot{v}}v & X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix}, \quad (2.20)$$

por lo tanto, la matriz (2.17) esta descrita por:

$$C(\nu) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w + mw & Y_{\dot{v}}v - mv \\ 0 & 0 & 0 & Z_{\dot{w}}w - mw & 0 & -X_{\dot{u}}u + mu \\ 0 & 0 & 0 & -Y_{\dot{v}}v + mv & X_{\dot{u}}u - mu & 0 \\ 0 & -Z_{\dot{w}}w - mw & Y_{\dot{v}}v + mv & 0 & -N_{\dot{r}}r + I_z r & M_{\dot{q}}q - I_y q \\ Z_{\dot{w}}w + mw & 0 & -X_{\dot{u}}u - mu & N_{\dot{r}}r - I_z r & 0 & I_x p - K_{\dot{p}}p \\ -Y_{\dot{v}}v - mv & X_{\dot{u}}u + mu & 0 & I_y q - M_{\dot{q}}q & K_{\dot{p}}p - I_x p & 0 \end{bmatrix} \quad (2.21)$$

Matriz de amortiguamiento hidrodinámico

La matriz de amortiguamiento $D(\nu)$ es debido a la fricción producida debido a las fuerzas de arrastre y sustentación que se oponen al movimiento del vehículo dentro del agua. Dichas fuerzas pueden llegar a producir no linealidades sobre la dinámica del vehículo [8] y son denotadas por la ecuación siguiente:

$$D(\nu) = D_P(\nu) + D_S(\nu) + D_W(\nu) + D_M(\nu), \quad (2.22)$$

donde $D_P(\nu)$ es ocasionada por el potencial de amortiguación, $D_S(\nu)$ es debido a la fricción del cuerpo, $D_W(\nu)$ es producida por la fuerza de amortiguamiento ocasionada por las olas y $D_M(\nu)$ es causada por el amortiguamiento debido al vórtice. Fossen en [44], propone considerar que para un cuerpo rígido que se mueve en un fluido la matriz de amortiguamiento $D(\nu) > 0 \quad \forall \nu \in \mathbb{R}^6$, es estrictamente definida positiva, real y asimétrica. Asumiendo que el UUV realiza un movimiento desacoplado, que tiene simetría en dos ejes y que se desplaza a bajas velocidades; se asume que los parámetros lineales de amortiguamiento son los más significativos, por lo tanto,

$$D = - \begin{bmatrix} X_u & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_w & 0 & 0 & 0 \\ 0 & 0 & 0 & K_p & 0 & 0 \\ 0 & 0 & 0 & 0 & M_q & 0 \\ 0 & 0 & 0 & 0 & 0 & N_r \end{bmatrix} \quad (2.23)$$

donde $X_u, Y_v, Z_w, K_p, M_q, N_r$ son los coeficientes lineales de amortiguamiento, que se pueden determinar mediante herramientas computacionales como ANSYS o WAMIT que utilizan conceptos hidrodinámicos y teoría de flujos potenciales.

Vector de fuerza gravitacional y empuje hidrostático

Los vehículos submarinos al sumergirse en un fluido son afectados por la fuerza de gravedad, la cuál produce una fuerza que contrarresta su efecto llamada fuerza de flotabilidad. Una manera de explicarlo es por el principio de Arquímedes: “cuando un objeto esta sumergido en un fluido, se presenta una fuerza vertical ascendente igual al peso del volumen del fluido desplazado por dicho cuerpo”. Esto se puede representar por las siguientes ecuaciones:

$$W = ma_g^I, \quad (2.24)$$

$$\beta = V\rho a_g^I, \quad (2.25)$$

donde W es el peso, m la masa, $a_g^I = (0 \ 0 \ 9.81)^T m/s^2$ la aceleración gravitacional, β la flotabilidad, ρ la densidad del agua y V el volumen del fluido desplazado. Por lo tanto, se define el vector de fuerzas debido a la aceleración de la gravedad en el marco (I) como:

$$f_W = (0 \ 0 \ W)^T, \quad (2.26)$$

y el vector de flotabilidad por,

$$f_\beta = -(0 \ 0 \ \beta)^T. \quad (2.27)$$

El efecto de estas fuerzas en el vehículo depende donde se encuentren localizados el centro de gravedad C_G y el centro de flotabilidad C_β , para determinar los torques producidos por el efecto de estas fuerzas y para evitar momentos o torques no deseados. Por ello, se diseñó la plataforma experimental de tal forma que el origen del marco fijo en el cuerpo (O_B) sea coincidente con el C_G y el C_β sea colíneal al eje z_B . De esta manera se puede definir,

$$C_\beta = (0 \ 0 \ -z_B)^T, \quad C_W = (0 \ 0 \ 0)^T. \quad (2.28)$$

Ahora, para analizar las fuerzas de restauración y empuje hidrostático con respecto al marco fijo al cuerpo, se utiliza la matriz de transformación propuesta por Fossen en [41] y se obtiene,

$$F_B = \begin{bmatrix} \beta \sin(\theta) \\ -\beta \cos(\theta) \sin(\phi) \\ -\beta \cos(\theta) \cos(\phi) \end{bmatrix}; \quad F_W = \begin{bmatrix} -W \sin(\theta) \\ W \cos(\theta) \sin(\phi) \\ -W \cos(\theta) \cos(\phi) \end{bmatrix}, \quad (2.29)$$

entonces, se define la fuerza de restauración por,

$$f_g = F_B + F_W = \begin{bmatrix} X_g \\ Y_g \\ Z_g \end{bmatrix} = \begin{bmatrix} (\beta - W) \sin(\theta) \\ (W - \beta) \cos(\theta) \sin(\phi) \\ (W - \beta) \cos(\theta) \cos(\phi) \end{bmatrix}, \quad (2.30)$$

se asume que por diseño, el robot submarino tiene flotabilidad positiva, es decir, $\beta > W$ para que en caso de fallas emerja a la superficie de forma natural. Entonces, para obtener una orientación deseada usando los pares de restauración, se define la siguiente ecuación

$$m_g = r_w \times F_W + r_b \times F_B, \quad (2.31)$$

donde $r_w = [x_w, y_w, z_w]^T$ denota la posición del centro gravedad, y $r_b = [x_b, y_b, z_b]^T$ la ubicación del centro de flotabilidad, ambos con respecto al marco fijo al cuerpo, entonces se puede obtener:

$$\begin{bmatrix} K_g \\ M_g \\ N_g \end{bmatrix} = \begin{bmatrix} (y_w W - y_b \beta) \cos(\theta) \cos(\phi) + (z_b \beta - z_w W) \cos(\theta) \sin(\phi) \\ (x_b \beta - x_w W) \cos(\theta) \cos(\phi) + (z_b \beta - z_w W) \sin(\theta) \\ (x_w W - x_b \beta) \cos(\theta) \sin(\phi) + (y_w W - y_b \beta) \sin(\theta) \end{bmatrix}, \quad (2.32)$$

de las ecuaciones (2.30) y (2.32) se puede escribir el vector de gravedad y flotabilidad en su forma general como:

$$g(\eta) = \begin{bmatrix} (W - \beta) \sin(\theta) \\ (\beta - W) \cos(\theta) \sin(\phi) \\ (\beta - W) \cos(\theta) \cos(\phi) \\ (y_w W - y_b \beta) \cos(\theta) \cos(\phi) + (z_b \beta - z_w W) \cos(\theta) \sin(\phi) \\ (x_b \beta - x_w W) \cos(\theta) \cos(\phi) + (z_b \beta - z_w W) \sin(\theta) \\ (x_w W - x_b \beta) \cos(\theta) \cos(\phi) + (y_w W - y_b \beta) \sin(\theta) \end{bmatrix}. \quad (2.33)$$

Analizando las propiedades físicas de este vehículo submarino, tenemos que el C_G por diseño esta localizado en el origen del marco fijo al cuerpo (B). Además, el C_β por elección esta ubicado en el eje $-z_B$ y $W - \beta = f_\beta$.

Por lo tanto, se puede reescribir el vector (2.33) como:

$$g(\eta) = \begin{bmatrix} f_\beta \sin(\theta) \\ -f_\beta \cos(\theta) \sin(\phi) \\ -f_\beta \cos(\theta) \cos(\phi) \\ -z_B \beta \cos(\theta) \sin(\phi) \\ -z_B \beta \sin(\theta) \\ 0 \end{bmatrix}. \quad (2.34)$$

Fuerzas y momentos de los actuadores

En la Figura 2.2, se puede observar las fuerzas y momentos que actúan en el submarino, considerando que el origen del marco fijo al cuerpo esta ubicado en el C_G del vehículo.

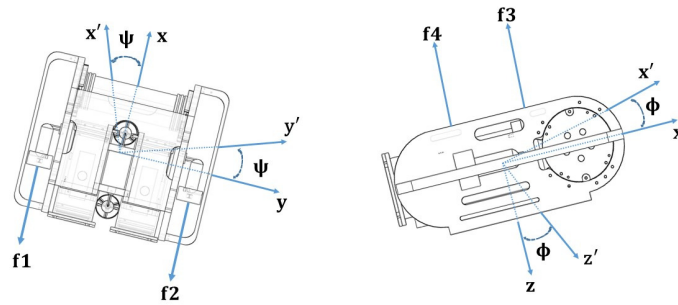


Figura 2.2: Fuerzas y momentos en LIRMIA III.

Para el análisis de como interactúan las fuerzas, se define $u_{L1} = f1 + f2$ como la sumatoria de las fuerzas horizontales y $u_{L2} = f3 + f4$ como la sumatoria de las fuerzas verticales que actúan sobre el vehículo. Entonces podemos escribir el vector de fuerzas externas que actúan sobre el submarino en el marco fijo al cuerpo (B) como:

$$\tau_1 = [X, Y, Z]^T = [u_{L1}, 0, u_{L2}]^T. \quad (2.35)$$

Para obtener el vector de momentos τ_2 que actúan en el LIRMIA III, se tiene que:

$$\tau_2 = \sum_{i=1}^4 l_i \times \hat{f}_i, \quad (2.36)$$

donde $l_i = (l_{ix}, l_{iy}, l_{iz})^T$ son las distancias respecto al C_G del vehículo y $f_i \quad \forall i = 1, \dots, 4$ es el vector de las fuerzas con respecto al marco fijo al cuerpo (B) del vehículo. Desarrollando la ecuación (2.36), los torques generados por los propulsores están descritos como:

$$\tau_2 = \begin{bmatrix} K \\ M \\ N \end{bmatrix} = \begin{bmatrix} 0 \\ l_{3x}(f3) - l_{4x}(f4) \\ l_{1y}(f1) - l_{2y}(f2) \end{bmatrix}, \quad (2.37)$$

por lo tanto, el vector de fuerzas y momentos que actúan en el vehículo es:

$$\tau = \begin{bmatrix} X \\ Y \\ Z \\ K \\ M \\ N \end{bmatrix} = \begin{bmatrix} u_{L1} \\ 0 \\ u_{L2} \\ 0 \\ l_{3x}(f3) - l_{4x}(f4) \\ l_{1y}(f1) - f2 \end{bmatrix}. \quad (2.38)$$

Modelo dinámico de LIRMIA III con respecto al marco inercial

La representación del sistema de ecuaciones (2.10) en el marco inercial, se obtiene mediante las siguientes transformaciones, asumiendo que $J(\eta)$ es no singular:

$$\begin{aligned} \dot{\eta} &= J(\eta)\nu & \Leftrightarrow \nu &= J^{-1}(\eta)\dot{\eta} \\ \ddot{\eta} &= J(\eta)\dot{\nu} + \dot{J}(\eta)\nu & \Leftrightarrow \dot{\nu} &= J^{-1}(\eta)[\ddot{\eta} - \dot{J}(\eta)J^{-1}(\eta)\dot{\eta}], \end{aligned} \quad (2.39)$$

entonces, para reescribir el sistema de ecuaciones (2.10) en función de η se definen:

$$\begin{aligned} M_\eta(\eta) &= J^{-T}(\eta)MJ^{-1}(\eta) \\ C_\eta(\nu, \eta) &= J^{-T}(\eta)[C(\nu) - MJ^{-1}(\eta)\dot{J}(\eta)]J^{-1}(\eta) \\ D_\eta(\nu, \eta) &= J^{-T}(\eta)D(\eta)J^{-1}(\eta) \\ g_\eta(\eta) &= J^{-T}(\eta)g(\eta) \\ \tau_\eta(\eta) &= J^{-T}(\eta)\tau \\ w_\eta(t) &= J^{-T}(\eta)w(t), \end{aligned} \quad (2.40)$$

por lo tanto, la representación vectorial en el marco inercial esta denotada por:

$$M_\eta(\eta)\ddot{\eta} + C_\eta(\nu, \eta)\dot{\eta} + D_\eta(\nu, \eta)\dot{\eta} + g_\eta(\eta) = \tau_\eta. \quad (2.41)$$

2.2. Plataforma experimental “BlueROV 2”

El modelo matemático simplificado del BlueROV 2 es expuesto en dos partes, la primera parte aborda el movimiento cinemático del vehículo y la segunda parte analiza el modelo dinámico del robot.

2.2.1. Modelo cinemático

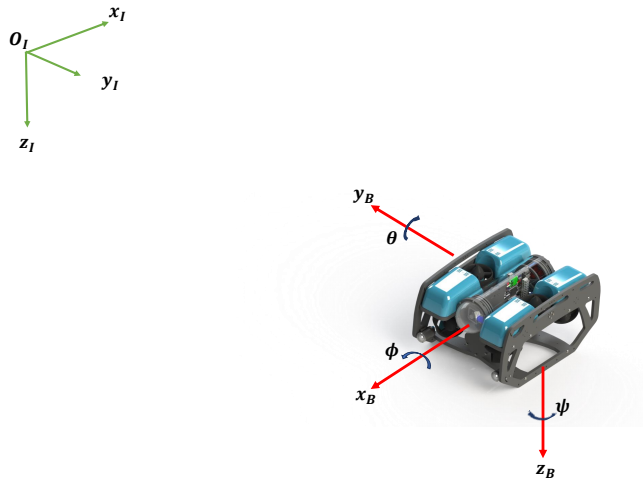


Figura 2.3: Marcos de referencia BlueROV 2.

Se consideran dos marcos de referencia: el sistema fijo al cuerpo del vehículo (B) y el sistema de coordenadas fijo en la tierra (I). Véase, Figura. 2.3. Además, se asume lo siguiente:

Suposición. El origen del sistema del marco fijo al cuerpo (O_B) está localizado en el centro de gravedad (C_G) del vehículo.

Suposición. Se establece que los ejes (x_B, y_B, z_B) son paralelos a los ejes principales de inercia.

Suposición. Por diseño, el robot submarino tiene simetría en dos de sus planos y es estable mecánicamente en los ángulos de alabeo y cabeceo, gracias a una cuidadosa selección del centro de flotabilidad, es decir, $\phi, \theta \approx 0$, lo que implica que el submarino tiene la capacidad de volver a un estado estable después de una perturbación (movimiento generado por el flujo de agua), por lo tanto, $K, M, p, q, \dot{p}, \dot{q} \approx 0$.

Entonces, la configuración del vehículo permite que se mueva libremente dentro de 4 GDL (x, y, z, ψ) . Por lo tanto, de ahora en adelante solo consideran los vectores de estado reducidos: $\bar{v} = [u, v, w, r]^T$ para denotar las velocidades lineales y angulares, $\bar{\eta} = [x, y, z, \psi]^T$ para describir la posición y orientación respectivamente.

Matriz de transformación entre el marco del cuerpo y el inercial

La matriz ecuación que describe la cinemática de un vehículo submarino de 4DOF es dada por,

$$\dot{\bar{\eta}} = J(\bar{\eta})\bar{v}, \quad (2.42)$$

donde la matriz de transformación reducida entre el marco de referencia del (B) y el marco inercial (I) esta denotada como,

$$J(\bar{\eta}) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 & 0 \\ \sin(\psi) & \cos(\psi) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2.43)$$

2.2.2. Modelo dinámico

La representación vectorial del modelo dinámico propuesto por Fossen en [41], se utiliza para analizar el modelo simplificado del BlueROV 2 y es denotado por las siguientes ecuaciones.

$$\begin{aligned} M\dot{\bar{v}} + C(\bar{v})\bar{v} + D(\bar{v})\bar{v} + g(\bar{\eta}) &= \bar{\tau} + \omega \\ \dot{\bar{\eta}} &= \bar{J}(\bar{\eta})\bar{v} \end{aligned} \quad (2.44)$$

donde $\bar{\tau} = [X, Y, Z, N]^T$ es el el vector de fuerza reducida. A continuación, se describen las matrices reducidas que conforman el modelo.

Matriz de inercia

La matriz de inercia para un vehículo de 4 GDL como el BlueROV 2 considerando sus características físicas, es definida por

$$M = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & -my_g \\ 0 & m - Y_{\dot{v}} & 0 & mx_g \\ 0 & 0 & m - Z_{\dot{w}} & 0 \\ -my_g & mx_g & 0 & I_{zz} - N_{\dot{r}} \end{bmatrix}, \quad (2.45)$$

donde m es la masa del vehículo, I_{zz} denota el momento de inercia sobre z -axis, $X_{\dot{u}}, Y_{\dot{v}}, Z_{\dot{w}}, N_{\dot{r}}$ son los parámetros de masa agregada y x_g, y_g son elementos del vector r_g .

Matriz de Coriolis

Para un cuerpo rígido que se mueve en un fluido, la matriz de Coriolis y fuerzas centrífugas para un vehículo de 4 GDL, es denotada por

$$C(\bar{v}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -X_{\dot{u}} \\ 0 & 0 & 0 & 0 \\ (m - Y_{\dot{v}})v & -(m - X_{\dot{u}})u & 0 & 0 \end{bmatrix}. \quad (2.46)$$

Matriz de amortiguamiento

La matriz de amortiguamiento hidrodinámico para un vehículo de 4 GDL que se desplaza en un fluido, es definida como,

$$D(\bar{v}) = - \begin{bmatrix} X_u + X_{u|u}|u| & 0 & 0 & 0 \\ 0 & Y_v + Y_{v|v}|v| & 0 & 0 \\ 0 & 0 & Z_w + Z_{w|w}|w| & 0 \\ 0 & 0 & 0 & N_r + N_{r|r}|r| \end{bmatrix}, \quad (2.47)$$

donde X_u, Y_v, Z_w, N_r son los coeficientes hidrodinámicos lineales; $X_{u|u}, Y_{v|v}, Z_{w|w}, N_{r|r}$ son los coeficientes cuadráticos de amortiguamiento hidrodinámico. Observe, que se consideran los coeficientes lineales, y los coeficientes cuadráticos de amortiguamiento no lineal causados por el aumento de la resistencia del cuerpo del vehículo y el flujo de arrastre cruzado.

Vector de fuerza gravitacional y empuje hidrostático

Para calcular la energía potencial debida al peso del vehículo y la fuerza de flotabilidad producida por el principio de Arquímedes, se tiene que considerar que el vehículo es estable

mecánicamente en ϕ, θ , lo que implica que podemos definir de manera reducida este vector como:

$$g(\vec{\eta}) = \begin{bmatrix} 0 \\ 0 \\ -(W - \beta) \\ 0 \end{bmatrix}, \quad (2.48)$$

donde W es el peso del vehículo y β es la fuerza de flotabilidad.

Vector de fuerzas y momentos

Analizando la Figura 2.4, el vector reducido de fuerzas y momentos $\vec{\tau}$ que actúa en el vehículo se puede expresar como:

$$\vec{\tau} = \begin{bmatrix} F_1 c(\varphi_1) + F_2 c(\varphi_2) + F_3 c(\varphi_3) + F_4 c(\varphi_4) \\ F_1 s(\varphi_1) - F_2 s(\varphi_2) + F_3 s(\varphi_3) - F_4 s(\varphi_4) \\ F_5 + F_6 \\ \xi \end{bmatrix}, \quad (2.49)$$

con $\xi = l_x(F_1 c(\varphi_1) - F_2 c(\varphi_2) - F_3 c(\varphi_3) + F_4 c(\varphi_4)) + l_y(F_1 s(\varphi_1) + F_2 s(\varphi_2) - F_3 s(\varphi_3) - F_4 s(\varphi_4))$. $F_{i:1,\dots,6}$ son las fuerzas generadas por cada propulsor; $\varphi_{i:1,\dots,4}$ son los ángulos entre el eje x_B y la fuerza $F_{i:1,\dots,4}$ aplicada en el robot; l_x y l_y son las distancias entre los propulsores horizontales y verticales con C_G . Se considera que las propelas están en el mismo plano que el centro de gravedad.

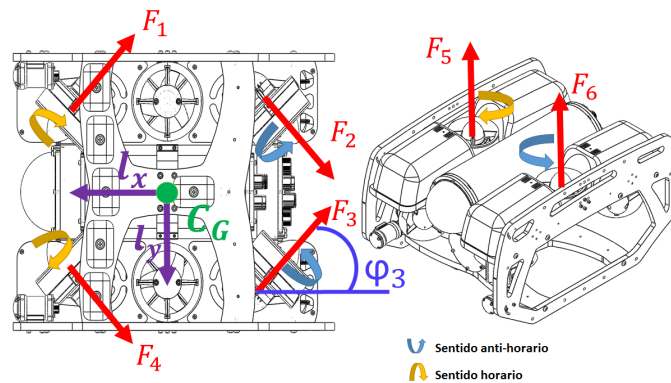


Figura 2.4: Fuerzas actuando en BlueROV 2, los propulsores 1 a 4 se utilizan para generar los movimientos de: avance, retroceso, el desplazamiento lateral y rotación en guiñada. Mientras que, los propulsores 5 y 6 se usan para emerger y sumergir el vehículo.

De acuerdo a la configuración y acción de los propulsores, el movimiento de avance se genera mediante las fuerzas generadas en la misma dirección por los propulsores 1 – 4. En

cambio, para generar el movimiento de retroceso se genera en dirección opuesta la fuerza en los propulsores 1 – 4. En el caso del movimiento lateral hacia la izquierda las fuerzas en los propulsores 3 y 4 se producen en una dirección, mientras que las fuerzas producidas por los propulsores 1 y 2 se generan en dirección opuesta. Por el contrario, para el movimiento lateral hacia el lado derecho se invierte la dirección de las fuerzas que generan el movimiento lateral hacia el lado izquierdo. Para cancelar los momentos reactivos, observe que los propulsores 1, 4 y 5 usan hélices en sentido horario, mientras que 2, 3 y 6 en sentido antihorario.

Análisis del modelo dinámico simplificado

Desarrollando el sistema de ecuaciones (2.44), el modelo matemático en el marco de coordenadas fijo al cuerpo de nuestro robot para 4 GDL se puede expresar como:

$$\begin{aligned}
 X &= (m - X_{\dot{u}})\dot{u} - (X_u + X_{u|u}|u|)u \\
 Y &= (m - Y_{\dot{v}})\dot{v} - X_{\dot{u}}r - (Y_v + Y_{v|v}|v|)v \\
 Z &= (m - Z_{\dot{w}})\dot{w} - (Z_w + Z_{w|w}|w|)w - f_{\beta} \\
 N &= (I_{zz} - N_{\dot{r}})\dot{r} + (mv - Y_{\dot{v}})u + (X_{\dot{u}}u - mu)v \\
 &\quad - (N_r + N_{r|r}|r|)r
 \end{aligned} \tag{2.50}$$

donde $r_g = [x_g, y_g, z_g]^T$ es el vector de distancia desde el origen del marco fijo del cuerpo O_B hasta el C_G del robot y $f_{\beta} = W - \beta$ es el vector de fuerzas de restauración. Ahora, considerando las suposiciones de la sección 2.2.1 y asumiendo que el vehículo se desplaza mediante movimientos lentos por debajo de 2 m/s; se puede simplificar el modelo del BlueROV 2 porque los parámetros hidrodinámicos dependen de la velocidad lineal y angular del vehículo. Por consiguiente, podemos escribir que estos parámetros son $\leq \delta_i$ con $i = x, y, z, \psi$ que puede ser considerada como incertidumbre no lineal en el modelo que depende de los estados del vehículo o como una perturbación externa desconocida. Entonces, se puede reescribir el sistema de ecuaciones (2.50) en:

$$\begin{aligned}
 X &= I_u \dot{u} - \delta_x \\
 Y &= I_v \dot{v} - \delta_y \\
 Z &= I_z \dot{w} - \delta_z - f_{\beta} \\
 N &= I_r \dot{r} - \delta_{\psi}.
 \end{aligned} \tag{2.51}$$

donde $I_u = m - X_{\dot{u}}$, $I_v = m - Y_{\dot{v}}$, $I_w = m - Z_{\dot{w}}$ e $I - r = I_{zz} - N_{\dot{r}}$ son constantes. Ahora, utilizando la matriz de transformación 2.42 se tiene que podemos representar el sistema en el marco inercial por,

$$\begin{aligned}\ddot{x} &= \frac{1}{I_u} (u_x \cos(\psi) + \delta_x) - \frac{1}{I_v} (u_y \sin(\psi)) \\ \ddot{y} &= \frac{1}{I_v} (u_y \cos(\psi) + \delta_y) + \frac{1}{I_u} (u_x \sin(\psi)) \\ \ddot{z} &= \frac{1}{I_w} (u_z + \delta_z + f_\beta) \\ \ddot{\psi} &= \frac{1}{I_r} (u_\psi + \delta_\psi).\end{aligned}\tag{2.52}$$

Para facilitar la lectura, X, Y, Z, N se reescriben como u_i para $i = x, y, z, \psi$ que representan las entradas de control.

2.3. Conclusión del capítulo

En este capítulo se describe el modelo cinemático y dinámico de dos plataformas experimentales, considerando sus características y limitaciones. El modelo matemático está basado en la representación vectorial propuesta por Fossen y la notación dada por la SNAME para vehículos marinos. En el análisis se detallan las fuerzas y pares aplicados al robot, así como, los parámetros hidrodinámicos debidos a la masa agregada, la amortiguación hidrodinámica y fuerzas restitutivas. Algunas suposiciones basadas en el diseño de los vehículos son asumidas para simplificar el modelo. A partir del modelo matemático obtenido se desarrollan las estrategias de control en el Capítulo 3.

Diseño del sistema de control

El sistema de control hace referencia a la metodología empleada para determinar las entradas de control que se necesitan para satisfacer un objetivo de control. Dentro de los objetivos más importantes están la estabilización a un punto, control de seguimiento de ruta y seguimiento de trayectoria [45, 46]. En la actualidad, la comunidad científica ha incrementado su interés en el desarrollo de algoritmos de control para realizar misiones autónomas en una amplia gama de aplicaciones, sin embargo, el control de UUV's presenta desafíos claves asociados al comportamiento dinámico no lineal del vehículo, las incertidumbres en los coeficientes hidrodinámicos, la dinámica acoplada y las perturbaciones desconocidas causadas por las olas y las corrientes marinas.

Este capítulo, presenta tres controladores para abordar diferentes acciones como: estabilizar el vehículo y el seguimiento de trayectoria, incluso en presencia de perturbaciones e incertidumbres en el modelo. El primer controlador presentado es un control proporcional derivativo con compensación de fuerzas restitutivas (PD+G), véase [47]. Para continuar se diseñó un controlador basado en las técnicas de modo deslizantes integral y super-twisting, para finalizar un controlador basado en el método de bakstepping integral adaptativo (BIA, por sus siglas) es presentado, para más detalles ver [48, 49]. Para verificar la estabilidad del sistema a lazo cerrado en cada caso, se utiliza la teoría de Lyapunov.

Antecedentes

En la literatura, se han propuesto un amplio número de estrategias de control basadas en enfoques clásicos para controlar un robot submarino, por ejemplo, un control proporcional integral derivativo (PID) es propuesto en [31] para regular la profundidad y guiñada de un vehículo submarino y evitar el sobreimpulso mediante algoritmos de selección para ajustar las ganancias del controlador. Mientras que, en [50] se presenta la comparación de controladores: PD, PID y PD saturado para estabilizar la guiñada y profundidad de un vehículo submarino miniatura. Sin embargo, los controladores convencionales no compensan las incertidumbres en el modelo y las perturbaciones del entorno. Por lo que, en [51] se propone el seguimiento de trayectoria de un perfil de profundidad utilizando un controlador PD y un filtro de segundo orden, lo que agrega robustez al sistema a lazo cerrado.

Entonces, de acuerdo al estado del arte se han estudiado con notable interés el diseño de controles basados en técnicas robustas que compensen las incertidumbres en el modelo y las perturbaciones externas. En [52] el control de modo deslizantes integral (ISMC, por sus siglas en inglés) es planteado para obtener una compensación efectiva contra las incertidumbres en el modelo y ante los efectos de perturbaciones debido a las corrientes oceánicas. Sus resultados demuestran que las perturbaciones desconocidas son compensadas desde el principio pero la variable discontinua $-\rho \operatorname{sign}(\sigma)$ produce un evidente efecto de conmutación en la entrada de control lo que puede producir desgaste mecánico y eléctrico en el vehículo submarino [53]. Para solucionar esta problemática, en [54, 55, 56], proponen el control super-twisting para atenuar la conmutación en la entrada de control y garantizar que el error de seguimiento converja exponencialmente mientras que la superficie deslizante converge en tiempo finito.

Otras técnicas empleadas, se basan en la teoría de Lyapunov y técnicas de backstepping. En [57], se propone un controlador de profundidad robusto basado en la teoría de Lyapunov y las técnicas de backstepping, sus resultados demuestran que el valor real converge a una región cercana al valor de referencia. Un trabajo más amplio, es presentado en [58] donde se diseña una estrategia de control que emplea la técnica de Lyapunov para obtener un lazo de control que mantenga al vehículo a una velocidad deseada. Después, se propone una técnica de backstepping en función del error de velocidad; no obstante, sus resultados muestran un

error de seguimiento por falta de robustez frente a las incertidumbres en los parámetros de amortiguación y de masa agregada. Por ello, en [59] se propone un controlador bakstepping con un observador no lineal basado en la teoría de Lyapunov con el fin de seguir un perfil de profundidad con un UUV incluso ante perturbaciones externas.

3.1. Control proporcional derivativo con compensación de fuerzas restitutivas

En la presente sección, se analiza el diseño de un controlador clásico basado en la estrategia de control proporcional derivativo (PD, del inglés Proportional Derivative). Esta propuesta compensa las fuerzas restitutivas causadas por la fuerza gravitacional y la fuerza de flotabilidad. Para verificar la estabilidad asintótica del sistema a lazo cerrado, se utiliza la teoría de Lyapunov. Con el fin de demostrar el rendimiento del controlador propuesto, en el Capítulo 6 se presentan un conjunto de experimentos.

3.1.1. Estrategia de control

A partir del modelo dinámico de un vehículo submarino dado por la ecuación (2.10). La ley de control proporcional derivativa con compensación de fuerzas restitutivas (PD+G) se define como:

$$\tau = g(\eta) - J^T(\eta)\tau_{PD}, \quad (3.1)$$

con

$$\tau_{PD} = k_p(\cdot)e + k_d(\cdot)\dot{e}, \quad (3.2)$$

donde $k_p(\cdot), k_d(\cdot) \in \mathbb{R}^{6 \times 6}$ son matrices diagonales definidas positivas, $e = \eta - \eta_d$ representa el error, $\dot{e} = \dot{\eta} - \dot{\eta}_d$ la derivada del error de estado y J^T la matriz de transformación dada por la ecuación (2.9). Considerando el caso de regulación a un punto $\eta_d = cte$, entonces podemos reescribir la ecuación (3.1) como

$$\tau = g(\eta) - J^T(\eta)[k_p(\cdot)e + k_d(\cdot)\dot{\eta}], \quad (3.3)$$

introduciendo la ecuación (3.3) en la ecuación (2.10), entonces el sistema a lazo cerrado está definido por:

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu = -J^T(\eta)[k_p(\cdot)e + k_d(\cdot)\dot{\eta}] + \omega, \quad (3.4)$$

introduciendo $\dot{\eta}$ dada en la ecuación (2.8) y asumiendo que $\omega \approx 0$ se puede reescribir la ecuación (3.4) como,

$$M\dot{\nu} + C(\nu)\nu + D(\nu)\nu = -J^T(\eta)[k_p(\cdot)e + k_d(\cdot)J(\eta)\nu], \quad (3.5)$$

entonces la ecuación previa puede ser escrita por,

$$M\dot{\nu} = -J^T(\eta)k_p(\cdot)e - [J^T(\eta)k_d(\cdot)J(\eta)]\nu - C(\nu)\nu - D(\nu)\nu, \quad (3.6)$$

reescribiendo, el sistema a lazo cerrado puede ser representado en espacio de estados como:

$$\frac{d}{dt} \begin{bmatrix} e \\ \nu \end{bmatrix} = \begin{bmatrix} J(\eta)\nu \\ M^{-1}[-J^T(\eta)k_p(\cdot)e - [J^T(\eta)k_d(\cdot)J(\eta)]\nu - C(\nu)\nu - D(\nu)\nu] \end{bmatrix}. \quad (3.7)$$

Observe que el origen del modelo en espacio de estados es el único punto de equilibrio. Ahora, para probar la estabilidad asintótica del sistema se propone la siguiente función candidata de Lyapunov,

$$V(e, \nu) = \frac{1}{2}\nu^T M\nu + \frac{1}{2}e^T k_p(\cdot)e, \quad (3.8)$$

como el vehículo submarino tiene un desplazamiento a baja velocidad y considerando que la matriz de inercia es simétrica $M = M^T > 0$. Entonces, según el Lema 2 de [60], tenemos que $V(e, \nu)$ es definida positiva y una función radialmente desacotada. Por lo tanto, la derivada temporal de la función candidata de Lyapunov a lo largo de las trayectorias del sistema es:

$$\dot{V}(e, \nu) = \nu^T M\dot{\nu} + \dot{e}^T k_p e, \quad (3.9)$$

como $\eta_d = cte$, entonces $\dot{e} = \nu^T J^T(\eta)$. Por lo tanto, se puede reescribir la ecuación (3.9) como:

$$\dot{V}(e, \nu) = \nu^T [M\dot{\nu} + J^T(\eta)k_p(\cdot)e], \quad (3.10)$$

al sustituir la ecuación de lazo cerrado (3.6) en la ecuación (3.10), obtenemos:

$$\begin{aligned}\dot{V}(e, \nu) &= \nu^T [-J^T(\eta)k_p(\cdot)e - [J^T(\eta)k_d(\cdot)J(\eta)]\nu - C(\nu)\nu - D(\nu)\nu + J^T(\eta)k_p(\cdot)e], \\ \dot{V}(e, \nu) &= \nu^T [-[C(\nu) - D(\nu) - [J^T(\eta)k_d(\cdot)J(\eta)]\nu],\end{aligned}\tag{3.11}$$

considerando que la matriz de Coriolis es antisimétrica, es decir, $C(\nu) = -C^T(\nu) \forall \nu$. Entonces, tenemos que $\nu^T C(\nu)\nu = 0 \forall \nu$. Por lo tanto, podemos reescribir la ecuación anterior como:

$$\dot{V}(e, \nu) = -\nu^T [D(\nu) + J^T(\eta)k_d(\cdot)J(\eta)]\nu,\tag{3.12}$$

como las matrices $D(\nu) > 0$ y $k_d(\cdot) > 0$. Esto implica que $\dot{V}(e, \nu) = 0$ cuando $\nu = 0$, entonces concluimos la estabilidad del punto de equilibrio. Para demostrar la estabilidad asintótica global, aplicamos el teorema de Krasovskii-LaSalle [61]. Entonces, definimos un conjunto Ω_{PD} como:

$$\Omega_{PD} = \left[(e, \nu) \in \mathbb{R}^n : \dot{V}(e, \nu) = 0 \right]\tag{3.13}$$

si $e(0) = 0$ y $\nu(0) = 0$. Note que $(e, \nu) = (0, 0)$, entonces es la única condición inicial en Ω_{PD} para cual $(e, \nu) \in \mathbb{R}^n \forall t \geq 0$. Esto implica que el punto de equilibrio es globalmente asintóticamente estable.

3.2. Control de modo de deslizantes integral con efecto de super-twisting

El objetivo de la presente sección, es el diseño de un algoritmo de control robusto para el seguimiento de trayectoria en 3D de un UUV de 4 GDL. La estrategia propuesta se basa en un control de modo deslizantes integral con un efecto de super-twisting (SISMC, del inglés Super-twisting Integral Sliding Mode Control), donde la parte integral garantiza la compensación de las perturbaciones y el efecto de super-twisting reduce el efecto de conmutación en la entrada de control. Para verificar la estabilidad del sistema a lazo cerrado, se utiliza la teoría de Lyapunov. Con el fin de demostrar el rendimiento satisfactorio del controlador propuesto incluso en presencia de perturbaciones, en el Capítulo 6 se presentan un conjunto de simulaciones y experimentos en tiempo real.

3.2.1. Estrategia de control

Considerando el modelo dinámico simplificado en el marco inercial del BlueROV 2 dado por la ecuación (2.52), asumiendo que las perturbaciones son acotadas $|\delta_i| \leq L_i$ y proponiendo las entradas de control $u_x = I_u v_x$, $u_y = I_v v_y$, $u_z = I_w(v_z - f_B)$ y $u_\psi = I_r v_\psi$. Donde v_i para $i = x, y, z, \psi$ son entradas de control virtual, entonces reescribiendo el sistema de ecuaciones (2.52) se tiene que,

$$\begin{aligned}\ddot{x} &= v_x \cos(\psi) - v_y \sin(\psi) + \xi_x \\ \ddot{y} &= v_y \cos(\psi) + v_x \sin(\psi) + \xi_y \\ \ddot{z} &= v_z + \xi_z \\ \ddot{\psi} &= v_\psi + \xi_\psi,\end{aligned}\tag{3.14}$$

donde $\xi_x = \frac{\delta_x}{I_u}$, $\xi_y = \frac{\delta_y}{I_v}$, $\xi_z = \frac{\delta_z}{I_w}$, $\xi_\psi = \frac{\delta_\psi}{I_r}$, observe que estas variables son acotadas, es decir, $|\xi_i| \leq \bar{L}_i$. Ahora, si proponemos cada v_i para $i = x, y, z, \psi$ como,

$$\begin{aligned}v_x &= (U_x + \ddot{x}_d) \cos(\psi) + (U_y + \ddot{y}_d) \sin(\psi) \\ v_y &= (U_y + \ddot{y}_d) \cos(\psi) - (U_x + \ddot{x}_d) \sin(\psi) \\ v_z &= U_z + \ddot{z}_d \\ v_\psi &= U_\psi + \ddot{\psi}_d,\end{aligned}\tag{3.15}$$

donde U_i con $i = x, y, z, \psi$ son entradas de control que se definirán más adelante. Considerando el sistema de ecuaciones (3.15), tenemos que

$$\begin{aligned}\ddot{x} &= U_x + \xi_x + \ddot{x}_d \\ \ddot{y} &= U_y + \xi_y + \ddot{y}_d \\ \ddot{z} &= U_z + \xi_z + \ddot{z}_d \\ \ddot{\psi} &= U_\psi + \xi_\psi + \ddot{\psi}_d.\end{aligned}\tag{3.16}$$

Por consiguiente, para el diseño del controlador se proponen el error como $e_i = i - i_d$ para $i = x, y, z, \psi$ donde i_d es la posición y orientación deseada. Entonces, se tiene el siguiente

sistema de ecuaciones:

$$\ddot{e}_i = U_i + \xi_i, \quad (3.17)$$

observe que el sistema de ecuaciones (3.17) son sistemas lineales perturbados desacoplados, donde el problema de la singularidad no aparece en la dinámica \dot{e}_{x_2} y \dot{e}_{y_2} , debido a la continuidad de $\cos(\psi)$, $\sin(\psi)$ dada en el sistema de ecuaciones (3.15). Entonces, su capacidad de control no se ve afectada por el ángulo de orientación ψ .

La estrategia de control está diseñada para lograr que las cuatro variables robustas U_i compensen las incertidumbres limitadas ξ_i desde la primera instancia si satisfacen $\xi_i(0) = 0$ y sin producir el efecto de conmutación que comúnmente se presenta en los algoritmos de control por modo deslizantes. Este hecho será posible si combinamos la ley de control de super-twisting con el algoritmo de modo deslizantes integral. Con la idea de lograr $e_{i_1} \rightarrow 0$ y $e_{i_2} \rightarrow 0$ asintóticamente, las superficies deslizantes S_i se proponen como,

$$S_i = e_{i_2} + c_i e_{i_1} \quad \text{con } c_i > 0, \quad (3.18)$$

donde $[e_{i_1}, e_{i_2}]^T = [e_i, \dot{e}_i]^T$ y $\dot{S}_i \rightarrow 0$ está dado por,

$$\dot{S}_i = U_i + \xi_i + c_i e_{i_2}. \quad (3.19)$$

Se propone U_i como,

$$U_i = -c_i e_{i_2} + \bar{u}_{1_i} + \bar{u}_{2_i}, \quad (3.20)$$

donde $\bar{u}_{1_i}, \bar{u}_{2_i}$ son entradas de control por definir más adelante. Entonces, se obtiene que \dot{S}_i está dado por,

$$\dot{S}_i = \bar{u}_{1_i} + \bar{u}_{2_i} + \xi_i \quad i : x, y, z, \psi. \quad (3.21)$$

Note que los términos $-c_i e_{i_2}$ cancelan la parte lineal de la ecuación (3.21). Ahora se propone un sistema auxiliar como:

$$\left\{ \begin{array}{l} \sigma_i = S_i - Z_i \\ \dot{Z}_i = \bar{u}_{2_i} \end{array} \right\} \quad (3.22)$$

considerando las ecuaciones (3.21) y (3.22), $\dot{\sigma}_i$ es dado por

$$\dot{\sigma}_i = \bar{u}_{1_i} + \xi_i, \quad (3.23)$$

donde cada \bar{u}_{1_i} es definida como

$$\bar{u}_{1_i} = -k_{1_i} |\sigma_i|^{1/2} \text{sign}(\sigma_i) - k_{2_i} \int_0^t \text{sign}(\sigma_i) d\tau. \quad (3.24)$$

Introduciendo la ecuación (3.24) en la ecuación (3.21), se tiene que

$$\dot{\sigma}_i = -k_{1_i} |\sigma_i|^{1/2} \text{sign}(\sigma_i) - k_{2_i} \int_0^t \text{sign}(\sigma_i) d\tau + \xi_i, \quad (3.25)$$

definiendo,

$$h_{1_i} = \sigma_i, \quad (3.26)$$

$$h_{2_i} = -k_{2_i} \int_0^t \text{sign}(\sigma_i) d\tau + \xi_i,$$

entonces,

$$\dot{h}_{1_i} = -k_{1_i} |h_{1_i}|^{1/2} \text{sign}(h_{1_i}) + h_{2_i}, \quad (3.27)$$

$$\dot{h}_{2_i} = -k_{2_i} \text{sign}(h_{1_i}) + \dot{\xi}_i.$$

Teorema. *Supongamos que la derivada de cada ξ_i es acotada, es decir, $|\dot{\xi}_i| < \xi_i^+ \in \mathbb{R}^+$ y las ganancias k_{1_i} , k_{2_i} son seleccionadas de acuerdo*

$$\begin{aligned} k_{1_i} &= \frac{(\lambda_{1_i} + \lambda_{2_i}^2 + 2\xi_i^+)^2}{4\lambda_{1_i}\lambda_{2_i}} \\ k_{2_i} &= \frac{1}{2}\lambda_{2_i}k_{1_i}, \end{aligned} \quad (3.28)$$

donde $\lambda_{1_i} > 0$ y λ_{2_i} son algún número real diferente de cero. Entonces la ley de control de super-twisting (3.24) proporciona la continuidad de σ_i y $\dot{\sigma}_i$, y produce que $\sigma_i = \dot{\sigma}_i = 0$ en tiempo finito.

Demostración. ¹ Se define

$$\nu = [\nu_{1_i} \ \nu_{2_i}]^T = \left[|h_{1_i}|^{1/2} \text{sign}(h_{1_i}) \ h_{2_i} \right]^T, \quad (3.29)$$

de la ecuación (3.29) se tiene que $|\nu_{1_i}| = |h_{1_i}|^{1/2}$ y $\text{sign}(h_{1_i}) = \frac{\nu_{1_i}}{|\nu_{1_i}|}$, considerando el

¹Algunas ideas para este análisis de estabilidad, se toman de [62, 63, 64].

sistema de ecuaciones (3.27) la derivada de cada componente de ν es representada como:

$$\begin{aligned}\dot{\nu}_{1_i} &= \frac{1}{2|\nu_{1_i}|} (-k_{1_i}\nu_{1_i} + \nu_{2_i}), \\ \dot{\nu}_{2_i} &= \frac{1}{2|\nu_{1_i}|} (-2k_{2_i}\nu_{1_i}) + \dot{\xi}_i.\end{aligned}\tag{3.30}$$

Se propone que

$$\dot{\xi}_i(t) = \chi_i(t) \text{sign}(h_{1_i}) = \chi_i(t) \frac{\nu_{1_i}}{|\nu_{1_i}|},\tag{3.31}$$

donde $\chi_i(t)$ es una función acotada tal que

$$0 < \chi_i(t) < \xi_i^+,\tag{3.32}$$

reescribiendo el sistema de ecuaciones (3.30) en forma matricial, resulta que

$$\begin{bmatrix} \dot{\nu}_{1_i} \\ \dot{\nu}_{2_i} \end{bmatrix} = \frac{1}{2|\nu_{1_i}|} \underbrace{\begin{bmatrix} -k_{1_i} & 1 \\ -2k_{2_i} + 2\chi_i & 0 \end{bmatrix}}_{A(\nu)} \begin{bmatrix} \nu_{1_i} \\ \nu_{2_i} \end{bmatrix},\tag{3.33}$$

por lo tanto, $\dot{\nu} = A(\nu)\nu$.

Observación 1. Note en la ecuación (3.29) que si $\nu_{1_i}, \nu_{2_i} \rightarrow 0$ en tiempo finito t_{r_i} , entonces $h_{1_i}, h_{2_i} \rightarrow 0$ también en tiempo finito t_{r_i} y por la ecuación (3.26) tendremos $\sigma_i(t \geq t_{r_i}) = 0$ y las incertidumbres ξ_i serán compensadas por todos los $t \geq t_{r_i}$.

Se propone la siguiente función candidata de Lyapunov

$$V(\nu) = \nu^T P_i \nu,\tag{3.34}$$

con

$$P_i = \begin{bmatrix} \lambda_{1_i} + \lambda_{2_i}^2 & -\lambda_{2_i} \\ -\lambda_{2_i} & 1 \end{bmatrix},$$

donde P_i es definida positiva sí $\lambda_{1_i} > 0$ y λ_{2_i} son cualquier número real. Por consiguiente, la derivada temporal de la función candidata de Lyapunov a lo largo de las trayectorias del

sistema es,

$$\dot{V}(\nu) = \nu^T [P_i A(\nu) + A^T(\nu) P_i] \nu = -\frac{1}{2|\nu_{1_i}|} \nu^T \beta_i \nu, \quad (3.35)$$

donde $\beta_i = \beta_i^T$ y es dada por:

$$\beta_i(\chi_i) = \begin{bmatrix} 2\lambda_{1_i} k_{1_i} + 2\lambda_{2_i} (\lambda_{2_i} k_{1_i} - 2k_{2_i}) + 4\lambda_{2_i} \chi_i & (2k_{2_i} - \lambda_{2_i} k_{1_i} - \lambda_{1_i} - \lambda_{2_i}^2) - 2\chi_i \\ (2k_{2_i} - \lambda_{2_i} k_{1_i} - \lambda_{1_i} - \lambda_{2_i}^2) - 2\chi_i & 2\lambda_{2_i} \end{bmatrix}, \quad (3.36)$$

usando el límite de $\chi_i(t)$, se deduce que

$$\dot{V}(\nu) \leq -\frac{1}{2|\nu_{1_i}|} \nu^T Q_i \nu, \quad (3.37)$$

con $Q_i = \beta_i(\xi_i^+)$, seleccionando las ganancias del sistema de ecuaciones (3.28) se tiene que

$$Q_i = \begin{bmatrix} \frac{(\lambda_{1_i} + \lambda_{2_i}^2 + 2\xi_i^+)^2}{2\lambda_{2_i}} + 4\lambda_{2_i} \xi_i^+ & -(\lambda_{1_i} + \lambda_{2_i}^2 + 2\xi_i^+) \\ -(\lambda_{1_i} + \lambda_{2_i}^2 + 2\xi_i^+) & 2\lambda_{2_i} \end{bmatrix}, \quad (3.38)$$

donde $\det(Q_i) = 8\lambda_{2_i}^2 \xi_i^+$ con λ_{2_i} como cualquier número real diferente de cero, entonces $\det(Q_i) > 0$, implica que Q_i es definida positiva, garantizando que $-\frac{1}{2|\nu_{1_i}|} \nu^T Q_i \nu < 0$. Como consecuencia $\nu \rightarrow 0$ asintóticamente, sin embargo, si $\dot{V}(\nu)$ satisface también la condición modificada $\dot{V}(\nu) \leq -\gamma_i V^{1/2}(\nu) < 0$, $\gamma_i > 0$, la convergencia ocurrirá en tiempo finito t_{r_i} , ver [65] página 4.

Para probar la convergencia de tiempo finito, considere las siguientes desigualdades

$$\lambda_{\min} \{P_i\} \|\nu\|^2 \leq V(\nu) \leq \lambda_{\max} \{P_i\} \|\nu\|^2 \quad (3.39)$$

$$-\lambda_{\max} \{Q_i\} \|\nu\|^2 \leq -\nu^T Q_i \nu \leq -\lambda_{\min} \{Q_i\} \|\nu\|^2 \quad (3.40)$$

$$|\nu_{1_i}| \leq \|\nu\| \leq \frac{V^{1/2}(\nu)}{\lambda_{\min}^{1/2} \{P_i\}}, \quad (3.41)$$

usando las ecuaciones (3.39) y (3.40), entonces la ecuación (3.37) se puede escribir como:

$$\dot{V}(\nu) \leq -\frac{1}{2|\nu_{1_i}|} \nu^T Q_i \nu \leq -\frac{1}{2|\nu_{1_i}|} \lambda_{\min} \{Q_i\} \|\nu\|^2 \leq -\frac{\lambda_{\min} \{Q_i\}}{2|\nu_{1_i}|} \frac{V(\nu)}{\lambda_{\max} \{P_i\}}, \quad (3.42)$$

de la ecuación (3.41), se deduce que la condición de alcance es

$$\dot{V}(\nu) \leq -\gamma_i V^{1/2}(\nu) \quad \text{con} \quad \gamma_i = \frac{\lambda_{\min}^{1/2}\{P_i\} \lambda_{\min}(Q_i)}{2\lambda_{\max}\{P_i\}}. \quad (3.43)$$

Separando variables e integrando la desigualdad (3.43) durante el intervalo de tiempo $0 \leq \tau \leq t$ produce que

$$V^{1/2}(\nu(t)) - V^{1/2}(\nu(0)) \leq -\frac{1}{2}\gamma_i t, \quad (3.44)$$

considerando en la ecuación (3.44) que $V(\nu(t))$ alcanza el cero en tiempo finito t_{r_i} , entonces t_{r_i} es acotada por

$$t_{r_i} \leq \frac{2V^{1/2}(\nu(0))}{\gamma_i}. \quad (3.45)$$

donde la ecuación (3.45) garantiza que $\nu = [\nu_{1_i} \ \nu_{2_i}]^T \rightarrow [0, \ 0]^T$ en tiempo finito t_{r_i} , por lo tanto, lo mencionado en la Observación 1 es cumplido. Ahora desarrollando y simplificando la función de Lyapunov (3.34) nosotros tenemos que $V(\nu) = (\lambda_{1_i} + \lambda_{1_i}^2)\nu_{1_i}^2 - 2\lambda_{2_i}\nu_{1_i}\nu_{2_i} + \nu_{2_i}^2$, y si proponemos $\sigma_i(0) = 0$, podemos verificar que $V^{1/2}(\nu(0)) = \xi_i(0)$. Entonces, t_{r_i} en la ecuación (3.45) es acotado por,

$$t_{r_i} \leq \frac{2\xi_i(0)}{\gamma_i}, \quad (3.46)$$

Por lo tanto, del sistema de ecuaciones (3.22) resulta que $S_i(t) = Z_i(t)$ para todo $t \geq t_{r_i}$. En consecuencia el sistema puede ser reescrito como,

$$\begin{cases} S_i = Z_i \\ \dot{Z}_i = \bar{u}_{2_i} \end{cases} \quad \text{para todo } t \geq t_{r_i} \text{ y con } Z_i(0) = S_i(0). \quad (3.47)$$

De lo anterior, se obtiene que

$$\dot{S}_i = \bar{u}_{2_i} \quad \forall t \geq t_{r_i}, \quad (3.48)$$

note que, \dot{S}_i ya no está perturbado. Ahora, se propone

$$\bar{u}_{2_i} = -k_{3_i}S_i, \quad k_{3_i} > 0, \quad (3.49)$$

entonces \dot{S}_i es dado como

$$\dot{S}_i = -k_{3_i}S_i. \quad (3.50)$$

Para resolver la ecuación (3.50), integramos sobre el intervalo de tiempo $t_{r_i} \leq \tau \leq t$,

obteniendo

$$S_i(t) = \exp(-k_{3_i}(t - t_{r_i})) S_i(t_{r_i}), \quad (3.51)$$

por lo tanto, $S_i(t) \rightarrow 0$ asintóticamente para todo $t > t_{r_i}$, y los errores estarán dados por

$$\begin{aligned} e_{i_1}(t) &\rightarrow \exp(-c_i(t - t_{r_i})) e_{i_1}(t_{r_i}) \\ e_{i_2}(t) &\rightarrow -c_i \exp(-c_i(t - t_{r_i})) e_{i_1}(t_{r_i}) \end{aligned} \quad (3.52)$$

Entonces $e_{i_1}(t) \rightarrow 0$ y $e_{i_2}(t) \rightarrow 0$ cuando $t \rightarrow \infty$ empezando desde $t = t_{r_i}$. \square

Por lo tanto, los principales resultados sobre \bar{u}_{i_1} y \bar{u}_{i_2} se describen a continuación

1. La ley de control super-twisting \bar{u}_{i_1} provee la continuidad de σ_i y $\dot{\sigma}_i$, produce que $\sigma_i = \dot{\sigma}_i = 0$ para todo $t \geq t_{r_i}$ y compensa ξ_i para todo $t \geq t_{r_i}$.
2. Una vez que las incertidumbres ξ_i han sido compensados el control lineal \bar{u}_{i_2} logra que $S_i(t) \rightarrow 0$ asintóticamente para todo $t \geq t_{r_i}$, $e_{i_1}(t) \rightarrow 0$ y $e_{i_2}(t) \rightarrow 0$ cuando $t \rightarrow \infty$ empezando desde $t = t_{r_i}$.

Observación 2. *Observe que si las incertidumbres acotadas ξ_i satisfacen $\xi_i(0) = 0$, entonces la ecuación (3.46) garantiza $t_{r_i} \leq 0$ y la ley de control super-twisting (3.24) logra que σ_i converja a cero en un tiempo finito como máximo después de $t_{r_i} = 0$ unidades de tiempo, además, cada ξ_i será compensado en $t \geq (t_{r_i} = 0)$. Por otro lado, el control lineal (3.49) implicará que $S_i(t) \rightarrow 0$ asintóticamente para todo $t > 0$, en consecuencia $e_{i_1}(t) \rightarrow 0$ y $e_{i_2}(t) \rightarrow 0$ cuando $t \rightarrow \infty$ empezando desde $t = 0$.*

Lema. *Si $\xi_i(0) = 0$ y $e_{i_1}(0) = 0$, $e_{i_2}(0) = 0$, entonces se obtiene $S_i(t) = 0 \forall t \geq 0$, también $e_{i_1}(t) = 0$ y $e_{i_2}(t) = 0 \forall t \geq 0$.*

Demostración. Como se mencionó en la Observación 2 con $\xi_i(0) = 0$ el tiempo de convergencia viene dado por $t_{r_i} = 0$, sustituyendo en la ecuación (3.51), se obtiene $S_i(t) = \exp(-k_{3_i}t)S_i(0)$, por la ecuación (3.18), se tiene que $S_i(0) = e_{i_2}(0) + c_i e_{i_1}(0)$. Ahora asumiendo que $e_{i_1}(0) = 0$, $e_{i_2}(0) = 0$, tenemos $S_i(0) = 0$, implica que $S_i(t) = 0 \forall t \geq 0$ que corresponde a la solución de equilibrio de la ecuación (3.50), luego en la ecuación (3.18), $e_{i_2} + c_i e_{i_1} = 0 \forall t \geq 0$. Por lo tanto, $e_{i_2} = -c_i e_{i_1} \forall t \geq 0$ cuya solución viene dada por,

$$\begin{aligned} e_{i_1}(t) &= \exp(-c_i t) e_{i_1}(0) \\ e_{i_2}(t) &= \exp(-c_i t) e_{i_2}(0) \end{aligned} \quad (3.53)$$

Entonces, sí asumimos que $e_{i_1}(0) = 0$, $e_{i_2}(0) = 0$, se obtiene $e_{i_1}(t) = 0$ y $e_{i_2}(t) = 0$ para todo $t \geq 0$. \square

Lo anterior significa que si las condiciones iniciales de posición y velocidad del vehículo coinciden con las condiciones iniciales de la trayectoria deseada, el seguimiento de la trayectoria se producirá desde el principio.

3.3. Bakstepping integral adaptativo

En esta sección, se introduce un esquema de control no lineal con propiedades adaptativas e integrales para realizar el seguimiento de trayectoria en un vehículo submarino. El diseño y análisis de estabilidad del controlador se basa en la teoría de Lyapunov. La idea de agregar un componente integral es obtener robustez en el sistema ante perturbaciones desconocidas e incertidumbres en el modelo, mientras que la propiedad adaptativa se utiliza para estimar las perturbaciones desconocidas. En el Capítulo 6, los resultados de validación numérica y experimentales son expuestos.

3.3.1. Estrategia de control

El siguiente controlador es desarrollado usando el modelo vectorial de un vehículo submarino respecto al marco fijo a la tierra, definido por,

$$M_{\bar{\eta}}(\bar{\eta})\ddot{\bar{\eta}} + C_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\bar{\eta}} + D_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\bar{\eta}} + g_{\bar{\eta}}(\bar{\eta}) = \tau_{\bar{\eta}}(\bar{\eta}) + \delta, \quad (3.54)$$

de la ecuación (3.54), se tiene que:

$$\ddot{\bar{\eta}} = M_{\bar{\eta}}^{-1}[-C_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\bar{\eta}} - D_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\bar{\eta}} - g_{\bar{\eta}}(\bar{\eta}) + \tau_{\bar{\eta}}(\bar{\eta}) + \delta], \quad (3.55)$$

Teorema. *Considere el sistema (3.54) y el error de seguimiento definido como:*

$$e = \bar{\eta} - \bar{\eta}_d, \quad (3.56)$$

donde $\bar{\eta} = [x, y, z, \psi]^T$ y $\bar{\eta}_d = [x_d, y_d, z_d, \psi_d]^T$. Ahora, denotando la entrada de control por

$$\tau_{\bar{\eta}}(\bar{\eta}) = M_{\bar{\eta}}[\dot{p} - e - K_v e_2] + C_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\bar{\eta}} + D_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\bar{\eta}} + g_{\bar{\eta}}(\bar{\eta}) - \hat{\delta}, \quad (3.57)$$

con $p = \dot{\eta}_d - k_s \int_0^t e d\tau - k_e e$; $\dot{\delta} = k_\delta M_\eta^{-1}$. Entonces el sistema es estable cuando los errores son cero para todo $t \geq 0$ y cuando los errores son diferentes de cero, entonces, $e \rightarrow 0$ y $e_2 \rightarrow 0$ asintóticamente. Esto implica, que $\bar{\eta} \rightarrow \bar{\eta}_d$ y $\dot{\bar{\eta}} \rightarrow \dot{\bar{\eta}}_d$ cuando $t \rightarrow \infty$.

Demostración. Para realizar el seguimiento de trayectoria con el vehículo submarino, se define el error de seguimiento de la siguiente manera:

$$e = \bar{\eta} - \bar{\eta}_d, \quad (3.58)$$

ahora, se plantea que:

$$\varsigma = \int_0^t e d\tau, \quad (3.59)$$

entonces $\dot{\varsigma} = e$. Después, se propone la siguiente función definida positiva para diseñar una velocidad virtual que asegure la convergencia a la posición deseada.

$$V_1 = \frac{1}{2} \varsigma^T k_s \varsigma + \frac{1}{2} e^T e, \quad (3.60)$$

donde $k_s > 0 \in \mathbb{R}^{4 \times 4}$ es una matriz diagonal definida positiva. Diferenciando la ecuación (3.60), da como resultado:

$$\dot{V}_1 = \varsigma^T k_s \dot{\varsigma} + e^T \dot{e}, \quad (3.61)$$

proponiendo la velocidad virtual como:

$$p = \dot{\eta}_d - k_s \varsigma - k_e e, \quad (3.62)$$

con $k_e > 0 \in \mathbb{R}^{4 \times 4}$ como una matriz diagonal definida positiva, entonces de la ecuación (3.62) se tiene que,

$$\dot{\eta}_d = p + k_s \varsigma + k_e e, \quad (3.63)$$

esto implica que la derivada del error de seguimiento es:

$$\dot{e} = \dot{\bar{\eta}} - p - k_s \varsigma - k_e e, \quad (3.64)$$

por lo tanto,

$$\dot{V}_1 = -e^T k_e e + e^T e_2, \quad (3.65)$$

con el error de velocidad denotado por $e_2 = \dot{\bar{\eta}} - p$, ahora se propone una segunda fun-

ción definida positiva para asegurar la convergencia del error de seguimiento a la posición deseada.

$$V_2 = V_1 + \frac{1}{2}e_2^T e_2, \quad (3.66)$$

diferenciando la función definida positiva (3.66), se tiene que:

$$\dot{V}_2 = \dot{V}_1 + e_2^T \dot{e}_2, \quad (3.67)$$

donde $\dot{e}_2 = \ddot{\eta} - \dot{p}$. Reescribiendo, la ecuación (3.67) e introduciendo la ecuación (3.55) tenemos que,

$$\dot{V}_2 = -e^T k_e e + e^T e_2 + e_2^T \{M_{\bar{\eta}}^{-1}[-C_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\eta} - D_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\eta} - g_{\bar{\eta}}(\bar{\eta}) + \tau_{\bar{\eta}}(\bar{\eta}) + \delta] - \dot{p}\}. \quad (3.68)$$

Eligiendo la siguiente entrada de control,

$$\tau_{\bar{\eta}}(\bar{\eta}) = M_{\bar{\eta}}[\dot{p} - e - K_v e_2] + C_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\eta} + D_{\bar{\eta}}(\bar{v}, \bar{\eta})\dot{\eta} + g_{\bar{\eta}}(\bar{\eta}) - \hat{\delta}, \quad (3.69)$$

donde $k_v > 0 \in \mathbb{R}^{4 \times 4}$ es una matriz diagonal definida positiva, $\hat{\delta}$ define la estimación de δ . Reescribiendo, la ecuación (3.68) y definiendo $e_\delta = \delta - \hat{\delta}$, se tiene que:

$$\dot{V}_2 = -e^T k_e e - e_2^T k_v e_2 + e_2^T M_{\bar{\eta}}^{-1} e_\delta, \quad (3.70)$$

Considerando la siguiente función candidata de Lyapunov

$$V_L = V_2 + \frac{1}{2}e_\delta^T k_\delta^{-1} e_\delta, \quad (3.71)$$

donde $k_\delta > 0 \in \mathbb{R}^{4 \times 4}$ es una matriz diagonal definida positiva. Entonces, la derivada temporal de la función candidata de Lyapunov a lo largo de las trayectorias del sistema es,

$$\dot{V}_L = \dot{V}_2 + e_\delta^T k_\delta^{-1} \dot{e}_\delta, \quad (3.72)$$

considerando que la perturbación de las olas cambia lentamente, debido a que, el vehículo opera por debajo de la zona de efecto de onda, esto significa que podemos asumir una

perturbación constante $\delta = cte$. Esto implica que:

$$\dot{e}_\delta = -\dot{\hat{\delta}}, \quad (3.73)$$

reescribiendo la ecuación (3.72) e introduciendo las ecuaciones (3.70) y (3.73)

$$\dot{V}_L = -e^T k_e e - e_2^T k_v e_2 + e_2^T M_{\bar{\eta}}^{-1} e_\delta + e_\delta^T k_\delta^{-1} (-\dot{\hat{\delta}}), \quad (3.74)$$

se propone la dinámica deseada para $\dot{\hat{\delta}}$ como,

$$\dot{\hat{\delta}} = k_\delta M_{\bar{\eta}}^{-1} e_2. \quad (3.75)$$

Por lo tanto, podemos reescribir la ecuación (3.74) de la siguiente manera,

$$\dot{V}_L = -e^T k_e e - e_2^T k_v e_2 \leq 0 \quad \forall t \geq 0. \quad (3.76)$$

En la expresión anterior, se observa que si $\dot{V}_L = 0 \quad \forall t \geq 0 \implies e = 0, e_2 = 0 \quad \forall t \geq 0$ entonces el sistema es estable, lo cual garantiza que la referencia de posición y velocidad son alcanzadas. Por otro lado, cuando e y e_2 son diferentes de cero; entonces $\dot{V}_L < 0 \quad \forall t \geq 0$ lo que garantiza estabilidad asintótica en los errores e y e_2 . Por tanto, los estados convergen a los vectores de referencia de posición y velocidad cuando $t \rightarrow \infty$. Es decir,

$$\begin{aligned} \bar{\eta} &\rightarrow \bar{\eta}_d \\ \dot{\bar{\eta}} &\rightarrow \dot{\bar{\eta}}_d \end{aligned} \quad (3.77)$$

□

3.4. Conclusión del capítulo

En este capítulo, se desarrollan tres estrategias de control para que un vehículo submarino realice las acciones de regulación a un punto y de seguimiento de trayectoria. El primer controlador es un control proporcional derivativo con compensación de fuerzas restitutivas (PD+G), para comprobar la estabilidad del sistema a lazo cerrado se realiza un análisis de estabilidad mediante el método de Lyapunov, el cuál garantiza la estabilidad del sistema en

el punto de equilibrio. Además, para demostrar la estabilidad asintótica global se utiliza el teorema de Krasovskii-LaSalle.

El siguiente controlador diseñado utiliza una técnica de control robusto basada en el control de modo deslizantes integral con una acción de super-twisting para compensar las perturbaciones y atenuar el efecto de conmutación en las entradas de control. Para su diseño se analiza el ISMC, se concluye que el término u_0 produce el fenómeno de conmutación. Por lo tanto, una forma de solucionar este fenómeno es reemplazar u_0 por el controlador de super-twisting, asegurando la continuidad de $\sigma = \dot{\sigma} = 0$ lo que atenuará el efecto de conmutación, consulte [55, 66, 67]. Para verificar la estabilidad del sistema a lazo cerrado, se utiliza la teoría de Lyapunov que garantiza la estabilidad asintótica.

Para finalizar, un controlador no lineal con propiedades adaptativas e integrales basado en la técnica de backstepping es presentado. La función integral da robustez al sistema ante perturbaciones desconocidas e incertidumbres en el modelo y disminuye el error de estado estable, mientras que la parte adaptativa estima las perturbaciones desconocidas para ser compensadas. El análisis de estabilidad del controlador se basa en la teoría de Lyapunov, garantiza que el sistema es estable cuando $e, e_2 = 0$ y cuando $e, e_2 \neq 0$ entonces se garantiza la estabilidad asintótica de los errores de seguimiento.

En el capítulo 6, un conjunto de simulaciones numéricas y pruebas experimentales se llevan a cabo con el fin de seguir una trayectoria de referencia con un UUV.

Integración del sistema de guía y navegación

Este capítulo, describe los sistemas de guía y navegación que conforman el sistema GNC. En la Figura 4.1, se observa la interacción entre sí de cada sistema a través de datos y señales de transmisión. Para más detalles, ver [41, 44].

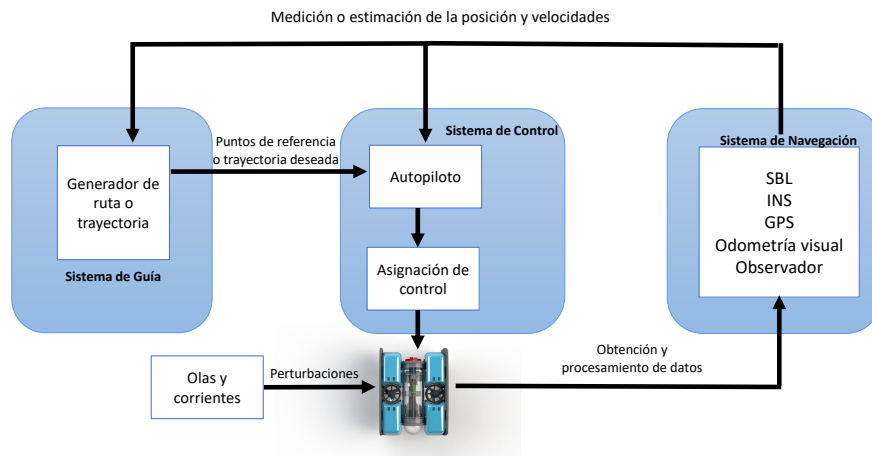


Figura 4.1: Interacción de los bloques en un sistema GNC.

Observe, que el sistema de guía calcula continuamente el valor de referencia deseado (posición, orientación y velocidad) considerando características como la evasión de colisiones. Esta señal es empleada por el sistema de control, que determina la fuerza y momentos necesarios para que se cumpla el seguimiento de una trayectoria de referencia. Mientras que, el sistema de navegación dirige al UUV por determinada posición y orientación durante el curso de una misión.

Considere que el sistema de control es desarrollado mediante las estrategias de control descritas en el Capítulo 3, entonces para el sistema de navegación se detallan los métodos de localización implementados para estimar la posición del vehículo submarino. Por otra lado, para el sistema de guía se desarrollan los algoritmos para generar rutas 2D y 3D mediante el uso puntos de referencia, y se describen las técnicas implementadas para la detección y evasión de obstáculos.

4.1. Sistema de navegación

4.1.1. Antecedentes de localización

En los vehículos submarinos un gran desafío es localizar el vehículo en un entorno acuático, diferentes técnicas se han implementado para adquirir la ubicación de un UUV. Donde el uso del modelo matemático para mejorar la medición es un enfoque común, como es el caso de la técnica de navegación por estima presentada en [68, 69, 70], donde utilizan el filtro de Kalman extendido (EKF, por sus siglas en inglés) para fusionar los datos de la unidad de medición inercial (IMU, por sus siglas en inglés) y un registro de velocidad Doppler (DVL, por sus siglas en inglés). Sin embargo, una posición inicial precisa rápidamente se vuelve incierta a través de variaciones en el movimiento del AUV y los errores de posición crecen ilimitadamente con el tiempo por la acumulación del error del giroscopio y acelerómetro; lo que implica que no pueda ser utilizado en misiones a largo plazo. Por ello, se han utilizado sistemas de posicionamiento acústico que utilizan la fusión de datos para mejorar el rango y la precisión de la localización del UUV. Estos sistemas utilizan dispositivos que reciben y emiten señales acústicas, su principio de funcionamiento se basa en medir la diferencia del tiempo de retardo en la llegada (TDoA, por sus siglas en inglés) de la señal acústica para triangular la posición del robot. Existen distintos tipos de sistemas de posicionamiento acústico: el sistema de línea de base corta (SBL, por sus siglas en inglés), el sistema de línea de base ultracorta (USBL, por sus siglas en inglés) y el sistema de línea de base larga (LBL, por sus siglas en inglés), véase [25, 46].

Otro método propuesto, es el diseñado en [71] donde se utiliza la aritmética de intervalos para resolver el problema de SLAM mediante una fusión de datos del sonar, IMU y características del entorno acuático. El principio del enfoque es convertir el problema SLAM en

un problema de satisfacción de restricciones para el cual los algoritmos de propagación de intervalos son particularmente poderosos. El método de propagación resultante se ilustra en la localización y construcción de mapas con un robot submarino real. Por otra parte, en la literatura se encuentran algoritmos de visión que estiman la pose del robot. En [72] se describe el uso de localización y mapeo visual simultáneo (vSLAM, por sus siglas en inglés), que se puede implementar mediante el uso de cámaras monoculares, proporcionando una cantidad significativa de información a bajo costo, por ejemplo: color, bordes, tamaño, textura, forma, etc. Estos datos permiten ubicar el movimiento del vehículo en una escena estática de cierta manera, mediante una secuencia de imágenes capturadas. La implementación de SLAM en entornos subacuáticos reales puede considerarse un problema abierto de gran interés en robótica [73]. Dentro de la literatura de vSLAM aplicada a vehículos submarinos, podemos encontrar en [74] el uso de un método de mapeo directo que estima en tiempo real una reconstrucción densa en 3D de una escena submarina y la posición del vehículo, usando una cámara monocular. En [75] proponen un SLAM submarino llamado DolphinSLAM, que es capaz de mejorar la percepción a baja resolución y las tareas a largo plazo para un plano 2D, utilizando una red neuronal. Sin embargo, la aplicación de estas técnicas depende principalmente del costo computacional, las condiciones de luz e imagen, es decir, iluminación, sombras, desenfoque, turbidez, escala, rotación, reflectancia, entre otros factores ambientales. Estos son efectos negativos que afectan la detección de características y alteran la estimación del desplazamiento de píxeles entre cuadros de imagen, lo que lleva a errores en la estimación [76]. Por lo cuál, en [77], se presenta una técnica que agrupa los puntos clave, con el propósito de validarla en un AUV, en ambientes marinos con bajas texturas. Basado en [78], la técnica de coincidencia de características robustas aceleradas (SURF, por sus siglas en inglés) es el mejor método para la detección de puntos característicos en imágenes turbias. Para conocer que algoritmo de odometría visual tiene mejor estimación en distintas escenas, en [79] varios algoritmos de odometría visual y SLAM se prueban fuera de línea en conjuntos de datos capturados por investigadores de todo el mundo utilizando vehículos terrestres, aéreos y marinos. Los autores encuentran que los mejores resultados para aplicaciones submarinas se logran con los algoritmos PTAM, ORB-SLAM, COLMAP, g2o y Ceres.

4.1.2. Localización acústica

GPS Water Linked

Es un sistema de posicionamiento acústico basado en el principio de línea de base corta (SBL) [80, 81], el UUV tiene una baliza de localización que envía una señal acústica hacia 4 hidrófonos receptores localizados cerca de la superficie. Los receptores escuchan el pulso desde la baliza de localización y usan la diferencia en el tiempo de llegada hacia cada receptor para triangular la posición del vehículo. Para obtener la posición global el sistema añade la información adquirida por el GPS fuera de borda y la IMU, este dispositivo funciona en aguas poco profundas (< 1 m de profundidad) y en entornos acústicos ruidosos como: albercas, tanques o jaulas para peces. El alcance del sistema GPS subacuático es de $200 \times 200 \times 100$ m, debido a que esta limitado por la frecuencia de los hidrófonos, en circunstancias normales, el localizador se colocará dentro de una bola de 1 metro de diámetro en un rango de 100 m. Véase, Figura 4.2.

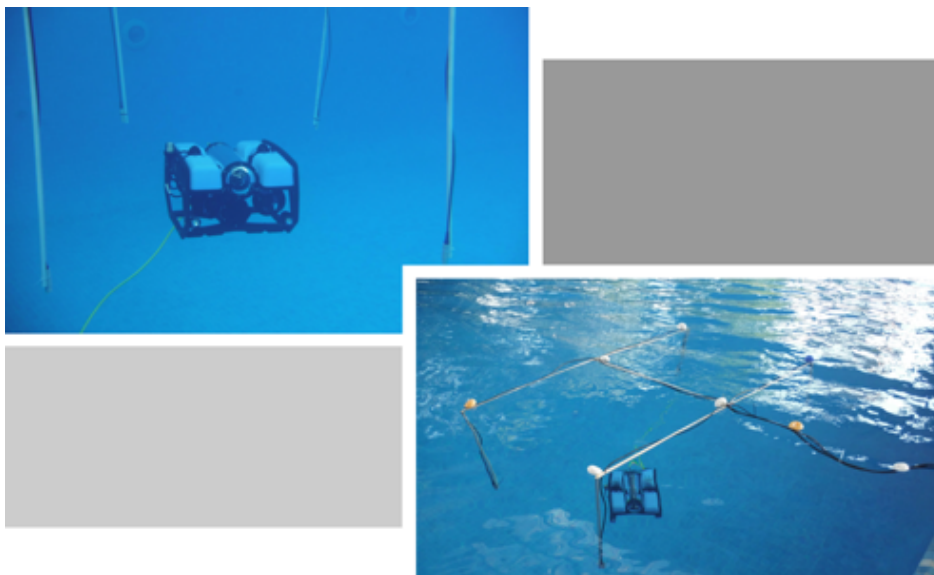


Figura 4.2: Sistema acústico Water Linked, el transceptor se localiza en el UUV y 4 transponder son ubicados cerca de la superficie. La computadora de posicionamiento utiliza algoritmos para estimar la posición del localizador en función de las señales recibidas de los transponder.

Las propiedades del GPS submarino, se describen en los puntos siguientes:

1. Precisión: se obtiene una precisión de 2 metros en la estimación global de la posición y de 1 metro en la estimación acústica. Esto depende de la ubicación y configuración correcta de los receptores mediante el uso de la GUI.

2. Robustez: el sistema funciona en piscinas, sin embargo, una piscina es el peor escenario para reflexiones acústicas lo que ocasiona ruido de fondo y esto puede afectar la precisión. Las actualizaciones del firmware han agregado filtros para las reflexiones pero han ocasionado retardo en la estimación de la pose.
3. Tasa de actualización: es de 4 Hz.

El GPS Water Linked tiene una interfaz gráfica de usuario (GUI, por sus siglas en inglés) pre-cargada, que se ejecuta en la estación terrestre mediante la computadora de posicionamiento Máster-D1 y se accede a través de un navegador web estándar. Para adquirir la localización acústica del GPS submarino, se analizan los algoritmos del SDK y la API del Water Linked. Se concluye que modificando el script de Python dado por el fabricante se puede crear en el sistema operativo robótico (ROS, por sus siglas en inglés) un nodo que adquiere la estimación de la posición acústica y la publica a un nodo que elimina los valores atípicos de la medición mediante un filtro pasa bajas de segundo orden. Para finalizar, se crea un tópicos que envía un mensaje que contiene las coordenadas de localización acústica filtrada. Véase, Figura 4.3.

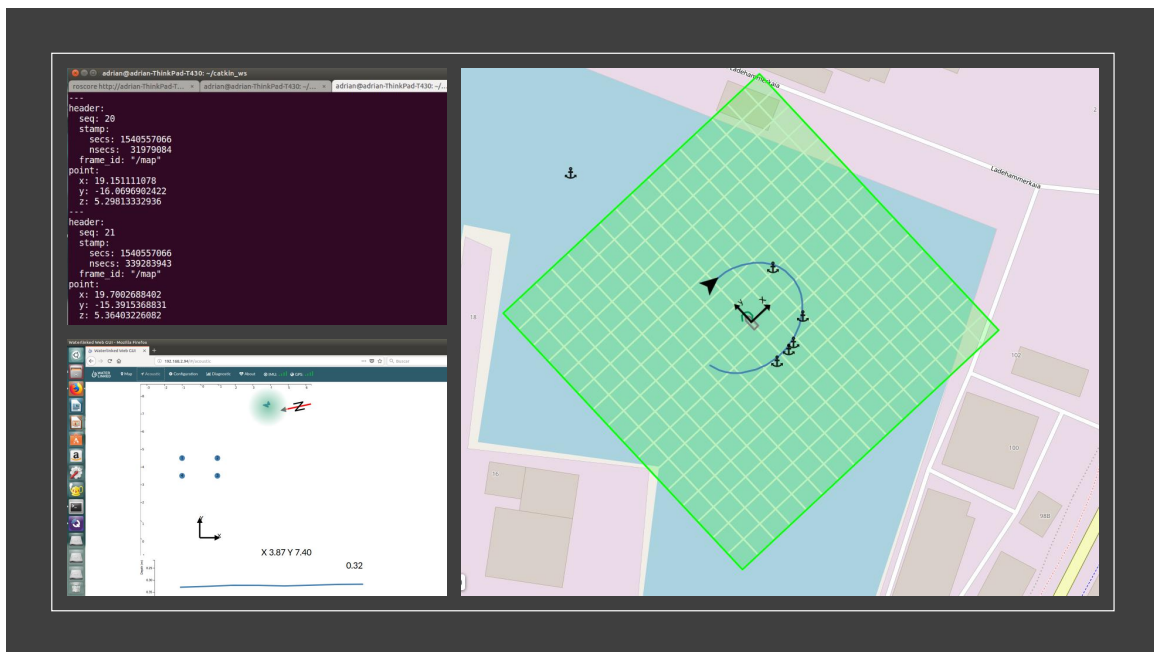


Figura 4.3: La GUI de Water Linked GPS nos permite visualizar la posición global y acústica del vehículo, además de configurar la distancia de cada una de las balizas receptoras. Observe, que el nodo de ROS publica en un tópicos la estimación de la posición.

Localización mediante un sonar acústico y el método de análisis de intervalos

Este método solo considera la localización 2D, porque asume que el vehículo submarino cuenta con un sensor de presión o altímetro. Para que el robot submarino pueda obtener su posición con respecto a los bordes de la alberca en la que se mueve, el submarino tiene un sonar Tritech Mini conectado a la computadora integrada mediante comunicación RS232. El sonar emite un haz acústico cada 2.4° hasta hacer un barrido en 360° , la distancia que puede recorrer este haz es de hasta 100 m, este dispositivo puede proporcionar datos como imágenes o nubes de puntos. Para calcular una estimación de la posición del robot en una alberca suponemos que conocemos las características del entorno, cuando el robot se detiene se realiza la localización estática usando aritmética de intervalos, ver [82, 83]. Una simple intersección entre los resultados que evalúan los mismos datos conduce a un intervalo con mayor precisión. El centro del intervalo se toma como la estimación de la pose, si el intervalo resultante está vacío (por ejemplo, debido a valores atípicos), se realiza una intersección de los datos con mayor compatibilidad, véase [84]. Sin embargo, cuando el AUV está en movimiento debe usar primero sus sensores internos (sensor de profundidad, AHRS y GPS cuando está en la superficie) junto con un modelo de estado para evaluar su posición mediante el método de navegación por estima. Pero, los errores de estimación de posición pueden ser muy altos al final (especialmente si hay fuertes corrientes aleatorias). Una representación visual de la localización estática y dinámica se muestra en la Figura 4.4.

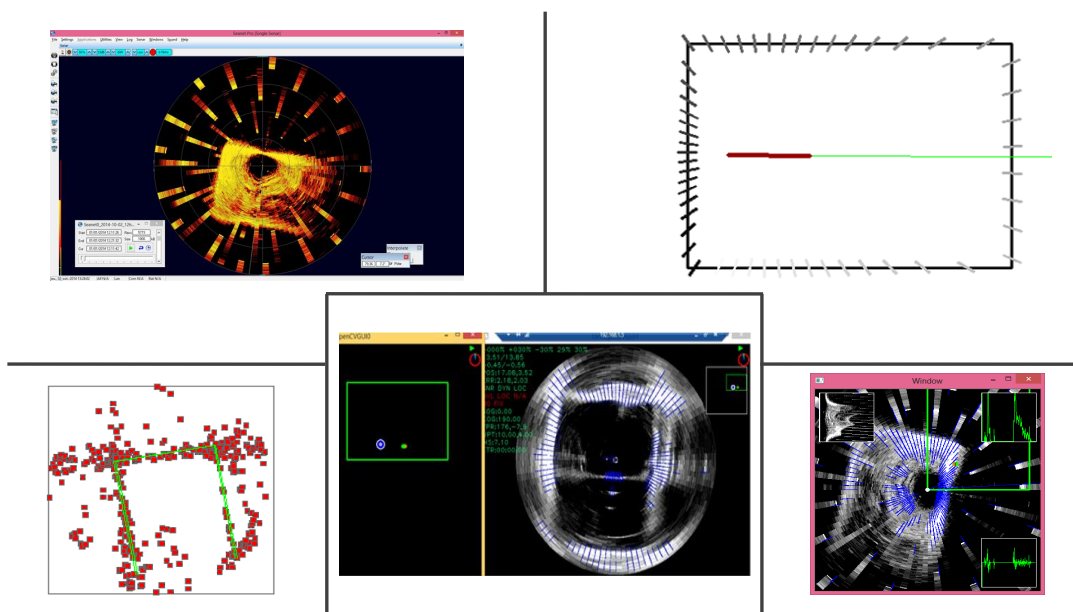


Figura 4.4: Estimación de la posición mediante análisis de intervalos de un vehículo submarino en una alberca utilizando las mediciones del sensor inercial y sonar acústico.

4.1.3. Localización mediante visión

Para estimar la posición del vehículo mediante odometría visual, se utiliza una cámara monocular HD de baja luminosidad. La postura es estimada utilizando el algoritmo de PTAM [85]. Originalmente concebido para la realidad aumentada se utiliza para estimar la postura sin escala de la cámara en una escena desconocida. Las principales ventajas de este algoritmo es que no requiere marcadores especiales, mapas previamente conocidos o escenarios estructurados, y solo requiere una sola cámara (visión monocular). El proceso consiste básicamente en separar el seguimiento y mapeo en dos tareas, procesadas en paralelo. En la Figura 4.5, podemos observar las etapas en cada tarea desde la extracción de puntos característicos, la generación del mapa y la estimación de la pose. Como se observa el resultado es un sistema que produce mapas detallados con puntos de referencia que pueden ser rastreados, ver Figura 4.6.

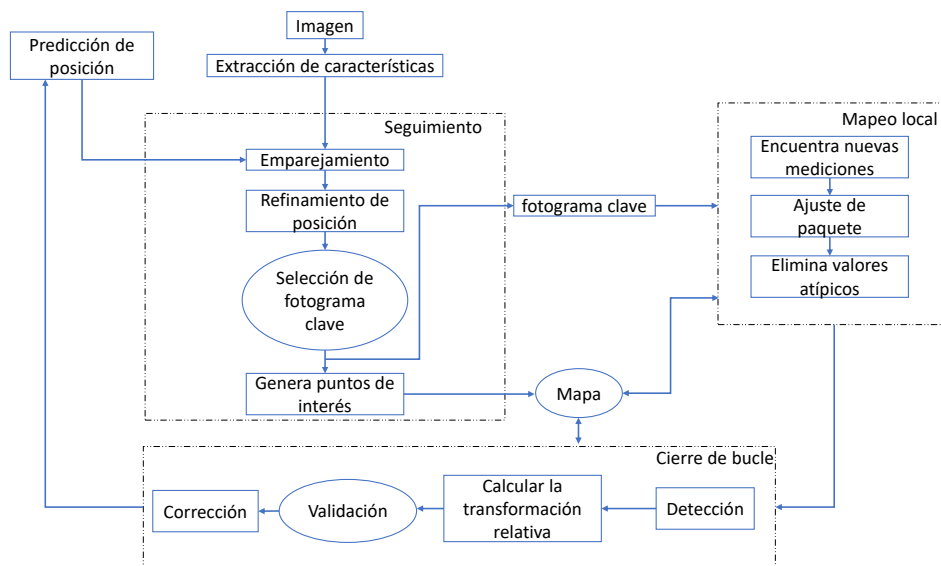


Figura 4.5: Etapas del algoritmo de mapeo y seguimiento paralelo (PTAM).

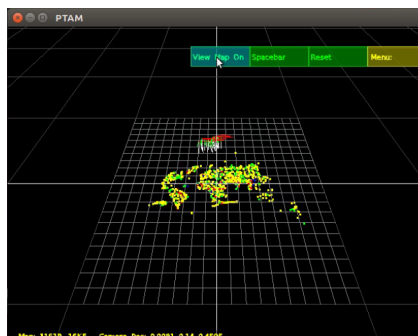


Figura 4.6: Predicción de la posición con respecto al mapa generado por el algoritmo de PTAM.

Sin embargo, el algoritmo de visión monocular proporciona una estimación sin escala de la pose de la cámara, sin deriva con el tiempo, a un ritmo lento. Entonces, para recuperar la escala de la posición y para mejorar la estimación, el algoritmo está fusionado con mediciones inerciales de la IMU (acelerómetros y giroscopios) utilizando un EKF. Esto permite incrementar la tasa de estimación, aprovechando que la medición de los sensores inerciales es a mayor frecuencia de muestreo. Las estimaciones de escala y deriva aseguran la operatividad a largo plazo a pesar de la deriva potencial de un sensor adicional como la odometría visual cuando se usa una cámara monocular. El EKF se divide en dos pasos: predicción y actualización, para la parte de actualización se utiliza la medición de la odometría visual (PTAM) y para la predicción utiliza los datos adquiridos por la IMU. Para detalles con respecto a las ecuaciones del EKF y de la fusión de datos realizada, véase [86]. Para implementar el algoritmo de visión y el filtro extendido de Kalman, se modifica un paquete de ROS para que el BlueROV 2 pueda operar de manera autónoma. De esta forma, se puede obtener la adquisición de datos de los sensores del UUV. Además, se considera la matriz de rotación entre los distintos sistemas de coordenadas y los parámetros del sesgo de giroscopios y acelerómetros. En la Figura 4.7, se pueden ver los principales nodos de ROS que se ejecutan, donde el nodo de BlueROV proporciona las mediciones de los dispositivos desde el vehículo, incluyendo el flujo de datos adquirido por el vídeo que alimenta al nodo PTAM. La fusión de datos inicia cuando el nodo adquiere la estimación de pose de PTAM y la fusiona con las mediciones crudas de los acelerómetros y giroscopios.

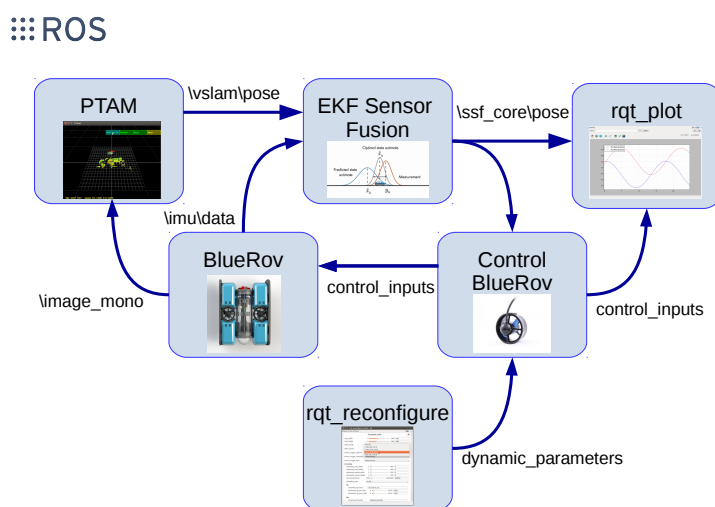


Figura 4.7: Principales nodos de ROS utilizados en la implementación PTAM+EKF.

El algoritmo de PTAM es implementado porque separa los procesos de seguimiento y mapeo, lo que permite realizar un procesamiento rápido en la estimación de la posición. En comparación con el método de localización y mapeo simultáneo usando funciones ORB (ORB-SLAM, por sus siglas en inglés) propuesto en [87] que tiene una gran costo computacional, debido a que incluye el cierre y la relocalización de bucles a gran escala para el mapeo y localización simultáneos, lo que necesita un ajuste de las poses y los puntos del mapa de ese bucle para su optimización.

4.2. Sistema de guía

El sistema de guía calcula la referencia en orientación y posición que se le manda al sistema de control mediante principios fundamentales. Schneider's en [88] define a el sistema de guía como: "El proceso para guiar el camino de un objeto en movimiento hacia un punto dado".

4.2.1. Generación de trayectoria 2D por puntos de referencia

El objetivo es generar una trayectoria en el plano $x - y$, que parte de un punto inicial p_0 hacia cinco distintos puntos p_i en diferentes intervalos de tiempo T_{Fi} considerando que el robot submarino permanezca un tiempo ϵ_i sobre dichos puntos con $i = 1, 2, 3, 4, 5$. Para ello, se tiene que

$$T_{Fi} = T_i + \epsilon_i, \quad (4.1)$$

donde cada tiempo de avance T_i es directamente proporcional a la distancia que hay entre los puntos p_i y p_{i+1} , es decir,

$$T_i = k_{t1} * \|(p_{i+1} - p_i)\|, \quad (4.2)$$

donde k_{t1} es una constante positiva. Para verificar la generación de la trayectoria, se realizó una simulación en Simulink donde las condiciones iniciales son $p_0 = [-1, 7]$, los puntos de referencia están definidos como $p_1 = (2, 6)$, $p_2 = (8, 6)$, $p_3 = (8, 2)$, $p_4 = (2, 2)$, $p_5 = (2, 5)$, mientras que, $\epsilon = 2$ y $k_{t1} = 1$. En la Figura 4.8, se puede observar que los puntos azules denotan la generación de la trayectoria y su paso por cada punto de referencia.

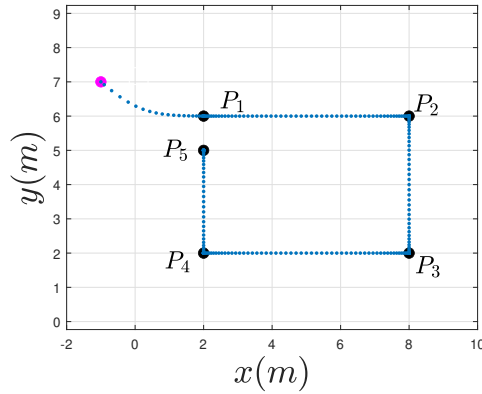


Figura 4.8: Trayectoria mediante seguimiento de puntos de referencia.

4.2.2. Generación de trayectoria 3D

El objetivo es diseñar una trayectoria capaz de ejecutar 3 acciones de manera autónoma: la inmersión a una profundidad deseada, avanzar y emerger. Cada una podrá ajustar la rapidez de la dinámica dependiendo de la tarea a realizar, la trayectoria a generar introduce tres dinámicas de movimiento (x_{1d}, y_{1d}, z_{1d}) en tres intervalos de tiempo específicos dados por $[0, T_1)$, $[T_1, T_2)$ y $[T_2, T_F]$ respectivamente. Las ecuaciones diferenciales que describen la trayectoria son,

$$\begin{aligned}\dot{\Omega}_{1d} &= \Omega_{2d} \\ \dot{\Omega}_{2d} &= u_{\Omega 1} g_1(t) + u_{\Omega 2} f_1(t) g_2(t) + u_{\Omega 3} f_2(t),\end{aligned}\tag{4.3}$$

donde $\Omega = x, y, z, \psi$ y las funciones $u_{\Omega i}$, $f_j(t)$ con $i = 1, 2, 3$ y $j = 1, 2$ son denotadas como

$$u_{\Omega i} = -k_{p\Omega i} (\Omega_{1d} - \Omega_{eq_i}) - k_{d\Omega i} \Omega_{2d}\tag{4.4}$$

$$f_j(t) = \begin{cases} e^{n(t-T_j)} & \text{for } 0 \leq t \leq T_j \\ 1 & \text{for } T_j \leq t \leq T_f, \end{cases}\tag{4.5}$$

con $g_j(t) = 1 - f_j(t)$, $0 < T_1 < T_2 < T_F$, $0 < n, k_{p\Omega i}, k_{d\Omega i}$ y Ω_{eq_i} son constantes. Para verificar el funcionamiento del algoritmo se realiza una simulación mediante Simulink, se parte de una condición inicial de $(0, 0)$ y los movimientos a realizar son los siguientes; el primer punto de referencia es $(0, 0, -2)$, el segundo movimiento es al punto $(5, 0, -2)$ y por último, el punto de referencia es $(5, 0, 0)$. Las ganancias utilizadas para generar la posición deseada en cada punto son: $k_{p_x} = 0.45$, $k_{d_x} = 1.85$, $k_{p_y} = 0.5$, $k_{d_y} = 0.5$, $k_{p_z} = 0.34$,

$k_{dz} = 1.75$ y los tiempos propuestos son $T_1 = 25$, $T_2 = 50$, $T_F = 75$. En la Figura 4.9, es presentada la posición deseada (x_{1r}, y_{1r}, z_{1r}) para los intervalos $[0, T_1]$, $[T_1, T_2]$ y $[T_2, T_F]$ respectivamente.

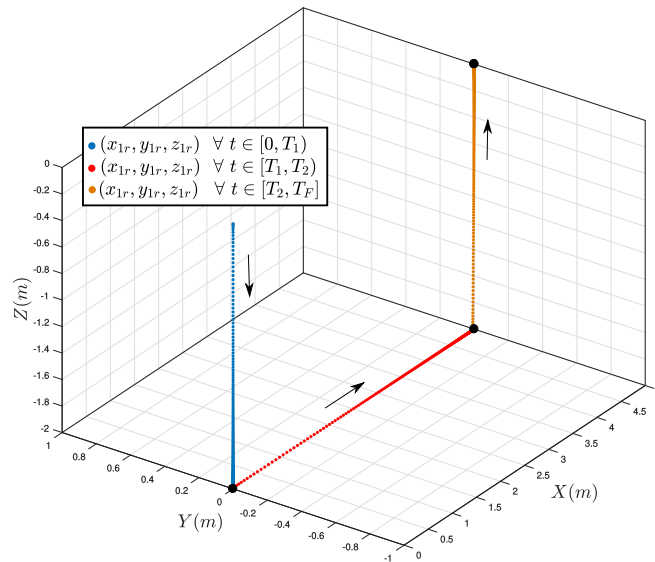


Figura 4.9: Generación de trayectoria 3D, las flechas indican la dirección y el sentido de como se llega a cada punto de referencia en un intervalo de tiempo específico.

4.2.3. Detección y evasión de obstáculos

Para detectar un objeto en un entorno acuático en este trabajo de tesis se usan dispositivos acústicos como el sonar y ecosonda, porque sus mediciones no se ven afectados por las condiciones del entorno acuático (iluminación y turbidez del agua). Estos dispositivos generalmente devuelven datos como imagen o una representación de nube de puntos, donde cada elemento corresponde a una distancia basada en el tiempo de vuelo del eco. Entonces, se presenta el uso de dos sensores acústicos de Trittech para la detección de objetos en un entorno acuático. Con el objetivo de analizar las ventajas y limitantes que tiene cada uno de ellos.

4.2.3.1. Detección de obstáculos

Echosounder Trittech

Este es uno de los sensores de rango más simples, la ecosonda emite un pulso desde su transductor cuando este pulso es reflejado por una superficie, regresa a la cabeza del sensor y se puede medir el tiempo de llegada, entonces, se puede estimar la distancia. Este tipo de

dispositivo es adecuado para medir la distancia del vehículo a estructuras u objetos. Cabe señalar que se coloca en posiciones estáticas y es importante contemplar su resolución y el ancho del haz acústico. Por lo tanto, consultando la hoja de datos del fabricante tenemos que:

- Rango: 50 m de alcance.
- Ancho del haz acústico: 6° hacia cada lado.

Lo que implica, que sí un objeto es detectado a 10 m el cono acústico podría detectar hasta un objeto de 2 m. Ver, Figura 4.10.

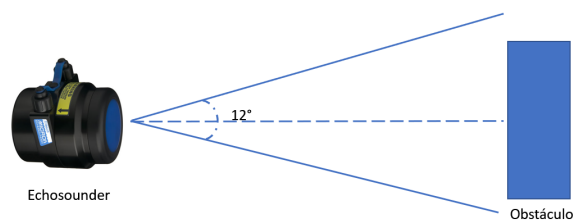


Figura 4.10: Rango de operación de la ecosonda.

Para obtener la medición de la ecosonda, se crea un paquete en ROS que esta conformado por el controlador del dispositivo, y un script en Python donde el algoritmo realiza la adquisición de datos y publica en un tópico la distancia hacia el objeto. Para verificar su funcionamiento se hace una prueba experimental en una pecera, el escenario experimental se puede ver en la Figura 4.11.

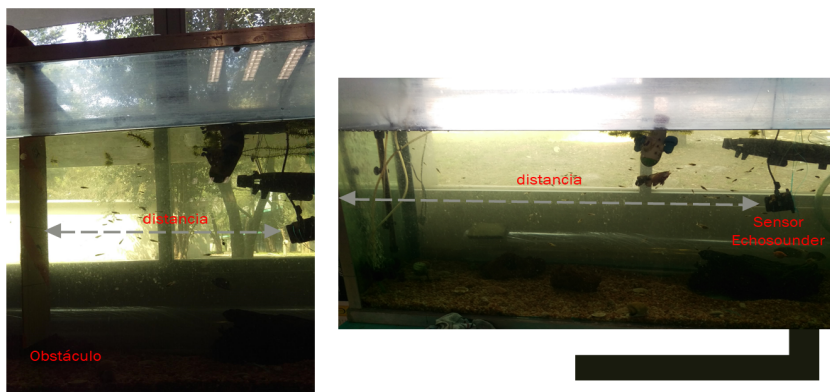


Figura 4.11: Escenario de prueba experimental, lado derecho se visualiza la pecera sin obstáculo y del lado izquierdo con el objeto dentro de la pecera.

El nodo desarrollado en ROS recibe cada medición en un período de adquisición de datos de 20 Hz, una vez adquirida la medición se genera un mensaje el cual es publicado en un tópico

que contiene el dato de la medición hacia el objeto. En la Figura 4.12, se puede visualizar el resultado de los datos adquiridos cuando se introduce una estructura que limita el eco a cierta distancia. Lo que nos permite visualizar el aumento de la distancia medida al retirar el obstáculo.

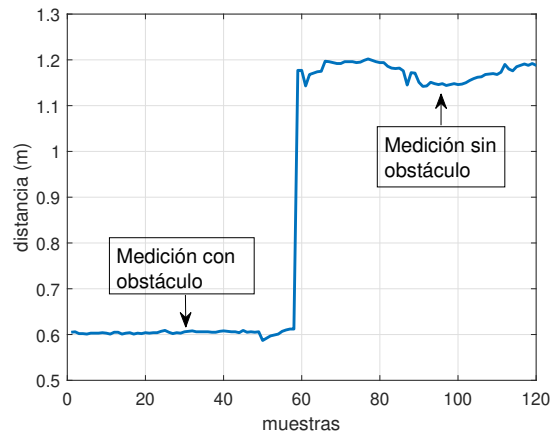


Figura 4.12: Resultados experimentales con la ecosonda mediante ROS.

El sensor permite tener una buena tasa de actualización, sin embargo, el haz acústico actúa de manera muy puntual lo que limita su uso en ambientes con más de un objeto que se encuentren en distintas posiciones.

Sonar Micron Tritech

El sonar Micron de Tritech, es un dispositivo mecánico de un haz acústico de un rango máximo de detección de 75 m, que puede hacer barridos en 360° o en una región específica. La resolución de paso puede configurarse en: 0.45° , 0.9° , 1.8° y 2.4° con un ancho de haz de 35° en vertical y de 3° en horizontal. Véase, Figura 4.13.

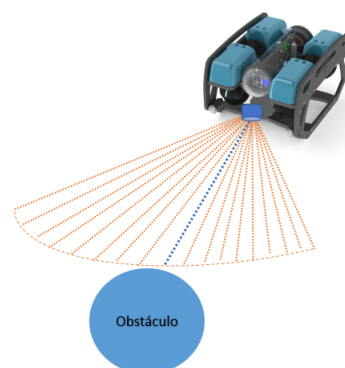


Figura 4.13: Escaneo en una región, la línea azul representa el haz acústico medido.

Para trabajar con este dispositivo, se utiliza el paquete de ROS desarrollado por Micron Tri-tech. El tipo de formato de dato que obtenemos en la medición del sensor es una nube de puntos que puede representar una estructura de datos que permite añadir diversos parámetros, por ejemplo, la intensidad que puede ser utilizada para obtener puntos basados en un cierto nivel de retorno del sensor. Un ejemplo de ello, se puede observar en la Figura 4.14 donde se realiza el escaneo de 360° en una alberca circular. Note que el color azul y morado representan el contorno de la alberca y podemos visualizarlo por el cambio de intensidad.

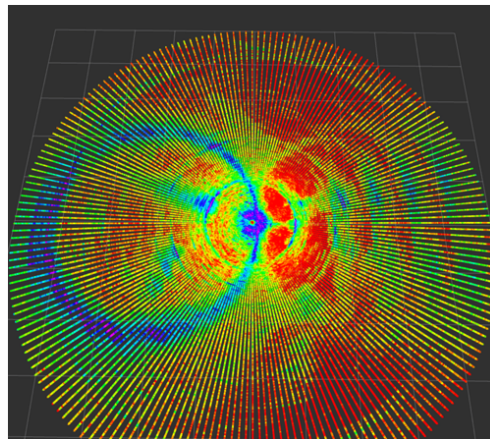


Figura 4.14: Escaneo de 360° en una alberca circular.

Entonces, el primer paso para implementar el filtro es adquirir los datos de la nube de puntos; por ello, se realizó un nodo en ROS que lee de cada haz acústico las 600 mediciones donde se encuentran las coordenadas de cada medición y su valor de intensidad. Para poder publicar un tópico en ROS que contiene un mensaje que incluye las coordenadas 2D de cada punto, su parámetro de intensidad, la orientación y el número del haz acústico desde que inició el barrido. Ver, Figura 4.15.

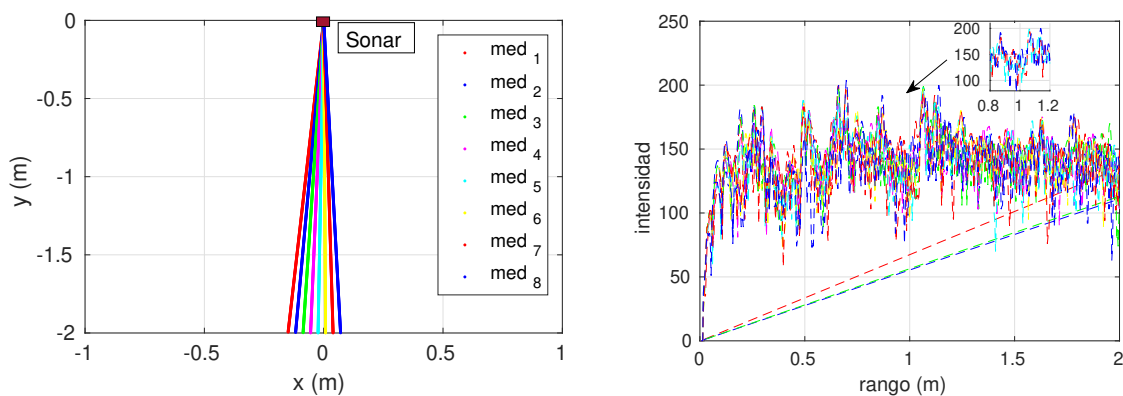


Figura 4.15: Mediciones de cada haz acústico y su respectiva intensidad.

Para continuar, se crea un nodo en ROS para la implementación de un filtro mediante el umbral de intensidad que recupere la posición de cada contenedor del haz acústico de la partición en la que se detecte un umbral de intensidad mayor al permitido [89]. Los pasos a seguir para generar el filtro son:

1. Adquisición de datos del sensor acústico.
2. Generar un histograma.
3. Etapa de filtro suave.
4. Calcular el mínimo local de las mediciones de un escaneo.
5. Calcular el máximo local de las mediciones.
6. Encuentra la similaridad entre las mediciones.
7. Aísla el pico con una ventana, guarda la posición $x-y$ y el valor del pico de intensidad.

Una vez que se obtiene la pose, se almacena en una variable y además se publica en un mensaje. Para verificar el funcionamiento del filtro se crea un rosbag en ROS para grabar los datos de la lectura del sonar en una alberca circular con un barrido solo en una región de 100° y con un obstáculo circular. En la Figura 4.16, se puede observar el resultado del filtro implementado a la señal de la nube de puntos con diferentes umbrales de intensidad.

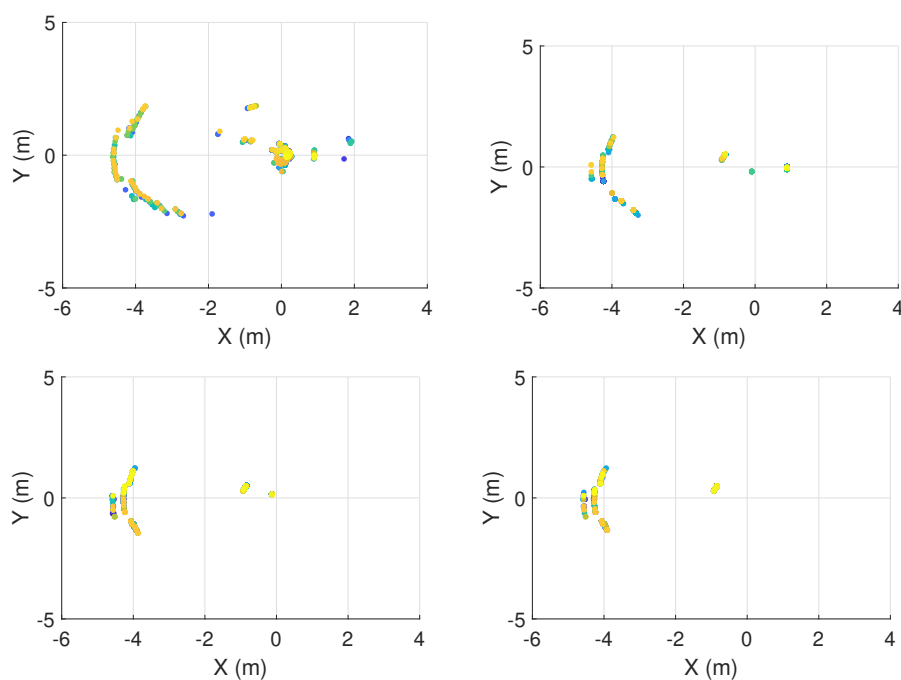


Figura 4.16: Diagramas de dispersión con un umbral de 100, 150, 180 y 190 respectivamente.

Note que con un umbral menor a 190 los valores atípicos causados por el eco no son removidos. En conclusión, con un umbral adecuado el filtro nos ayuda a reducir el ruido de la señal adquirida por el sonar y el algoritmo implementado nos permite tener puntos dispersos del contorno de la alberca y del objeto. Por ello, se tiene que implementar un método para agrupar los datos dispersos obtenidos en la etapa de filtrado.

Técnicas de agrupamiento (Clustering)

Clustering, es el proceso de aprendizaje no supervisado para agrupar datos en clases contemplando una similitud alta entre ellos y baja entre otras clases. Esta clasificación puede ser probabilística, jerárquica o utilizando métodos euclidianos para obtener la distancia numérica entre los datos, lo que implica, encontrar clúster esféricos. Entre las técnicas más utilizadas se encuentran las siguientes:

- **K-means** : tiene como objetivo la partición de un conjunto de n datos en k grupos en el que cada dato pertenece al grupo cuyo valor medio es más cercano.
- **Jerárquico** : es un algoritmo que busca construir una jerarquía de grupos mediante el uso de dos métodos primero el aglomerativo donde cada dato comienza en su propio grupo y el divisivo que considera que todos los datos comienzan en un grupo.
- **Modelo de mezcla gaussiana** : enfoque probabilístico para la agrupación de datos que utiliza la media, covarianza y peso para ajustar un conjunto de datos.
- **Fuzzy c-means** : es un procedimiento que consiste en minimizar iterativamente la forma de una función objetivo (criterio de agrupamiento) hasta obtener una partición difusa óptima.
- **Gustafson-Kessel** : es un algoritmo de agrupación difusa que se ocupa de la medida de pertenencia de un punto a un clúster, en lugar de la clasificación binaria 0/1 (pertenece o no pertenece). Entonces, cada punto no se asigna a un grupo sino que se le da un vector de medidas de membresía. Por lo tanto, un punto generalmente pertenece a todos los grupos de manera simultánea, pero su membresía tiene un peso diferente para los diferentes grupos.

Implementación de técnica de agrupamiento

Asumiendo que la detección es de objetos estáticos en línea y analizando el estado del arte. Se elige el algoritmo de K-means, debido a que permite trabajar con conjuntos de datos grandes o pequeños, es fácil de adaptar y su proceso computacional es rápido. Para construir una partición del conjunto de datos dispersos de n objetos en un conjunto de k clústers, en la Figura 4.17 se presentan los pasos a seguir para el agrupamiento de datos mediante esta técnica, para más detalles ver [90, 91]. El algoritmo itera entre la asignación de cada dato en un clúster y la actualización de los centroides hasta que se cumple un criterio de detención (es decir, ningún punto de datos cambia los clústers, la suma de las distancias se minimiza o se alcanza un número máximo de iteraciones).

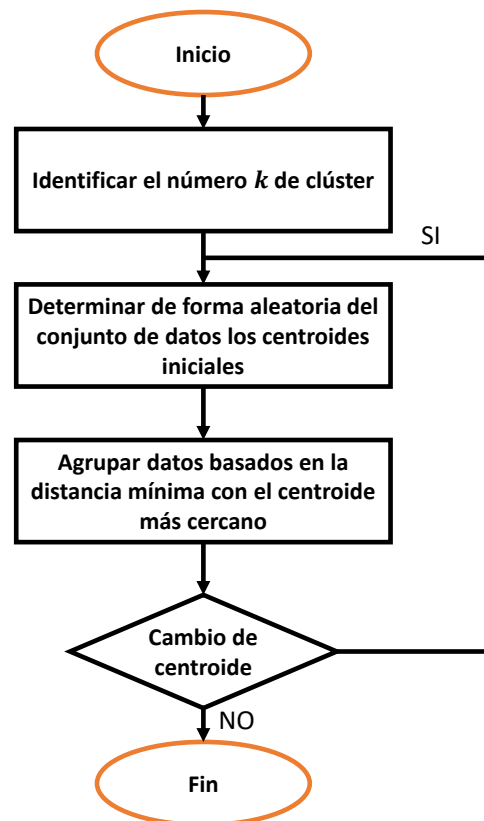


Figura 4.17: Algoritmo K-means, pasos para implementación.

Para comprobar el agrupamiento de datos del algoritmo de K-means con el conjunto de datos obtenidos de la etapa de filtrado. Se realizó una primer prueba utilizando el Clustering Toolbox de Matlab, en la Figura 4.18 se muestra la agrupación de los datos del sonar. Podemos observar que hay 2 clúster, uno en color rojo que corresponde al objeto y el otro en color azul que pertenece al contorno de la alberca.

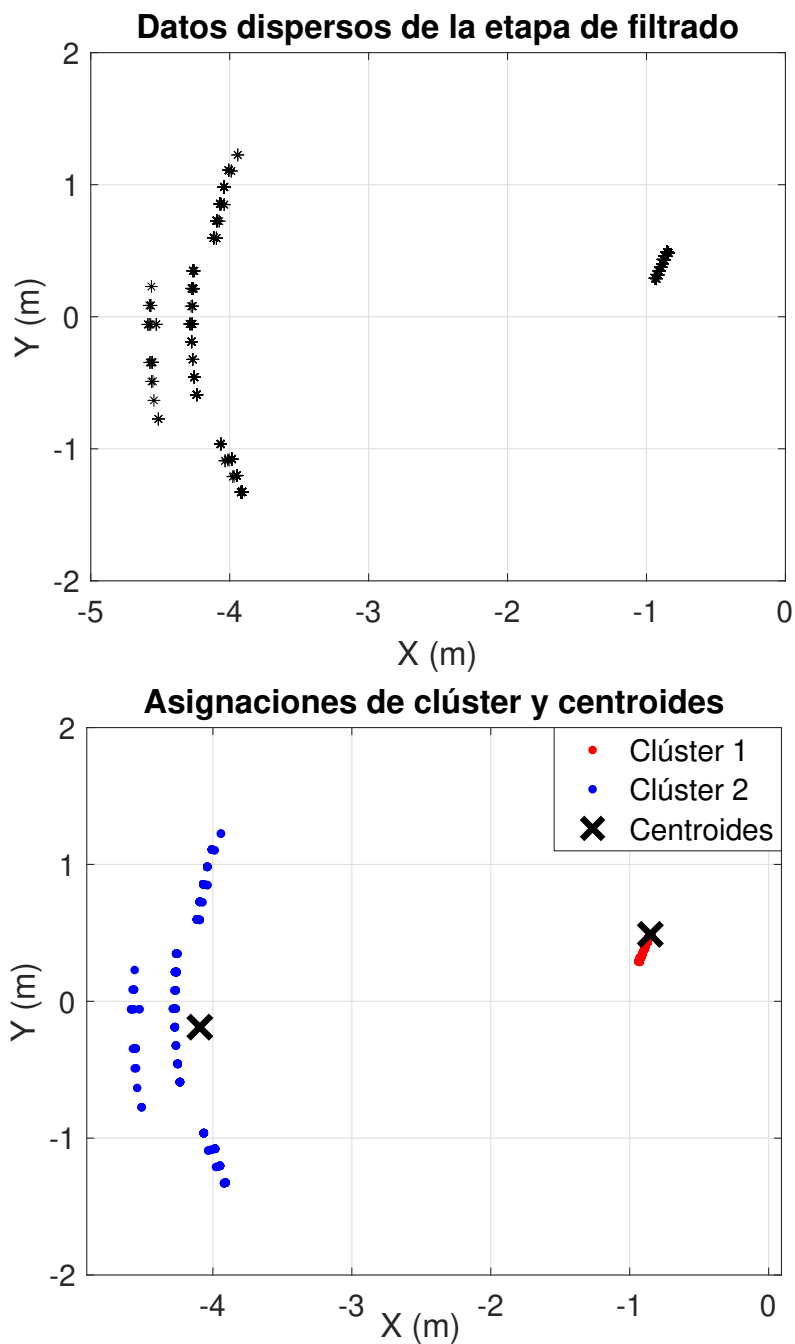


Figura 4.18: Agrupamiento de datos K-means utilizando los datos filtrados del sonar.

4.2.3.2. Técnicas de evasión de obstáculos

Para que un vehículo submarino realice una misión autónoma, tiene que ser capaz de detectar y evadir obstáculos. Esto se debe a que no hay un conocimiento a priori del entorno. En la literatura, se encuentran trabajos que abordan esta problemática como en [28] y [30], donde comparan diversos métodos de planificación de misiones y evasión de obstáculos utilizando diversos vehículos para conocer las ventajas y desventajas al implementar cada

técnica. Además, los autores proponen una clasificación de los métodos que se muestra a continuación:

1. Algoritmos basados en la búsqueda del modelo geométrico

- Dijkstra
- A*
- Algoritmos de marcha rápida

2. Algoritmos probabilísticos basados en muestreo

- Método de ruta probabilística
- Exploramiento rápido de un árbol al azar

3. Campo potencial artificial

- Campo potencial artificial
- Algoritmo del error.

4. Algoritmos de inteligencia

- Optimización de enjambre de partículas
- Algoritmo genético
- Red neuronal artificial

Analizando la información de ambos trabajos, además, considerando que el vehículo cuenta con distintos sensores: IMU, sensor de presión, sistema de posicionamiento acústico y sonar, y asumiendo que el procesamiento del algoritmo se realiza en línea. Se decide trabajar con las técnicas de campo potencial artificial, debido a la rápida velocidad de procesamiento y a su baja dependencia del entorno.

Algoritmo de Campos Potenciales

Asume que los obstáculos y el robot tienen una carga eléctrica del mismo signo, con la finalidad de generar una fuerza de repulsión, mientras que el objetivo tiene asociada una carga eléctrica de signo opuesto, para atraer el robot al punto de destino. La trayectoria es determinada por la resultante de estas fuerzas de atracción o repulsión [92]. Sin embargo,

los campos potenciales pueden ser no convexos, presentando mínimos locales que dejaría estancado al robot en un punto diferente al objetivo. Otra desventaja, es la estimación de los parámetros del campo potencial para lograr una adecuada evasión de obstáculos, ver Figura 4.21. Para evadir mínimos locales, se puede modificar el algoritmo de campos potenciales mediante una fuerza de escape, que aplique una fuerza adicional que le permita al robot, evitar el mínimo local.

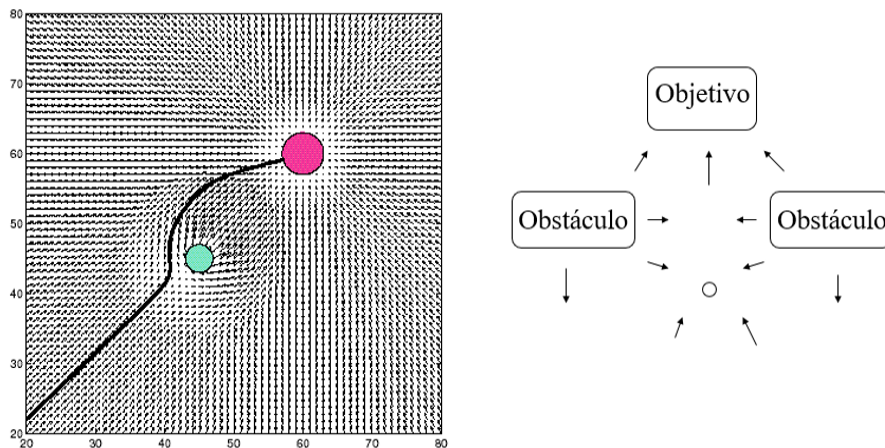


Figura 4.19: Algoritmo de campo potencial, en la imagen izquierda note que el robot evita la colisión con el objeto debido a la fuerza de repulsión. Sin embargo, como se ve en la imagen derecha en un entorno con múltiples obstáculos se puede presentar la problemática de mínimo local dejando parado al robot debido a que cada obstáculo genera una fuerza de repulsión.

Campo Potencial Artificial

El método del campo de potencial artificial (APF, por sus siglas en inglés) propuesto en [93], es un enfoque donde el obstáculo es un punto no deseado, que produce un campo potencial repulsivo para rechazar un objeto que se aproxima. Donde la fuerza de repulsión es igual a la fuerza de atracción dada por el controlador para converger a los valores deseados. Para generar la trayectoria con estos potenciales, sólo se requiere calcular los gradientes. Debido a que no se cuenta con un un sonar de múltiples haces acústicos, se asume que la evasión de obstáculos tiene lugar en el plano (x, y) , es decir, a una profundidad constante.

Entonces, se puede definir la posición en el espacio de trabajo $q = [x, y]^T$, el punto de referencia deseado $q_d = [x_d, y_d]^T$ y la posición del obstáculo $q_{obs} = [x_{obs}, y_{obs}]^T$. De manera

sucesiva, se puede definir el campo potencial artificial (U_{tot}) por,

$$U_{tot} = U_{att} + U_{rep}, \quad (4.6)$$

con

$$U_{att}(q) = \frac{1}{2}k_o\rho_o^j(q, q_d) \quad \forall k_o, j \in \mathbb{R}^+, \quad (4.7)$$

donde k_o es una constante de escalamiento, $\rho_o = \|q_d - q\|$ define la distancia entre la posición del vehículo q y la posición deseada q_d . Mientras que, el campo potencial de repulsión esta denotado como:

$$U_{rep} = \frac{1}{2}\beta_o\left(\frac{1}{\rho(q, q_{obs})} - \frac{1}{\rho_o}\right)^2 \quad \forall \beta_o, \rho_o \in \mathbb{R}^+, \quad (4.8)$$

sí y solo sí,

$$\rho_o(q, q_{obs}) \leq \rho_o \quad U_{rep} = 0 \quad \text{si} \quad \rho_o(q, q_{obs}) > \rho_o \quad (4.9)$$

donde β_o denota un factor de escala y ρ_o define una constante relacionada con la distancia eficiente en la cual actúa el campo de repulsión.

Evasión de obstáculo horizontal mediante campo potencial tangencial

El método APF adopta las superficies atractivas y repulsivas para guiar el robot; sin embargo, hay algunas desventajas: el APF podría producir un problema de mínimos locales. Pero sí utilizamos campos potenciales tangenciales alrededor del objeto, el problema de los mínimos es inherente a este método [94]. En el caso donde se necesita evitar obstáculos en el plano (x, y) , las ecuaciones para generar la fuerza repulsiva son:

$$F_{repx} = q_{y_{obs}} + q_{x_{obs}}(r^2 - q_{x_{obs}}^2 - q_{y_{obs}}^2), \quad (4.10)$$

sí y solo sí $\|q - q_{obs}\| < r$ y $F_{repx} = 0$ sí $\|q - q_{obs}\| > r$. Donde r es el radio de la zona de repulsión, ahora para el caso de F_{repy} ,

$$F_{repy} = q_{x_{obs}} + q_{y_{obs}}(r^2 - q_{x_{obs}}^2 - q_{y_{obs}}^2), \quad (4.11)$$

sí y solo sí $\|q - q_{obs}\| < r$ y $F_{repy} = 0$ sí $\|q - q_{obs}\| > r$. Tenga en cuenta que cuando el campo de rechazo está activo, la fuerza tiene la forma del ciclo límite pero contiene la

distancia entre el obstáculo y el vehículo. Por lo tanto, la entrada de control estará dada por:

$$\tau = U + F_{rep}, \quad (4.12)$$

donde $U = [U_x \ U_y]^T$ es el controlador propuesto y $F_{rep} = [F_{repx} \ F_{repy}]^T$ son fuerzas de repulsión. Pero en el caso de evadir múltiples obstáculos se propone que \vec{q}_{obs_α} talque $\|\vec{q} - \vec{q}_{obs_\alpha}\| = \min[\|\vec{q} - \vec{q}_{obs_{\alpha_1}}\|, \dots, \|\vec{q} - \vec{q}_{obs_n}\|]$, donde n es el número de obstáculos y \vec{q} la posición en el espacio de trabajo del robot submarino. Las ecuaciones para generar las fuerzas repulsivas se pueden definir como:

$$F_{repx_\alpha} = [q_{y_{obs_\alpha}} + q_{x_{obs_\alpha}}(r^2 - q_{x_{obs_\alpha}}^2 - q_{y_{obs_\alpha}}^2)] * \Lambda, \quad (4.13)$$

sí y solo sí $\|q - q_{obs_\alpha}\| < r$ y $F_{repx} = 0$ sí $\|q - q_{obs_\alpha}\| > r$. Donde r es el radio de la zona de repulsión y Λ denota un factor de escala, ahora la siguiente ecuación denota el caso de la fuerza de repulsión en el eje y :

$$F_{repy_\alpha} = [q_{x_{obs_\alpha}} + q_{y_{obs_\alpha}}(r^2 - q_{x_{obs_\alpha}}^2 - q_{y_{obs_\alpha}}^2)] * \Lambda, \quad (4.14)$$

sí y solo sí $\|q - q_{obs_\alpha}\| < r$ y $F_{repy} = 0$ sí $\|q - q_{obs_\alpha}\| > r$. Nótese que cuando el campo de rechazo está activo, la fuerza tiene la forma del ciclo límite pero contiene la distancia entre el obstáculo y el vehículo.

Los algoritmos de error

Una comparación de las distintas técnicas empleadas en la implementación del algoritmo de BUG se puede ver más a detalle en [95]. En los párrafos siguientes se describen estos métodos:

BUG1: es un método que encuentra el obstáculo, una vez localizado hace que el robot rodee completamente el objeto para encontrar la distancia más corta al siguiente punto, luego calcula el límite del obstáculo al punto. La problemática presentada de este método es que puede rodear el objeto más de una vez, además considera que el robot tiene un punto de posicionamiento perfecto y puede medir la distancia $d(x, y)$ entre dos puntos cualquiera. Véase, Figura 4.20.

BUG2: es un método que hace que el robot inicié siguiendo el límite del obstáculo, pero lo deja tan pronto se cruza con el segmento de línea que conecta el punto de partida y el objetivo. Sin embargo, tiene desventajas como: no considerar la cinemática del robot, solo tomar en cuenta las mediciones actuales del sensor lo que ocasiona que el proceso sea lento, ver Figura 4.20.

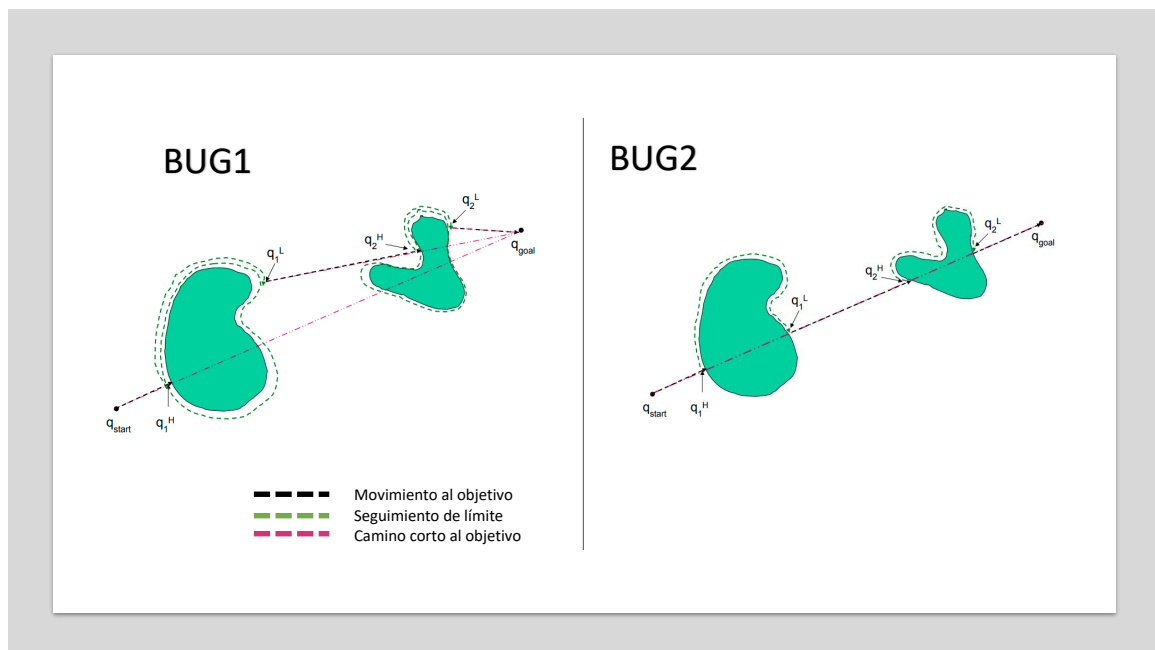


Figura 4.20: Evasión de obstáculos mediante algoritmos de BUG, imágenes adaptadas de [3].

Algoritmo BUG tangencial

El algoritmo de BUG tangencial propuesto en [96], utiliza un sensor de rango y asume al robot como un punto que encuentra los puntos finales de los obstáculos donde puede moverse libremente. El algoritmo hace que el robot se mueva hacia la meta si no hay obstáculo en el camino, si hay obstáculo, se mueve hacia su alrededor hasta que evada al objeto. La problemática básica es calcular la tangente a la curva, tomando el límite del obstáculo en cualquier punto y conducir el robot en esa dirección. Los pasos para realizar el algoritmo son:

- Inicialmente, se crea una línea recta entre el punto de inicio y el objetivo. Para ello, se calcula la distancia mínima entre los dos puntos.
- El robot calcula los puntos radialmente alejados del objeto considerando alguna distancia segura de seguimiento, para desplazarse hacia un punto utiliza los valores de coordenadas obtenidos X_g , Y_g y la orientación dada por θ_g .

- Sí el robot encuentra un obstáculo cerca de la posición actual y en la misma dirección hacia la meta. El límite seguido por el robot se activará para moverse a lo largo de la circunferencia del obstáculo usando X_u, Y_u y la dirección a seguir mediante θ_f .
- La distancia entre la posición actual del robot móvil con respecto al objetivo se calculará usando d_f y d_r .

Donde, las ecuaciones de planificación de movimiento están definidas como:

$$\begin{aligned}
 d_r &= \sqrt{(x_{i+1} - x_g)^2 + (y_{i+1} - y_g)^2} \\
 X_g &= x_{i+1} + \cos \theta_g \\
 Y_g &= y_{i+1} + \sin \theta_g \\
 \theta_g &= \arctan^2((y_{i+1} - y_g), (x_{i+1} - x_g)),
 \end{aligned}
 \tag{4.15}$$

mientras que las ecuaciones acotadas se denotan por:

$$\begin{aligned}
 d_f &= \sqrt{(x_i - x_{i+1})^2 + (y_i - y_{i+1})^2} \\
 X_u &= x_i + \cos \theta_f \\
 Y_u &= y_i + \sin \theta_f \\
 \theta_f &= \arctan^2((y_i - y_{i+1}), (x_i - x_{i+1})).
 \end{aligned}
 \tag{4.16}$$

Para verificar el funcionamiento del algoritmo de BUG tangencial se simula en ROS un escenario con obstáculos donde un robot tiene que llegar a un objetivo, ver Figura 4.21.

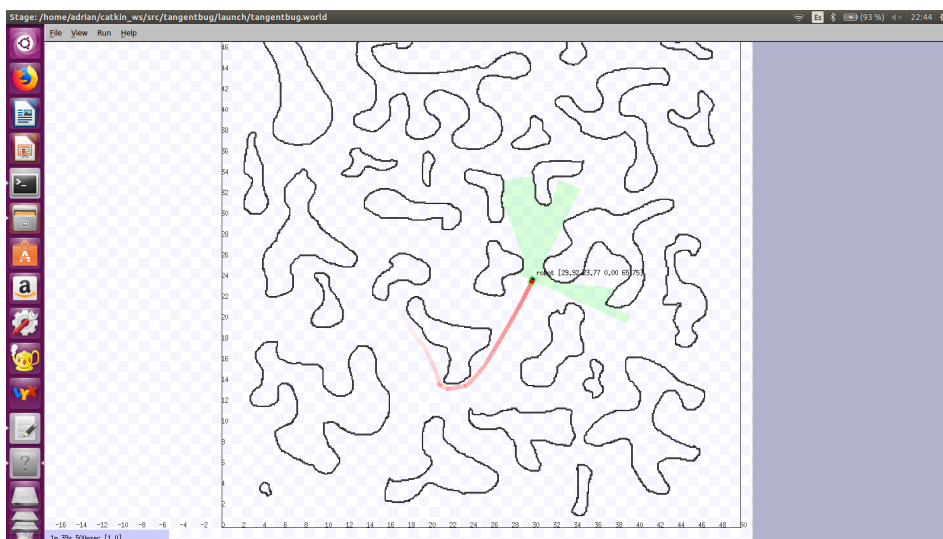


Figura 4.21: Simulación de algoritmo de errores BUG tangencial.

4.3. Conclusión del capítulo

Para conformar los sistemas de guía y navegación, se describen las técnicas de localización, planificación de trayectoria y evasión de obstáculos. Para estimar la posición de un vehículo submarino se presentan técnicas que usan los datos recabados por los sensores internos del vehículo para su localización en un entorno acuático. Dentro de las técnicas presentadas se elige el sistema de posicionamiento acústico basado en SBL, debido a que la técnica de odometría visual (PTAM+EKF) esta limitada por el ángulo visual de la cámara y por factores del entorno; y en el caso de la localización de un vehículo acuático mediante aritmética de intervalos se utiliza el método de navegación por estima para estimar la localización dinámica, lo que implica que los errores de estimación pueden ser muy altos especialmente si hay corrientes.

Para que el sistema de guía pueda enviar un valor de referencia se desarrollan algoritmos para generar trayectorias 2D y 3D mediante el uso puntos de referencia. Considerando que el vehículo se mueve en un entorno desconocido es necesario que el vehículo tenga la capacidad de detectar objetos en su entorno circundante para evitar una colisión. Por ello, se elige el uso del sonar acústico debido a que la ecosonda realiza una lectura muy puntual lo que limita su uso en entornos con múltiples objetos. Entonces, para que el sonar detecte objetos se desarrolla un algoritmo basado en un filtro de umbral de intensidad y la técnica de agrupamiento de K-means. Los resultados demuestran que se pueden agrupar datos dispersos y obtener las coordenadas de los centroides para localizar objetos en un entorno acuático.

Para evadir una colisión, se propone el uso de algoritmos basados en la técnica de campo potencial artificial. Comparando los algoritmos, se puede concluir que la técnica de campo potencial tangencial tiene baja dependencia del entorno y evita el problema del mínimo local. Por otra lado, los métodos de BUG no consideran la cinemática del robot, solo toman en cuenta las mediciones actuales de los sensores. Además, los algoritmos de BUG pueden llegar a rodear un obstáculo hasta que ubiquen la pendiente de la línea recta de la distancia más corta al objetivo. Una mejora es propuesta mediante el algoritmo de BUG tangencial pero la problemática presentada se da en el calculo de la tangente a la curva tomando la distancia hacia el obstáculo en cualquier punto esto hace que el proceso sea lento.

Plataformas experimentales

Este capítulo, describe las plataformas experimentales utilizadas para realizar el conjunto de simulaciones y pruebas experimentales. Se presentan dos prototipos: el LIRMIA III que es un AUV desarrollado en el Laboratorio UMI-LAFMIA y el BlueROV 2 que es un robot submarino comercial desarrollado por Blue Robotics. A continuación, se especifican algunas características de los vehículos, la arquitectura electrónica y se describe el funcionamiento de su estación terrestre.

5.1. LIRMIA III

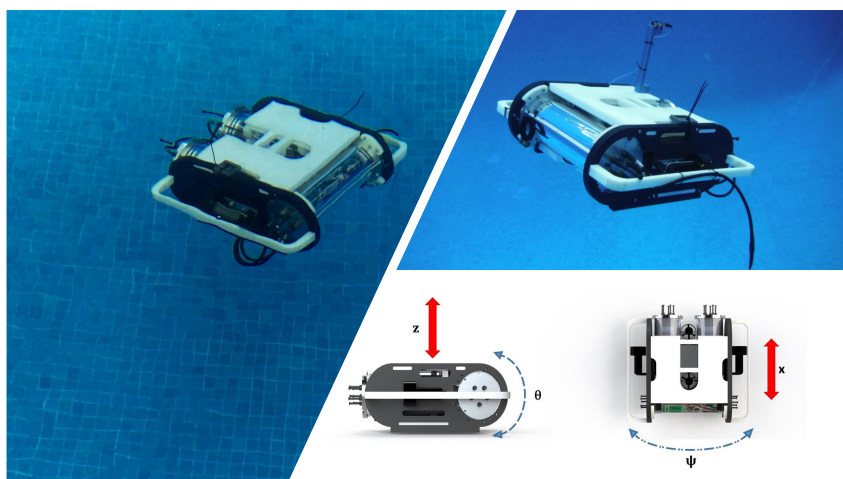


Figura 5.1: LIRMIA III.

Es un vehículo submarino que puede operar en 2 modos: ROV o AUV, tiene una dimensión de $60\text{cm} \times 70\text{cm} \times 30\text{cm}$ con un peso de 35.8 kg. Es importante mencionar que es un vehículo que se desplaza a bajas velocidades, que el origen del sistema de coordenadas fijo al

cuerpo O_B coincide con el centro de gravedad C_G , por diseño es estable mecánicamente en ϕ , cuenta con 4 propulsores Seabotix BD150 que le permiten movimientos en 4 GDL, su autonomía es de 6 horas y soporta una profundidad de operación de 100 m que es determinada por la profundidad máxima soportada por los conectores. En la Figura 5.1, se puede observar un vehículo submarino modular de configuración abierta con la habilidad de tener un control total de los dispositivos mecánicos, esta fabricado a base de naylamid y polietileno con tres tubos de acrílico de 6 pulgadas de diámetro. En la Tabla 5.1, se pueden ver los parámetros conocidos del vehículo LIRMIA III.

Parámetro	Valor	Parámetro	Valor	Parámetro	Valor
W	$107.91(N)$	$X_{\dot{u}}$	$-17.5(kg)$	$Y_{\dot{v}}$	$-36.7(kg)$
\vec{r}_g	$[0, 0, -0.05]^T(m)$	$Z_{\dot{w}}$	$-38.57(kg)$	I_{zz}	$0.16(kg\ m^2)$
β	$109.87(N)$	l_x	$0.156(m)$	l_y	$0.11(m)$
$K_{\dot{p}}$	$-0.12(kg\ m^2/rad)$	$M_{\dot{q}}$	$-0.12(kg\ m^2/rad)$	$N_{\dot{r}}$	$-0.12(kg\ m^2/rad)$

Tabla 5.1: Parámetros físicos y dinámicos del LIRMIA III.

5.1.1. Sistema de propulsión

El LIRMIA III esta equipado con cuatro propulsores DC BTD150 de Seabotix que permiten asegurar que el vehículo submarino se mueva donde el controlador autónomo u operador necesite. En la Figura 5.2, se observa que cuenta con dos propulsores verticales y dos horizontales. Para minimizar las reacciones del momento producido por el torque, los propulsores son capaces de girar en sentido horario y antihorario. Por medio de señales de modulación de ancho de pulso (PWM, por sus siglas en inglés) generadas desde el arduino-uno y enviadas a los controladores de velocidad electrónicos MD-03 que son responsables de generar la señal apropiada para manipular la magnitud y dirección de la fuerza generada por los propulsores.

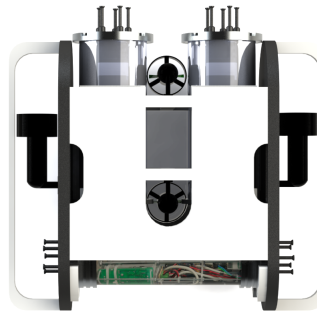


Figura 5.2: CAD LIRMIA III, sistema de propulsión mediante 4 propulsores Seabotix BTD150.

5.1.2. Arquitectura electrónica

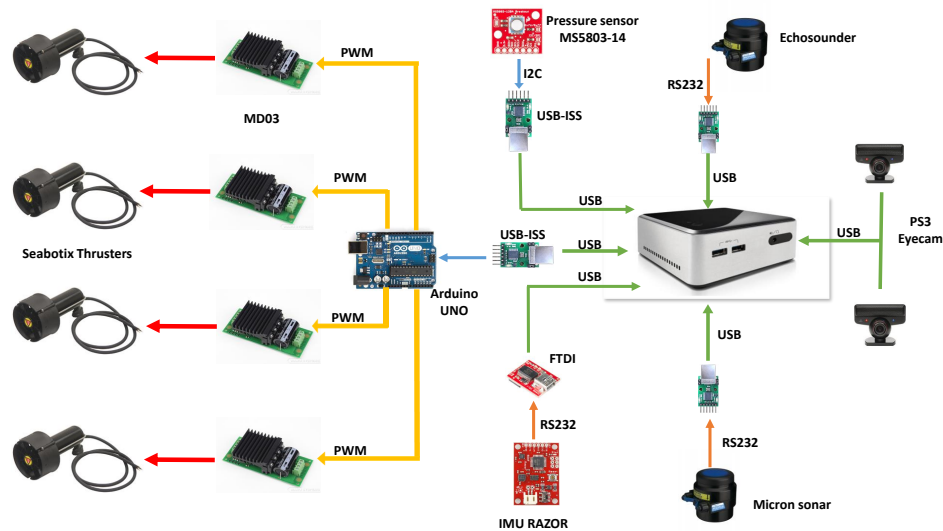


Figura 5.3: Arquitectura electrónica del LIRMIA III.

La arquitectura electrónica del LIRMIA III está basada en un ordenador INTEL NUC Core i5 que calcula los algoritmos propuestos y adquiere datos de los sensores para que el vehículo realice una acción o misión autónoma, mientras que la percepción del entorno del robot se realiza utilizando varios sensores y dispositivos. Una IMU de 9 GDL es instrumentada para conocer la orientación y velocidad angular de nuestro vehículo, el sensor de presión permite calcular la profundidad, mientras que dos cámaras (PlayStation Eye) se utilizan para adquirir información de las imágenes, no obstante, su uso depende de la misión que sea asignada al vehículo. Además, puede incorporar dos sensores acústicos (sonar y echosounder) para localizar el vehículo y detectar obstáculos. Como interfaz de comunicación entre los controladores de velocidad y el ordenador incorporado se utiliza el arduino-uno, mientras que el USB-ISS comunica todos los dispositivos que utilizan el protocolo de comunicación I2C al ordenador, véase Figura 5.3.

5.1.3. Estación terrestre

El LIRMIA III utiliza el sistema operativo Windows. Para obtener una comunicación bidireccional con la estación en tierra se utiliza la comunicación SSH mediante el escritorio remoto. En la Figura 5.4, se puede visualizar que la estación en tierra es una netbook que se comunica vía WiFi a un módem y éste a su vez se comunica con la computadora del robot mediante conexión Ethernet.

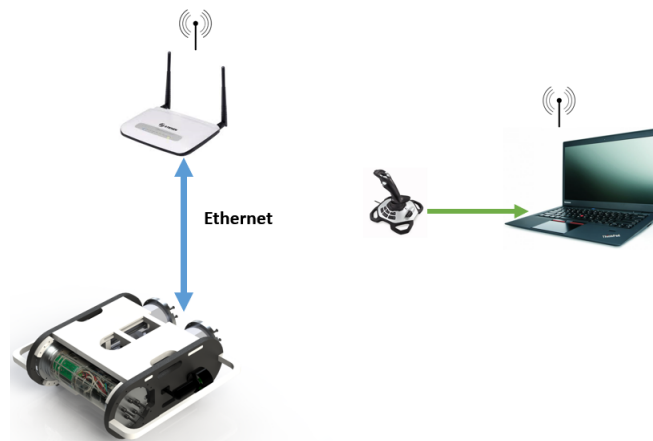


Figura 5.4: Estación en tierra LIRMIA III.

Para realizar pruebas de diversos algoritmos y de la interfaz de usuario se utiliza el lenguaje C++ para programar los algoritmos de control y navegación, sin embargo, para realizar la localización acústica por el método de análisis de intervalos se implementó el software UvxCtrl desarrollado por el equipo de robótica de ENSTA-Bretagne, ver Figura 5.5. El software contiene un programa utilizado para controlar y configurar diferentes tipos de robot entre ellos UUV's. Está programado mediante hilos de manera que disminuye el coste computacional y gestiona la comunicación entre cada nodo, esto facilita la comprobación del funcionamiento de los dispositivos y permite realizar cambios en los algoritmos de control, visión y planificación de misión. Sus herramientas y librerías de análisis de intervalos permiten implementar técnicas de localización acústica.

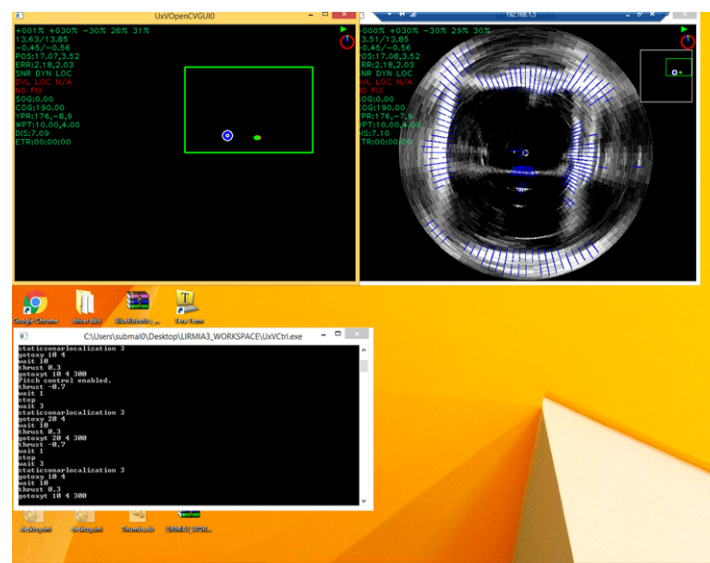


Figura 5.5: UvxCtrl, interfaz de usuario.

5.2. BlueROV 2

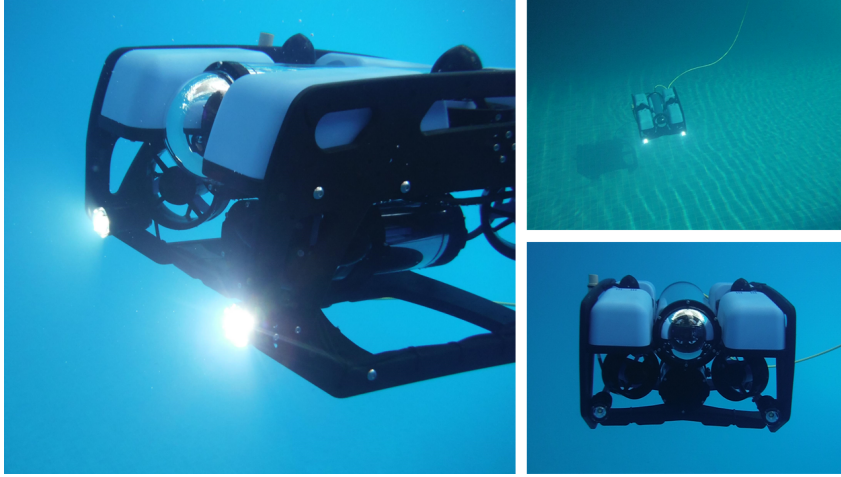


Figura 5.6: BlueROV 2.

El BlueROV 2 es un vehículo submarino que opera de manera remota, sin embargo, podemos modificar su configuración para que opere de manera autónoma. Es un vehículo modular de estructura abierta con una dimensión de $45.71\text{cm} \times 33.81\text{cm} \times 22.1\text{cm}$, un peso de 11.5 kg y una profundidad de operación máxima de 100 m . Este robot submarino tiene 6 propulsores Brushless T200 que le permiten realizar movimientos desacoplados en 4 GDL, como se mencionó anteriormente es estable mecánicamente en ϕ y θ . Tiene una autonomía de 2 h que depende del uso de las luces. Su estructura es fabricada de polietileno de alta densidad y se complementa con dos tubos de acrílico que almacenan los componentes electrónicos y la batería tipo LiPo. Ver, Figura 5.6.

Los parámetros conocidos del sistema se definen en la Tabla 5.2 para más detalles ver [97].

Parámetro	Valor	Parámetro	Valor	Parámetro	Valor
W	$112.8(N)$	$X_{\dot{u}}$	$-5.5(kg)$	$Y_{\dot{v}}$	$-12.7(kg)$
\vec{r}_b	$[0, 0, -0.02]^T(m)$	$Z_{\dot{w}}$	$-14.57(kg)$	I_{xx}	$0.16(kg\ m^2)$
β	$114.8(N)$	I_{yy}	$0.16(kg\ m^2)$	I_{zz}	$0.16(kg\ m^2)$
$K_{\dot{p}}$	$-0.12(kg\ m^2/rad)$	$M_{\dot{q}}$	$-0.12(kg\ m^2/rad)$	$N_{\dot{r}}$	$-0.12(kg\ m^2/rad)$
X_u	$-4.03(Ns/m)$	Y_v	$-6.22(Ns/m)$	Z_w	$-5.18(Ns/m)$
K_p	$-0.07(Ns/rad)$	M_q	$-0.07(Ns/rad)$	N_r	$-0.07(Ns/rad)$
X_{u_c}	$-18.18(Ns^2/m^2)$	Y_{v_c}	$-21.66(Ns^2/m^2)$	Z_{w_c}	$-36.99(Ns^2/m^2)$
K_{p_c}	$-1.55(Ns^2/rad^2)$	M_{q_c}	$-1.55(Ns^2/rad^2)$	N_{r_c}	$-1.55(Ns^2/rad^2)$

Tabla 5.2: Parámetros físicos y dinámicos del BlueROV 2.

5.2.1. Arquitectura electrónica

El BlueROV 2 incorpora un piloto automático Pixhawk como computadora embebida que adquiere los datos de los sensores: el sensor de presión Bar30 permite calcular la profundidad del vehículo con una precisión de centímetros, los sensores incorporados al Pixhawk (giroscopio, acelerómetro y magnetómetro) le permiten estimar la orientación del vehículo y el sensor de filtraciones de agua nos alerta si entra agua al contenedor electrónico. Una computadora a bordo Raspberry Pi 3 se encarga de la adquisición de información mediante la cámara Hd de baja luminosidad y de la interfaz de comunicación Fathom-X. El sistema de comunicación se compone de 2 módems que se comunican mediante un cable Ethernet de 100 m a una velocidad de ancho de banda de 80 Mb/s. Es importante mencionar que el algoritmo de control se calcula en la estación terrestre y devuelve el valor de la modulación de ancho de pulso (PWM) a los 6 controladores de velocidad electrónicos y estos producen el voltaje adecuado a cada propulsor del vehículo. Véase, Figura 5.7.

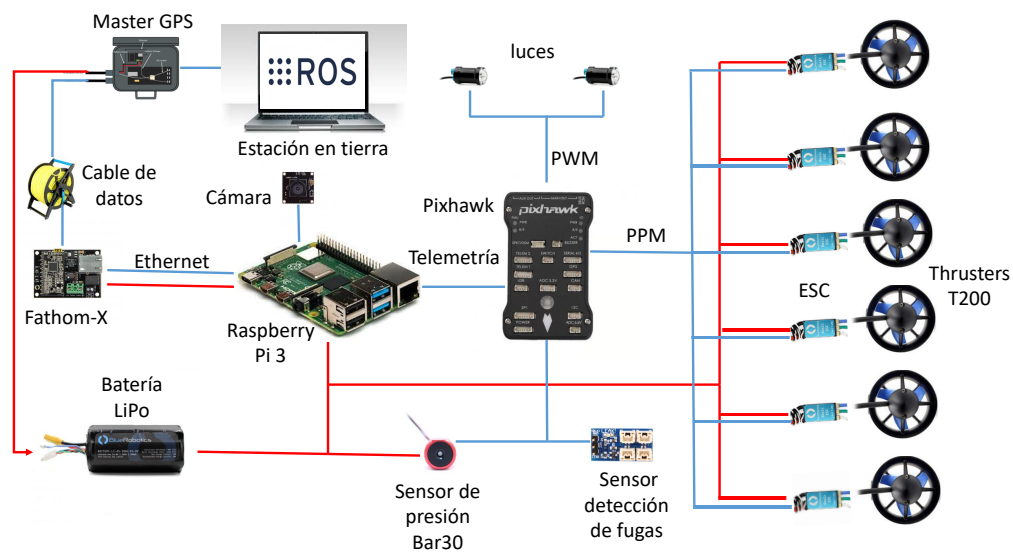


Figura 5.7: Arquitectura electrónica del BlueROV 2.

5.2.2. Sistema de propulsión

BlueROV 2 tiene seis propulsores del tipo T200 (BlueRobotics), tiene cuatro propulsores horizontales y dos verticales como la configuración ilustrada en Figura 5.8. Los propulsores T200 tienen propelas con orientación en sentido horario y antihorario, lo que permite minimizar las reacciones del torque. En la Figura 5.8, los propulsores verdes ilustran la con-

figuración de la propela en sentido antihorario y los propulsores azules indican las propelas en sentido horario. Estos propulsores se controlan mediante señales de modulación de ancho de pulso (PWM) enviadas desde el piloto automático hacia los controladores electrónicos de velocidad (ESC's, por sus siglas en inglés).

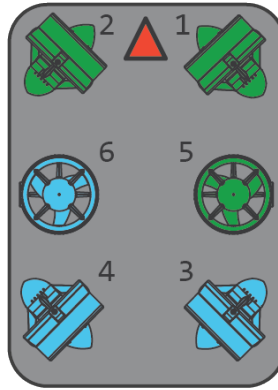


Figura 5.8: Esquemático del sistema de propulsión del BlueROV 2.

5.2.3. Estación terrestre

En la Figura 5.9, se observa que la estación terrestre esta compuesta por una notebook Lenovo Thinkpad T430 que opera bajo el sistema operativo Linux Xenial 16.04 LTS y ROS Kinetic. También, incorpora las tarjetas Fathom-x que sirven de interfaz de comunicación con el BlueROV 2 y con la tarjeta máster del GPS Water Linked. Para la comunicación el estándar de redes Ethernet es utilizado y el uso del protocolo de comunicación SSH. Además, se tiene un sistema de localización acústica descrito a detalle en el Capítulo 4.

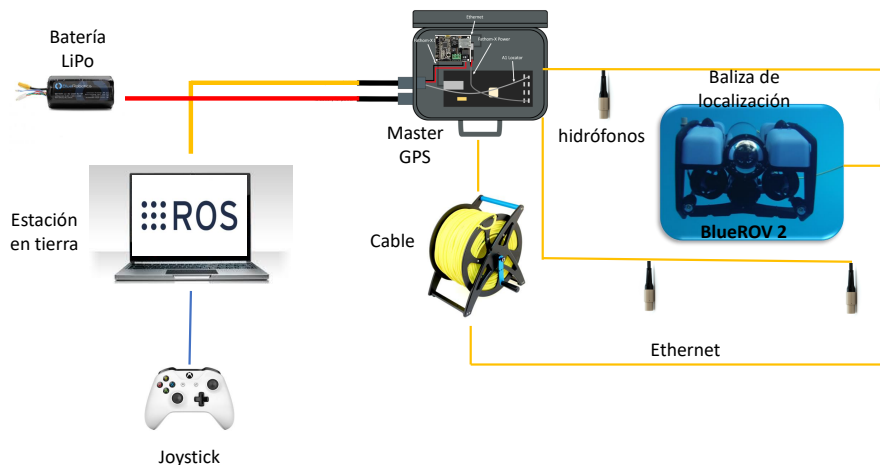


Figura 5.9: Estación terrestre del BlueROV 2.

El robot submarino esta basado en un sistema operativo base Linux de código abierto llamado ArduSub, se puede manipular de manera remota mediante el planificador de misiones QGroundControl. Sin embargo, para realizar tareas de manera autónoma se utiliza el sistema operativo robótico (ROS, por sus siglas en inglés), debido a que podemos programar en múltiples lenguajes y hacer uso de código abierto. En la Figura 5.10, se pueden ver a detalle los nodos y tópicos que interactúan para comunicar la estación terrestre y el piloto automático en ROS.

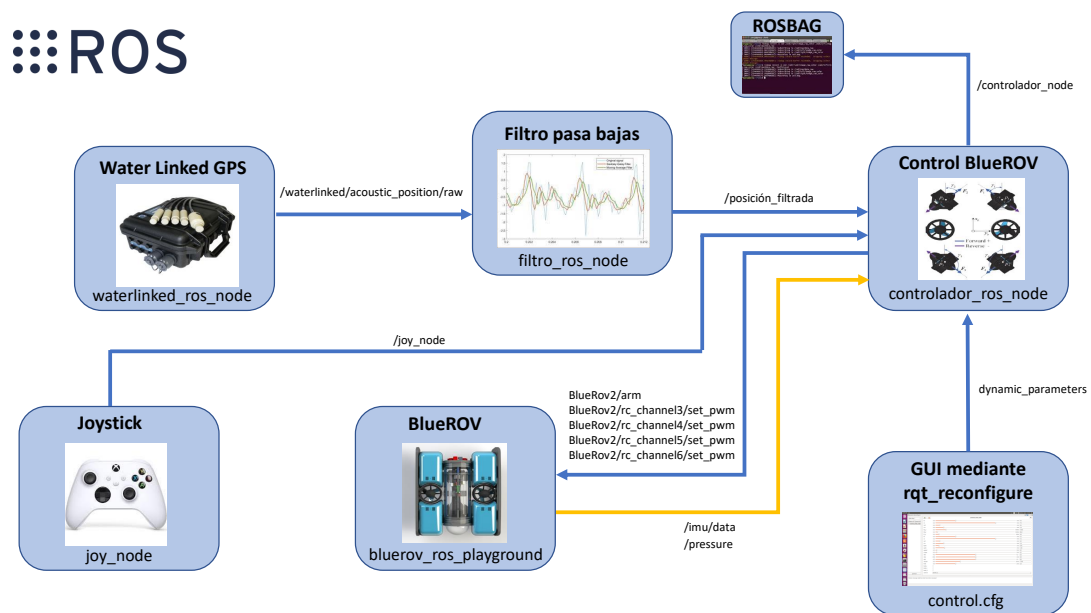


Figura 5.10: Diagrama de bloques sobre la interacción de los nodos del BlueROV y del Water Linked.

Donde el nodo `bluerov_ros_playground` nos permite comunicarnos con el vehículo para obtener las mediciones del sensor inercial, del sensor de presión y de la cámara, además, envía a los ESC's el PWM adecuado para que el robot se desplace o rote. Por otra parte, para localizar el vehículo mediante el sistema de posicionamiento acústico hacemos uso del nodo de Water Linked que publica la estimación de la posición acústica al nodo del filtro el cuál publica la posición filtrada al nodo del controlador que calcula las fuerzas y momentos para lograr un seguimiento de trayectoria autónoma. Finalmente, los nodos `rqt_reconfigure` y `joy_node` se utilizan respectivamente para el ajuste de parámetros en línea mediante una interfaz gráfica de usuario (GUI, por sus siglas en inglés) y el envío de comandos mediante el joystick. Es importante señalar que los nodos de control y guía se procesan en la estación en tierra con un período de muestreo de hasta 50 Hz.

5.3. Conclusión del capítulo

Se describen dos plataformas experimentales que pueden operar como ROV o AUV. Estas plataformas son utilizadas para realizar la implementación de los algoritmos propuestos en los capítulos 3 y 4. El LIRMIA III es un UUV desarrollado en Laboratorio UMI-LAFMIA, mientras que el BlueROV 2 es un UUV comercial desarrollado por Blue Robotics. La diferencia parte, en que el LIRMIA III utiliza el sistema operativo WINDOWS y el software UvxCtrl basado en lenguaje C++ para programar los algoritmos de guía, navegación y control en su computadora interna. En cambio el BlueROV 2 utiliza ROS para comunicar la estación en tierra y el vehículo, de esta forma la estación en tierra procesa la información de los sensores para calcular la señal de control que se envía al vehículo.

Otra diferencia importante, es que utilizan distintos dispositivos y técnicas para estimar la posición del vehículo submarino. Mientras el LIRMIA III utiliza aritmética de intervalos para estimar su posición a partir de la medición del sonar, GPS e IMU. El BlueROV 2 utiliza un sistema de posicionamiento acústico basado en el método de SBL para triangular la posición del vehículo utilizando el TDoA.

Resultados

En este Capítulo, se analizan los resultados obtenidos de un conjunto de simulaciones y experimentos en tiempo real. Los algoritmos a comprobar son: métodos de localización, estrategias de control, técnicas de detección y evasión de obstáculos propuestos en los Capítulos previos. Es importante señalar, que se utilizaron los UUV's descritos en el Capítulo 5 para realizar las pruebas experimentales.

6.1. Simulaciones numéricas

6.1.1. Sistema de control

En los escenarios de simulación propuestos se proponen dos tipos de perturbaciones, una función sinusoidal para simular el efecto que producen las olas en un vehículo submarino y perturbaciones de alta frecuencia generadas para comprobar la robustez de los controladores en ambientes donde el mar de viento y el mar de fondo incrementan la frecuencia de las olas.

6.1.1.1. Control de modos deslizante integral de segundo orden

El diseño del controlador propuesto en la sección 3.2.1, se comprueba con un conjunto de pruebas para abordar el seguimiento de trayectoria en el BlueROV 2. Observe que los controladores aplicados v_i son los definidos en en el sistema de ecuaciones (3.15). Según la metodología de control, cada variable de control U_i viene dada por:

$$U_i = -c_i e_{i2} - k_{1_i} |\sigma_i|^{1/2} \text{sign}(\sigma_i) - k_{2_i} \int_0^t \text{sign}(\sigma_i) d\tau - k_{3_i} S_i, \quad (6.1)$$

con

$$Z_i = -k_{3_i} \int_0^t S_i d\tau,$$

donde $i = x, y, z, \psi$ y las condiciones iniciales son dadas como $S_i(0) = Z_i(0) = 0$. Para

introducir las perturbaciones acotadas $\xi_i = \frac{\delta_i}{I_i}$, se define que

$$\xi_i = \frac{A_i \text{sign}(\sigma_i) \sin(B_i t)}{I_i}. \quad (6.2)$$

Para comparar la eficiencia del control por modos deslizantes integral de segundo orden con efecto de super-twisting con un control de modo deslizante integral (ISMC, por sus siglas en inglés) propuesto por [65]. La trayectoria de referencia en esta prueba es $(x_{1r}, y_{1r}, z_{1r}, \psi_{1r})$, cuyas dinámicas fueron descritas en la sección 4.2.2. En la Tabla 6.1, se pueden ver los valores de los parámetros utilizados durante las pruebas. Para la simulación $\lambda_{1_i}, \lambda_{2_i}$ se sintonizan teniendo en cuenta las condiciones propuestas en el Teorema 3.2.1 y para hacer $S_i = 0$ para todo tiempo, cada k_{3_i} es propuesta igual a 1. Las condiciones iniciales para el UUV son $\bar{\eta} = [0, 0, 0, 0.1]$, los parámetros para generar la posición deseada en cada punto son dados por $k_{p_{\Omega x}} = 0.45, k_{d_{\Omega x}} = 1.85, k_{p_{\Omega y}} = 0.5, k_{d_{\Omega y}} = 0.5, k_{p_{\Omega z}} = 0.34, k_{d_{\Omega z}} = 1.75$. Por otro lado, los tiempos propuestos son $T_1 = 25, T_2 = 50, T_F = 75$ y los puntos están definidos por $P_1 = (0, 0, -2), P_2 = (5, 0, -2), P_3 = (5, 0, 0)$.

	A_i	B_i	ξ_i^+	L_i	λ_{1_i}	λ_{2_i}	k_{1_i}	k_{2_i}	I_i
x	0.3	6	0.11	0.01	1	1	1.29	0.64	17
y	0.7	8	0.24	0.03	3	1.5	1.82	1.36	34.2
z	0.6	3	0.07	0.02	1.5	1.2	0.79	0.07	26.07
ψ	0.4	4	5.71	1.42	2.5	3	17.52	26.28	0.28

Tabla 6.1: Datos de simulación del controlador SISMIC

Las Figuras 6.1 y 6.2, muestran los resultados del seguimiento de la trayectoria. Note que ambas estrategias de control alcanzaron la trayectoria de referencia, sin embargo, el error de seguimiento es menor con SISMIC incluso cuando la dinámica del vehículo se ve afectada por perturbaciones limitadas a altas frecuencias, ver Tabla 6.2.

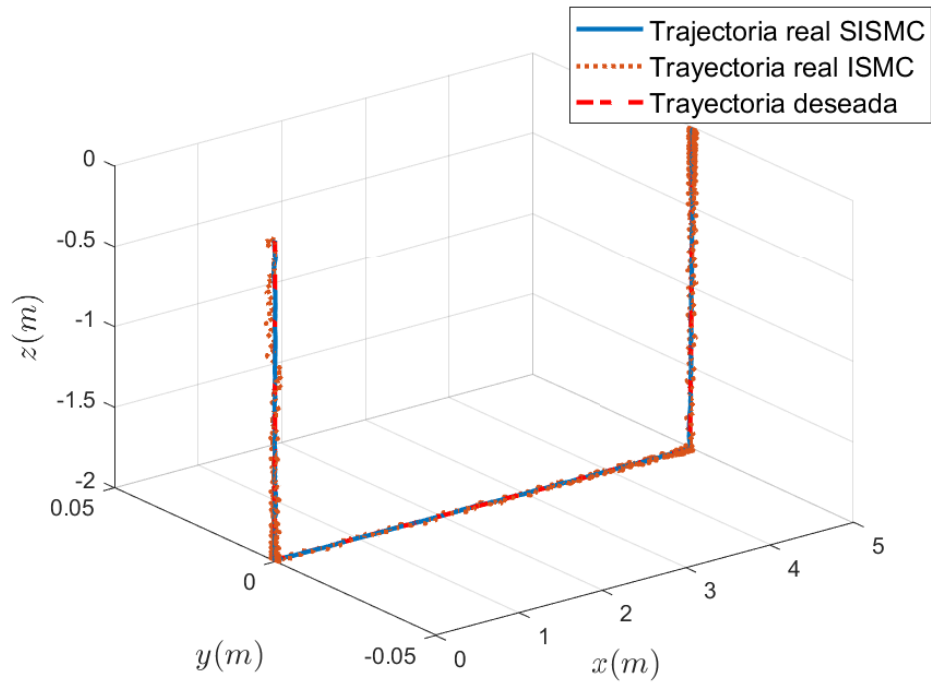


Figura 6.1: Vista 3D del seguimiento de trayectoria.

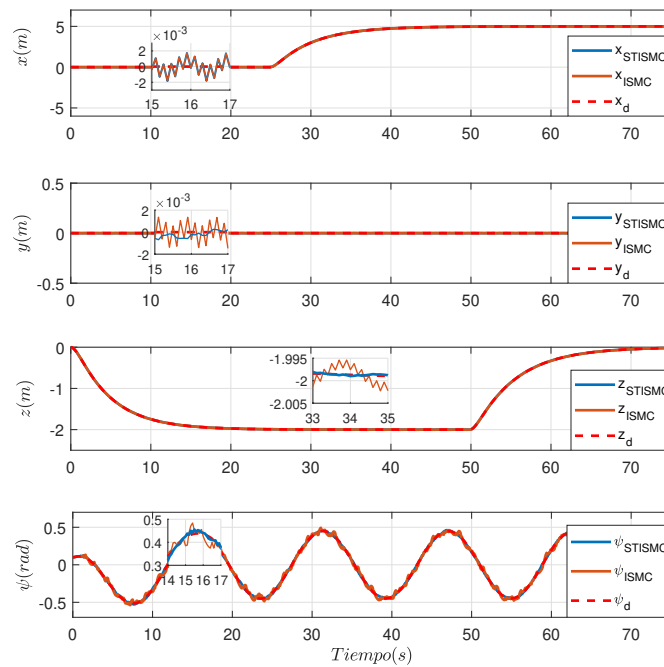


Figura 6.2: Comparaciones de seguimiento de trayectoria en posición y orientación. Trayectoria deseada (línea roja discontinua), trayectoria con SISMIC (línea azul sólida) y trayectoria con controlador ISMC (línea naranja sólida).

RMSE		
	ISMC	SIMC
x	0.0011(m)	0.0001667(m)
y	0.0010(m)	0.000216(m)
z	0.0019(m)	0.000163(m)
ψ	0.0304($^{\circ}$)	0.0069($^{\circ}$)

Tabla 6.2: Criterio de error cuadrático medio para SIMC e ISMC.

En la Figura 6.3, se puede observar que σ_i y S_i con SIMC permanece cercanas a cero todo el tiempo, donde $\sigma_i(t \geq 0) = 0$ implica que cada perturbación acotada ξ_i es compensada desde $t = 0$, y $S_i(t \geq 0) = 0$, implica que cada error $e_i(t \geq 0) = 0$, esta es una clara indicación del seguimiento de la dinámica deseada. Por otro lado, podemos ver que no se aprecia el efecto de conmutación en las entradas de control, debido a la continuidad del controlador SIMC, es decir, una forma de lograr robustez desde el primer momento. Sin embargo, en la Figura 6.4 nosotros observamos que σ_i y S_i presentan oscilaciones cuando converge a una región cercana a cero. Además, tenga en cuenta que el control de modo deslizante integral tiene el efecto de conmutación en las entradas de control debido al control de retroalimentación discontinua.

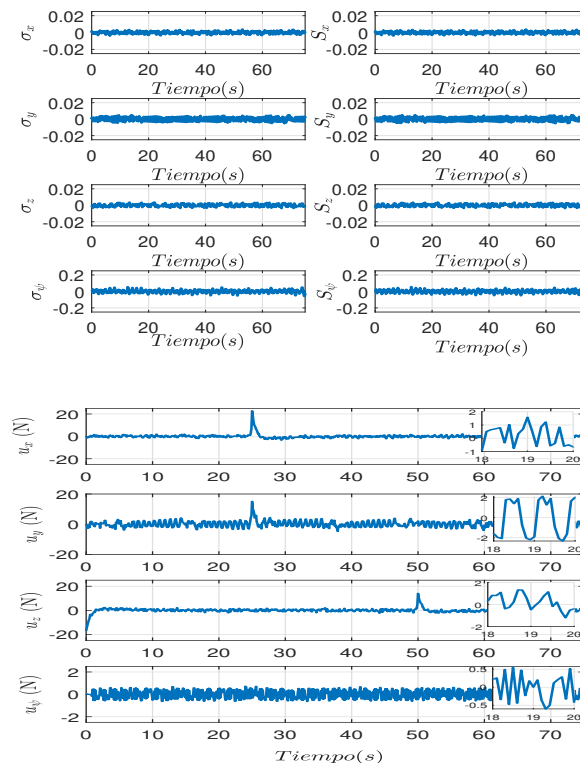


Figura 6.3: Entradas de control SIMC con atenuación del efecto de conmutación, mientras que las superficies deslizantes y variables auxiliares que convergen en una región cercana a cero.

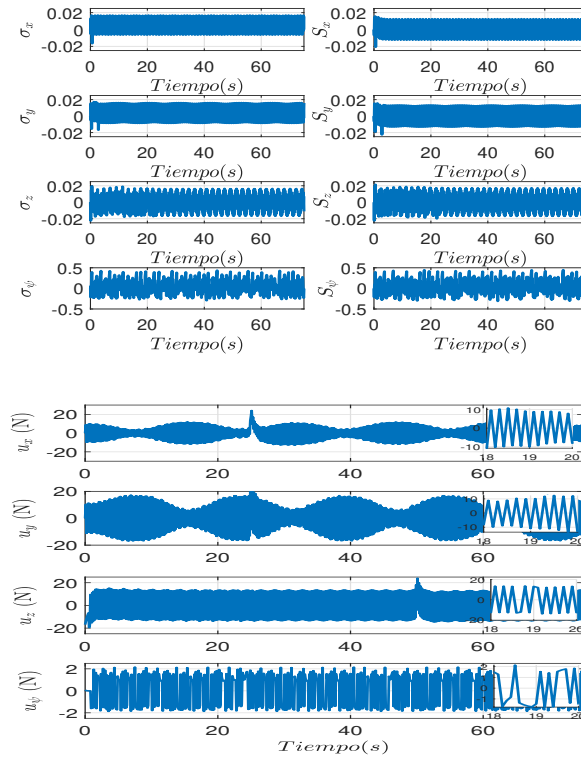


Figura 6.4: Las entradas de control ISMC tienen efecto de conmutación, en tanto que las superficies de deslizamiento y las variables auxiliares oscilan en una región muy cercana a cero..

6.1.1.2. Bakstepping integral con efecto adaptativo

Para comprobar el funcionamiento del controlador propuesto en la sección 3.3.1, se llevó a cabo un conjunto de simulaciones para abordar el problema de regulación y seguimiento de trayectoria, para ello se consideró el modelo vectorial del BlueROV 2 dado en (3.54) y los parámetros de la Tabla 5.2. Las perturbaciones para simular las corrientes y olas, son propuestas por: $\delta = [0.5 + 0.5 * c(0.6 * t - \pi/3); 0.3 + 0.4 * c(0.3 * t - \pi/6); 0.5 + 0.3 * s(0.5 * t - \pi/4); 0.2 + 0.2 * s(0.1 * t + \pi/8)]^T$ y las ganancias utilizadas para todos los escenarios de simulación son: $k_e = \text{diag}[10, 10, 10, 8]$; $k_v = \text{diag}[25, 25, 20, 15]$; $k_s = \text{diag}[2.5, 2.5, 2.0, 1.5]$; $k_w = \text{diag}[60, 100, 100, 15]$. Cabe mencionar que para determinar las ganancias de esta simulación, se consideran que las matrices son definidas positivas y la sintonización se realiza de forma empírica.

La primer prueba consiste en abordar el problema de regulación. Para esto el vehículo tiene las siguientes condiciones iniciales $\bar{\eta}(0) = [-1, 0, 0, 0]^T$, mientras que su posición y orientación deseada es $\bar{\eta}_d = [1, 2, -1.5, 0]^T$. En las Figuras 6.5-6.6, se puede visualizar el

comportamiento del vehículo en la prueba, observe que en todos los estados se alcanza la referencia y que el aporte de energía en las entradas de control es mínimo. Para comprobar la eficiencia del controlador se utiliza el criterio del error cuadrático medio (RMSE, por sus siglas en inglés), ver Tabla 6.3. Por otra parte, en la Figura 6.7 se muestra que la estimación de la perturbación converge a la simulada; lo que implica que la retroalimentación de la estimación en el controlador compense las perturbaciones introducidas.

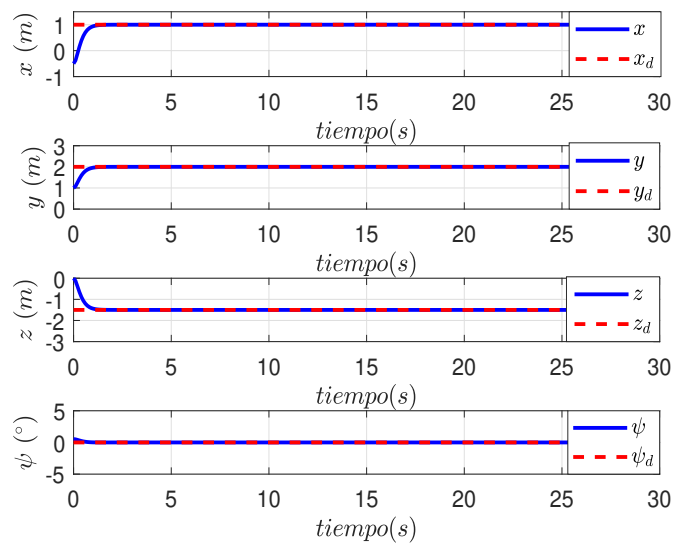


Figura 6.5: Evolución de la posición y guiñada, problema de regulación.

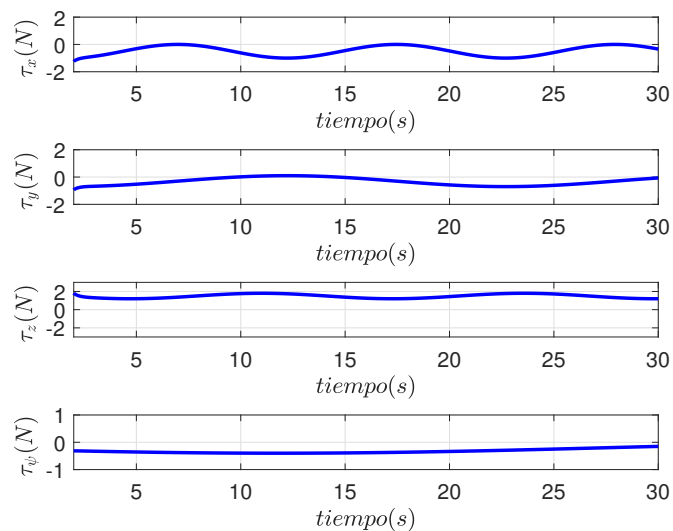


Figura 6.6: Entradas de control.

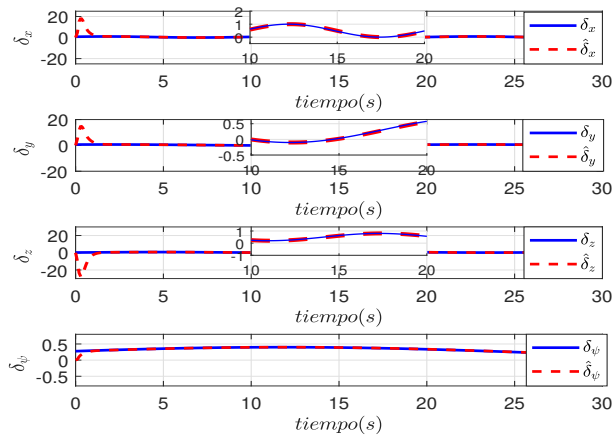


Figura 6.7: Estimación de la perturbación δ_i con $i : x, y, z, \psi$.

Una segunda prueba es realizada para abordar el seguimiento de trayectoria de una elipsoide mientras la profundidad sigue una referencia sinusoidal y el ángulo de guiñada se queda estabilizado, donde $\bar{\eta}_d = [\sin(0.6 * t); \cos(0.6 * t); \sin(0.05 * t) + 1; \pi/6]^T$ y las condiciones iniciales son: $\bar{\eta}(0) = [-0.5, -1, 0, 0]^T$. En la Tabla 6.3, se muestran que los valores calculados mediante RMSE, note que son valores muy cercanos a cero esto indica que el vehículo realiza el seguimiento en una región muy cercana la trayectoria de referencia como se ve en la Figura 6.8. En el caso de las entradas de control, podemos observar que su aporte energético se modificó muy poco con respecto a la primer prueba como se ve en la Figura 6.9. Mientras que, la Figura 6.10 muestra que el estimador de perturbaciones propuesto alcanza el valor de las perturbaciones simuladas rápidamente.

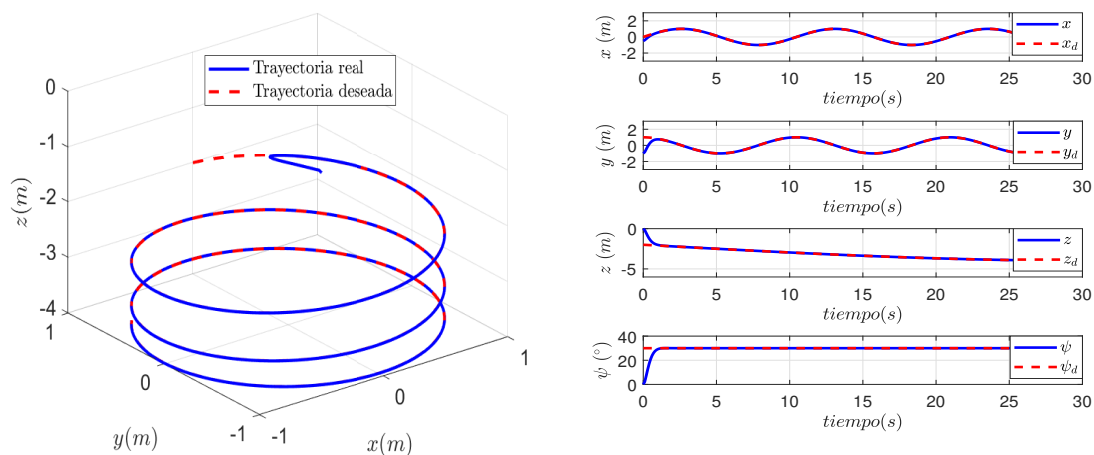


Figura 6.8: Seguimiento de una trayectoria elipsoidal.

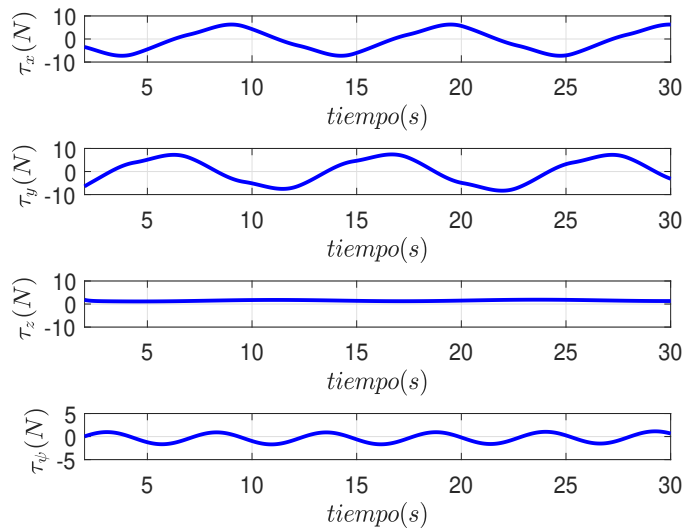


Figura 6.9: Entradas de control.

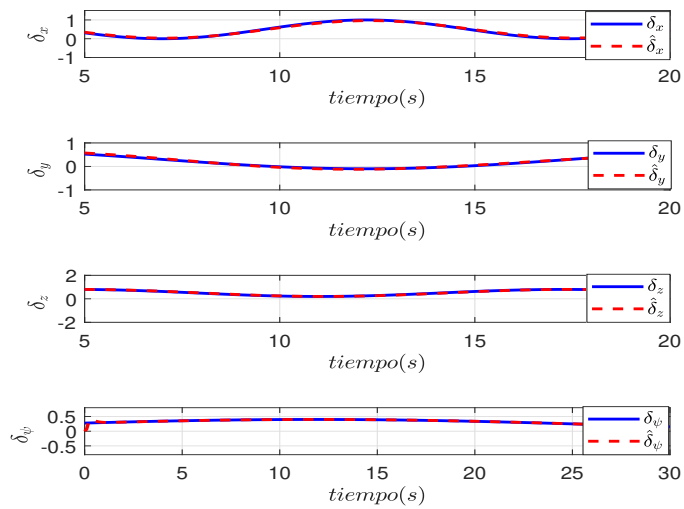


Figura 6.10: Estimación de perturbaciones, para apreciar de mejor forma la estimación solo se muestran los valores de un intervalo de tiempo de la prueba.

RMSE BIA		
	Prueba 1	Prueba 2
$x(m)$	0.0727	0.0493
$y(m)$	0.0558	0.0686
$z(m)$	0.0797	0.0724
$\psi(^{\circ})$	0.0035	0.0535

Tabla 6.3: Criterio de error cuadrático medio para BIA.

Para comprobar la generación de la trayectoria propuesta en la sección 4.2.1, se implementó una tercer prueba donde se puede observar el rendimiento y la robustez del sistema ante perturbaciones externas; las condiciones iniciales son $\bar{\eta}(0) = [-1, 7, 0, 1.6]^T$, mientras que los puntos de referencia son $p_1 = (2, 6)$, $p_2 = (8, 6)$, $p_3 = (8, 2)$, $p_4 = (2, 2)$ y $p_5 = (2, 5)$ con una profundidad deseada de $2m$. En la Figura 6.11, podemos observar la generación mediante el uso de puntos de referencia y el avance del vehículo a lo largo de toda la trayectoria.

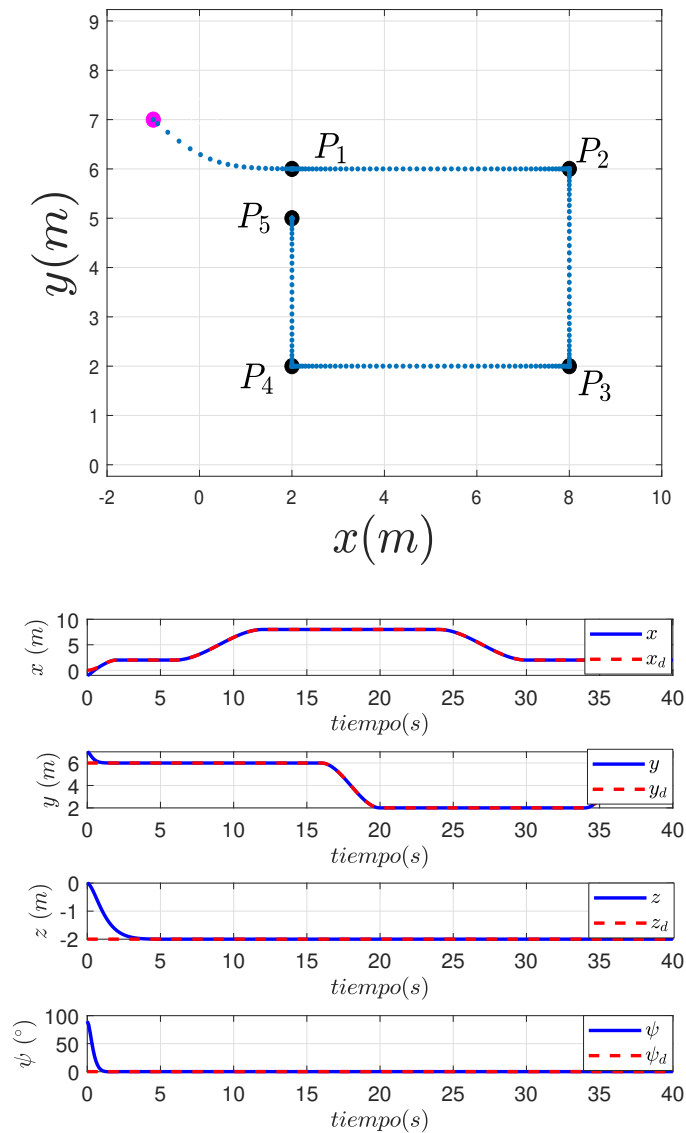


Figura 6.11: Comportamiento del controlador durante el seguimiento de puntos de referencia.

6.1.1.3. Comparativa del PD+G, BIA y SISMC

Para comparar el rendimiento de los controladores propuestos en el Capítulo 3, se realiza una prueba de seguimiento de trayectoria considerando perturbaciones de alta frecuencia. La trayectoria de referencia en esta prueba es la descrita en la sección 4.2.2, las perturbaciones a introducir están dadas por (6.2). Las ganancias de los controladores SISMC y BIA son las utilizadas en las simulaciones previamente descritas. En tanto que, las ganancias para el control PD+G son: $k_p = \text{diag}[30, 80, 6, 4]$ y $k_d = \text{diag}[30, 80, 6, 4]$.

En las figuras 6.12-6.14, se muestra el desempeño de los controladores propuestos al seguir una trayectoria mientras se aplican perturbaciones al sistema. Observe que tanto el BIA como el SISMC convergen a la trayectoria de referencia incluso ante las perturbaciones, sin embargo, el control BIA es el que presenta el menor error de seguimiento como se ven en la Figura 6.15 y el RMSE de la Tabla 6.4. Para el caso del PD+G, el sistema no puede hacer frente a los efectos dinámicos ocasionados por las fuerzas externas lo que ocasiona que no siga de buena manera la trayectoria y que presente el mayor error de seguimiento. Además, en la Figura 6.16, se muestra que las entradas de control generadas por SISMC presentan el efecto de conmutación lo que puede causar mayor gasto energético en comparación con los controles PD+G y BIA.

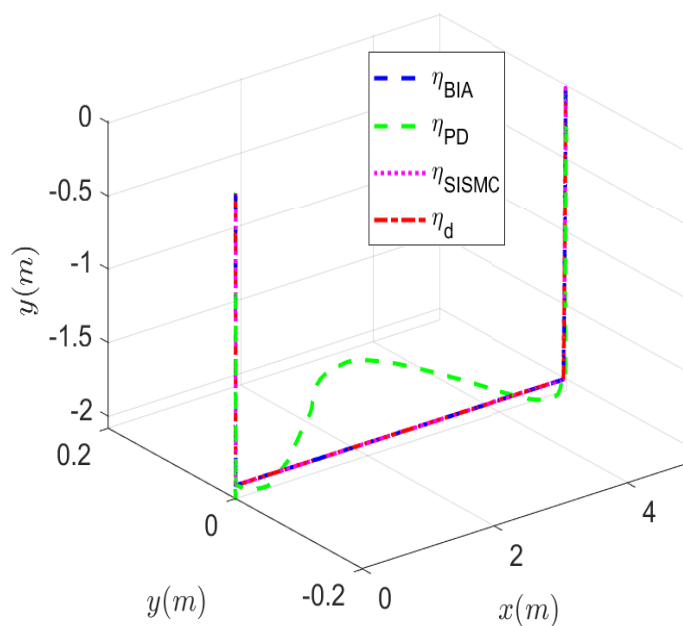


Figura 6.12: Seguimiento de trayectoria mediante puntos de referencia, resultados de simulación con perturbaciones externas.

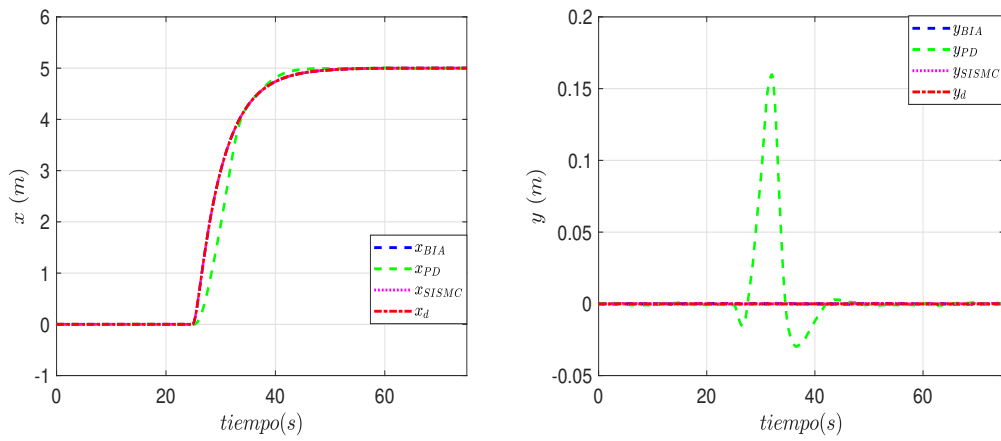


Figura 6.13: Comportamiento en posición (x, y) usando SISM, BIA y PD+G.

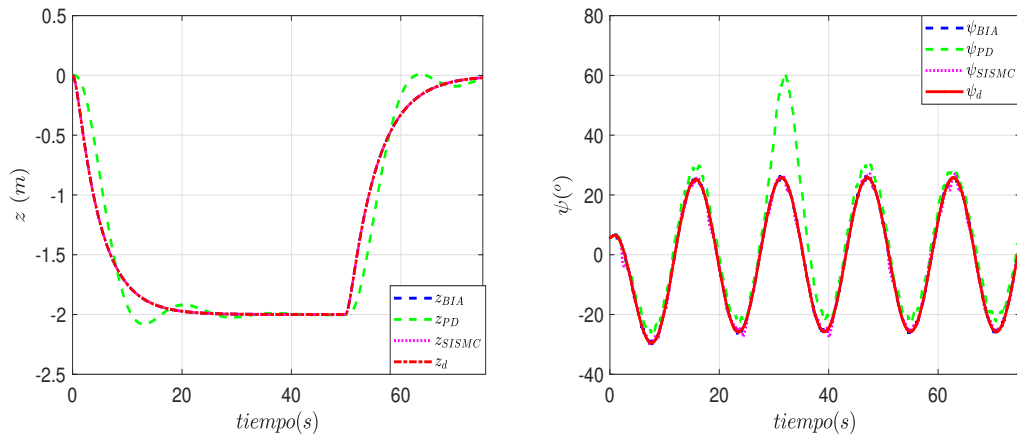


Figura 6.14: Comportamiento en profundidad (z) y guiñada (ψ) usando SISM, BIA y PD+G.

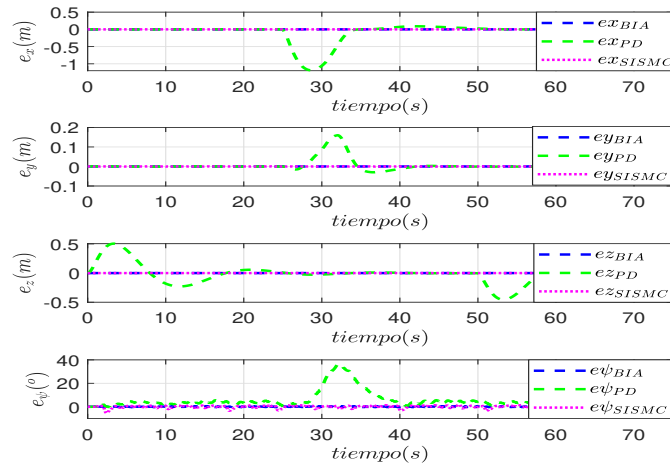


Figura 6.15: Errores de seguimiento a lo largo de la trayectoria.

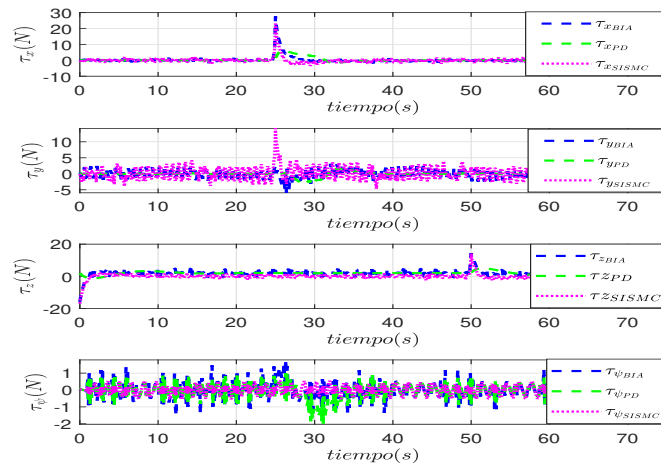


Figura 6.16: Entradas de control, note que el SISMC tiene efecto de conmutación lo que podría generar desgaste mecánico en los propulsores.

RMSE			
	PD+G	BIA	SISMC
$x(m)$	0.2747	0.00015	0.00016
$y(m)$	0.0358	0.00014	0.00021
$z(m)$	0.1697	0.00013	0.00016
$\psi(^{\circ})$	0.987	0.0083	0.0121

Tabla 6.4: Criterio de error cuadrático medio para prueba comparativa.

6.1.2. Sistema de guía

6.1.2.1. Detección y evasión de obstáculos

Detección de obstáculos

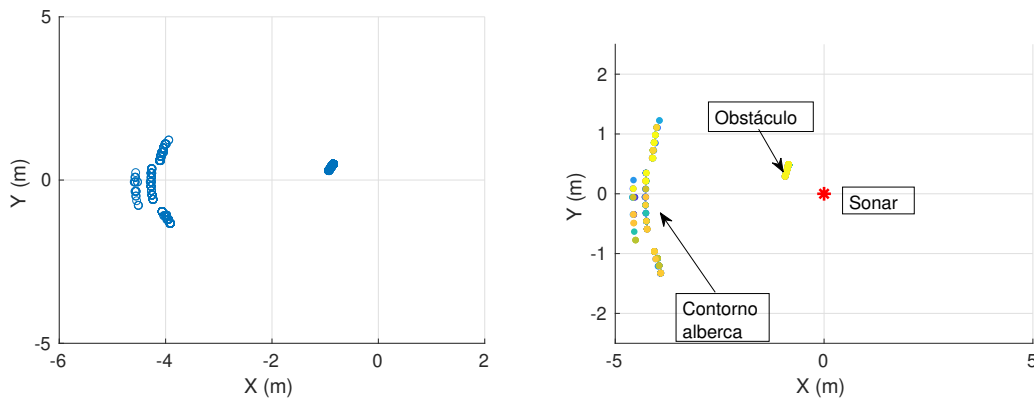


Figura 6.17: Diagramas de dispersión de los datos filtrados.

Un filtro mediante el umbral de intensidad es desarrollado en la sección 4.2.3.1. En la Figura 6.17, se puede observar el resultado del filtro implementado a la señal de la nube de puntos. Observe que se detecta el contorno de la alberca y un obstáculo a un metro, sin embargo, los datos son puntos dispersos a los que tenemos que aplicar un método de agrupamiento de datos para detectar el obstáculo. Por ello, el método de K-means para agrupar datos dispersos es implementado en ROS mediante un script de Python. En la Figura 6.18, se puede visualizar el resultado obtenido en Python y comparándolo con el obtenido mediante Clustering Toolbox de Matlab, se llegó a la conclusión que el resultado es similar.

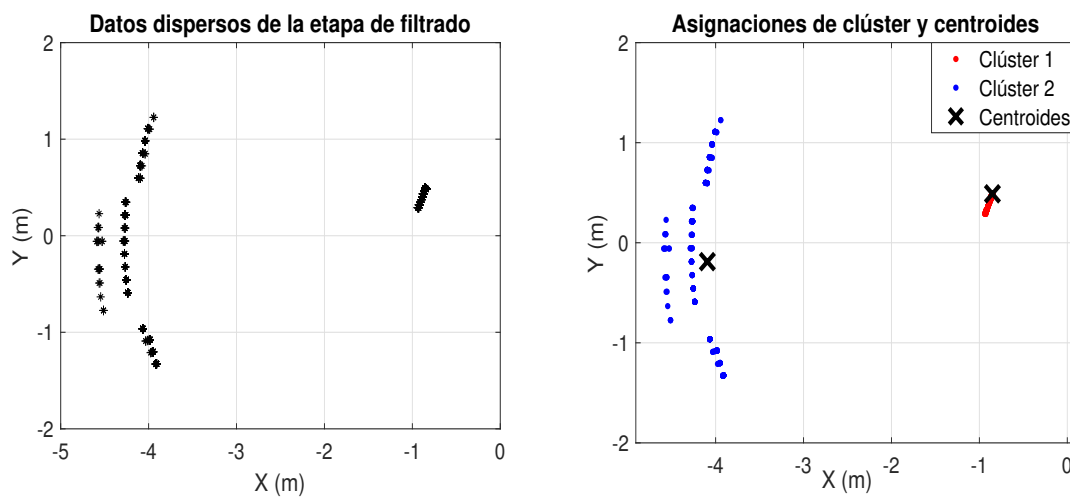


Figura 6.18: Agrupación de datos con su centroide utilizando K-means en Python.

Este algoritmo proporciona las coordenadas de los centroides, que podemos utilizar para obtener la distancia euclidiana con los puntos de cada grupo, con el objetivo de calcular la superficie esférica ocupada por el clúster. Para ello, se tiene que

$$d(i, j) = \sqrt{|(x_{i_1} - x_{j_1})|^2 + |(x_{i_2} - x_{j_2})|^2 + \dots + |(x_{i_p} - x_{j_p})|^2}, \quad (6.3)$$

Entonces, se agrega al algoritmo de K-means la ecuación (6.3) para el cálculo de la longitud máxima de los centroides a cada punto del clúster con el objetivo de obtener un radio y las cotas de la superficie que ocupa el obstáculo. En la Figura 6.19, podemos observar el resultado de la implementación del algoritmo. Note que se agrupan dos clúster y sus centroides; con las coordenadas obtenidas se calcula el radio de la superficie que ocupa el obstáculo en dos dimensiones. Para comprobar el resultado del algoritmo, se obtuvo la distancia máxima de los puntos con el centroide del clúster 1 y es igual a 0.18 m, ver Figura 6.20. Ahora sí se considera que el objeto utilizado para esta prueba es de un diámetro similar,

se puede concluir que la detección del objeto permitirá que utilicemos las coordenadas y los datos calculados en el método de evasión de obstáculos.

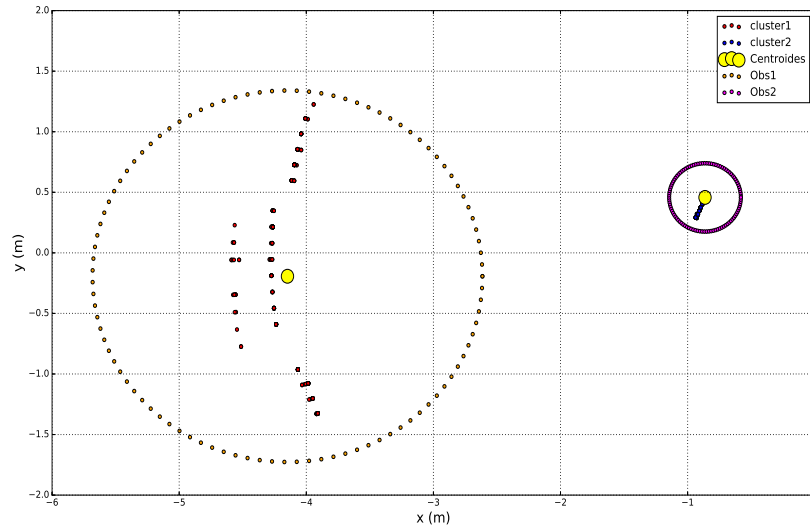


Figura 6.19: Técnica de agrupamiento, puntos rojos definen al clúster 1 mientras que los puntos azules son el clúster 2, los centroides están definidos en círculos amarillos, los puntos naranja y morado definen la ubicación del obstáculo en un entorno acuático.

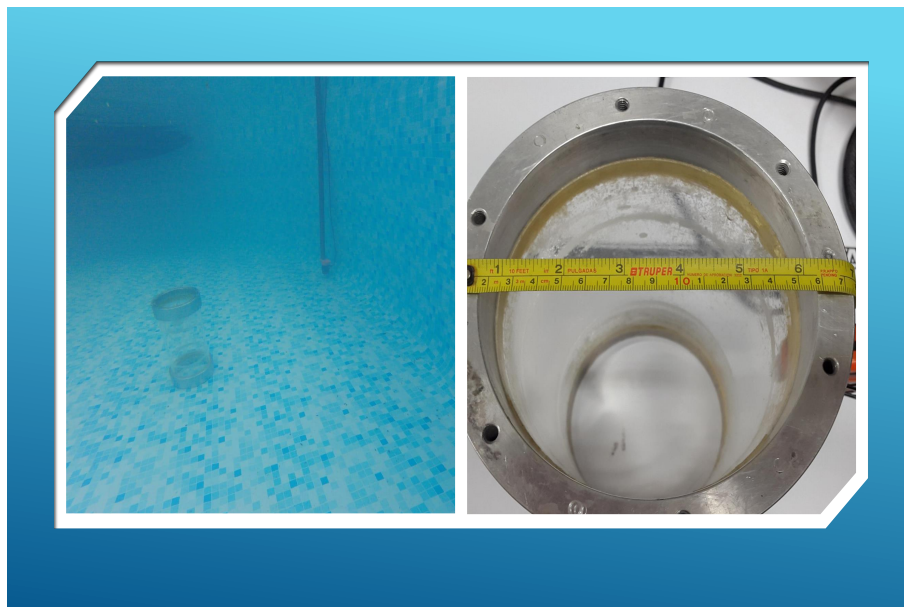


Figura 6.20: Posición del objeto durante la prueba, y medición de la longitud del objeto.

Técnicas de evasión de obstáculos

Simulación de evasión de obstáculos mediante el algoritmo de BUG.

En la sección 4.2.3.2, se describen los métodos BUG para evadir obstáculos. Para comprobar el algoritmo BUG1 se simuló en Matlab un entorno de prueba con dos obstáculos donde el vehículo inicia con condiciones iniciales $(0, 0)$ y tiene que llegar al punto objetivo P_0 $(4, 6)$. Para ello se considera que el vehículo tiene un sensor echosounder implementado y conocemos la localización del vehículo. En la Figura 6.21, se puede visualizar que la evasión se realiza con éxito, pero note que al encontrar un obstáculo circunnavega alrededor del objeto hasta que encuentra el punto de salida para generar una nueva ruta hacia el punto P_0 $(4, 6)$.

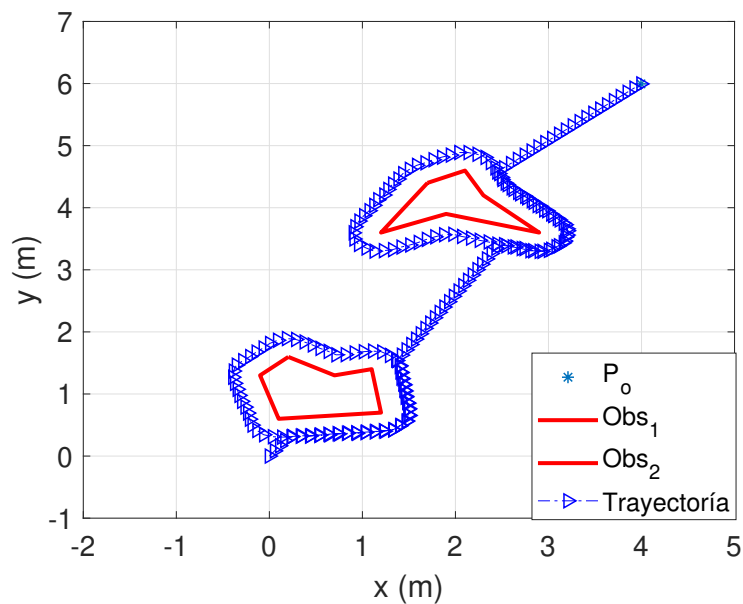


Figura 6.21: Simulación de algoritmo BUG1 para evasión de obstáculos, cuando encuentra un obstáculo lo rodea hasta ubicar la pendiente de la línea recta de la distancia más corta al objetivo.

BUG tangencial.

Para comprobar el funcionamiento del algoritmo se realizó una simulación en Matlab, donde se introducen las ecuaciones del algoritmo tangencial y el modelo del sonar propuesto en [98] por KUC que está definido por la ecuación:

$$h_R(t_s, z_c, a, \alpha) = \frac{2c_s \cos \alpha}{\pi a \sin \alpha} \sqrt{1 - \frac{c_s^2 (t_s - 2z_c/c_s)^2}{a^2 \sin^2 \alpha}}, \quad (6.4)$$

donde c_s denota la velocidad del sonido, a es el diámetro del elemento acústico (sonar), t_s denota el tiempo, z_c define la distancia ortogonal y α es el ángulo de la superficie con el cono acústico. Entonces, para la simulación se contempló un entorno acuático con un obstáculo el cuál es una embarcación con las dimensiones de un barco petrolero, además se asume que se tiene la localización del vehículo, que cuenta con un sonar acústico con lectura de 360° y un sensor de límite. En la Figura 6.22, se puede ver que el robot se dirige en línea recta hacia el objetivo hasta que detecta un obstáculo. Para evadirlo, el vehículo sigue los límites del obstáculo para moverse a lo largo de la circunferencia pero note que esto provoca grandes giros en las esquinas del obstáculo lo que causa que el vehículo se acerque al objeto, una vez que el sensor de límite no detecta obstáculo cercano se inicia el cálculo de las ecuaciones de planificación de movimiento para llegar al objetivo.

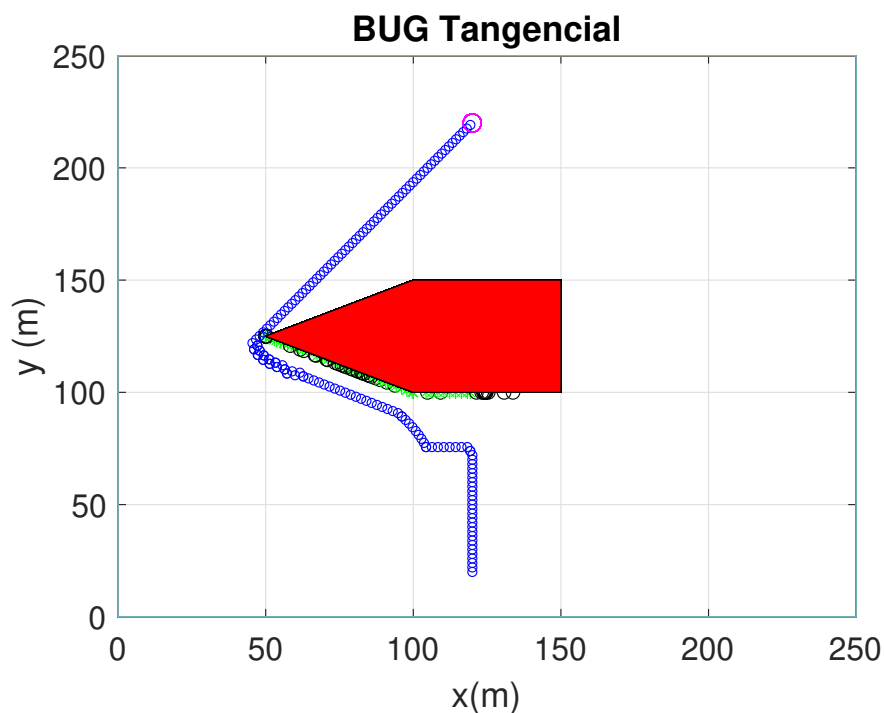


Figura 6.22: Simulación del algoritmo de BUG Tangencial en Matlab, los círculos azules definen la trayectoria del robot, los asteriscos verdes denotan la detección del obstáculo mediante el sensor acústico, la superficie roja muestra el obstáculo y el círculo magenta el punto objetivo.

Campo potencial tangencial.

Para comprobar el funcionamiento del método propuesto en la sección 4.2.3.2, se realizó un conjunto de simulaciones en Simulink de Matlab, para ello se consideró el modelo dinámico del BlueROV 2 y el uso del controlador PD+G propuesto en (3.3) para que el vehículo

se dirija a la referencia. La primer prueba considera un entorno con un obstáculo, donde las condiciones iniciales del robot son $(-1, -1)$ y el objetivo se ubica en el punto $(7, 6)$. La Figura 6.23, presenta la evolución de la trayectoria donde el obstáculo es representado con un punto rojo, la zona de repulsión en una elipsoide roja, la trayectoria real sin un método de evasión se muestra en magenta y la línea punteada azul es la trayectoria con el método de evasión de obstáculos. Note que, sin el algoritmo de campo potencial tangencial el vehículo colisiona con el objeto, en cambio con el método de evasión de obstáculos el vehículo evade el obstáculo y se dirige al punto deseado.

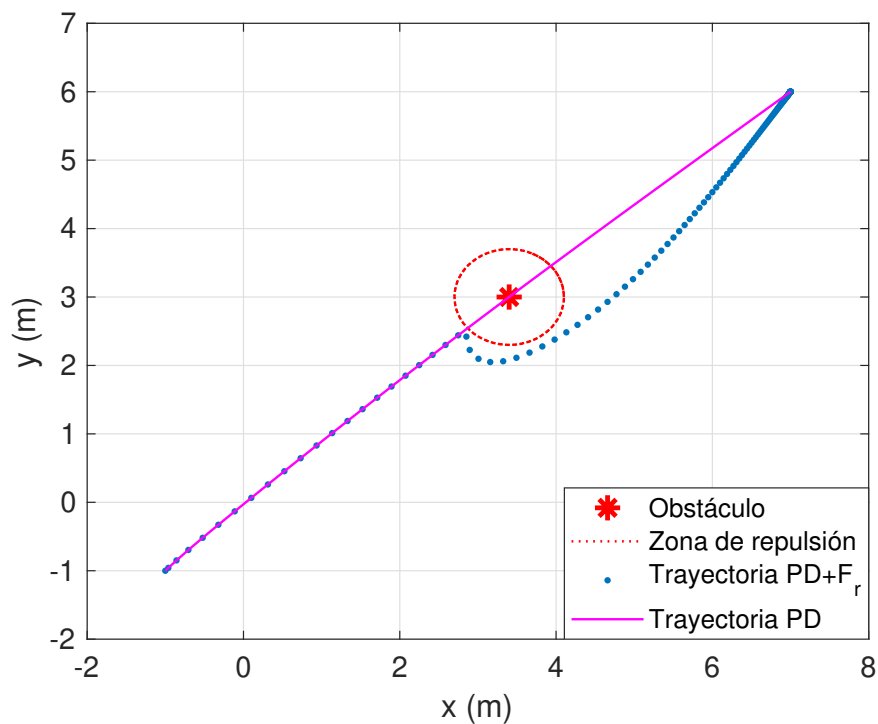


Figura 6.23: Simulación del algoritmo de evasión de obstáculos por campo potencial tangencial.

Para simular la evasión de múltiples obstáculos se utilizan las ecuaciones propuestas por (4.13) y (4.14). Se asumen dos escenarios, el primer escenario de pruebas consiste en un entorno con 3 obstáculos estáticos. En la Figura 6.24, se puede observar la evolución de la trayectoria en la simulación considerando que el robot submarino sigue en línea recta hasta que se encuentra cerca de la zona de repulsión del obstáculo, entonces la fuerza de repulsión actúa para que el vehículo evite el primer obstáculo, una vez evadido sigue con su camino hasta evadir el segundo obstáculo y así continúa hasta llegar al punto de referencia.

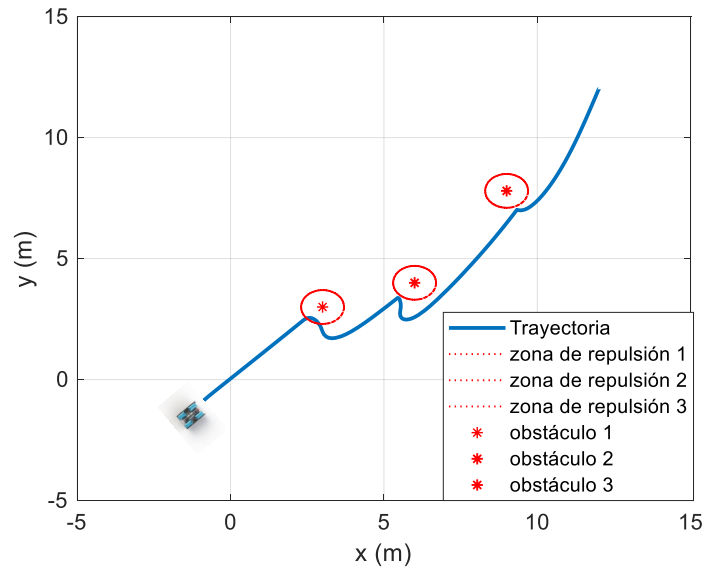


Figura 6.24: Evasión de múltiples obstáculos estáticos.

En el siguiente escenario, el vehículo debe evadir múltiples objetos mientras sigue la trayectoria generada mediante el algoritmo propuesto en la sección 4.2.1. Las condiciones iniciales del vehículo son $(1, 0)$ y los obstáculos se encuentran en $(4, 10)$, $(10, 4)$ y $(4, -1)$. La Figura 6.25, muestra que se llega al objetivo evadiendo múltiples objetos mientras el robot realiza el seguimiento de camino mediante puntos de referencia.

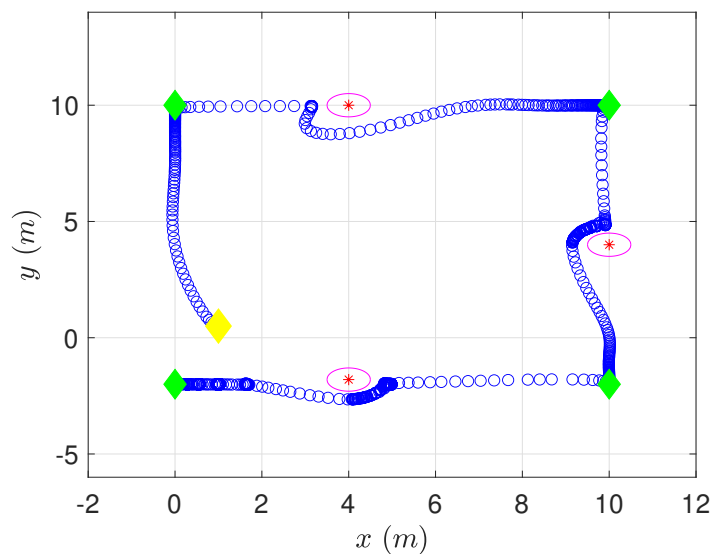


Figura 6.25: Seguimiento de camino mediante puntos de referencia con evasión de múltiples obstáculos, donde los rombos verdes son los puntos de referencia a lo largo de la trayectoria y el rombo amarillo es la ubicación inicial del vehículo.

Observe que el círculo de azul muestra la evolución de la trayectoria considerando el algoritmo de evasión, note que la fuerza de repulsión evita que el robot colisione con los obstáculos representados por los círculos magenta.

Análisis de resultados de las técnicas de evasión de obstáculos

En conclusión, los resultados de las simulaciones demuestran el algoritmo de BUG1 es lento porque rodea al objeto completamente hasta encontrar el punto de partida hacia el objetivo. Mientras que, el algoritmo de BUG tangencial se basa en las mediciones actuales de los sensores; para activar la planificación de movimiento o el seguimiento de límite, lo que implica que el vehículo tenga contacto al girar por la cercanía al objeto circundante hasta tocar su sensor de límite. En el caso, del método de campo potencial tangencial evita los objetos durante la trayectoria sin acercarse al objeto demasiado y no muestra el problema de mínimo local, por lo que, se elige el método para la evasión de obstáculos.

6.2. Experimentos en tiempo real

En esta sección, se exponen los resultados experimentales de las pruebas en tiempo real de los algoritmos previamente explicados en los Capítulos 3, 4 y se utilizan los prototipos descritos en el Capítulo 5.

6.2.1. Sistema de navegación

6.2.1.1. Sistema de posicionamiento acústico

Para comprobar el funcionamiento en tiempo real del sistema de posicionamiento acústico GPS Water Linked, se utilizó el script de Python dado por la API del fabricante para hacer un nodo en ROS que publica la posición acústica del vehículo. El escenario experimental es realizado en una alberca de dimensiones $20 \times 20 \times 2$ m con el BlueROV 2, donde los receptores del GPS están montados en una estructura y se utiliza el controlador PD+G dado por (3.3). La primer prueba consiste en regular la posición en distintos puntos hasta formar un rectángulo 6.26, observe en los recuadros amarillos que la estimación de la posición del submarino tiene una variación de hasta de 1.5 m, debido a la reflectancia del entorno. La solución propuesta para mejorar la medición obtenida con el Water Linked es implementar

un filtro pasa bajas de segundo orden, para ello utilizamos el software Micromodeler como se muestra en la Figura 6.27.

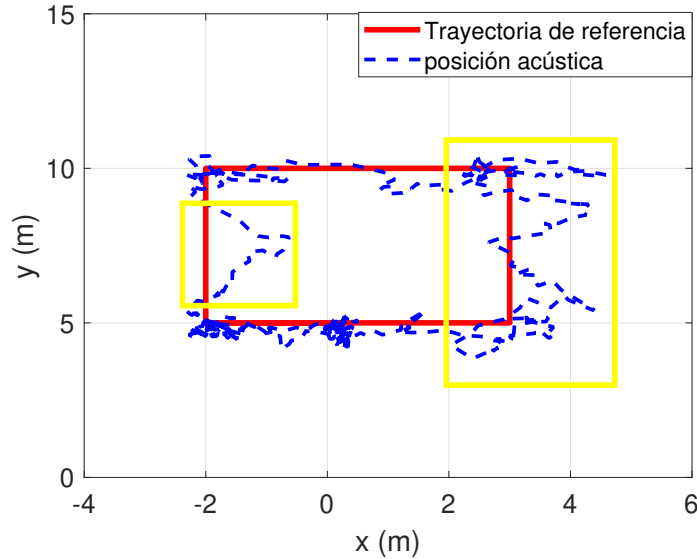


Figura 6.26: Pruebas experimental, la línea azul representa la localización acústica y la línea roja la posición deseada.

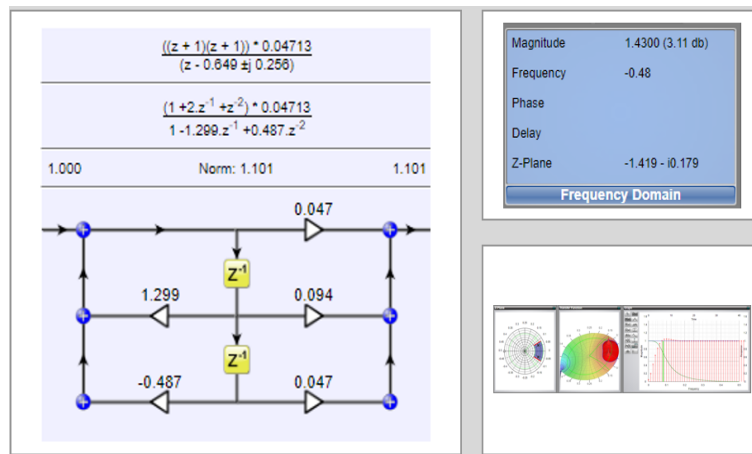


Figura 6.27: Diseño del filtro pasa bajas de segundo orden mediante herramienta computacional.

Para comprobar el funcionamiento del filtro diseñado, se realizó una prueba donde el Blue-ROV 2 sigue una trayectoria elíptica dada por: $x_d = 4 \cdot \cos(2\pi t/25)$ y $y_d = 2.5 \cdot \sin(2\pi t/25)$. La Figura 6.28 muestra que no hay variación en la medición acústica en comparación con los resultados mostrados en la Figura 6.26; esto permite que el vehículo pueda alcanzar la referencia. Entonces, podemos utilizar el sistema de localización acústico en misiones de entornos acuáticos para localizar a un vehículo submarino hasta en un rango de 100 m como lo señala el fabricante en la hoja de datos del Water Linked.

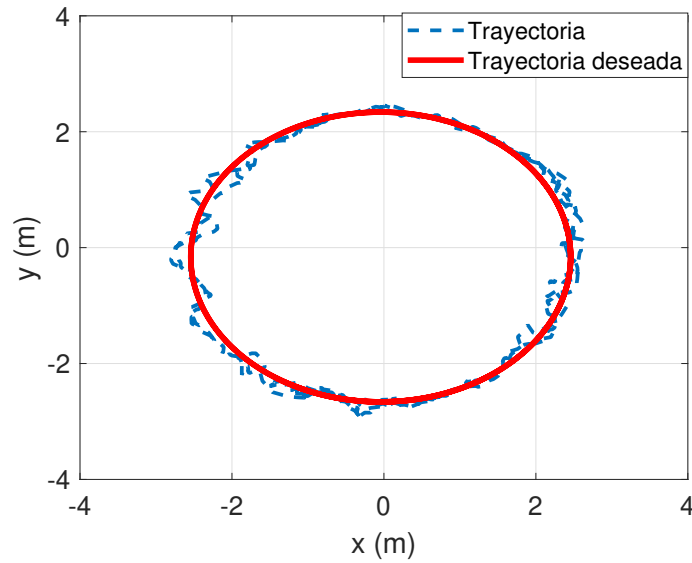


Figura 6.28: Seguimiento de trayectoria utilizando la localización acústica del sistema de posicionamiento Water Linked.

Odometría visual PTAM+EKF

Para implementar el algoritmo propuesto en la sección 4.1.3, se llevaron a cabo distintos análisis y pruebas experimentales previas para comprobar si el retardo en el cable del Blue-ROV 2 es significativo, para ello se realizó un script en Python que nos ayudó a sincronizar los relojes de la estación en tierra y el sistema embebido (Raspberry) para obtener el retardo en la comunicación mediante el cable de 100 m, los resultados obtenidos en esta prueba fueron de 54 milisegundos. Para compensar el retraso se disminuyó la resolución y la tasa de cuadros del vídeo a 640×480 a 15 fps, esta configuración tiene un retardo de 27 milisegundos y ocupa un ancho de banda de 1.179 Mb por cada segundo de vídeo. Considerando que se utilizan módems especiales para atenuar la latencia en la comunicación y mantener un ancho de banda de 80 Mb/s, muy por encima del tamaño del vídeo transmitido. En conclusión, dado que la dinámica del vehículo es lenta bajo el agua y los datos transmitidos están por debajo del ancho de banda máximo, consideramos que el retraso no es significativo.

Una vez hecho este análisis, para verificar el funcionamiento del algoritmo propuesto se presentan resultados de experimentos en tiempo real, donde el objetivo principal es estimar la posición mediante PTAM+EKF de un vehículo submarino y el análisis de las limitaciones del algoritmo. Las pruebas se realizaron en una alberca olímpica cuyas dimensiones son $25 \times 50 \times 1.6$ m, se implementó el algoritmo de control dado en (3.3), las ganancias de

control se sintonizaron de manera empírica y son definidas por: $k_{p_x} = 130$, $k_{d_x} = 20$, $k_{p_y} = 130$, $k_{d_y} = 15$, $k_{p_z} = 400$, $k_{d_z} = 80$, $k_{p_\psi} = 4$ y $k_{d_\psi} = 30$. Por otra parte, para la fusión de datos la matriz de covarianza para el paso de actualización EKF es proporcionada por el algoritmo PTAM, mientras que para el paso de predicción las variaciones de ruido para los acelerómetros son $\sigma_a^2 = 0.25 \frac{m^2}{s^3}$, para los giroscopios $\sigma_\omega^2 = 0.013 \frac{rad^2}{s}$ y su sesgo respectivamente es $\sigma_{b_a}^2 = 0.025 \frac{m^2}{s^5}$ y $\sigma_{b_\omega}^2 = 0.0013 \frac{rad^2}{s^3}$ respectivamente. Para más detalles acerca de la implementación del EKF ver [99, 100].

La primer prueba consiste en estabilizar la orientación y la profundidad en una referencia constante, para comprobar la obtención de puntos característicos de los datos capturados por la cámara del vehículo y para comprobar la estimación de la posición en el mapa local hecho por el algoritmo de PTAM sin el EKF, ver Figura 6.29. El siguiente experimento, consiste en verificar el resultado de la fusión de datos realizada con el EKF, para ello se estabiliza la orientación y profundidad en una referencia constante dada por $\psi_d = 110^\circ$ y $z_d = -0.35$ m. Una vez que se estabiliza el vehículo, se realiza un seguimiento de trayectoria en (x, y) donde la trayectoria es dada como $x_d = 0.7 * \cos(2\pi t/25)$ y $y_d = 0.5 * \sin(2\pi t/25)$, véase Figura 6.30. Note que la trayectoria de referencia es alcanzada en posición (x, y) , mientras que la profundidad y orientación permanecen convergen al valor de referencia.

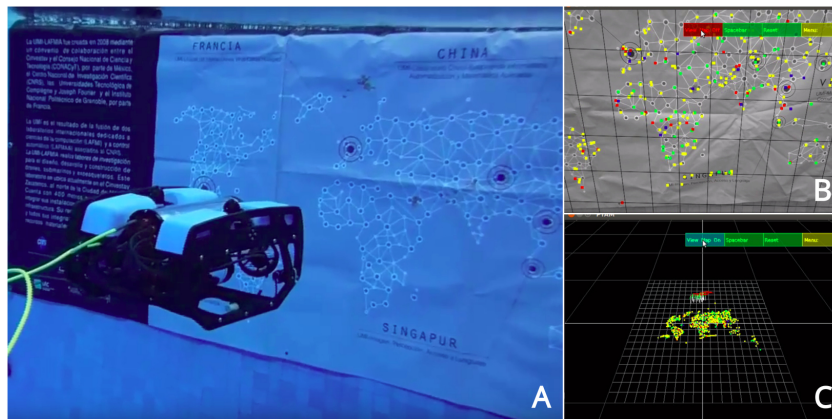


Figura 6.29: A) BlueROV 2 utilizando localización basada en visión. B) Se utiliza una cámara frontal en el vehículo para obtener los puntos característicos para la localización. C) Mapa y localización generado por el algoritmo PTAM.

Como se puede observar en los experimentos previos el algoritmo localiza al vehículo y permite que la trayectoria sea alcanzada con un error de seguimiento significativo, véase Tabla 6.5. Para analizar las limitaciones de PTAM+EKF se realizaron diversas pruebas experimentales para ver las limitaciones que tiene el algoritmo ante diferentes problemáticas como la

turbidez, el cambio de iluminación, las características no estructuradas desordenadas, las características en movimiento como los peces, etc. Para ello, se realizó una prueba experimental en la cuál se perturbó el vehículo submarino de diferentes formas, la primer perturbación es en orientación (ϕ, θ, ψ), después se obstruye la cámara por unos instantes para verificar que el vehículo puede recuperar el mapa, para continuar se pasa un objeto móvil frente al mapa y se perturbó jalando el cable del vehículo. Ver Figuras 6.31 y 6.32.

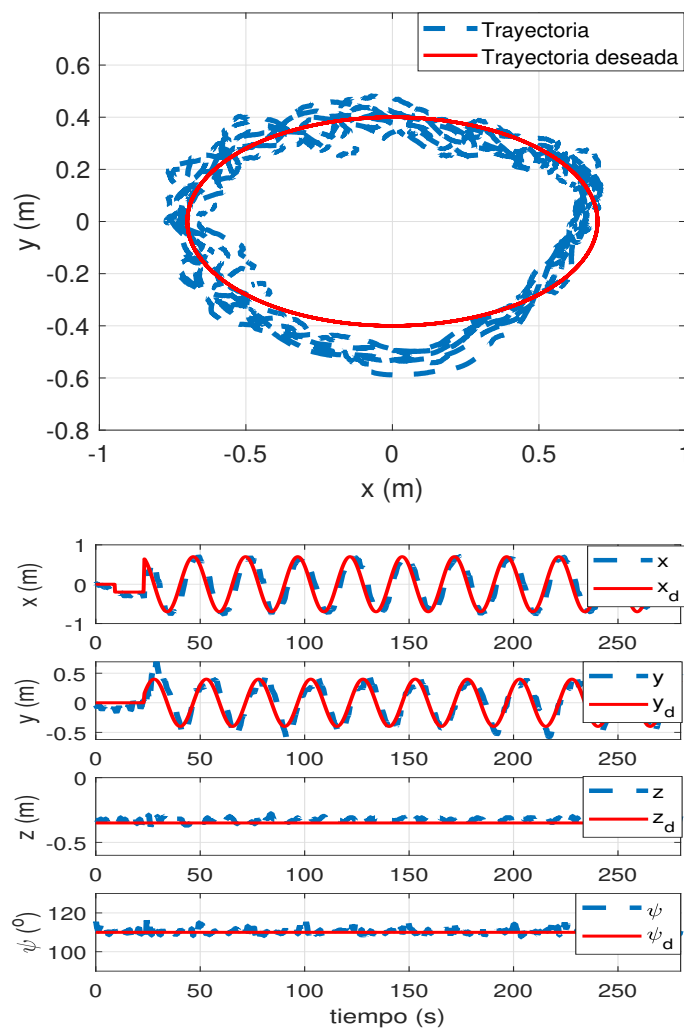


Figura 6.30: Seguimiento de trayectoria autónomo, regulación de profundidad y orientación.

RMSE	
	PD+G
$x(m)$	0.2235
$y(m)$	0.1032
$z(m)$	0.0903
$\psi(^{\circ})$	1.579

Tabla 6.5: Criterio de error cuadrático medio, PTAM.

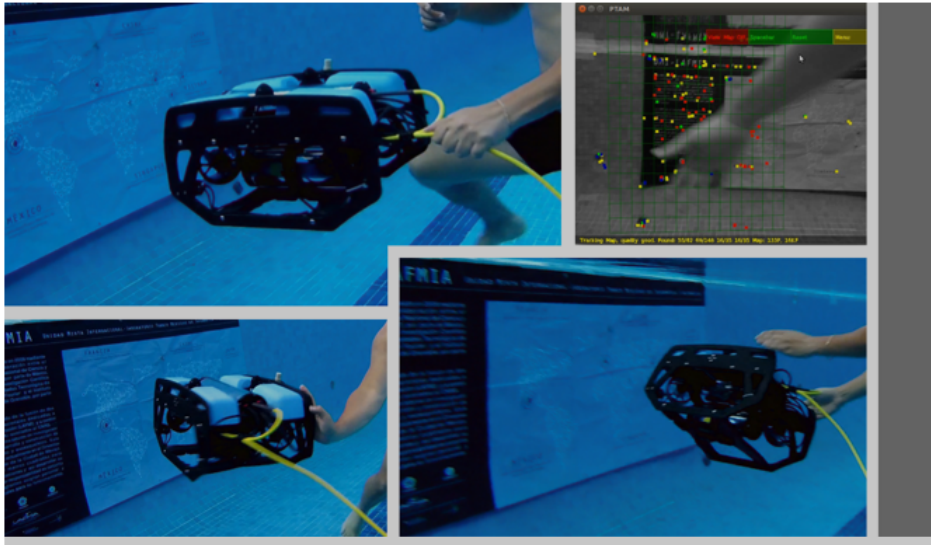


Figura 6.31: Pruebas para comprobar las limitaciones del algoritmo de visión.

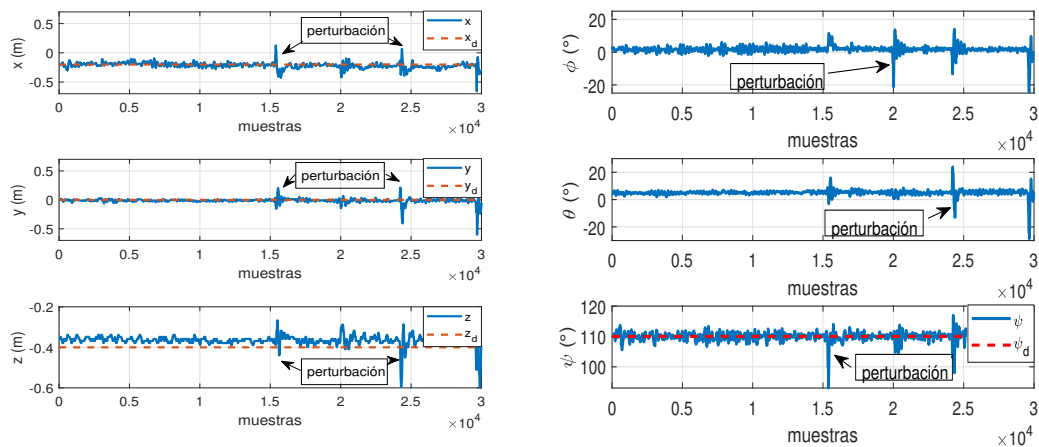


Figura 6.32: Prueba experimental con distintas perturbaciones externas.

En el experimento realizado se demostró que ante diferentes perturbaciones el vehículo permanece estable, esto debido a la estimación del EKF cuando la estimación del PTAM falla debido a la pérdida de puntos de interés. Para finalizar, se realizó un experimento para comparar las mediciones obtenidas por el PTAM y el EKF. La prueba consiste en estabilizar la orientación y la profundidad en una referencia constante $\psi_d = 110^\circ$, $z_d = -35\text{cm}$, mientras el vehículo sigue una trayectoria de referencia dada por: $x_d = 0.7 * \cos(2\pi t/25)$ y $y_d = 0.5 * \sin(2\pi t/25)$. Además, se perturba varias veces al vehículo durante la prueba para probar aún más la robustez de los algoritmos propuestos, en los segundos 310, 400 y 475.

En la Figura 6.33, se muestra la posición (x, y, z) del vehículo durante el seguimiento de la trayectoria, según lo estimado por PTAM (puntos de color rosa), EKF (guiones azules) contra

la trayectoria deseada (línea roja continua). Podemos apreciar que utilizando la estimación del EKF la localización no se pierde incluso ante las perturbaciones externas, además de que el control PD+G implementado logra que se alcance la posición y orientación de referencia. La orientación del UUV se muestra en la Figura 6.34, a través de ésta se puede apreciar la propiedad de estabilidad mecánica del vehículo en ángulos ϕ y θ , obtenidos a través de una selección adecuada del centro de flotabilidad.

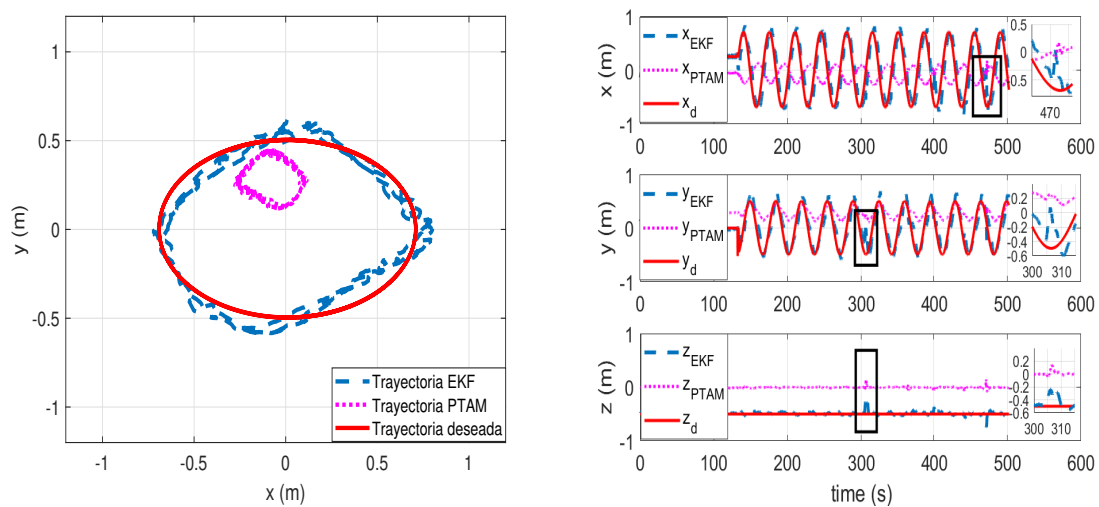


Figura 6.33: Seguimiento de trayectoria mediante odometría visual, comparación de estimación por PTAM y EKF.

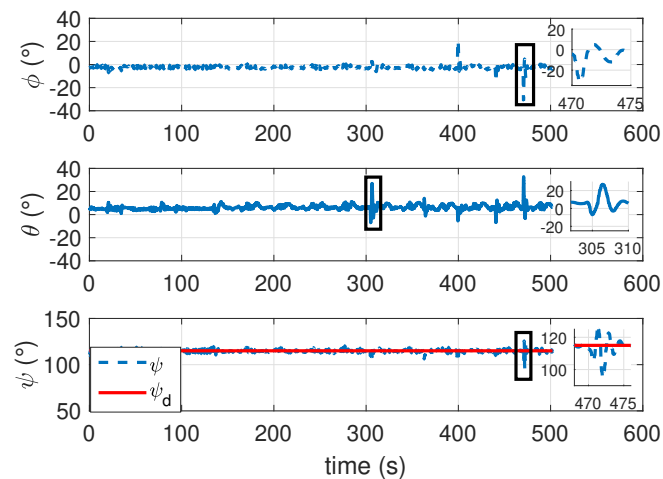


Figura 6.34: Seguimiento de trayectorias utilizando localización basada en visión: orientación.

El experimento realizado se puede ver en el siguiente vídeo: <https://www.youtube.com/watch?v=v0Uj-IuztIs>.

6.2.2. Sistema de control

Para las pruebas experimentales se utilizará el LIRMIA III y el BlueROV 2. Para implementar algoritmos de control, se tiene que seguir las indicaciones de los pseudocódigos del Apéndice A.

6.2.2.1. Control de modos deslizante integral de segundo orden

Para comprobar experimentalmente la ley de control propuesta en la sección 3.2, se realizaron pruebas experimentales con el BlueROV 2 para comprobar la eficiencia del controlador en profundidad, guiñada y en el movimiento de traslación en x . Para las pruebas λ_{1_i} , λ_{2_i} y la ganancia k_{3_i} son propuestas considerando el Teorema 3.2.1, entonces se eligen con los valores siguientes:

$$\begin{aligned} \lambda_{1_x} &= 1.8; & \lambda_{2_x} &= 2.3 \\ \lambda_{1_z} &= 1.5; & \lambda_{2_z} &= 2.0 \\ \lambda_{1_\psi} &= 2.2; & \lambda_{2_\psi} &= 3.0 \\ k_{3_x} = k_{3_z} &= 4.0; & k_{3_\psi} &= 2 \end{aligned}$$

La primer prueba regula el ángulo de guiñada (ψ) del vehículo en diferentes valores constantes. La Figura 6.35, muestra que el vehículo alcanza el ángulo de guiñada en distintos valores sin presentar oscilaciones y con un error de seguimiento muy cercano a cero, el cuál se corrobora mediante el RMSE que es de 1.872° . Por otra parte, en la Figura 6.36 se visualiza que la superficie de deslizamiento converge a una región cercana a cero y que hay atenuación del efecto de conmutación en la entrada de control.

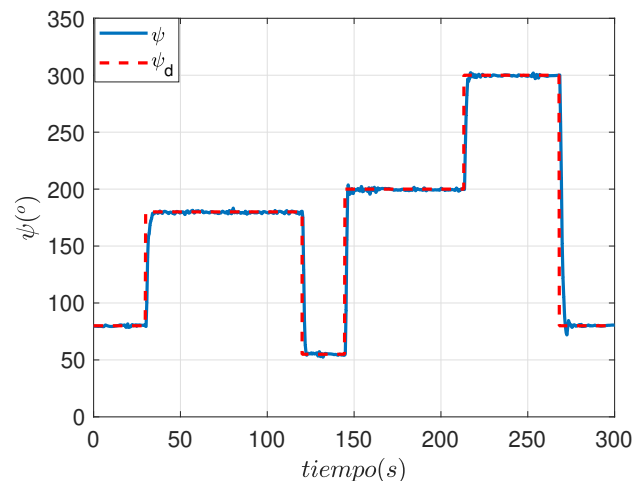


Figura 6.35: Regulación de guiñada (ψ) con SISMC.

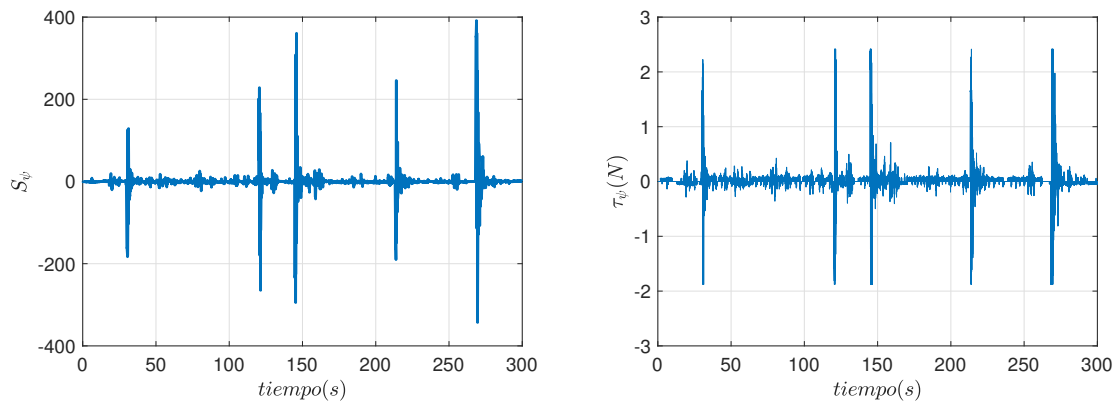


Figura 6.36: Evolución de la superficie deslizante y la entrada de control.

El siguiente escenario, propone que el vehículo se estabiliza en profundidad a -0.5 m y 60° en guiñada, a su vez realiza la regulación a un punto del movimiento traslacional en el eje x primero a 1 m y después a -1 m. En la Figura 6.37 podemos observar que la orientación en ψ y la profundidad permanecen estables en el valor de referencia, mientras que la posición en x converge al valor deseado. Note que la entrada de control no manifiesta el efecto de conmutación.

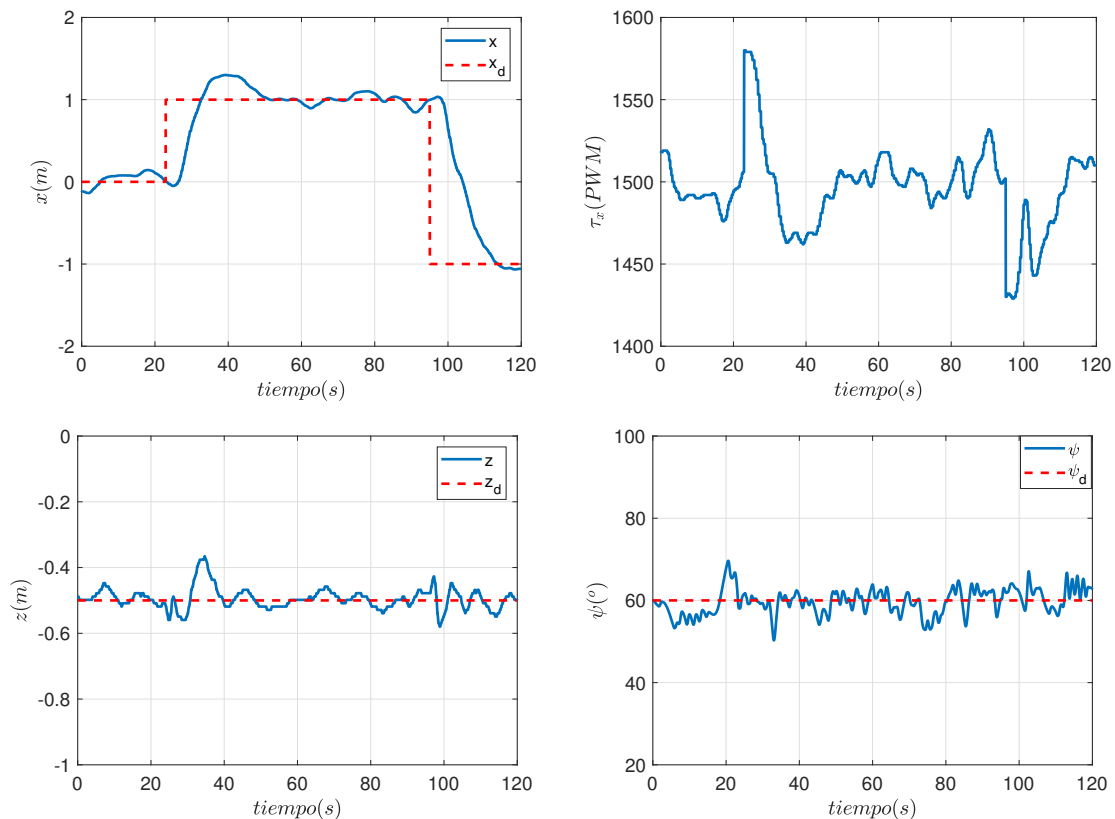


Figura 6.37: Regulación de profundidad, guiñada y del movimiento traslacional x .

Es importante mencionar que los experimentos se realizan mediante el uso del sistema acústico de localización Water Linked, su implementación tanto del nodo de localización como el nodo de control se realiza en ROS con una tasa de muestreo de 10 Hz. Para la elección de la acción de control (regulación o seguimiento), el ajuste de los parámetros y las ganancias en tiempo real; se diseñó una GUI mediante la herramienta de reconfiguración dinámica.

6.2.2.2. Bakstepping integral con efecto adaptativo caso escalar

El diseño del controlador en el caso escalar y el análisis de estabilidad a lazo cerrado se presentan en [48] en base a la teoría de Lyapunov y a la técnica de bakstepping. Para la prueba experimental, se considera que el vehículo es estable mecánicamente en cabeceo ϕ , entonces solo se estabiliza la orientación (θ, ψ) y la profundidad del LIRMIA III mediante las siguientes entradas de control:

$$\begin{aligned}\tau_\psi &= I_{\psi_1}(\dot{\rho}_\psi - e_\psi - k_{\dot{e}_\psi}\dot{e}_\psi) - I_{\psi_2}\dot{\psi}\dot{\theta} - \hat{\delta}_\psi \\ \tau_\theta &= \bar{I}_q(\dot{\rho}_\theta - e_\theta - k_{\dot{e}_\theta}\dot{e}_\theta) - \bar{k}_{uv}uv - \hat{\delta}_\theta \\ \tau_z &= \bar{I}_z(\dot{\rho}_z - e_z - k_{\dot{e}_z}\dot{e}_z) - \hat{\delta}_z.\end{aligned}\tag{6.5}$$

con $e_i = i - i_d$, $\rho_i = \dot{i}_d - k_{\varsigma_i}\varsigma_i - k_i e_i$, $\hat{\delta}_i = \frac{1}{I_{i_1}}k_{\delta_i}\dot{e}_i$ y $\varsigma_i = e_i$ con $i : z, \theta, \psi$. Donde $k_{\delta_i}, k_{\varsigma_i}, k_{\dot{e}_i} > 0$; $\bar{I}_z, \bar{I}_q, \bar{I}_r = 0.018$; $I_{\psi_1} = \bar{I}_r \cos\theta$ e $I_{\psi_2} = \bar{I}_r \sin\theta$. Las ganancias seleccionadas para la guiñada son: $k_{\varsigma_\psi} = 9 \times 10^{-4}$, $k_{\psi} = 14.11$, $k_{\dot{e}_\psi} = 12.6$, $k_{\delta_\psi} = 1.5 \times 10^{-5}$, para pitch: $k_{\varsigma_\theta} = 9.89 \times 10^{-5}$, $k_{\theta} = 14.15$, $k_{\dot{e}_\theta} = 11.75$, $k_{\delta_\theta} = 1.5 \times 10^{-4}$ y para la profundidad: $k_{\varsigma_z} = 9.9 \times 10^{-4}$, $k_z = 2$, $k_{\dot{e}_z} = 2.5$, $k_{\delta_z} = 1.1 \times 10^{-4}$.

El experimento se llevó a cabo en una piscina, en la que se generó una corriente transversal con la ayuda de una bomba hidráulica para producir una perturbación externa. La prueba consiste en estabilizar el ángulo de cabeceo $\theta_d = 4^\circ$ y la profundidad de nuestro vehículo a 1 m, mientras que se sigue una referencia sinusoidal $\psi_d = (50 * \sin(0.1 * t) + 100)^\circ$ en el ángulo de guiñada. La Figura 6.2.2.2 muestra el rendimiento del controlador en tiempo real, note que converge al valor deseado aún cuando se está perturbando el sistema externamente. Y mediante el RMSE de la Tabla 6.6 podemos decir que el error de seguimiento es pequeño considerando que se utiliza un sensor inercial de bajo costo en el prototipo. Por otra parte, observe en las entradas de control que el aporte del PWM es mínimo considerando que su límite es 254.

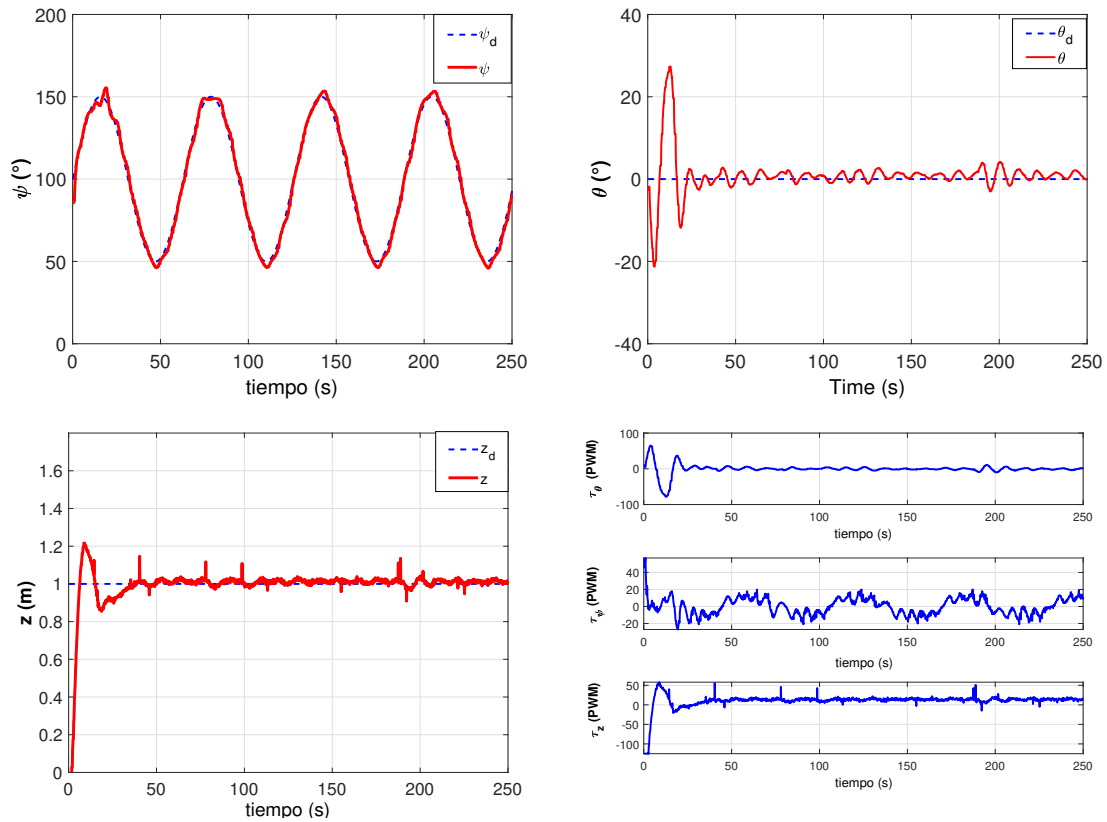


Figura 6.38: Respuesta del control en θ , ψ , z y evolución de las entradas de control.

RMSE	
	BIA
$z(m)$	0.083
$\theta(^{\circ})$	5.84
$\psi(^{\circ})$	4.579

Tabla 6.6: Criterio de error cuadrático medio, BIA implementado en LIRMIA III.

6.2.2.3. Prueba experimental comparativa de PD+G, BIA, SISMC.

Un conjunto de pruebas experimentales se llevaron a cabo para comparar la eficiencia de los controladores al realizar el seguimiento de trayectoria en profundidad y en guiñada. El escenario experimental es una alberca de 5 m de diámetro y una profundidad máxima de 0.7 m. El BlueROV 2 se utiliza para esta prueba, se propone que siga una trayectoria sinusoidal en profundidad y en guiñada dada por las ecuaciones siguientes:

$$z_d = (.15 * \sin(0.1 * t) + .30)m$$

$$\psi_d = (90 * \sin(0.1 * t) + 180)^{\circ}$$

En las Figuras 6.39, 6.41 y 6.43, se puede observar el seguimiento de los perfiles de profundidad y guiñada, observe que el BIA y SISMC convergen a la trayectoria, sin embargo, el PD+G no alcanza a converger al valor deseado. Para analizar que controlador es más eficiente se obtiene el RMSE de cada controlador, véase Tabla 6.7. Analizando los resultados la técnica por bakstepping tiene el error cuadrático medio más pequeño en profundidad, lo que implica que el error de seguimiento es menor que el de los otros controladores. No obstante, el SISMC presenta el RMSE más pequeño en el ángulo de guiñada.

Ahora, las Figuras 6.40, 6.42 y 6.44 muestran las entradas de control; observe que la magnitud de la fuerza dada por los controladores es muy pequeña, sin embargo, la menor magnitud es dada por el SISMC. Pero presenta el efecto de conmutación a alta frecuencia lo que podría generar mayor gasto energético.

Por lo tanto, para misiones autónomas en entornos no controlados se elige el uso del BIA porque puede compensar perturbaciones del entorno y por cuestiones de energía es más eficiente que el SISMC debido a que presenta el efecto de conmutación en la entrada de control.

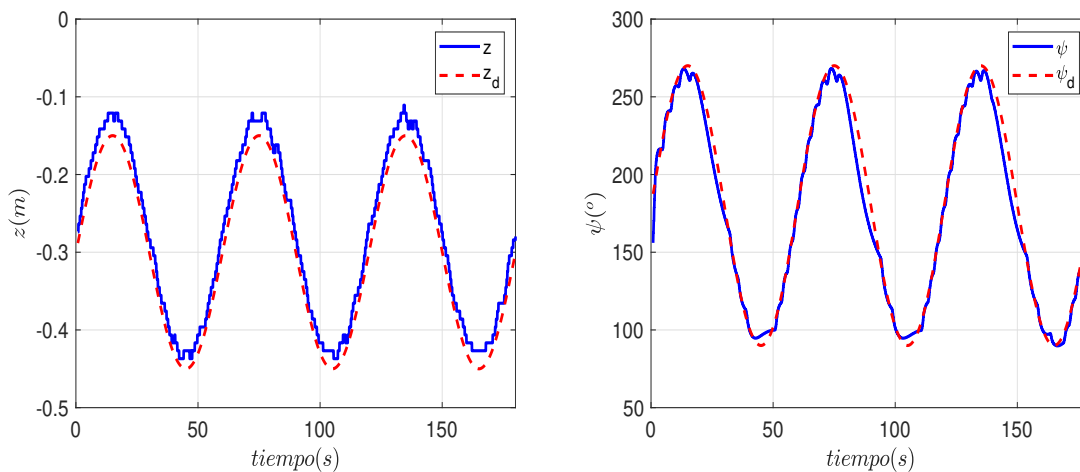


Figura 6.39: Respuesta del control PD+G al seguir un perfil sinusoidal en z, ψ .

RMSE			
	PD+G	BIA	SISMC
$z(m)$	0.1767	0.0183	0.0386
$\psi(^{\circ})$	8.67	4.25	2.81

Tabla 6.7: Criterio de error cuadrático medio para pruebas experimentales comparativas.

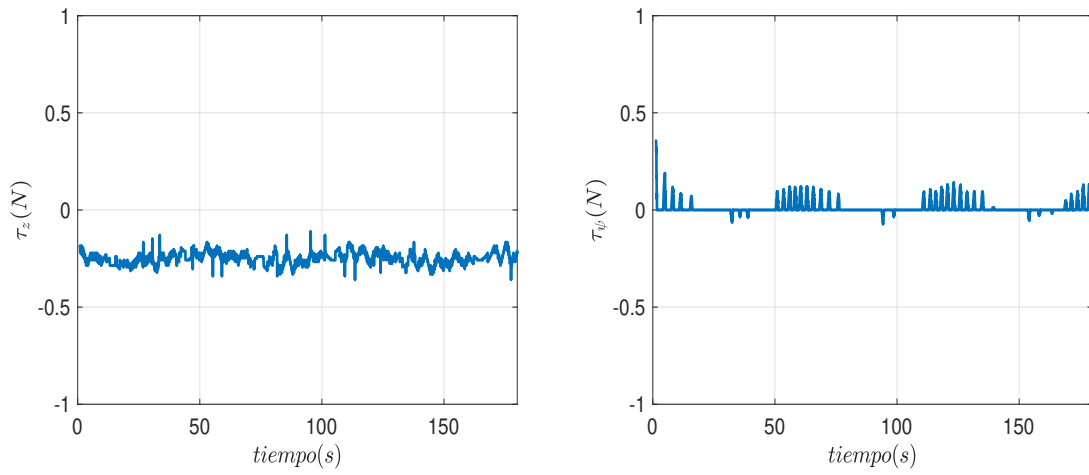


Figura 6.40: Evolución de las entradas de control PD+G.

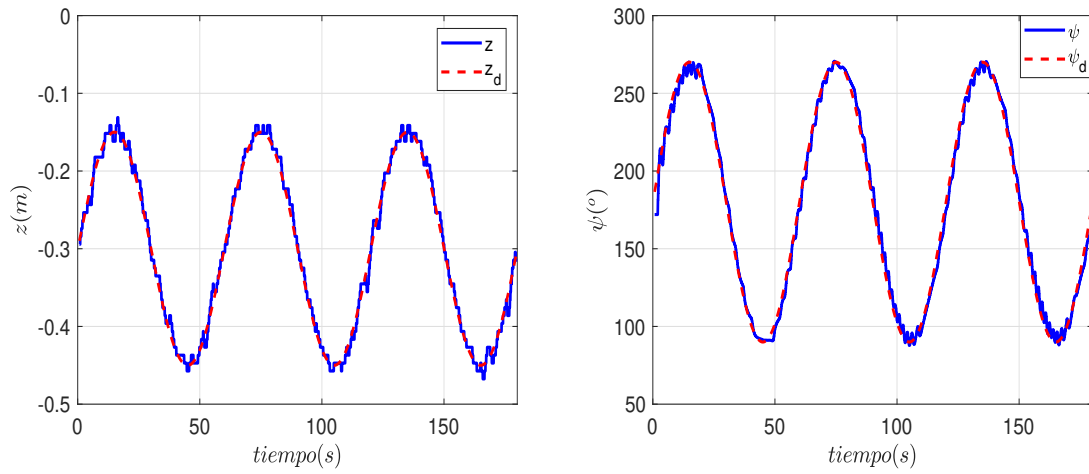


Figura 6.41: Respuesta del control BIA al seguir un perfil sinusoidal en z, ψ .

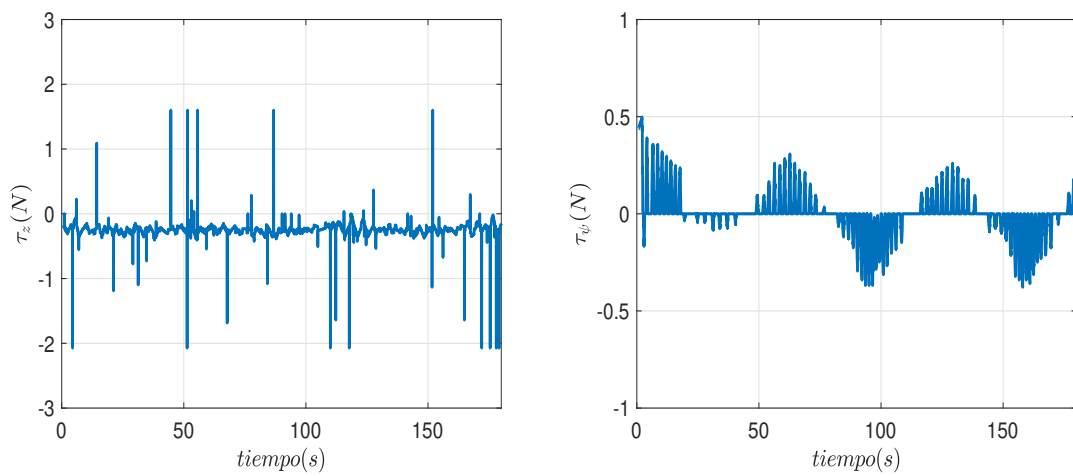


Figura 6.42: Evolución de las entradas de control BIA.

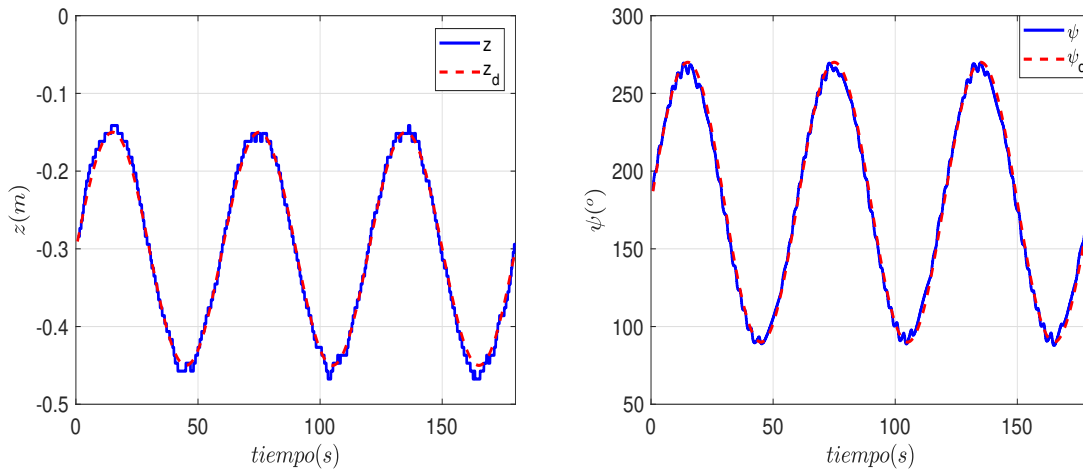


Figura 6.43: Respuesta del control SISM a seguir un perfil sinusoidal en z , ψ .

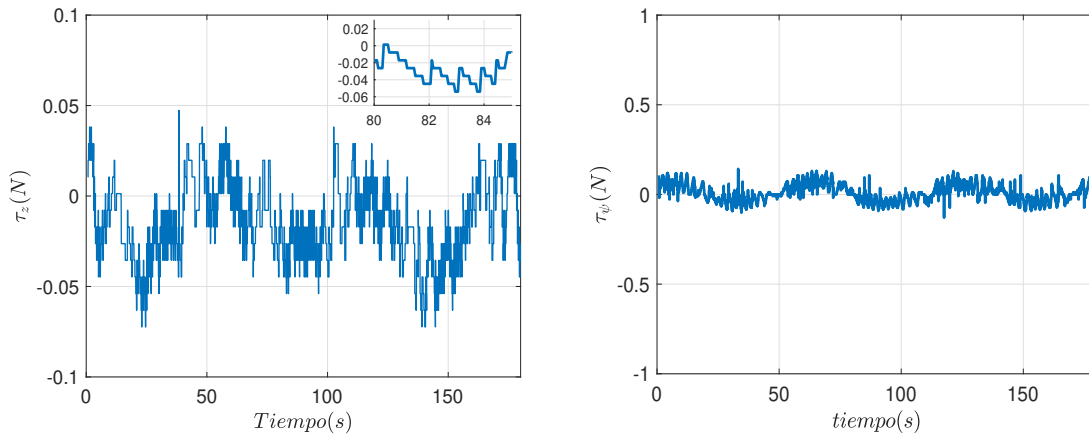


Figura 6.44: Evolución de las entradas de control SISM.

6.2.3. Sistema de guía

6.2.3.1. Detección de dos obstáculos estáticos

Una prueba experimental para la detección de obstáculos estáticos en tiempo real es realizada en una alberca de 5 m de diámetro por 0.7 m de profundidad. El escenario experimental es un entorno acuático altamente reflectivo en donde se colocan dos objetos, el objetivo es comprobar el funcionamiento de el método de detección propuesto en el Capítulo 4. En la Figura 6.45, se puede ver la ubicación del sonar acústico y de los dos objetos a detectar. Para ello, se utiliza el sonar Micron de Tritech y su paquete de ROS para realizar la lectura de la nube de puntos. Una vez que tenemos la medición quitamos los valores atípicos mediante el filtro de umbral propuesto, después nos suscribimos al tópic generado para utilizar esos datos en el algoritmo de K-means modificado donde introducimos el número de clúster

que necesitamos agrupar. Finalmente, la Figura 6.46 muestra que se pueden detectar ambos obstáculos incluso en entornos altamente reflectivos como lo son las albercas.



Figura 6.45: Escenario de prueba experimental con dos obstáculos estáticos.

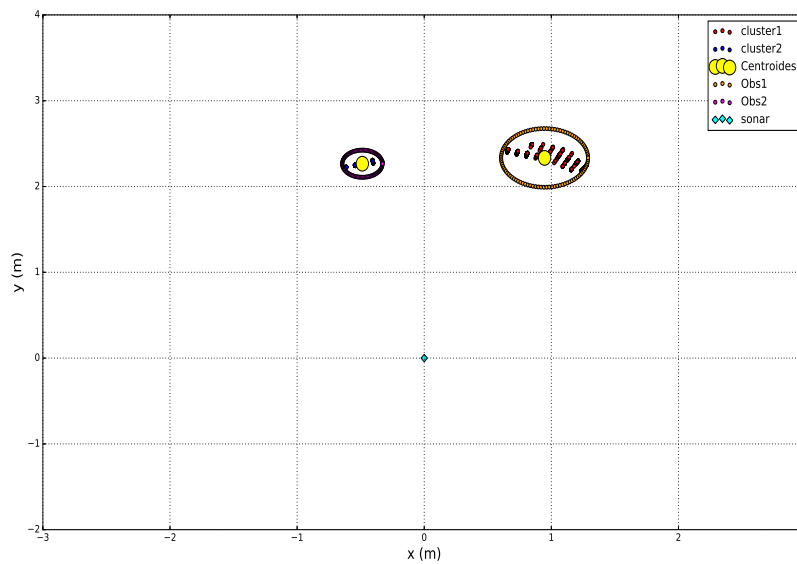


Figura 6.46: Detección de múltiples objetos mediante el filtro de umbral de intensidad y la técnica de K-means.

6.3. Conclusión del capítulo

De los resultados obtenidos de las simulaciones numéricas y experimentos. Se concluye que el SISMC sigue una trayectoria de referencia incluso ante perturbaciones de alta frecuencia, además de que reduce el efecto de conmutación en las entradas de control presentado por el ISMC. Mientras que el BIA realiza el seguimiento de trayectoria incluso ante perturbaciones como las causadas por una ola. En la simulación se introducen señales sinusoidales para agregar este efecto, durante la prueba la parte adaptativa estima rápidamente las perturbaciones lo que permite compensarlas. Note que la señal de control es suave lo que implica menor aporte energético. En el caso del PD+G, el sistema no puede hacer frente a los efectos dinámicos ocasionados por las perturbaciones lo que ocasiona un error de seguimiento mayor al obtenido con el SISMC y el BIA.

Acerca de los algoritmos de localización, los resultados experimentales demuestran que pueden estimar la posición del robot submarino. En el caso particular del algoritmo de odometría visual el vehículo se puede localizar con respecto a un mapa visual utilizando PTAM y el EKF para recuperar la escala del mapa y mejorar la estimación de la pose. Por otra parte, el filtro implementado para quitar los valores atípicos causados por la reflectancia del entorno en la estimación realizada por el sistema de posicionamiento acústico Water Linked nos permite reducir las variaciones en la medición lo que implica que podemos localizar a un vehículo submarino con mayor precisión.

Los resultados de las pruebas muestran que se pueden detectar obstáculos en entornos reflectivos utilizando el filtro de umbral de intensidad para una nube de puntos y la técnica de agrupamiento de datos K-means. Para evadir estos obstáculos se realizó un conjunto de simulaciones en Simulink de Matlab, que ponen a prueba los algoritmos de campos potenciales. Los resultados demuestran que los algoritmos de BUG circunnavegan alrededor del objeto hasta que encuentra el punto de salida, en el caso del BUG tangencial el vehículo se acerca demasiado al objeto y para llegar al objetivo el vehículo realiza grandes giros en las esquinas lo que provoca un contacto. Sin embargo, con el algoritmo de campo artificial tangencial los escenarios propuestos demostraron que podemos evitar múltiples obstáculos cuando el vehículo sigue una trayectoria por puntos de referencia.

Conclusiones y trabajo futuro

Este Capítulo proporciona las conclusiones del trabajo presentado a lo largo de esta tesis, así como el trabajo futuro que puede continuar desarrollándose como resultado de esta investigación.

7.1. Conclusión general

El funcionamiento del sistema del sistema de Guía, Navegación y Control (GNC) es comprobado mediante simulaciones y escenarios experimentales con el uso de dos UUV. En base a los resultados obtenidos se concluye que el desarrollo adecuado de cada uno de los módulos que componen un sistema GNC permiten el seguimiento de una trayectoria con evasión de obstáculos en un UUV con los algoritmos propuestos en este trabajo de tesis. Por lo tanto, la hipótesis es probada.

7.2. Conclusiones específicas

7.2.1. Sistema de guía

- Los algoritmos para generar trayectorias mediante puntos de referencias simulados en el presente trabajo, permiten que el vehículo pueda seguir un camino para realizar una tarea autónoma.
- Los resultados experimentales prueban que se pueden detectar objetos incluso en entornos altamente reflectivos. Mediante el algoritmo desarrollado basado en un filtro de umbral para una nube de puntos y la técnica de agrupamiento de datos K-means.

- Las técnicas de evasión de obstáculos mediante el uso de campos potenciales permiten que el vehículo evada un objeto, sin embargo, los resultados de simulación indican que el campo potencial tangencial permite realizar evasión de múltiples obstáculos sin necesidad de aproximarse y no presenta el problema de mínimo local.

7.2.2. Sistema de navegación

El sistema de navegación, incluye la localización del vehículo. Por ello, se implementan distintas soluciones mediante el uso de técnicas como el tiempo de retardo en la llegada, análisis de intervalos y odometría visual con un filtro de Kalman extendido.

- Los resultados de implementar la técnica de fusión de datos por análisis de intervalos para detectar la localización estática y dinámica de un vehículo acuático, demuestran que podemos localizar un vehículo con una precisión de 0.5 m. Pero tenemos que asumir que conocemos las características del entorno y para localizar el vehículo cuando este se encuentra en movimiento se utiliza el método de navegación por estima. Lo que implica que la estimación de la posición tenga deriva a lo largo del tiempo lo que disminuye la precisión.
- El sistema de posicionamiento acústico Water Linked con un filtro de segundo orden, nos permite localizar al vehículo en un rango de $200 \times 200 \times 100$ m. Sin embargo, la estimación depende de la configuración de la API del sistema y el uso del filtro para entornos altamente reflectivos ocasiona un retardo.
- La fusión mediante un EKF de una técnica de odometría visual (PTAM) y de mediciones inerciales de la IMU (acelerómetros y giroscopios). Implicó retos debido a diferentes problemáticas que se pueden presentar como la turbidez del agua, el cambio de iluminación, características móviles como peces, etc. Los resultados experimentales demuestran que el vehículo se puede localizar con respecto a un mapa visual utilizando PTAM y el EKF para recuperar la escala del mapa y mejorar la estimación de la pose.

7.2.3. Sistema de control

Para el diseño de los controladores presentados en este trabajo de tesis, se analiza el modelo matemático de dos robots submarinos mediante el modelo vectorial propuesto por

Fossen. Los controladores implementados se basan en técnicas convencionales, en el método de modo deslizantes y la teoría de bakstepping. Para el análisis de estabilidad se utiliza la teoría de Lyapunov.

- Los resultados en pruebas experimentales demuestran que el controlador PD+G, presenta errores de estado estacionario que no pueden ser eliminados, causados por las fuerzas restitutivas y los momentos de torsión. Además, no compensan las incertidumbres del modelo dinámico ni las perturbaciones del entorno. Por ello, su error de seguimiento en las pruebas es mayor al presentado por el BIA y SISMC.
- La ley de control de modo deslizante integral de segundo orden con efecto de super-twisting (SISMC) demostró mediante los resultados la eficiencia del controlador a lazo cerrado con un error de seguimiento pequeño incluso ante las perturbaciones simuladas, además de la reducción del efecto de conmutación en las entradas de control comparado con el control de modos deslizante integral (ISMC). Sin embargo, aunque se reduce el efecto de conmutación el gasto energético es mayor que el presentado por BIA y PD+G.
- El controlador bakstepping integral adaptativo (BIA), presenta un error de seguimiento menor en los resultados experimentales a la técnica por modos deslizantes en el seguimiento de un perfil sinusoidal en la profundidad. Y las pruebas de simulación demuestran la estimación en corto tiempo de las perturbaciones introducidas.

7.3. Trabajo futuro

Se presentan algunos trabajos que pueden desarrollarse o ser vistos más a detalle sobre esta investigación.

- Para evitar el retardo en la estimación de la posición del sistema acústico Water Linked, se puede implementar un predictor.
- Para la detección de múltiples obstáculos mediante la técnica de agrupamiento modificada de K-means, se puede implementar una mejora para que el proceso de agrupación no dependa de la identificación inicial de la representación de agrupaciones.

- En este trabajo solo se considera el uso de vehículos de 4 *GDL*, por ello se simplifica el modelo dinámico y se utiliza para el diseño o implementación de distintas estrategias de control. No obstante, los UUV's actuales tienen 6 *GDL* por lo que debemos extender las ecuaciones de movimiento.
- Para obtener resultados de simulación más precisos, tenemos que incluir la ecuación dinámica de los efectos de las corrientes y olas.
- Realizar experimentos en entornos no controlados, para verificar que el sistema de control de movimiento funcione correctamente al realizar una misión autónoma.

Apéndice

Este Apéndice incluye los pseudocódigos que resumen el procedimiento para la implementación de las estrategias de control en los vehículos LIRMIA III y BlueROV 2.

Algorithm 1 Procedimiento para la implementación del controlador mediante el LIRMIA III y UVxCtrl.

- 1: **Requiere:** Instalar opencv, C++, UvxCtrl y todas las dependencias necesarias de cada paquete.
 - 2: Inicializar la comunicación SSH con el vehículo LIRMIA III modificando la configuración de la red y seleccionando una IP estática.
 - 3: Configurar los dispositivos y el entorno del robot.
 - 4: Inicializar el programa UvxCtrl.
 - 5: El Programa espera la entrada del usuario a través de la GUI o el intérprete de comandos.
 - 6: **while** No hay error **do**
 - 7: Obtener lecturas del sensor de presión, inercial y del observador de estados. Para conocer el estado actual del robot.
 - 8: Calcular la ley de control para que el LIRMIA III alcance la posición, orientación o siga los puntos de referencia.
 - 9: Enviar señal de control a los ESC's para modifique la velocidad de los propulsores según la ley de control implementada.
 - 10: **end while**
 - 11: Cerrar el programa UvxCtrl y cerrar la comunicación SSH.
-

Algorithm 2 Procedimiento para la implementación del controlador mediante el uso de BlueROV 2, ROS y Water Linked GPS.

- 1: **Requiere:** Instalar mavros, bluerov-ros-playground, Water-Linked Underwater-GPS-pkg y todas las dependencias necesarias de cada paquete.
 - 2: Inicialice la comunicación con el vehículo BlueROV2 modificando la configuración de la red y seleccionando una IP fija.
 - 3: Inicializar y configurar el sistema de posicionamiento acústico Water Linked GPS mediante la interfaz gráfica de usuario en la API.
 - 4: Ejecutar el lanzador del paquete de bluerov-ros-playground y el nodo Water-Linked.
 - 5: Definir la trayectoria deseada para la posición y orientación del BlueROV 2 a través de la interfaz gráfica de usuario.
 - 6: **while** No hay error **do**
 - 7: Obtener lecturas del sensor inercial del BlueROV2 y del sistema acústico; posiciones, orientaciones y velocidades.
 - 8: Calcular la ley de control para que el BlueROV2 alcance la trayectoria en posición y orientación.
 - 9: Enviar señal de control a los ESC's para modifique la velocidad de los propulsores según la ley de control implementada.
 - 10: **end while**
 - 11: Cerrar todas las comunicaciones y paquetes activos en ROS.
-

Bibliografía

- [1] W. Commons, “Main page — wikimedia commons,,” 2020. [Online; accessed 02-mayo-2020].
- [2] J. Pike, “Program page — federation of american scientists,,” 2020. [Online; accessed 03-mayo-2020].
- [3] C. M. U. Biorobotics Laboratory, “Robotic motion planning: Bug algorithms,,” 2020. [Online; accessed 08-mayo-2020].
- [4] N. A. Raineault and J. Flanders, “New frontiers in ocean exploration: The e/v nautilus, noaa ship okeanos explorer, and r/v falkor 2018 field season.,” 2019.
- [5] G. Griffiths, *Technology and applications of autonomous underwater vehicles*, vol. 2. CRC Press, 2002.
- [6] S. Chutia, N. M. Kakoty, and D. Deka, “A review of underwater robotics, navigation, sensing techniques and applications,” *Proceedings of the Advances in Robotics*, pp. 1–6, 2017.
- [7] G. Antonelli and G. Antonelli, *Underwater robots*, vol. 3. Springer, 2014.
- [8] H. A. Moreno, R. Saltarén, L. Puglisi, I. Carrera, P. Cárdenas, and C. Álvarez, “Robótica submarina: Conceptos, elementos, modelado y control,” *Revista Iberoamericana de Automática e Informática industrial*, vol. 11, no. 1, pp. 3–19, 2014.
- [9] C. Von Alt, “Autonomous underwater vehicles,” in *Autonomous Underwater Lagrangian Platforms and Sensors Workshop*, vol. 3, 2003.
- [10] F. M. Barber, *Lecture on the Whitehead Torpedo*. US Torpedo Station, 1874.

- [11] T. Ewart, “Observations from straightline isobaric runs of spurv,” *Proc. IAP-SO/IAMAP PSII*, pp. 1–18, 1976.
- [12] T. Ewart and W. Bendiner, “An observation of the horizontal and vertical diffusion of a passive tracer in the deep ocean,” *Journal of Geophysical Research: Oceans*, vol. 86, no. C11, pp. 10974–10982, 1981.
- [13] D. Endicott and G. Kuhl, “The fast area search system (fass). a feasibility study.,” tech. rep., NAVAL COMMAND CONTROL AND OCEAN SURVEILLANCE CENTER RDT AND E DIV SAN DIEGO CA, 1992.
- [14] I. Busby Associates, *Undersea Vehicles Directory*. Busby Associates, Incorporated, 1990.
- [15] J. Bellingham, C. Goudey, T. Consi, J. Bales, D. Atwood, J. Leonard, and C. Chrysostomidis, “A second generation survey auv,” in *Proceedings of IEEE Symposium on Autonomous Underwater Vehicle Technology (AUV’94)*, pp. 148–155, IEEE, 1995.
- [16] B. Allen, R. Stokey, T. Austin, N. Forrester, R. Goldsborough, M. Purcell, and C. von Alt, “Remus: a small, low cost auv; system description, field trials and performance results,” in *Oceans’ 97. MTS/IEEE Conference Proceedings*, vol. 2, pp. 994–1000, IEEE, 1997.
- [17] V. Bobkov, A. Kudryashov, S. Melman, and A. Scherbatyuk, “Visual navigation and 3d reconstruction of underwater objects with autonomous underwater vehicle,” in *2017 24th Saint Petersburg International Conference on Integrated Navigation Systems (ICINS)*, pp. 1–5, IEEE, 2017.
- [18] R. Fezzani, B. Zerr, A. Mansour, M. Legris, and C. Vrignaud, “Fusion of swath bathymetric data: Application to auv rapid environment assessment,” *IEEE Journal of Oceanic Engineering*, vol. 44, no. 1, pp. 111–120, 2018.
- [19] S. Chatzichristofis, A. Kapoutsis, E. Kosmatopoulos, L. Doitsidis, D. Rovas, and J. B. De Sousa, “The noptilus project: Autonomous multi-auv navigation for exploration of unknown environments,” in *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 45, pp. 373–380, Elsevier, 2016.

- [20] E. M. Robador, G. M. Hansen, L. Acha, S. Pedre, *et al.*, “Design of an auv for visual inspection of nuclear power plants,” in *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pp. 1–6, IEEE, 2018.
- [21] M. Sung, S.-C. Yu, and Y. Girdhar, “Vision based real-time fish detection using convolutional neural network,” in *OCEANS 2017-Aberdeen*, pp. 1–6, IEEE, 2017.
- [22] H. Wu, S. He, Z. Deng, L. Kou, K. Huang, F. Suo, *et al.*, “Fishery monitoring system with auv based on yolo and sgbm,” in *2019 Chinese Control Conference (CCC)*, pp. 4726–4731, IEEE, 2019.
- [23] Y. Li, K. Cai, Y. Zhang, Z. Tang, and T. Jiang, “Localization and tracking for auvs in marine information networks: Research directions, recent advances, and challenges,” *IEEE Network*, 2019.
- [24] J. A. Catipovic, “Performance limitations in underwater acoustic telemetry,” *IEEE Journal of Oceanic Engineering*, vol. 15, no. 3, pp. 205–216, 1990.
- [25] L. Paull, S. Saeedi, M. Seto, and H. Li, “Auv navigation and localization: A review,” *IEEE Journal of Oceanic Engineering*, vol. 39, no. 1, pp. 131–149, 2014.
- [26] J. Hwang, N. Bose, and S. Fan, “Auv adaptive sampling methods: A review,” *Applied Sciences*, vol. 9, no. 15, p. 3145, 2019.
- [27] J. González-García, A. Gómez-Espinosa, E. Cuan-Urquizo, L. G. García-Valdovinos, T. Salgado-Jiménez, and J. A. E. Cabello, “Autonomous underwater vehicles: Localization, navigation, and communication for collaborative missions,” *Applied Sciences*, vol. 10, no. 4, p. 1256, 2020.
- [28] M. N. Zafar and J. Mohanta, “Methodology for path planning and optimization of mobile robots: A review,” *Procedia computer science*, vol. 133, pp. 141–152, 2018.
- [29] B. Patle, A. Pandey, D. Parhi, A. Jagadeesh, *et al.*, “A review: On path planning strategies for navigation of mobile robot,” *Defence Technology*, 2019.
- [30] D. Li, P. Wang, and L. Du, “Path planning technologies for autonomous underwater vehicles-a review,” *IEEE Access*, vol. 7, pp. 9745–9768, 2018.

- [31] J. Lee, M. Roh, J. Lee, and D. Lee, “Clonal selection algorithms for 6-dof pid control of autonomous underwater vehicles,” in *International Conference on Artificial Immune Systems*, pp. 182–190, Springer, 2007.
- [32] J. Bak, H.-N. Nguyen, S. Park, D. Lee, T. Seo, S. Jin, and J. Kim, “Positioning control of an underwater robot with tilting thrusters via decomposition of thrust vector,” *International Journal of Control, Automation and Systems*, vol. 15, no. 5, pp. 2283–2291, 2017.
- [33] L. Lapierre and D. Soetanto, “Nonlinear path-following control of an auv,” *Ocean engineering*, vol. 34, no. 11-12, pp. 1734–1744, 2007.
- [34] Z. M. Zain and N. F. Harun, “Backstepping control strategy for an underactuated x4-auv,” *Jurnal Teknologi*, vol. 74, no. 9, 2015.
- [35] T. Elmokadem, M. Zribi, and K. Youcef-Toumi, “Control for dynamic positioning and way-point tracking of underactuated autonomous underwater vehicles using sliding mode control,” *Journal of Intelligent & Robotic Systems*, vol. 95, no. 3-4, pp. 1113–1132, 2019.
- [36] K. Shojaei, “Three-dimensional neural network tracking control of a moving target by underactuated autonomous underwater vehicles,” *Neural Computing and Applications*, vol. 31, no. 2, pp. 509–521, 2019.
- [37] F. S. Tabataba’i-Nasab, A. Keymasi Khalaji, and S. A. A. Moosavian, “Adaptive non-linear control of an autonomous underwater vehicle,” *Transactions of the Institute of Measurement and Control*, vol. 41, no. 11, pp. 3121–3131, 2019.
- [38] F. Muñoz, J. S. Cervantes-Rojas, J. M. Valdovinos, O. Sandre-Hernández, S. Salazar, and H. Romero, “Dynamic neural network-based adaptive tracking control for an autonomous underwater vehicle subject to modeling and parametric uncertainties,” *Applied Sciences*, vol. 11, no. 6, p. 2797, 2021.
- [39] J. G. Graver, *Underwater gliders: Dynamics, control and design*. PhD thesis, Citeseer, 2005.
- [40] P. Bailey, “A unified mathematical model describing the maneuvering of a ship travelling in a seaway,” *Trans RINA*, vol. 140, pp. 131–149, 1997.

- [41] T. I. Fossen, *Guidance and control of ocean vehicles*. John Wiley & Sons Inc, 1994.
- [42] T. SNAME, “Nomenclature for treating the motion of a submerged body through a fluid,” *The Society of Naval Architects and Marine Engineers, Technical and Research Bulletin*, pp. 1–5, 1950.
- [43] T. I. Fossen, *Marine control system-guidance, navigation and control of ships, rigs and underwater vehicles*. 2002.
- [44] T. I. Fossen, *Handbook of marine craft hydrodynamics and motion control*. John Wiley & Sons, 2011.
- [45] P. Encarnação and A. Pascoal, “Combined trajectory tracking and path following: an application to the coordinated control of autonomous marine craft,” in *Proceedings of the 40th IEEE Conference on Decision and Control (Cat. No. 01CH37228)*, vol. 1, pp. 964–969, IEEE, 2001.
- [46] L. Stutters, H. Liu, C. Tiltman, and D. J. Brown, “Navigation technologies for autonomous underwater vehicles,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 38, no. 4, pp. 581–589, 2008.
- [47] M. E. Campos, *Design, modelling and control of AUV’s using artificial vision*. PhD thesis, Université Montpellier II, 2014.
- [48] A. Manzanilla, P. Castillo, and R. Lozano, “Nonlinear algorithm with adaptive properties to stabilize an underwater vehicle: real-time experiments,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6857–6862, 2017.
- [49] P. Castillo-García, L. E. M. Hernandez, and P. G. Gil, “Chapter 6-nonlinear control algorithms with integral action*,” *Indoor Navigation Strategies for Aerial Autonomous Systems*. Ed. by Pedro Castillo-García et al. Butterworth-Heinemann, p. 133, 2017.
- [50] J. A. Monroy, E. Campos, and J. A. Torres, “Attitude control of a micro auv through an embedded system,” *IEEE Latin America Transactions*, vol. 15, no. 4, pp. 603–612, 2017.
- [51] A. E. Z. Suarez, A. M. Magallanes, M. A. G. Rangel, R. L. Leal, S. S. Cruz, and F. M. Palacios, “Depth control of an underwater vehicle using robust pd controller: real-time

- experiments,” in *2018 IEEE/OES Autonomous Underwater Vehicle Workshop (AUV)*, pp. 1–6, IEEE, 2018.
- [52] M. Kim, H. Joe, J. Kim, and S.-c. Yu, “Integral sliding mode controller for precise manoeuvring of autonomous underwater vehicle in the presence of unknown environmental disturbances,” *International Journal of Control*, vol. 88, no. 10, pp. 2055–2065, 2015.
- [53] A. Levant, “Chattering analysis,” *IEEE Transactions on Automatic Control*, vol. 55, no. 6, pp. 1380–1389, 2010.
- [54] A. Levant, “Higher-order sliding modes, differentiation and output-feedback control,” *International journal of Control*, vol. 76, no. 9-10, pp. 924–941, 2003.
- [55] J. A. Moreno and M. Osorio, “Strict lyapunov functions for the super-twisting algorithm,” *IEEE Transactions on Automatic Control*, vol. 57, pp. 1035–1040, April 2012.
- [56] A. Chalanga, S. Kamal, L. M. Fridman, B. Bandyopadhyay, and J. A. Moreno, “Implementation of super-twisting control: Super-twisting and higher order sliding-mode observer-based approaches,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 6, pp. 3677–3685, 2016.
- [57] L. Lapierre, “Robust diving control of an auv,” *Ocean Engineering*, vol. 36, no. 1, pp. 92–104, 2009.
- [58] Y. Li, C. Wei, Q. Wu, P. Chen, Y. Jiang, and Y. Li, “Study of 3 dimension trajectory tracking of underactuated autonomous underwater vehicle,” *Ocean Engineering*, vol. 105, pp. 270–274, 2015.
- [59] C. Wei, W. Yanhui, and Z. Jianhui, “Back-stepping control of underactuated auv’s depth based on nonlinear disturbance observer,” in *2015 34th Chinese Control Conference (CCC)*, pp. 6061–6065, IEEE, 2015.
- [60] R. Kelly and R. Carelli, “A class of nonlinear pd-type controllers for robot manipulators,” *Journal of Robotic Systems*, vol. 13, no. 12, pp. 793–802, 1996.
- [61] R. Kelly and V. Santibáñez, *Control de movimiento de robots manipuladores*. Pearson educación, 2003.

- [62] E. Ibarra and P. Castillo, “Nonlinear super twisting algorithm for uav attitude stabilization,” in *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 640–645, IEEE, 2017.
- [63] J. A. Moreno and M. Osorio, “Strict lyapunov functions for the super-twisting algorithm,” *IEEE transactions on automatic control*, vol. 57, no. 4, pp. 1035–1040, 2012.
- [64] Y. Shtessel, M. Taleb, and F. Plestan, “A novel adaptive-gain supertwisting sliding mode controller: Methodology and application,” *Automatica*, vol. 48, no. 5, pp. 759–769, 2012.
- [65] Y. Shtessel, C. Edwards, L. Fridman, and A. Levant, *Sliding mode control and observation*. Springer, 2014.
- [66] J. A. Moreno, “A linear framework for the robust stability analysis of a generalized super-twisting algorithm,” in *2009 6th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, pp. 1–6, Jan 2009.
- [67] J. A. Moreno, “Lyapunov analysis of non homogeneous super-twisting algorithms,” in *2010 11th International Workshop on Variable Structure Systems (VSS)*, pp. 534–539, June 2010.
- [68] Z. Yan, S. Peng, J. Zhou, J. Xu, and H. Jia, “Research on an improved dead reckoning for auv navigation,” in *2010 Chinese Control and Decision Conference*, pp. 1793–1797, IEEE, 2010.
- [69] M. T. Sabet, H. M. Daniali, A. Fathi, and E. Alizadeh, “A low-cost dead reckoning navigation system for an auv using a robust ahrs: Design and experimental analysis,” *IEEE Journal of Oceanic Engineering*, vol. 43, no. 4, pp. 927–939, 2017.
- [70] J. Snyder, “Doppler velocity log (dvl) navigation for observation-class rovs,” in *OCEANS*, pp. 1–9, IEEE, 2010.
- [71] L. Jaulin, “A nonlinear set membership approach for the localization and map building of underwater robots,” *IEEE Transactions on Robotics*, vol. 25, no. 1, pp. 88–98, 2009.
- [72] T. Taketomi, H. Uchiyama, and S. Ikeda, “Visual slam algorithms: a survey from 2010 to 2016,” *IPSN Transactions on Computer Vision and Applications*, vol. 9, no. 1, p. 16, 2017.

- [73] F. Guth, L. Silveira, S. Botelho, P. Drews, and P. Ballester, “Underwater slam: Challenges, state of the art, algorithms and a new biologically-inspired approach,” in *Biomedical Robotics and Biomechatronics, 5th IEEE RAS & EMBS International Conference on*, pp. 981–986, IEEE, 2014.
- [74] A. Concha, P. Drews-Jr, M. Campos, and J. Civera, “Real-time localization and dense mapping in underwater environments from a monocular sequence,” in *OCEANS, Genova*, pp. 1–5, IEEE, 2015.
- [75] L. Silveira, F. Guth, P. Drews-Jr, P. Ballester, M. Machado, F. Codevilla, N. Duarte-Filho, and S. Botelho, “An open-source bio-inspired solution to underwater slam,” *IFAC-PapersOnLine*, vol. 48, no. 2, pp. 212–217, 2015.
- [76] M. O. Aqel, M. H. Marhaban, M. I. Saripan, and N. B. Ismail, “Review of visual odometry: types, approaches, challenges, and applications,” *SpringerPlus*, vol. 5, no. 1, p. 1897, 2016.
- [77] P. L. Negre, F. Bonin-Font, and G. Oliver, “Cluster-based loop closing detection for underwater slam in feature-poor regions,” in *Robotics and Automation (ICRA), International Conference on*, pp. 2589–2595, IEEE, 2016.
- [78] R. Garcia and N. Gracias, “Detection of interest points in turbid underwater images,” in *OCEANS*, pp. 1–9, IEEE, 2011.
- [79] C. Kahlefeldt, “Implementation and Evaluation of Monocular SLAM for an Underwater Robot,” Master’s thesis, University of Western Australia Robotics and Automation Lab, January 2018.
- [80] W.-H. Cheng, “Research for enhancing the precision of asymmetrical sbl system for any vessels,” *Ocean engineering*, vol. 33, no. 10, pp. 1271–1282, 2006.
- [81] Z. Yuyi, G. Zhenbang, W. Lei, Z. Ruiyong, and L. Huanxin, “Study of underwater positioning based on short baseline sonar system,” in *2009 International Conference on Artificial Intelligence and Computational Intelligence*, vol. 2, pp. 343–346, IEEE, 2009.

- [82] J. Sliwka, F. Le Bars, O. Reynet, and L. Jaulin, “Using interval methods in the context of robust localization of underwater robots,” in *2011 Annual Meeting of the North American Fuzzy Information Processing Society*, pp. 1–6, IEEE, 2011.
- [83] J. Sliwka, F. Le Bars, and L. Jaulin, “Calcul ensembliste pour la localisation et la cartographie robustes,” *JD JN MACS*, 2009.
- [84] L. Jaulin, “Robust set-membership state estimation,” *Automatica*, *To appear*, 2008.
- [85] G. Klein and D. Murray, “Parallel tracking and mapping for small ar workspaces,” in *6th IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR)*., pp. 225–234, IEEE, 2007.
- [86] S. Lynen, M. W. Achtelik, S. Weiss, M. Chli, and R. Siegwart, “A robust and modular multi-sensor fusion approach applied to mav navigation,” in *2013 IEEE/RSJ international conference on intelligent robots and systems*, pp. 3923–3929, IEEE, 2013.
- [87] R. Mur-Artal, J. M. M. Montiel, and J. D. Tardos, “Orb-slam: a versatile and accurate monocular slam system,” *IEEE transactions on robotics*, vol. 31, no. 5, pp. 1147–1163, 2015.
- [88] N. A. Shneydor, *Missile guidance and pursuit: kinematics, dynamics and control*. Elsevier, 1998.
- [89] H. K. Heidarsson and G. S. Sukhatme, “Obstacle detection and avoidance for an autonomous surface vehicle using a profiling sonar,” in *2011 IEEE International Conference on Robotics and Automation*, pp. 731–736, IEEE, 2011.
- [90] K. Wagstaff, C. Cardie, S. Rogers, S. Schrödl, *et al.*, “Constrained k-means clustering with background knowledge,” in *Icml*, vol. 1, pp. 577–584, 2001.
- [91] A. Likas, N. Vlassis, and J. J. Verbeek, “The global k-means clustering algorithm,” *Pattern recognition*, vol. 36, no. 2, pp. 451–461, 2003.
- [92] Y. Koren and J. Borenstein, “Potential field methods and their inherent limitations for mobile robot navigation,” in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, pp. 1398–1404 vol.2, April 1991.

- [93] C. W. Warren, “Global path planning using artificial potential fields,” in *Proceedings, 1989 International Conference on Robotics and Automation*, pp. 316–321, IEEE, 1989.
- [94] P. Castillo-Garcia, L. E. M. Hernandez, and P. G. Gil, *Indoor Navigation Strategies for Aerial Autonomous Systems*. Butterworth-Heinemann, 2016.
- [95] J. Ng and T. Bräunl, “Performance comparison of bug navigation algorithms,” *Journal of Intelligent and Robotic Systems*, vol. 50, no. 1, pp. 73–84, 2007.
- [96] H. M. Choset, S. Hutchinson, K. M. Lynch, G. Kantor, W. Burgard, L. E. Kavraki, and S. Thrun, *Principles of robot motion: theory, algorithms, and implementation*. MIT press, 2005.
- [97] C.-J. Wu, *6-DoF Modelling and Control of a Remotely Operated Vehicle*. PhD thesis, Flinders University, College of Science and Engineering., 2018.
- [98] R. Kuc and M. W. Siegel, “Physically based simulation model for acoustic sensor robot navigation,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, no. 6, pp. 766–778, 1987.
- [99] S. Weiss and R. Siegwart, “Real-time metric state estimation for modular vision-inertial systems,” *IEEE International Conference on Robotics and Automation (ICRA)*., 2011.
- [100] S. Weiss, M. W. Achtelik, M. Chli, and R. Siegwart, “Versatile distributed pose estimation and sensor self-calibration for autonomous mavs,” *IEEE International Conference on Robotics and Automation (ICRA)*, 2012.