

Universidad Central “Marta Abreu” de Las Villas

Facultad de Ingeniería Eléctrica

Departamento de Automática y Sistemas Computacionales



TRABAJO DE DIPLOMA

Propuesta de adaptación del Software Mission Planner para supervisión de vehículos autónomos acuáticos de superficie

Autor: Luis Alain Cabreja López

Tutor: Ing. Rubén Eduardo Carles Barrero

Santa Clara

2017

"Año 59 de la Revolución"

Universidad Central “Marta Abreu” de Las Villas

Facultad de Ingeniería Eléctrica

Departamento de Automática y Sistemas Computacionales



TRABAJO DE DIPLOMA

Propuesta de adaptación del Software Mission Planner para supervisión de vehículos autónomos acuáticos de superficie

Autor: Luis Alain Cabreja López

Email: (lcabreja@uclv.cu)

Tutor: Ing. Rubén Eduardo Carles Barrero

Email: (ruben_eduardo@uclv.edu.cu)

Santa Clara

2017

"Año 59 de la Revolución"



Hago constar que el presente TRABAJO DE DIPLOMA fue realizado en la Universidad Central "Marta Abreu" de Las Villas como parte de la culminación de estudios de la especialidad de Ingeniería en Automática, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

Luis Alain Cabreja López

Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Luis Alain Cabreja López

Autor

Ing. Rubén E. Carles Barrero

Tutor

Firma del Responsable de
Información Científico-Técnica

PENSAMIENTO

“Nunca consideres el estudio como una obligación, sino como una oportunidad para penetrar en el bello y maravilloso mundo del saber.”

Albert Einstein.

DEDICATORIA

A mi hijo

Por ser el último motor impulsor para lograr mi objetivo, y por traerme paz y armonía de nuevo a mi vida

A mi madre

Por estar siempre presente en los buenos y malos momentos de mi vida, por comprenderme y apoyarme y siempre brindarme los consejos más sabios.

A mi esposa

Por estar presente en los últimos momentos de mi formación y brindarme apoyo incondicional, por darme consejos y ánimos para seguir y formarme, por comprenderme, por estar en los buenos y malos momentos de mi vida.

A mi padre

Por brindarme apoyo incondicional en mi camino.

A mi Hermana

Por brindarme su apoyo y darme fuerzas para continuar.

A todos ellos, por ser los responsables de mi formación y educación como profesional, por ser ejemplo a seguir, por dar su mejor esfuerzo para lograr mi formación.

AGRADECIMIENTOS

Especialmente le agradezco a mi madre por todo, ya que si no fuese por ella no estuviese donde estoy ni fuese quien soy hoy, además que me ha sabido guiarme por los buenos senderos de la vida.

A mi tutor Rubén Eduardo por su apoyo desde los inicios de mi preparación del trabajo de diploma, por su apoyo incondicional para la preparación de la Tesis.

A mi esposa Leidys por su comprensión y apoyo incondicional y darme alegría, y además por realizarme el regalo máspreciado para un ser humano MI HIJO.

A mis compañeros de estudio, por compartir conmigo los buenos y malos momentos de estos últimos años de mi formación como profesional, sobre todo a Vicente, Rolando, Addiel, Luis Alberto, Guillermo, Claudia.

A todos los profesores que han estados involucrados en mi formación como profesional.

De forma general quisiera agradecer a todo aquel que, de una forma u otra, estuvo involucrado en la realización de este trabajo de diploma y a mi formación como profesional.

El seguimiento y control de vehículos autónomos acuáticos de superficie utilizando el software Mission Planner es un tema ampliamente discutido e implementado en la actualidad. Tanto la planificación de misiones como el control de velocidad y rumbo de este tipo de vehículos son elementos importantes para lograr un cumplimiento satisfactorio y estable de una misión. Por ende, en este trabajo se proporciona toda la información relacionada con este planificador de misiones, así como el procedimiento para planificar y ejecutar una misión, también se brinda información de cómo configurar los principales parámetros para la navegación de un USV, entre otros temas importantes. Para lograr estabilizar el control de la trayectoria del vehículo se propone ajustar un controlador P-D y sintonizar el mismo en el Mission Planner, el ajuste de este controlador se efectúa teniendo en cuenta el código fuente para el control de rumbo. Se proporciona, además, los principales parámetros que se deben configurar para ejecutar correctamente una misión. Los resultados teóricos obtenidos son apoyados primeramente mediante simulaciones con el Mission Planner utilizando el simulador Software In The Loop (SITL) y luego mediante una prueba real con el vehículo Krick Félix.

ÍNDICE DE FIGURAS

<i>Figura</i>	<i>Página</i>
1.1. USV AutoCat, Alrededor del 2000-----	9
1.2. Interfaz Gráfica del Software Mission Planner-----	14
1.3. Modelo Krick Félix-----	15
1.4. Componentes electrónicos-----	18
2.1. Sección correspondiente a la calibración del vehículo -----	23
2.2. Calibración autónoma mediante el microcontrolador-----	23
2.3. Ventana auxiliar para realizar la calibración manual -----	24
2.4. Subsección establecer la calibración Accel-----	25
2.5. Subsección para la calibración del radio control -----	26
2.6. Planificación de misión autónoma-----	27
2.7. Registro de Waypoints y Comandos-----	28
2.8. Ventana auxiliar para ejecutar Software In The Loop-----	29
2.9. Modos de trabajo de Mission Planner-----	30
2.10. Emisora DEVO F7-----	32
2.11. Diagrama de conexiones-----	34
2.12. Algoritmo de guiado L1-----	35
2.13. Configuraciones básicas del L1 Control -----	36
3.1. Lugar Geométrico de las raíces-----	39
3.2. Respuesta de Frecuencia-----	39
3.3. Simulación con parámetros estándares-----	41
3.4. Simulación sin KI-----	41
3.5. Simulación con el controlador diseñado-----	42
3.6. Simulación aumentando el radio de los Waypoints-----	43
3.7. Trayectoria programada con parámetros estándares-----	44

3.8. Trayectoria programada con valores modificados-----	45
3.9. Respuesta del giro en X con valores estándares-----	45
3.10. Respuesta del giro en X con valores modificados-----	46
3.11. Respuesta de los valores de L1 Control deseado (Azul) y registrado (Naranja) con parámetros estándares-----	47
3.12. Respuesta de los valores de L1 Control deseado (Azul) y registrado (Naranja) con parámetros modificados-----	47

ÍNDICE DE TABLAS

<i>Tabla</i>	<i>Página</i>
1.1.Especificaciones Geométricas del Krick Félix-----	16
1.2.Sensores integrados en el Pixhawk-----	17
1.3.Especificaciones geométricas del Pixhawk-----	17
1.4.Especificaciones del Módulo GPS + Compass-----	18
1.5.Especificaciones Técnicas del Módulo de Telemetría-----	19
3.1. Parámetros del Lazo de control de rumbo-----	40

ÍNDICE DE ANEXOS

<i>Anexos</i>	<i>Página</i>
A.1. Diagrama en Bloques del Código-----	55
B.1. Pantalla Plan de Vuelo-----	56
B.2. Radio Calibración-----	56
C.1. Código Fuente-----	57
D.1. Hardware del Vehículo-----	60
D.2. Vehículo Diseñado-----	60
E.1. Estanque de pruebas-----	61
E.2. Test de validación-----	61

ÍNDICE

PENSAMIENTO	i
DEDICATORIA.....	ii
AGRADECIMIENTOS	iii
RESUMEN	iv
ÍNDICE DE FIGURAS	v
ÍNDICE DE TABLAS	vii
ÍNDICE DE ANEXOS	viii
INTRODUCCIÓN	1
CAPÍTULO 1. TENDENCIA Y DESARROLLO DE LOS VEHÍCULOS AUTÓNOMOS ACUÁTICOS (USV) DE SUPERFICIE Y DEL SOFTWARE MISSION PLANNER.	6
1.1 Introducción.....	6
1.2 Vehículos Autónomos acuáticos.....	6
1.3 Desarrollos de los USV	7
1.4 Información General de los USV	9
1.5 Aplicaciones	9
1.5.1 Programa Sea Grant College MIT.....	10
1.5.2 Desarrollos en Europa	10
1.5.3 Desarrollos en China.....	11
1.6 Mission Planner	12

1.6.1	¿Qué es Mission Planner?	12
1.6.2	Planificación de una misión con puntos de referencia	13
1.6.3	Funciones y pantallas	13
1.7	Modelo Krick Félix	14
1.8	Pixhawk Px4	16
1.9	Conclusiones Parciales	20
CAPÍTULO 2. INTRODUCCIÓN AL MISSION PLANNER Y A LA APLICACIÓN.		
		21
2.1	Introducción	21
2.2	Configuración inicial del Mission Planner para vehículos acuáticos de superficie.	21
2.3	Calibraciones	22
2.3.1	Compass Calibration	22
2.3.2	Accelerometer Calibration	24
2.3.3	Radio Control Calibration	25
2.4	Procedimiento de planificación de una misión	26
2.5	Sistema de simulación Software In The Loop (SITL)	28
2.6	Modos de trabajo del Mission Planner para vehículos de superficie	29
2.7	Reportes e Informes de misiones	31
2.8	Plataforma Móvil	31
2.8.1	Navegación Manual	31
2.8.2	Navegación Autónoma	32
2.8.3	Hardware del vehículo	33
2.9	Algoritmo de Guiado L1 Control	34
2.9.1	Controlador de Rumbo	35

2.10	Conclusiones parciales.....	36
CAPÍTULO 3. PROPUESTA DE ADECUACIÓN		38
3.1	Introducción.....	38
3.2	Ajuste del Controlador de rumbo	38
3.3	Pruebas experimentales	40
3.3.1	Simulaciones	40
3.3.2	Test de Validación	43
3.4	Análisis Económico.....	47
3.5	Conclusiones parciales	49
CONCLUSIONES		50
RECOMENDACIONES.....		52
REFERENCIAS BIBLIOGRÁFICAS		53
ANEXOS		57
Anexo A	Controlador de Rumbo	57
Anexo B	Funciones y pantallas del Mission Planner.....	58
Anexo C	Código para implementación del Control de Rumbo.....	59
Anexo D	Hardware del Vehículo	62
Anexo E	Test de validación	63

INTRODUCCIÓN

Un Vehículo Autónomo Acuáticos (USV por sus siglas en inglés), es aquel vehículo marino capaz de operar de manera independiente durante el cumplimiento de misiones, sin que medie intervención humana alguna. Hoy en día existen diversas aplicaciones de estos vehículos; dado que pueden ser empleados en condiciones ambientales extremas en las cuales el hombre no podría sobrevivir. Los USVs pueden desarrollar tareas de reconocimiento, seguimiento y exploración, por lo que por sus amplias propiedades pueden ser utilizados en arqueología marina y navegación tanto marítima como portuaria. Además de ser usados en el ámbito civil es considerable su uso en el ámbito militar ([Sehuveret, 2016](#)).

Se puede decir que el desarrollo de los USVs comenzó en la década del 60 con vehículos como el UARS y el SPURV construido por la “*Universidad de Washington*”. A este le siguieron otros como el Skat del “*Instituto de Oceanología de Shirshov*” en Rusia ([Blidberg, 2001](#)).

Después del año 2000 el comercio de estos vehículos ha crecido continuamente y hoy en día brindan nuevas ventajas para la sociedad, al convertirse en una de las herramientas más potentes utilizadas en múltiples aplicaciones marinas.

Los USVs se han convertido en una herramienta poderosa para aplicaciones científicas, medioambientales y militares. En la actualidad los investigadores se han enfocado en perfeccionar los sistemas automáticos que poseen este tipo de vehículo y que les permite cumplir las misiones con mayor precisión.

El desarrollo de estos vehículos ha sido una tarea científico-técnica desarrollada por múltiples universidades, institutos y centros de investigación y producción de todo

el mundo ([Blidberg, 2001](#)), quienes llevan a cabo proyectos de desarrollo con un alto nivel científico.

En la actualidad por la diversidad que poseen estos dispositivos en aplicaciones, universidades e instituciones científicas del mundo entero se proponen como tarea la investigación y el desarrollo de nuevos prototipos de USV, cada vez con más prestaciones. Como ejemplo de ello se puede citar el “*Instituto Tecnológico Nacional*” (Itenas), el cual desarrolló un USV que posee un sistema de navegación y posicionamiento para inspeccionar profundidades marinas (Batimetría) ([Ratih C y Suwardhi, 2014](#)), en la “*University of Engineering and Technology Peshawar*”, se desarrolló un prototipo para medir e inspeccionar profundidades marinas ([Ullah, 2014](#)), en la “*Universidad Nacional del Centro provincia de Buenos Aires*” en conjunto con la “*Universidad de las Islas Baleares*”, proponen el diseño de un vehículo con el fin de establecer o compartir información y recursos de energías a través de estos dispositivos, ahorrándose así el trabajo de desplegar infraestructuras submarinas para lograr tal objetivo ([Villar y Acosta, 2013](#)); En la “*Universidad Politécnica de Cataluña*”, se propone el diseño de un ROV o aerodeslizador con microcontrolador *Pixhawk*, con el fin de recolectar información medioambiental en lugares donde no se pueda acceder tan fácilmente por el hombre ([Rabuffetti, 2016](#)).

En los últimos años el Grupo de Automatización Robótica y Percepción (GARP) de la Universidad Central “Martha Abreu” de Las Villas ha acumulado un número importante de investigaciones dirigidas a la supervisión y ejecución de estrategias de guiados a vehículo aéreo, terrestres y acuáticos con la implementación del Software Mission Planner para dichas tareas, ejemplo de ello se puede ver en la tesis de ([Larquin, 2016](#)) donde se propone la “*Implementación del Algoritmo de guiado I-LOS sobre el autopiloto ArduPlane*” y en la misma tesis realizan la modificación de la programación del *firmware ArduPlane*, otro ejemplo de implementación del Mission Planner es en la tesis de ([L. E. Hernández, 2016](#)) donde en esta investigación se realiza el estudio sobre la estrategia de guiado que se utiliza en el *firmware Ardupilot*, otro ejemplo de utilización del Mission Planner en el GARP es en la tesis de ([Rodríguez, 2015](#)), donde se realiza un estudio también al control

de rumbo L1 implementado en el *firmware Ardupilot* del Mission Planner. En la tesis de ([Peláez, 2016](#)) también se utiliza el Mission Planner para la realización de un análisis del Filtro Extendido de *Kalman* presentes en el Pixhawk.

El Mission Planner en el ámbito internacional es usado por su gran fiabilidad en diferentes aplicaciones, puesto que el mismo es una estación de tierra con todas las funciones para el proyecto piloto automático de código abierto APM y la serie *Pixhawk* ([“Mission Planner”, 2017](#)). Además de que en el centro no se cuenta con una aplicación para la supervisión de un vehículo autónomo acuático de superficie, por ende, es de vital importancia realizar una investigación sobre el tema antes mencionado para plasmar una propuesta de cómo se podría configurar el software Mission Planner para efectuar el guiado de este tipo de vehículo.

La situación del Problema encontrada para la realización de la investigación es ¿Cómo llegar a contribuir a la planificación y navegación de vehículos autónomos acuáticos de superficie teniendo en cuenta las principales características que brinda este tipo de sistemas?

Con la realización de esta investigación se expone como impacto posible, colaborar con diferentes instituciones en el desarrollo de aplicaciones con vehículos autónomos acuáticos utilizados en la exploración de mares, y así poder supervisarlos y controlarlos con el software Mission Planner.

La aplicabilidad que presenta esta investigación es que los resultados poseen una aplicación práctica y teórica de gran trascendencia para los vehículos autónomos acuáticos que está investigando el GARP. Esta investigación contribuirá a lograr establecer la supervisión mediante el Mission Planner a vehículos acuáticos.

Objetivo General

Adecuar el software Mission Planner para lograr la supervisión y control de un vehículo autónomo acuático de superficie.

Objetivos Específicos

- 1- Estudiar los aspectos teóricos relacionados con la temática a tratar que se presentan en la bibliografía especializada.

- 2- Implementar una arquitectura de hardware que permita el empleo de Mission Planner en vehículos autónomos acuáticos de superficie.
- 3- Definir parámetros relacionados con la configuración y funcionamiento del software Mission Planner y de la aplicación.
- 4- Adecuar en el software Mission Planner los parámetros necesarios para la supervisión y control de vehículos autónomos acuáticos de superficie.

Tareas de investigación

Para el desarrollo de esta investigación se proponen las siguientes tareas a realizar:

- Realización de un estudio detallado sobre las principales características técnicas del software Mission Planner para la supervisión y guiado de vehículos autónomos acuáticos de superficie.
- Caracterización de las aplicaciones existentes hoy en día en el mundo para vehículos autónomos acuáticos de superficie que usan Mission Planner.
- Realización de un estudio de los parámetros y funcionamiento de Planificador de Misiones que se utilizará en la aplicación.
- Sintonización de un controlador adecuado para el lazo de control de rumbo.
- Modificación de los parámetros en el software Mission Planner para estabilizar el USVs en su trayectoria.
- Elaboración del informe final.

Estructura del Trabajo

El informe de la investigación presenta la siguiente estructura:

INTRODUCCIÓN: En la Introducción se aborda sobre la importancia, actualidad y necesidad del tema a investigar y se presentan los elementos teóricos principales de la investigación, así como el problema científico a resolver en la investigación, la hipótesis y los objetivos que se persiguen con la misma.

CAPITULARIO

CAPÍTULO I: En este capítulo se aborda temas y conceptos principales relacionados con los USVs, su evolución y desarrollo. Se abordan las principales características del planificador de tareas escogido para ejecutar dicha aplicación.

Además de mencionar algunas características técnicas del modelo de USV Krick Félix y del Microcontrolador *Pixhawk*.

CAPÍTULO II: En este capítulo se realiza una descripción detallada de todos los componentes del software Mission Planner y se definen los principales parámetros que se deben configurar en dicho software para el correcto funcionamiento de un USV. Además, se aborda sobre el control de rumbo utilizado para la supervisión y control del USV. También, se explica en que consiste el algoritmo de guiado *L1 Control*. Además, se describe el hardware implementado para el USV.

CAPÍTULO III: En este capítulo se muestran las pruebas realizadas al vehículo, además se definen las adecuaciones que se le deben realizar al Mission Planner para supervisar vehículos acuáticos de superficie. También se realiza un análisis económico para este tipo de aplicaciones.

CAPÍTULO 1. TENDENCIA Y DESARROLLO DE LOS VEHÍCULOS AUTÓNOMOS ACUÁTICOS (USV) DE SUPERFICIE Y DEL SOFTWARE MISSION PLANNER.

1.1 Introducción

En este capítulo se aborda sobre temas y conceptos principales relacionados con los *USVs*, su evolución y desarrollo. Además de abordar sobre las principales características que presenta el planificador de tareas escogido para ejecutar dicha aplicación. También se brinda información sobre el modelo de embarcación *Krick Félix*, la cual es la que se utiliza como vehículo autónomo acuático de superficie, y se brindan datos relacionados con el *Pixhawk*.

1.2 Vehículos Autónomos acuáticos

En la actualidad los *USV* o vehículos de superficie autónomos, poseen una gran diversidad de clasificaciones, pero teniendo en cuenta su forma de control se puede clasificar en Vehículos Operados Remotamente (*ROV*, Remotely Operated Vehicles) y Vehículos Autónomos Subacuáticos (*AUV*, Autonomous Underwater Vehicles). Los primeros presentan una conexión física con la estación remota, desde la que se controla su movimiento. Esto puede resultar una desventaja a medida que se requiere alcanzar una mayor profundidad en las misiones, debido a la tensión impuesta al cable de conexión. Por su parte, los *AUV* poseen un sistema de control autónomo que les permite manejarse independientemente, sin que medie intervención humana alguna ([Sehuveret,2016](#)).

Hoy en día los *USVs* se han desarrollado y se han realizado demostraciones por Laboratorios académicos, sociedades anónimas y usuarios gubernamentales, en

los diferentes campos de aplicación de estos dispositivos como se podría mencionar el levantamiento de planos de batimetría, la defensa e investigación general de robótica, entre otras(Manley, 2008).

1.3 Desarrollos de los USV

El uso de los USV en las operaciones navales no es nuevo, puesto que después de la Segunda Guerra Mundial, estos dispositivos fueron creados y utilizados para fines tales como la detección de minas e intrusos. Por ejemplo, en 1946, durante la operación denominada “Operación Crossroads”, barcos y aviones autónomos se utilizaron para obtener muestras de aguas radioactivas después de cada una de las explosiones atómicas. Años más tardes, en la década de los ‘60 un barco de fibra de vidrio de 7 metros, impulsados por un motor de gas dentro del casco V-8, fue modificado para funcionar como un detector de minas por control remoto. No fue sino hasta la década de los ‘90 que se empezaron a crear proyectos de esta índole. Esto es debido a los progresos de la ciencia y tecnología, sin embargo, la mayor razón es que la marina de los Estados Unidos presta más atención a la guerra contra el terrorismo. En la segunda Guerra del Golfo, los barcos autónomos atrajeron la atención de la marina, porque estos dispositivos realizaban las tareas establecidas con éxito. Otros países e instituciones europeas principalmente han introducido los barcos autónomos en su marina moderna a partir de entonces también (Yan, Pang, Sun, y Pang, 2010).

El “Programa Sea Grant College MIT” en 1993 fue una de las primeras instituciones en crear este tipo de vehículo autónomo, donde los primeros prototipos fueron diseñados para cumplir diversas misiones. El primer prototipo creado fue nombrado ARTEMIS, el modelo que presentaba era una réplica a pequeña escala de un barco de pesca, y el mismo fue utilizado como una plataforma capaz de poner a prueba los sistemas de navegación y control requeridos para este tipo de vehículos, su primera aplicación fue para recopilar datos de las profundidades marinas en el río Charles en Boston. Las limitantes de estos vehículos principalmente fueron que sus dimensiones eran muy pequeñas y por ende limitaba su resistencia y comportamiento en mar abierto. El campo de aplicación del ARTEMIS estaba limitado al río Charles, una región de límite de interés científico limitada. Por las

limitantes que presentaba estos prototipos se crea a partir de *ARTEMIS* un nuevo diseño, un *Kayak*. Este nuevo prototipo se sometió a una serie de ensayos en el río Charles, y los mismos tenían equipado un sistema de seguimiento acústico y se utilizaban para el seguimiento de peces marcados ([Manley, 2008](#)).

Para continuar con los estudios de Batimetría automatizados iniciados con el *ARTEMIS*, se decide realizar un nuevo diseño de USV. Las especificaciones del siguiente dispositivo se basaron en el deseo de crear un sistema tan versátil y útil como una pequeña embarcación tripulada, manteniendo al mismo tiempo un tamaño pequeño para permitir las operaciones de despliegue. Los nuevos USV o ACES también llamados por sus siglas en inglés (*Autonomous Coastal Exploration System*) fueron desarrollados en 1996 y 1997, las primeras pruebas realizadas a este nuevo prototipo se realizaron a campo abierto en *Gloucester*, durante el verano de 1997. Luego completadas las pruebas se equiparon estos dispositivos con sensores adecuados para la realización de estudios hidrográficos y luego se realizó otra prueba con éxito en el puerto de Boston en diciembre de este propio año. A partir de enero de 1998, este nuevo USV fue devuelto al laboratorio para actualizarle el sistema mecánico. Las nuevas modificaciones e iteraciones de diseño fueron probadas en el verano de 2000, cuando se renombró la nueva plataforma de USV, como AutoCat ([Manley, 2008](#)), la cual se muestra en la Figura 1.1.

Estos primeros desarrollos de los USV fueron un motor inspirador para otros programas más allá del MIT. Pero es solo en los últimos años que los USV han proliferado y han alcanzado a tener un impacto en muchas áreas.

En el 2001, la “*Oficina de investigación Naval*” (ONR) de los Estados Unidos propuso el concepto de “*Buque de Combate Litoral*” (LCS por sus siglas en inglés), este prototipo se ha optimizado para la flexibilidad en los litorales como un sistema reconfigurable de misiones de los sistemas tanto tripulados como no tripulados. Este buque de superficie inteligente no tripulado de alta velocidad, es totalmente mencionado claramente en los denominados LCS, por ende los USV constituyen un sistema de combate autónomo junto con los vehículos aéreos y terrestres([Yan, Pang, Sun y Pang, 2010](#)).



Figura 1.1. USV AutoCat, alrededor del 2000

El éxito que presentan las misiones con los USVs ha aumentado el interés dentro de la Armada de E.U.A en estos dispositivos, además, varias marinas de guerras modernas han seguido su ejemplo. Los USVs tienen el potencial para reducir el riesgo en las fuerzas tripuladas, también para llevar a cabo tareas que vehículos tripulados no pueden ejercer. Estos vehículos autónomos podrían variar, dependiendo el tipo de misión que se desee, desde pequeños recolectores de datos a grandes barcos o buques no tripulados ([Yan, Pang, Sun y Pang, 2010](#)).

1.4 Información General de los USV

El desarrollo de estos vehículos ha sido una tarea científico técnica desarrollada por múltiples universidades, institutos y centros de investigación y producción de todo el mundo, quienes llevan a cabo proyectos de desarrollo con un alto nivel científico. Para la realización del estudio de estos vehículos autónomos es necesario realizar un estudio de las diferentes aplicaciones en donde están vinculados los mismo y además del área donde se encuentra.

1.5 Aplicaciones

Hoy en día existen disímiles aplicaciones de estos dispositivos en el mundo. A continuación, se mencionan varias aplicaciones típicas de los USVs.

1.5.1 Programa Sea Grant College MIT

Los USVs desarrollados en el MIT para aplicaciones educativas y civiles, consistentes en el *ARTEMIS*, este prototipo presentaba un modelo similar al de los pesqueros existente en 1997 y los catamaranes *ACES* luego desarrollados para la exploración de las costas y por último el *AutoCat*. Estos USVs demostraron la viabilidad del control automático, así como la posibilidad de operar de manera autónoma en la recogida de datos hidrográficos. Luego de la integración, desde el punto de vista de la interfaz del operador, la planificación de misión y de la arquitectura de PC, con los valores específicos de clase *MIT Odyssey*, los vehículos desarrollados nombrados *Kayaks SCOUT* sirven de apoyo para el desarrollo y la realización de pruebas de algoritmos de navegación acústicas distribuidas para vehículos subacuáticos ([Caccia, Bibuli, Bono, y Bruzzone, 2008](#)).

1.5.2 Desarrollos en Europa

- En Europa muchas entidades y universidades de la región desarrollan este tipo de vehículo autónomo, ejemplo de esto se podría mencionar la “*Universidad de Rostock*”, en Alemania, en la cual desarrollaron un modelo para el posicionamiento de alta precisión, guía de caminos y la realización de mediciones en aguas poco profundas ([Majohr y Buch, 2006](#)).
- En el laboratorio de *DSOR* de Lisboa IST-ISR, desarrollaron un modelo de catamarán autónomo nombrado *Delfim*, que sirve como un relé de comunicación entre un AUV. Este Proyecto de la Unión Europea fue financiado por el “*Sistema Avanzado de Integración para la Gestión de la Operación Coordinada de Vehículos Robóticos Oceánicos*”(ASIMOV), luego del financiamiento este proyecto fue explotado como una unidad independiente para la recogida de datos de mapas batimétricos y marinos ([Caccia, Bibuli, Bono y Bruzzone, 2008](#)).
- En Génova, Italia, un grupo de especialista desarrolló un catamarán autónomo nombrado *Charlie CND-Issia*, que originalmente lo diseñaron y explotaron para la recogida de microcapa superficial del mar, y luego se

acondiciono para la investigación robótica en los buques autónomos([Caccia, Bibuli, Bono y Bruzzone, 2008](#)).

1.5.3 Desarrollos en China

- Contramedidas de mina (MCM)

Los requisitos de esta aplicación son impulsados por la necesidad de la flota china en establecer rápidamente grandes áreas seguras de operación, rutas y carriles de tránsito. El objetivo de la creación de este tipo de dispositivo es encontrar áreas para la operación de la flota que se encuentren limpias de mina, sin requerir plataformas tripuladas para entrar a las zonas sospechosas minadas. Para establecer la misión, se desarrolló un modelo utilizando un patrón sencillo de escalera para la búsqueda de minas colocadas al azar. Para la implementación de la aplicación se supone que cada mina que se encuentra también es neutralizada con un misil desplegado por el USV ([Yan, Pang, Sun y Pang, 2010](#)).

- Antiterroristas

En esta aplicación los USV operan por sí mismo o integrados a una red de seguridad más amplia, además en dicha aplicación estos vehículos establecen una seguridad de ribera frente a las amenazas encubiertas o no convencionales. En comparación con otras aplicaciones, esta establece un único USV, el cual es el encargado de establecer de forma remota la vigilancia de día y noche en el puerto ([Yan, Pang, Sun y Pang, 2010](#)).

- Vigilancia y Reconocimiento

En esta aplicación el USV puede controlar el curso del agua mediante una orientación de un operador en el mar o desde la orilla. El dispositivo tiene montado una cámara que permite el monitoreo diurno y nocturno, además de poseer un láser infrarrojo con la capacidad de detectar y seguir blancos en un entorno próximo. Posee también una unidad de comunicación que permite la constante comunicación entre el vehículo y la estación de control. Las capacidades de navegación, vigilancia y reconocimiento permiten que los marineros detecten comportamientos sospechosos y salvaguardar instalaciones marítimas y buques de alto valor ([Yan, Pang, Sun y Pang, 2010](#)).

- Guerra antisubmarina (ASW)

Esta aplicación fue creada en vista a las crecientes amenazas submarinas que hoy en día existen. El enfoque para la capacidad de las misiones incluye un enfoque mono-estándar (transmisor-receptor en un solo dispositivo), enfoque bi-estándar y multi-estáticas (transmisor(s) y recetor(s) encontrados en diferentes plataformas) y numerosas variaciones en la ubicación relativa de los sensores ([Yan, Pang, Sun y Pang, 2010](#)).

1.6 Mission Planner

1.6.1 ¿Qué es Mission Planner?

Mission Planner es una completa aplicación de estación terrestre para el proyecto de piloto automático de código abierto (APM) *ArduPlane*, *ArduCopter* y *ArduRover* y de la serie *Pixhawk*. Es un software libre de fácil adquisición en internet. Este software fue desarrollado por Michael Osborne. Solamente es compatible con las diferentes versiones de Windows y Mac OS ([Mission Planner Home, 2017](#)). Con este software se pueden realizar disímiles tareas tales como:

- Cargar el *firmware* (software) en el piloto automático (APM, PX4...) que está controlando el vehículo automático.
- Configuración y ajuste del vehículo automático para un rendimiento óptimo.
- Planificar, guardar y cargar misiones autónomas en el piloto automático con una sencilla interfaz.
- Descargar y analizar los registros de las misiones creadas por el piloto automático.
- Con el hardware de telemetría adecuado se puede:
 - ✓ Controlar el estado del vehículo mientras está en funcionamiento.
 - ✓ Registrar los registros de telemetría que contienen mucha más información que los de piloto automático.
 - ✓ Ver y analizar los registros de telemetría.
 - ✓ Operar el vehículo en FPV o en vista de primera persona.

Estas especificaciones y muchas más características se encuentra en ([DRONES.MX, 2017](#)).

1.6.2 Planificación de una misión con puntos de referencia

- Configuración de la posición inicial

Para el helicóptero o el módulo *ArduCopter* la posición inicial se establecerá como la posición donde el helicóptero está armado o situado.

Para el avión o el módulo *ArduPlane* la posición inicial será la ubicación del avión donde se le bloqueó el GPS. Es decir que hay que encender el vehículo donde se desea que sea su posición inicial y armarlo con el Mission Planner.

Para el carro o el módulo *ArduRover* la posición inicial será igual que la del avión.

1.6.3 Funciones y pantallas

Para comprender mejor esta sección en la Figura 1.2 se muestra la interfaz gráfica del software. A continuación se describirá algunas de las funciones de las mismas([Mission Planner Home, 2017](#)).

- Conexión (*Connect*): Este botón está situado en la esquina superior derecha y con el mismo es con el que se puede conectar el planificador de misiones a su piloto automático. Además, se puede seleccionar también los tipos de dispositivos y la velocidad de comunicación.
- Datos de vuelo (*Flight Data*): Esta es la pantalla principal, y en la misma se puede observar toda la información sobre lo que se está ejecutando y las cosas que puede realizar en una misión.
- Plan de vuelo (*Flight Plan*): Esta es la próxima pantalla y en la misma se brinda información sobre los diversos aspectos de la preparación del plan de vuelo para cumplir las misiones.
- Configuración inicial (*Initial Setup*): En esta pantalla se ofrece información sobre lo que se puede realizar en la misión además de la configuración de los diferentes aspectos.
- Sintonización y configuración (*Configuration Tuning*): En esta pantalla se muestra toda la información adecuada para realizar los ajustes iniciales y la sintonización de los controladores en los lazos de control.
- Simulación (*Simulation*): En esta pantalla se muestra información acerca de cómo se puede utilizar el Planificador de misiones y además se brinda un

simulador de vuelos denominado *Software In The Loop* para simular misiones sin la necesidad de estar conectado físicamente con el dispositivo.

- Terminal: En esta pantalla se muestra información de los diferentes detalles de la misión.
- Ayuda (Help): En esta pantalla se muestra información sobre el software.



Figura 1.2. Interfaz Gráfica del Software Mission Planner

1.7 Modelo Krick Félix

El prototipo *Krick Félix* (Figura 1.3) fue construido en la década de 1960. Fue usado como un bote de transporte y carga, usualmente son usado como vehículos transbordadores alrededor del puerto. Estos botes son tan prolíferos que pueden encontrarlos en casi todos los refugios alemanes. Originalmente el diseño de este bote salió del modelo de un buque del puerto de Hamburgo, Alemania. El objetivo del diseño de este barco de pequeño porte era producir un prototipo de fácil ensamblado, pero a su vez de gran parecido al buque original.



Figura 1.3. Modelo Krick Félix

El casco del barco está fabricado de un material de alta calidad usado en Alemania con estos fines como es el ABS (*Acrilonitrilo Butadieno Estireno*), además de presentar una quilla acondicionada y estructurada para facilitar el montaje del timón del barco. El prototipo presenta en su diseño una marca de agua palpada en el casco, la cual facilita la instalación de la cubierta del dispositivo. Para el movimiento de la embarcación el prototipo cuenta con una propela acoplada y además un motor eléctrico ([Cornwall Model Boats, 2015](#)). Algunas de las especificaciones geométricas que presenta este dispositivo se muestran en la Tabla 1.1.

Este vehículo originalmente presenta la posibilidad del piloto manual. Para ello presenta en su estructura una estación de radio control *DEVO RX701*. Este módulo es necesario para establecer la operación manual del vehículo. Este módulo posee un Espectro de Expansión de Secuencias Directa (DSSS) de 2.4 GHz de frecuencia lo cual le provee al dispositivo una reacción rápida y además de protegerlo contra el atascamiento o bloqueo. Para establecer la comunicación de este módulo es usado un codificador PPM (*Pulse Position Modulation*) *V2-Fishbonne*. El mismo tiene como función agrupar diferentes señales PWM (*Pulse Width Modulation*) en una única señal ([DEVO F7, 2017](#)).

Tabla 1.1. Especificaciones Geométricas del Krick Félix

Parámetros	Símbolos	Valor	Unidad de medida
Longitud	L	0.634	m
Ancho	B	0.128	M
Calado	T	0.035	M
Masa	m	0.8	Kg

El sistema de comunicación y control que presenta el dispositivo cuenta con un equipo informático o una computadora (PC), la cual se encarga de monitorizar la embarcación, transmitirle órdenes para lograr la navegación autónoma y obtener los datos de los sensores mediante el Planificador de tareas escogido para implementar dichas acciones.

Para la administración de la potencia necesaria al controlador y a los demás elementos de hardware se cuenta con un *Motor DC* de 7.4 Volts y el mismo puede girar a una razón de 600 revoluciones por minuto (rpm), y el mismo tiene acoplado un Servo Motor digital *Robbe FS61bb* de 4.8 - 6 Volts ([Cornwall Model Boats, 2015](#)).

1.8 Pixhawk Px4

Para controlar el USV se usa el microcontrolador *Pixhawk Px4*. Este microcontrolador es un elemento de hardware diseñado para establecer el control remoto de los dispositivos autónomos, y por ende el mismo puede procesar los datos de los sensores que presente el vehículo instalado a bordo, de motores y de la superficie de control. También cuenta con una serie de sensores integrados para así poder conocer mejor o con mayor precisión la ubicación del vehículo. Estos sensores se muestran en la Tabla 1.2.

El *Px4* posee en su estructura un microprocesador de 32 bits *ARM Cortex M4* que trabaja a 168 MHz, con 256 KB de memoria RAM y 2 MB de memoria flash, además

presenta en su estructura un coprocesador a pruebas de fallos de 32 bits. Este dispositivo viene acompañado de elementos como adaptadores y cables Micro-SD y Micro-USB, además presenta un módulo de alimentación, un zumbador y un *switch* de seguridad, y también cuenta con varios cables de conexión ([Robotics, 2014](#)). Además de contar con los componentes antes mencionados, este dispositivo cuenta con otros elementos los cuales complementan el funcionamiento óptimo de este dispositivo (Figura 1.4).

Tabla 1.2. Sensores integrados en el Pixhawk

Sensores
Giróscopo ST Micro 16-bits
Acelerómetro/Magnetómetro St Micro 14 bits
Barómetro MEAS
Acelerómetro/Magnetómetro MPU6000

En la Tabla 1.3 se brinda una serie de especificaciones geométricas las cuales presenta este dispositivo.

Uno de los módulos importantes dentro del microcontrolador es el Módulo GPS + Compass (UbloxM8N), el cual tiene como propósito geolocalizar el vehículo autónomo en todo momento de la misión, y el cual proporciona la posición, velocidad y tiempo del dispositivo. Algunas de las características técnicas de este dispositivo se muestran a continuación en la Tabla1.4 ([A.G, 2017](#)).

Tabla 1.3. Especificaciones Geométricas del Pixhawk Px4

Parámetros	Valor	U/M
Peso	38	G
Ancho	50	Mm
Alto	15.5	Mm
Longitud	81.5	Mm

El Pixhawk también cuenta con un Módulo de telemetría 3DR, el cual presenta una conexión de datos a tierra entre el autopiloto y la estación en tierra (PC) gracias a los radios 3DR(Cierzo, 2013) . El piloto automático se conecta mediante cables que van directamente al puerto de telemetría del dispositivo. Para una mejor comprensión de las características que posee este dispositivo se muestra a continuación en la Tabla 1.5 una serie de especificaciones técnicas que presenta este dispositivo.

*Figura 1.4. Componentes electrónicos*

Tabla 1.4. Especificaciones del Módulo GPS + Compass

Características Técnicas	Datos
Velocidad de actualización	10 Hz
Batería recargable de litio	3 V
Regulador de Ruido	3.3 V
Indicadores de Potencia LEDs	
Diámetro	60 mm
Peso	32 g

El sistema de software que se implementará en la PC será el planificador de misiones Mission Planner y así con el mismo se podrá lograr la supervisión y configuración a distancia del vehículo.

Tabla 1.5. Especificaciones Técnicas del Módulo de Telemetría

Propiedades	Dato
Potencia de salida	100 mW
Corrección de errores	25 %
Frecuencia de trabajo	915 o 433 mHz
Puerto Micro-USB	
Voltaje de suministro	3.7-6 VDC

1.9 Conclusiones Parciales

En la referencia relacionada a la parte de los vehículos acuáticos de superficie, la implementación del software Mission Planner constituye un tema de interés. Además las aplicaciones de USV a nivel mundial son implementadas en condiciones extremas utilizando el software Mission Planner. Específicamente, el seguimiento y guiado de un vehículo acuático de superficie mediante el Mission Planner ha planteado interrogantes por los integrantes del GARP, las cuales es necesario solucionar.

Los vehículos autónomos acuáticos de superficie han traído desde su inicio, por la década de los '60, grandes resultados y buena aceptación por varias compañías del mundo, y esto trae consigo que se realicen investigaciones importantes tanto en el ámbito nacional como el internacional.

En el ámbito internacional estos dispositivos tienen un amplio uso para disímiles aplicaciones, tanto en el área civil como en la militar, y en dichas aplicaciones muchas instituciones realizan el seguimiento y guiado mediante el software Mission Planner, aunque no es el único planificador de misiones existente, pero es un software libre y de código abierto. Pero en el GARP no se cuenta con una aplicación para vehículos acuáticos de superficie usando este planificador de misiones.

Por todo lo anterior mencionado, resulta factible realizar un estudio de este tipo de aplicación existente en el mundo, para así poder realizar una propuesta de cómo se podría utilizar el Mission Planner para guiar un vehículo acuático de superficie. El objetivo principal sería lograr realizar la propuesta para que dicho software supervise y guíe este tipo de vehículo.

CAPÍTULO 2. INTRODUCCIÓN AL MISSION PLANNER Y A LA APLICACIÓN.

2.1 Introducción

En el presente capítulo se aborda sobre los parámetros de seguimiento y guiado que utilizan los USV de superficie. Además, se describen las principales características del planificador de tareas Mission Planner para establecer la comunicación y la ejecución de la misión programada para un USV. También se mencionan los principales parámetros de configuración que deben ser ajustados para vehículos autónomos acuáticos de superficie. Además, se alude sobre el tipo de navegación que utiliza la plataforma móvil y los controladores tanto de rumbo como de velocidad que se encuentran programados en el *firmware* del Mission Planner.

2.2 Configuración inicial del Mission Planner para vehículos acuáticos de superficie.

En la configuración del vehículo autónomo acuático de superficie existen 7 parámetros fundamentales los cuales se necesitan para que el dispositivo ejecute correctamente la misión deseada en el modo automático ([Cierzo, 2013](#)). Estos parámetros son:

- STEER2SRV_P: Con este parámetro se define el radio de giro para el dispositivo. Además es el valor de ganancia Proporcional del Lazo de control. El valor para comenzar una misión debe ser 2.
- TURN_MAX_G: Es el indicador de la fuerza de gravedad que el vehículo puede soportar sin dejar ser inestable. Si el valor de este parámetro es muy alto entonces el vehículo podría volcarse cuando esté realizando giros. El valor para comenzar una misión debe ser 1.

- NAVL1_PERIOD: Este parámetro controla la agresividad del algoritmo de navegación. Mientras más pequeño sea el número más agresivo será el giro en el modo AUTO. El valor para comenzar una misión debe ser 6.
- SPEED_TURN_GAIN: Este parámetro controla el porcentaje de la velocidad de destino actual. Este valor por defecto debe encontrarse en 100.
- CRUISE_SPEED: Este parámetro es el encargado de controlar la velocidad en metros por segundos en modo AUTO. El valor para comenzar una misión debe ser 2.
- CRUISE_THROTTLE: Parámetro encargado de proporcionarle el valor inicial al acelerador para establecer la marcha. Este valor debe ser adecuado para lograr un buen control de velocidad en el dispositivo. El valor para comenzar una misión debe ser 20.
- STEER2SRV_MINSPD: Parámetro encargado de proporcionar la velocidad mínima en m/s del motor.

Existen también otros parámetros que son esenciales para comenzar una misión, dichos parámetros están vinculados con la calibración del vehículo, puesto que si no se tiene calibrado el vehículo correctamente no se podrá ejecutar ninguna misión.

2.3 Calibraciones

2.3.1 Compass Calibration

Uno de los parámetros imprescindibles que se debe calibrar es la brújula del vehículo. Sin la correcta calibración de este parámetro no es posible realizar ninguna acción con el dispositivo desde el Misión Planner.

Para calibrar este parámetro se debe estar en la sección *Initial Setup*, y dentro de esta sección en el apartado *Mandatory Hardware*, luego de encontrado en este apartado se debe seleccionar *Compass* dentro de la gama de opciones que se presenta. Luego de seleccionada la opción *Compass* se mostrarán una serie de opciones como las que se pueden ver en la Figura 2.1. Dentro de las opciones que aparecen en este apartado para vehículos autónomos de superficie se debe activar el *compass #1* y desactivar los restantes *compass*. Además, si se está usando cualquier microcontrolador moderno se debe seleccionar el botón “*Pixhawk / PX4*”,

en nuestro caso se usará un *Pixhawk*. Para completar la calibración del vehículo existen dos opciones, una de las opciones es la calibración en el mismo controlador automáticamente como se muestra en la Figura 2.2 (*Onboard Calibration*) y la otra es la calibración manual (*Offboard Calibration o Live Calibration*), esta última más antigua y tediosa puesto que se debe mover el vehículo en todas las direcciones hasta completar la calibración.



Figura 2.1. Sección correspondiente a la calibración del vehículo.

Para realizar la calibración mediante el controlador se debe dar clic en “*Onboard Mag Calibration*” de esta sección y presionar el botón “*Start*” como se puede ver en la Figura 2.2. Luego de comenzado el proceso de calibración se deberá sostener el vehículo en el aire y luego girarlo de tal modo que todas sus caras apunten hacia el suelo por algunos segundos, luego de realizado esto el microcontrolador habrá completado la calibración de forma autónoma.

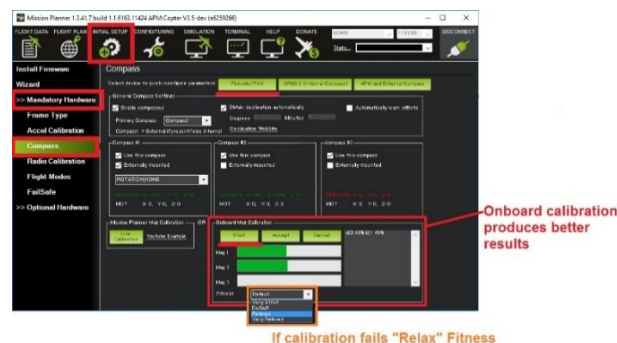


Figura 2.2. Calibración autónoma mediante el microcontrolador.

Si por alguna razón falla el proceso de calibración o no se establece la calibración correctamente se escuchará un tono de fallo en el microcontrolador y las barras

verdes que aparecen se pueden restablecer y la rutina de calibración puede reiniciarse. Si no completa la calibración el Mission Planner luego de varios intentos se debe cancelar la acción y en el menú desplegable “*Fitness*” se debe escoger otro parámetro y reiniciar la operación.

Para realizar la calibración “*Offboard Calibration*” se debe presionar el botón “*Live Calibration*” y se abrirá una nueva ventana como la que se muestra en la Figura 2.3.

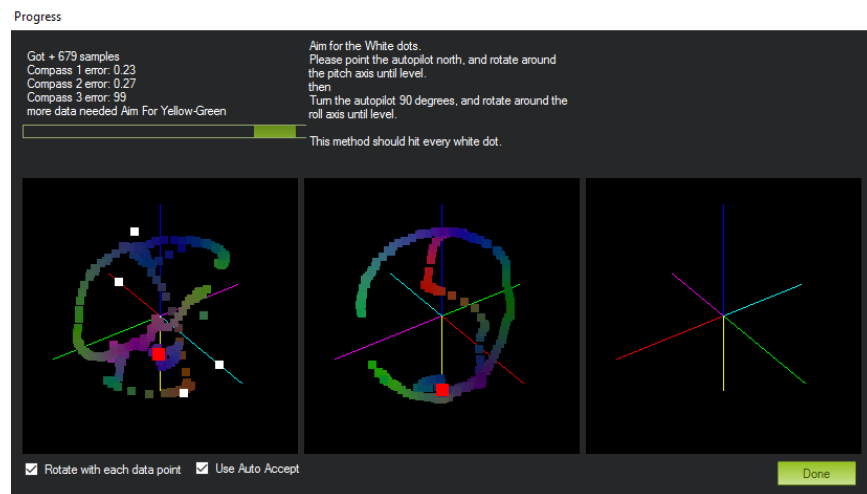


Figura 2.3. Ventana auxiliar para realizar la calibración manual.

Luego de iniciado el proceso de calibración es necesario girar el vehículo hasta lograr que las tres esferas se completen. La calibración se completará de forma automática cuando se obtengan datos en todas las posiciones.

En nuestro caso la calibración usada fue la *Live Compass Calibration*, puesto que el modelo *Krick Félix* no presenta grandes dimensiones, como se mencionó en el capítulo 1.

2.3.2 Accelerometer Calibration

El acelerómetro es otro de los parámetros que se debe calibrar para poder iniciar cualquier misión. Para calibrar este parámetro se debe estar en la sección *Initial Setup*, y dentro de esta sección en el apartado *Mandatory Hardware*, luego de encontrado en este apartado se debe seleccionar *Accel Calibration* dentro de la gama de opciones que se presenta (ver Figura 2.4).

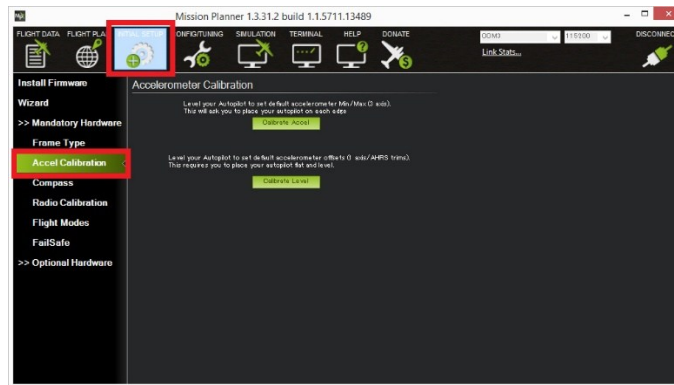


Figura 2.4. Subsección establecer la calibración Accel.

Para comenzar la calibración se debe presionar primeramente el botón “*Calibrate Accel*”. Luego de iniciado el proceso de calibración el Mission Planner le pedirá colocar el vehículo en cada posición de calibración. Las posiciones de calibración son: el barco debe encontrarse perfectamente anivelado; hacia el lado derecho; hacia el lado izquierdo; la parte delantera mirando hacia abajo; la parte delantera mirando hacia arriba. Luego de completado el proceso el software mostrará un botón “*Calibration Success full*” y se habrá completado la calibración de este parámetro.

2.3.3 Radio Control Calibration

La calibración del radiocontrol es otro de los parámetros necesario que se debe calibrar, para así poder controlar remotamente el USV. La calibración de este parámetro se realiza de forma sencilla, solamente se debe mover los interruptores en su rango completo y se registrará la posición máxima y mínima de los mismos. Para realizar la calibración del radiocontrol se debe apagar el vehículo desconectándole la batería por razones de seguridad. Luego de desconectado el vehículo entonces se procede a realizar la calibración. Para comenzar la calibración se debe estar en la sección *Initial Setup*, y dentro de esta sección en el apartado *Mandatory Hardware*, luego de encontrado en este apartado se debe seleccionar *Radio Calibration* dentro de la gama de opciones que se presenta (ver Figura 2.5).



Figura 2.5. Subsección para la calibración del radio control.

Luego de encontrado en este apartado el Mission Planner mostrará un mensaje para comprobar que el equipo de radiocontrol esta encendido y que el vehículo está apagado. Si el receptor y el transmisor están conectados, al mover los controles del radiocontrol se deberán mover las barras verdes que se encuentran en el Mission Planner.

2.4 Procedimiento de planificación de una misión

Para realizar la planificación de una misión es necesario dirigirse a la sección *Flight Plan* del Mission Planner, ver [Anexo B.1](#). En esta sección es donde se realiza la planificación de la misión que se desea ejecutar.

Esta sección esta subdividida en tres partes, el área del mapa en el centro, el panel de Acciones en el lado derecho y en la parte inferior es donde se registran todos los *Waypoints* ([Mission Planner Home, 2017](#)). En la esquina superior izquierda del mapa se muestran los valores de latitud y longitud actuales de la posición del *mouse* y la distancia entre puntos de paso.

Existen dos formas de planificar una misión en esta sección, manualmente o de forma autónoma. Si se desea realizar la planificación manual de la misión solamente es necesario dar un clic en el mapa donde se desea que se dirija el dispositivo y seguido a ello se mostrará en el mapa un *Waypoints* en la ubicación especificada, y así sucesivamente hasta lograr planificar la misión deseada. Los valores de latitud, longitud, altura, entre otros parámetros de los *Waypoints* serán registrados en la parte inferior. El parámetro relacionado con la altura del dispositivo se debe cambiar puesto que por defecto el Mission Planner establece una altura de 100 metros y esto

en la aplicación de un USV no tiene ningún sentido puesto que este vehículo está a en el agua y no en el aire . De esta forma se puede crear una misión manualmente. Además, en esta sección se puede ingresar otro comando dando clic derecho sobre el mapa y seguido a esto aparecerá un menú con diferentes opciones. La otra función de planificación de misiones automáticamente es muy útil para funciones de mapeo, donde el vehículo debe ir de un lado a otro en un patrón de cortadora de césped sobre un área para recoger fotografía si se utiliza un avión u otra acción se utiliza otro vehículo autónomo. Para realizar la planificación autónoma se escoge en el menú contextual la opción *Polygon* y luego de ello se procede a dibujar un cuadro alrededor del área que se desea asignar, luego se selecciona la opción del menú contextual *Auto Wp*, y seguido a esto la opción *Grid*, luego de escogido esta opción se procede a seguir las instrucciones pertinentes que proporciona el Mission Planner. Luego de completada las instrucciones el planificador de misiones generará una misión automática como la que se muestra en la Figura 2.6 ([Mission Planner Home, 2017](#)).

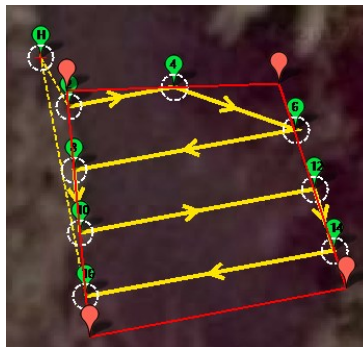


Figura 2.6. Planificación de misión autónoma.

Luego de planificada la misión en la parte del registro de los *Waypoints* se proporciona una lista detallada de los comandos apropiados para cada uno de los vehículos. Estos incluyen órdenes de navegación para dirigirse a *Waypoints* o localizarlos en la proximidad. Estos comandos junto con los *Waypoints* quedan registrados en la parte inferior de esta sección como se muestra en la Figura 2.7.

Comandos	Radio WP	Radio Perder	Tiempo	por	Default	Absolute	Verify Height	Agregar	Abajo	Alt	Wam	Spline	Borrar	Amb	Abajo	Grad %	Angle	Dist	AZ
3 DO_SET_CAM_TRIGG_DIST	4.61999988555908	0	0	0	0	0	0	0	0	0	0	0	X			0	0	0	0
4 WAYPOINT	0	0	0	0	0	0	0	22.4342297	-79.8958876	10	0	0	X			0.0	0.0	8.5	80
5 DO_SET_CAM_TRIGG_DIST	0	0	0	0	0	0	0	0	0	0	0	0	X			0	0	0	0
6 WAYPOINT	0	0	0	0	0	0	0	22.4342001	-79.8957961	10	0	0	X			0.0	0.0	10.0	109
7 DO_SET_CAM_TRIGG_DIST	4.61999988555908	0	0	0	0	0	0	0	0	0	0	0	X			0	0	0	0
8 WAYPOINT	0	0	0	0	0	0	0	22.4341714	-79.8959642	10	0	0	X			0.0	0.0	17.6	260

Figura 2.7. Registro de Waypoints y Comandos

Para completar la planificación de la misión es necesario presionar el botón “Escribir Waypoints” en el área de *Acciones* para así poder mandar la misión para la sección principal.

2.5 Sistema de simulación Software In The Loop (SITL)

El simulador *SITL* permite la ejecución de *Plane*, *Rover* o *Copter* sin la necesidad de estar conectado con el hardware. Es una compilación del código del piloto automático que usa un compilador C++ normal.

Cuando un ingeniero o profesional, pretende realizar el diseño de un sistema físico; sea automóvil, avión, etc., es difícil que se reúnan todos los componentes (llámese motor, acelerómetro, sensor de cualquier tipo, material de construcción) necesarios para llevar a cabo el diseño de dicho sistema o que existan las condiciones necesarias para la realización de las pruebas reales.

Software In The Loop, básicamente es una validación computacional que se realiza mediante un modelo matemático o de comportamiento de un componente físico o un fenómeno, para probar su comportamiento y la adecuación que éste podría tener para la realización del diseño de algún experimento y luego probarlo posteriormente en algo físico; tomando en cuenta que las características que se modelan, deben ser lo más cercanas a la realidad posible para que así se pueda llevar a la realidad estos modelos (de la Rosa, 2016).

La idea de la utilización de la técnica SITL en la presente tesis, es debido a que no se cuenta con el área lo suficientemente grande y adecuada en la universidad para realizar las pruebas al barco, pero si se cuenta con los componentes físicos y con el sistema de control de rumbo para vehículos autónomos terrestres, el cual será utilizado para el control del vehículo acuático.

Para ejecutar SITL en el Mission Planner primeramente debe estar conectado el software Mission Planner a *Internet* para así poder descargar esta opción, luego de conectado el software a internet se debe ir a la sección *Simulation*, y dentro de esta sección presionar el botón “*Start SITL*”, luego de presionado el botón se abrirá una nueva ventana como la que se muestra en la Figura 2.8 y luego de esto se debe escoger el modelo del vehículo que se esté usando, en nuestro caso se escoge el del *Rover*, puesto que los parámetros de configuración y los controles que utilizan en este *firmware* es similar a los que se utilizan para USV. Luego de escogido el tipo de vehículo que se usará en la simulación, el Mission Planner descargara de internet el simulador *SITL* y el mismo estará listo para la realización de las simulaciones mediante la técnica SITL.

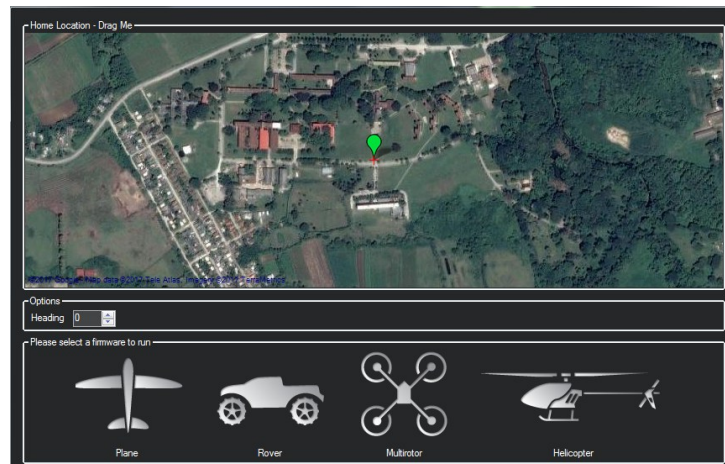


Figura 2.8. Ventana auxiliar para ejecutar Software In The Loop.

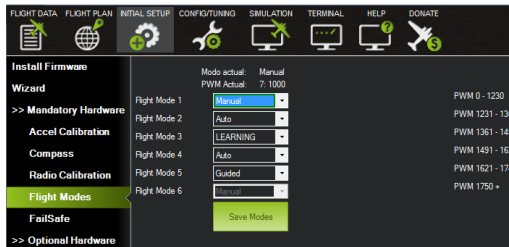
2.6 Modos de trabajo del Mission Planner para vehículos de superficie

Los modos de control del piloto automático se controlan a través de un conmutador del transmisor de radio, a través de comandos de misiones o usando comandos del Mission Planner.

Los modos de trabajo de Mission Planner (Cierzo, 2013) soportados para el rover son:

- MANUAL Mode
- LEARNING Mode
- AUTO Mode
- STEERING Mode
- HOLD Mode
- GUIDED Mode
- RTL

Estos modos de trabajo se pueden encontrar en la sección “*Initial Setup*”, luego en el apartado “*Mandatory Hardware*” y luego en la opción “*Flight Mode*”, como se puede ver en la Figura 2.9 (a). Además de encontrados en la sección “*Initial Setup*”, también se puede encontrar los modos de trabajo en la sección “*Flight Data*”, Luego en el “Área de Control”, y por último en la pestaña acciones como se puede ver en la Figura 2.9 (b).



(a) Sección de configuración



(b) Interfaz Principal

Figura 2.9. Modos de trabajo de Mission Planner.

El modo manual consiste en la conducción del USV con el transmisor RC o radiocontrol, mediante el mismo se puede realizar todas las acciones pertinentes que brinda la emisora de radio que se esté utilizando. El modo de aprendizaje o *Learning* consiste en la conducción manual del vehículo para que mediante la emisora de radio guardar los *waypoints*. Una vez que la misión haya terminado se puede cambiar al modo *Auto* y el *rover* debe replegarse por todos los *waypoints* que grabó. El modo automático consiste en seguir de forma automática la ruta generada por el Mission Planner mediante la planificación de misiones o mediante los *Waypoints* guardados en el modo *Learning*. El modo “*Steering*” o dirección es un poco como el modo de *Learning*, excepto que los controles utilizan el mismo código de dirección, aceleración y de evitación de obstáculos como el modo automático. El modo “*Hold*” solamente se activará o será de uso cuando exista algo que bloquee al vehículo o que falle en la misión. El modo “*Guide*” o guía no requiere de un conmutador dedicado, así que no se recomienda seleccionarlo como un modo de vuelo. Con el modo de trabajo “*Guide*” activado se puede guiar el vehículo a cualquier parte del mapa, para esto es necesario presionar el clic derecho y luego en el menú contextual que aparece escoger la acción “*Fly Here*” o “*Volar aquí*”.

2.7 Reportes e Informes de misiones

El reporte e informe de misiones es un tema importante, puesto que mediante el registro de las misiones se pueden diagnosticar y verificar los problemas que pudieron haber existido en una misión. El registro de la misión lo realiza el software Mission Planner de forma automática luego de ejecutado una misión.

El registro de las misiones se puede realizar de dos maneras, aunque los dos métodos realizan el registro de las misiones de diferentes maneras pero los datos que registran son muy similares. Una de las maneras es utilizando la memoria interna del *Pixhawk* mediante “*Dataflash log*”, los registros de este método se realizan de forma automática para el USV poco después de la ejecución de una misión. Otra forma de registrar los datos es mediante “*Telemetry logs*”, donde el registro de la misión se realiza mediante el propio Mission Planner, realizando el guardado automático en la PC.

En el caso de estudio realizado, primeramente se utilizará la segunda variante, puesto que las pruebas que se realizaran se efectuaran mediante el simulador del Mission Planner “*Software In The Loop*” y luego se procederá a realizar las pruebas reales y se utilizará entonces el registro de las misiones mediante “*Dataflash log*”.

Los valores de los parámetros del USV que se guardarán se pueden ver en ([Cierzo, 2013](#)), en donde se muestra una lista detallada de todos los parámetros que se utilizan en una misión con un USV.

2.8 Plataforma Móvil

El controlador Pixhawk dentro de sus prestaciones permite dos tipos de navegación: Autónoma guiada a través de la plataforma Mission Planner o bien Manual mediante un control remoto (a radiocontrol). A continuación, se definen las dos maneras de navegación que presenta este dispositivo.

2.8.1 Navegación Manual

Para ejecutar el pilotaje manual del dispositivo se requiere una emisora que proporcione dos canales, uno para la potencia del motor y el otro para la dirección del timón y deberá tener un enlace no superior al de la telemetría (500 m) para así

no perder la comunicación. Por otra parte, las bandas de frecuencias a la que trabaje deberán ser libres para así poder cumplir con los estándares (Rabuffetti, 2016).

Para efectuar el pilotaje manual se escogió la emisora *Devo F7*, la cual se muestra en la Figura 2.10, y la misma está diseñada para efectuar el pilotaje de vehículos autónomos acuáticos y aéreos. Algunas de las características que presenta esta emisora es que utiliza un encoder con 7 canales, trabaja con una banda de frecuencia de 2.4 GHz y FPV 5.8 de tiempo real de transmisión. Posee un conector USB para así poder cargar versiones más actualizadas de *firmware*, posee una pantalla LCD, su potencia de salida es de 100mW (DEVO F7, 2017).



Figura 2.10. Emisora DEVO F7

Para efectuar el pilotaje manual, primeramente, se debe instalar el receptor del radiocontrol en el vehículo autónomo (Figura 1.4 (d)). Este receptor cuenta con 7 canales, una sensibilidad de 105 dbm, peso 11.6 g, la dimensión del dispositivo es de 43x28x16 mm (DEVO F7, 2017).

2.8.2 Navegación Autónoma

Para efectuar la navegación autónoma se realiza a través del planificador de misiones Mission Planner. Antes de realizar la una misión, al igual que en el modo manual, se deberá armar el motor del vehículo manteniendo pulsado el pulsador de seguridad que presenta en su diseño el vehículo en la parte superior o dirigiéndose al menú principal en la sección “*Flight Data*”, luego en la subsección “*Acciones*” y seleccionar el botón “*Arm/Disarm*” para completar dicha acción (Rabuffetti, 2016). Luego de armado el dispositivo se procede a planificar la misión siguiendo las instrucciones para la planificación de misión planteada anteriormente.

2.8.3 Hardware del vehículo

Para la realización del proyecto el GARP adquirió el modelo del barco de pequeño porte *Krick Félix*, y el equipamiento sensorial y electrónico que presenta la embarcación es de bajo costo.

Los componentes electrónicos con que se cuenta para el montaje del Hardware de la aplicación son:

- Un microcontrolador *Pixhawk*.
- Un módulo GPS + Compasss
- Un módulo de Radio
- Un módulo de Telemetría
- Un Codificador PPM
- Un Motor DC de 7.4 Volts
- Un servo motor digital
- Una batería

Para establecer la conexión de todos los componentes con el microcontrolador se tuvieron en cuenta los modelos existentes en las aplicaciones mencionadas en el Capítulo 1. Además para la correcta instalación de los componentes se tuvo como guía a (*"Mission Planner Overview", 2017*). En la Figura 2.11 se muestra con detalles el diagrama de las conexiones que se establecieron para el montaje del *Hardware* del vehículo.

Como se puede ver en la Figura 2.11, el módulo de GPS + Compass se conectó directamente al puerto predeterminado del *Pixhawk* para el GPS. El módulo de Telemetría se conectó a uno de los puertos predeterminados para la telemetría, dicho puerto es *Telem 1*. Para la alimentación del *Pixhawk* y del motor con la batería se utiliza un módulo de potencia y alimentación, el cual suministra energía limpia a partir de una batería Li-Po, este módulo también permite realizar mediciones de voltajes, y gracias al sensor amperimétrico que posee, controlar el consumo en tiempo real. Del módulo de potencia se obtiene 5.3v, dependiendo el modelo de dicho módulo (*"DronModelling", 2017*). Este módulo de potencia le suministra directamente el mismo voltaje que demanda la batería al motor mediante una conexión con el Driver del mismo, puesto que este Driver del motor realiza algunas de las funciones de un microordenador de control mediante la utilización de dos circuitos integrados, uno para el manejo de la corriente y el otro para el control de las secuencias de conmutación y el control de la corriente de bobinado del motor (*Peters, 2013*).

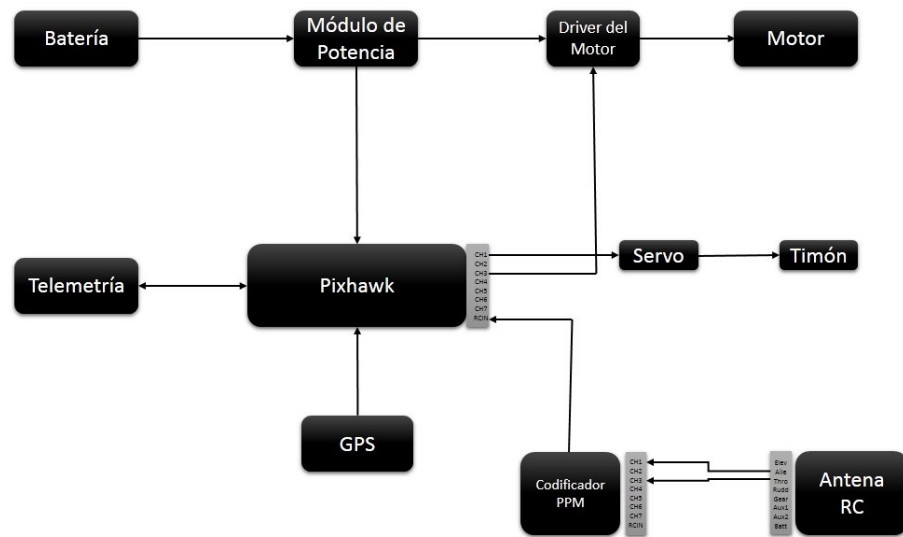


Figura 2.11. Diagrama de conexiones

El módulo de Radio control o Antena RC como viene señalizada, fue conectada para establecer el control manual del vehículo, y el mismo fue acoplado a un decodificador *PPM*. Este módulo fue conectado al decodificador para realizar la conversión de señal continua a discreta, para así poder enviarle al *Pixhawk* una señal de entrada en la forma *PWM* y dicha señal es la que se envía al servo, puesto que dicha señal es la que entienden los servos u otros dispositivos usados para el control por Radiocontrol (RC). Para entender mejor el principio de funcionamiento del decodificador se puede acceder a ([“E-Radiocontrol”, 2009](#)) en la sección *PPM* y si se desea conocer sobre la señal *PWM* se accede a la sección *PWM*. El Servo que mueve al timón del vehículo se conecta directamente al *Pin* del *Pixhawk* correspondiente para el servo, en nuestro caso se utilizó el CH1 para el servo y el CH3 para el motor. El vehículo diseñado se puede observar en el [Anexo D.2](#)

2.9 Algoritmo de Guiado L1 Control

La lógica de guiado L1 Control se basa en escoger un punto de referencia en la trayectoria deseada y generar un comando de aceleración lateral usando el punto de referencia escogido. El punto de referencia está en la trayectoria deseada a una distancia L1 hacia delante del vehículo (Figura 2.10) ([L. E. Hernández, 2016](#)).

La aceleración lateral no es más que la aceleración centrípeta necesaria para seguir el camino circular de radio R definido por la posición del punto de referencia, la posición del vehículo y tangencialmente al vector de velocidad del vehículo (Figura 2.11)(L. E. Hernández, 2016).

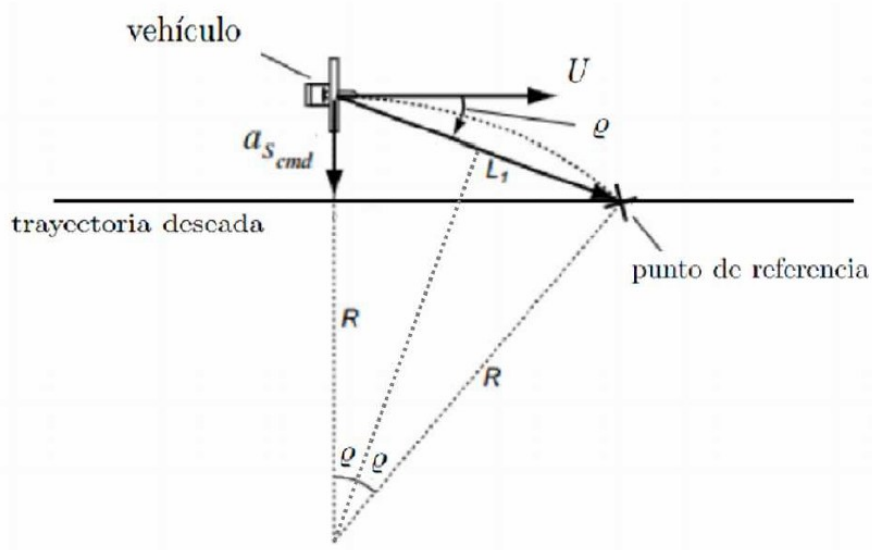


Figura 2.12. Algoritmo de guiado L1

El valor de $L1$ se calcula mediante el método de ajuste que presenta el *firmware* “ardupilot” a partir de dos parámetros ajustables por el operador, los cuales son el “Período” y el “Damping” o coeficiente de amortiguamiento como se aprecia en la Figura 2.12. Variando estos parámetros se puede mejorar o empeorar la respuesta de la embarcación, puesto que dichos parámetros están directamente relacionados con el controlador de rumbo.

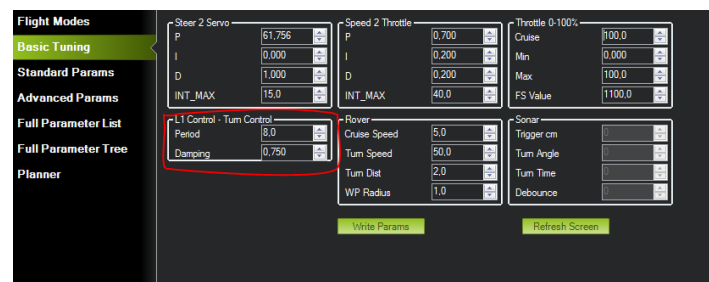


Figura 2.13. Configuraciones básicas del L1 Control

2.9.1 Controlador de Rumbo

Para establecer el control del vehículo se implementa en el *firmware* la biblioteca APM_CONTROL, donde en la misma se ejecutan diferentes controles, pero los más

importantes para la aplicación son el de rumbo y el de velocidad, ambos controles se encuentran en la Librería *AP_SteerController*. En el modo automático la velocidad del vehículo se mantiene constantes pues la misma se fija en la interfaz gráfica del Mission Planner como parámetro principal.

Un controlador de rumbo, o autopiloto de rumbo, como también se le suele llamar, tiene el objetivo de asegurar un adecuado desempeño del vehículo durante la ejecución de maniobras donde ocurren cambios en la dirección del mismo (A. Hernández, 2014). Para establecer el control del vehículo en el diseño del software Mission Planner se escoge un controlador PD, debido al estudio del código para el control de rumbo que se utiliza en dicha aplicación, ver [Anexo C](#), en la misma se utiliza un controlador PD para realizar este tipo de función.

En la configuración del software Mission Planner por defecto los valores del controlador PID de rumbo los proporciona dicho software y esto trae como consecuencia que el vehículo no realice de forma estable la ejecución de la misión y por ende para esto en esta investigación se procede a realizar el ajuste de los parámetros de este controlador mediante la función de transferencia que presenta el vehículo (ecuación 2.1) y se valida el diseño del mismo con pruebas experimentales, teniendo en cuenta el comportamiento del USV en cada una de las modificaciones de los parámetros del lazo de control.

$$\frac{\psi(s)s}{\delta T(s)} = \frac{0.0437}{0.1945s+1} \quad (2.1)$$

Donde ψ es el ángulo de rumbo y δT son las variaciones en el timón horizontal (Balanza, 2017).

2.10 Conclusiones parciales

El planificador de tareas Mission Planner brinda en su estructura muchas funciones para el seguimiento y guiado de vehículos autónomos, muchas de estas características son de gran ayuda para realizar el guiado de un vehículo autónomo sin la necesidad de controlarlo remotamente.

En el código fuente del *firmware ArduRover* existe también lazos de control de rumbo y de velocidad los cuales son los encargados del comportamiento del

vehículo y es de gran vitalidad conocer su funcionamiento para así poder realizar los cambios necesario para que cuando se realice la planificación de la misión mediante el software Mission Planner el vehículo ejecute la misión de forma correcta. También es necesario conocer los principales parámetros de configuración del rover para la navegación.

Luego de conocido todo lo anterior mencionado es necesario realizar la adecuación del Mission Planner para que el vehículo ejecute las misiones de forma estable.

CAPÍTULO 3. PROPUESTA DE ADECUACIÓN

3.1 Introducción

En el presente capítulo se aborda sobre las pruebas realizadas al vehículo mediante la planificación de la misión deseada por el software Mission Planner, para el seguimiento de la misma de forma automática. También se procede a realizar el ajuste del controlador para que el vehículo ejecute las misiones estables, puesto que con los valores predeterminados al ejecutar una misión el dispositivo se encuentra de forma inestable, pero no sigue la ruta de forma adecuada, el mismo posee un error que se puede corregir mediante un adecuado ajuste del regulador. Además, se realizará un análisis económico al final del capítulo.

3.2 Ajuste del Controlador de rumbo

La ecuación 2.1 describe de manera simplificada la dinámica lateral del *Krick Félix*. A partir de esta función de transferencia se ajusta un controlador PD. El ajuste se realiza utilizando la técnica de ubicación de polos y ceros en la herramienta *sisotool*. Para realizar el ajuste del controlador también se tuvo en cuenta el código fuente de ArduRover.

Luego de un estudio del código fuente ArduRover, ver [Anexo C](#), se procedió a llevar el código fuente a un diagrama en bloques en el software Matlab, el cual se puede ver en el [Anexo A.1](#), luego de procesado el código se llegó a la conclusión que el controlador de rumbo que se debe utilizar es un P-D.

En la Figura 3.1 se representa el lugar geométrico de las raíces y a partir de allí se selecciona los valores de $K_p=10$ y $K_d=1$.

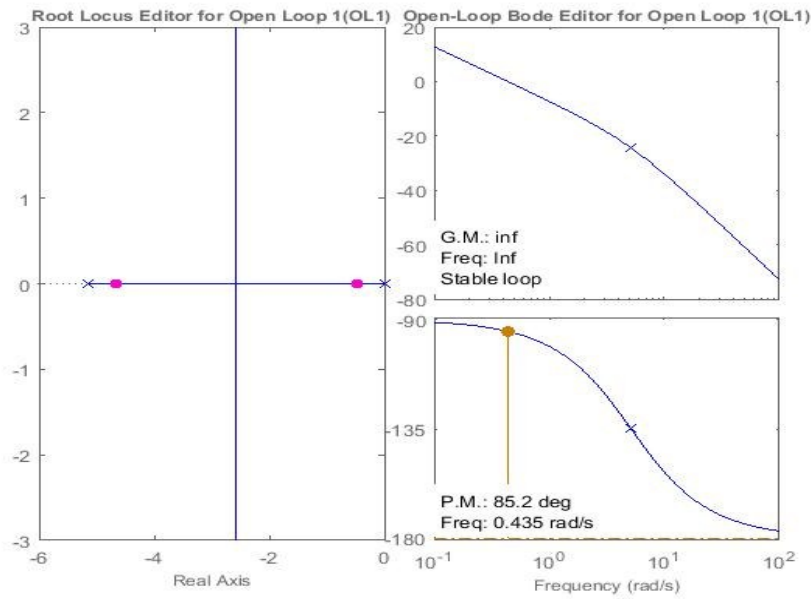


Figura 3.1. Lugar Geométrico de las raíces

En la Figura 3.2 se muestra la respuesta de frecuencia simulada con el controlador de rumbo ajustado. Como se puede ver en la figura el tiempo de subida es de 4.6 s debido a que en el *software Mission Planner* la *velocidad predeterminada que posee el vehículo es de 2 m/s*, lo cual el vehículo realiza la misiones de forma lenta y por ende el tiempo de establecimiento debe estar alto.

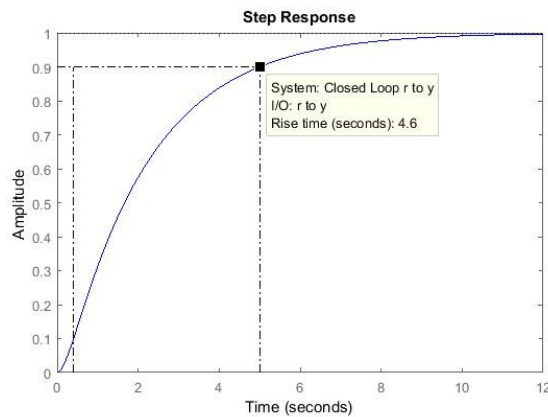


Figura 3.2. Respuesta de frecuencia

3.3 Pruebas experimentales

Para la realización de las pruebas experimentales, primeramente, se utilizó la opción que brinda el Mission Planner Software In The Loop (SITL), donde con la misma se procede a la realización de experimentos con el vehículo sin la necesidad de estar físicamente conectado con el mismo.

3.3.1 Simulaciones

Antes de la puesta en marcha o del experimento real del software Mission Planner con el Krick Félix, se realizaron simulaciones para así poder acondicionar el software para el correcto funcionamiento a la hora de la ejecución de una misión con un vehículo autónomo acuático de superficie.

La primera simulación realizada fue con los parámetros estándares que proporciona el software para el control de rumbo, dichos parámetros se pueden observar en la Tabla 3.1.

Tabla 3.1. Parámetros del Lazo de control de rumbo

Controlador	
KP	1.8
KI	0.2
KD	0.005
INT_MAX	180
Control de giro	
Periodo	8
Damping	0.75

En la Figura 3.3 se pueden ver los resultados obtenidos en dicha simulación. En dicha figura se puede observar que el dispositivo sigue la trayectoria de forma

correcta, aunque no de forma directa, pues como se ve cuando el dispositivo se encuentra dentro de la segunda recta, al realizar el giro no sigue satisfactoriamente por la ruta trazada.

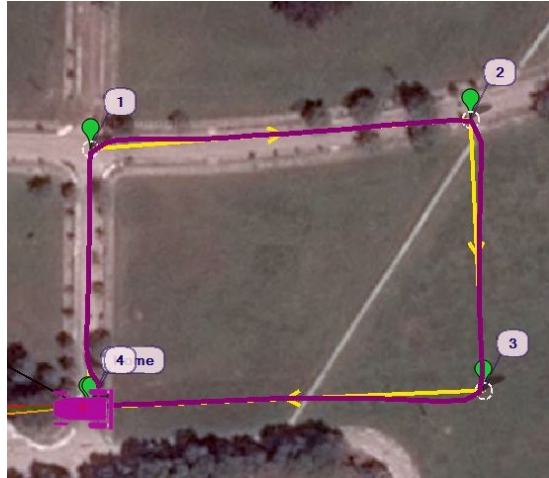


Figura 3.3. Simulación con parámetros estándares.

Para tratar de compensar este pequeño detalle se procede a eliminar la ganancia KI, luego del estudio código fuente ArduRover, se procedió a llevar ese código a diagramas en bloque en el Matlab, Ver [Anexo A.1](#) y el controlador que se utiliza en el mismo es un PD.

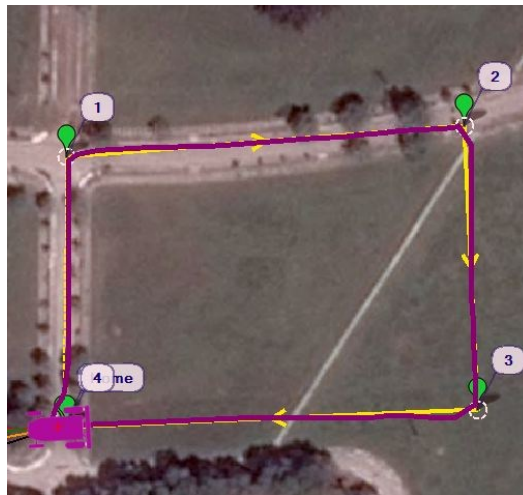


Figura 3.4. Simulación sin KI.

La Figura 3.4 muestra los resultados obtenidos luego de ajustado el controlador. Como se puede observar la respuesta del vehículo a seguir la trayectoria mejora,

puesto que cuando el vehículo realiza el giro en los *waypoints* procede a seguir la trayectoria correctamente y no con el pequeño desfase luego del giro como antes lo realizaba. Luego de realizado los cambios la respuesta del vehículo mejoró por lo tanto se procede a realizar el ajuste del controlador PD ajustado en el *sisotool*.

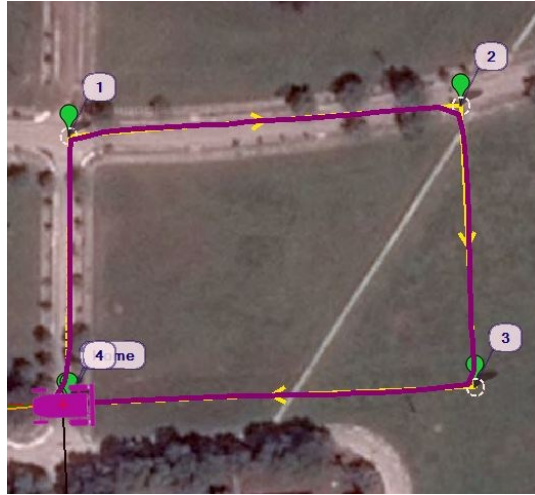


Figura 3.5. Simulación con el controlador diseñado.

Los resultados obtenidos luego de realizada la modificación se muestran en la Figura 3.5, como se puede observar ya el dispositivo mejoró el comportamiento del vehículo en las curvas principalmente. También como se puede observar el dispositivo en las trayectorias rectas se comporta adecuadamente siguiendo una trayectoria recta, no como antes lo realizaba con los parámetros estándares lo cual era el problema de la inestabilidad del vehículo.

Para comprobar el correcto funcionamiento del dispositivo se realizaron otras pruebas de simulaciones antes de la realización de la prueba real con el dispositivo, una de estas pruebas fue modificar el radio de los waypoints para ver el comportamiento del vehículo.

En la Figura 3.6 se muestra los resultados obtenidos ante el cambio del radio de los *waypoints*. Como se puede ver el vehículo realiza el giro menos brusco, puesto que este parámetro es el encargado de proporcionar el área alrededor del *waypoint* donde el vehículo puede ejecutar el giro o alguna otra acción, por ende, a medida que se aumente este parámetro el vehículo va a realizar el giro más suave y a medida que se disminuya el giro se realizara más brusco.

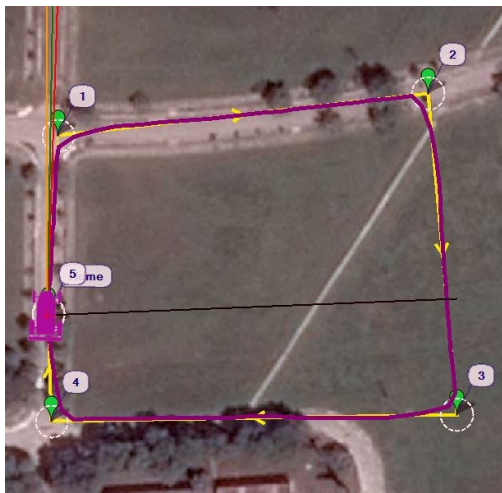


Figura 3.6. Simulación aumentando el radio de los waypoints.

Otra prueba simulada fue la modificación de los parámetros del *L1 Control*, puesto que dichos parámetros, como antes se ha mencionado, son los encargados de proporcionar el ángulo de giro del vehículo. Con la modificación del *Damping* o el *Periodo* se puede variar el ángulo de giro del vehículo, y por ende con un aumento o disminución de cualquiera de estos dos parámetros, podría o mejorar o empeorar el giro del vehículo.

3.3.2 Test de Validación

Para la comprobación del correcto funcionamiento del Mission Planner en la planificación de misiones para vehículos autónomos acuáticos de superficie, se realizó la prueba con el vehículo en una laguna cercana a los linderos de la universidad ([Anexo E.1](#)). Con esta prueba se comprobó el estado del vehículo al ejecutar misiones. Antes de la ejecución de las misiones primeramente se procedió a configurar los parámetros fundamentales y luego se tuvo que invertir los controles de giros en el Mission Planner, puesto que con los valores normales el barco realizaba los giros en sentido contrario ya que en el rover el giro se realiza por la parte frontal del vehículo y en esta aplicación el servo se encuentra en la parte posterior, y por ende es necesario invertir los controles de giro. Para invertir los controles de giro se debe ir a la sección *Initial Setup* y luego en la subsección *Radio Calibration*, encontrados en la subsección *Radio Calibration* se procede a invertir el control denominado *Roll*, como se puede ver en el [Anexo B.2](#).

La primera puesta en marcha del USV, se realizó con los parámetros estándares que proporciona el Mission Planner en el lazo de control de rumbo ([Anexo E.2](#)). Con esta prueba se comprobó que el vehículo no se comportaba correctamente a la hora de la realización de los giros y del seguimiento de la ruta deseada. Para contrarrestar estos resultados se procedió a modificar los valores de los parámetros del lazo de control, con el objetivo de mejorar el estado del barco y eliminar el balanceo con que se encontraba a la hora de la ejecución de la misión. Luego de varias modificaciones se llegó a la conclusión de que sin el valor de ganancia K_I o el valor integral, el barco realizaba los giros menos brusco, además, mientras más pequeño se hacía el valor de la acción proporcional, el giro se realizaba más suave y el vehículo navegaba con mayor estabilidad.

Luego de realizados los experimentos reales se procede a comprobar los resultados obtenidos mediante el software Matlab. En el software Matlab se plotearon los valores de *GPS*, *Giro en el eje x* y los valores de L1 Control, guardados mediante la ejecución de las misiones.

En la Figura 3.7 se muestra la ruta seguida por el barco en la primera misión, en dicha misión, como se puede observar, el vehículo se trasladaba de forma inestable, puesto que la trayectoria que estaba realizando la estaba haciendo con un pequeño balanceo en la dirección del mismo. Para contrarrestar este balanceo se procedió a modificar los parámetros del lazo de control y la respuesta del barco mejoró, como se puede ver en la Figura 3.8, la trayectoria que siguió el barco fue sin balanceo aunque aún no ejecutaba los giros de una forma Óptima.

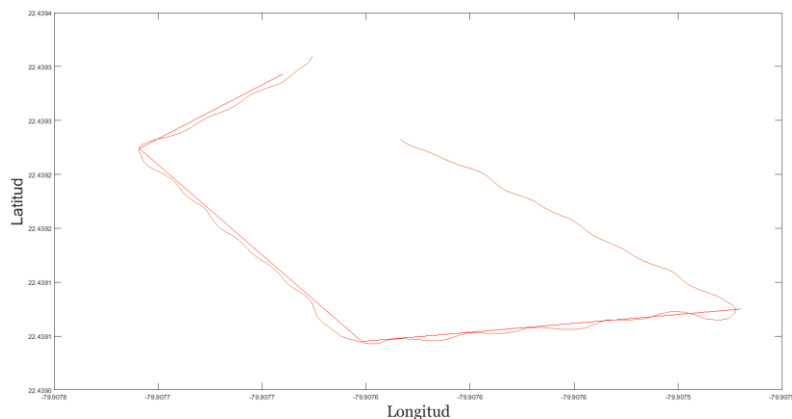


Figura 3.7. Trayectoria programada con valores estándares.

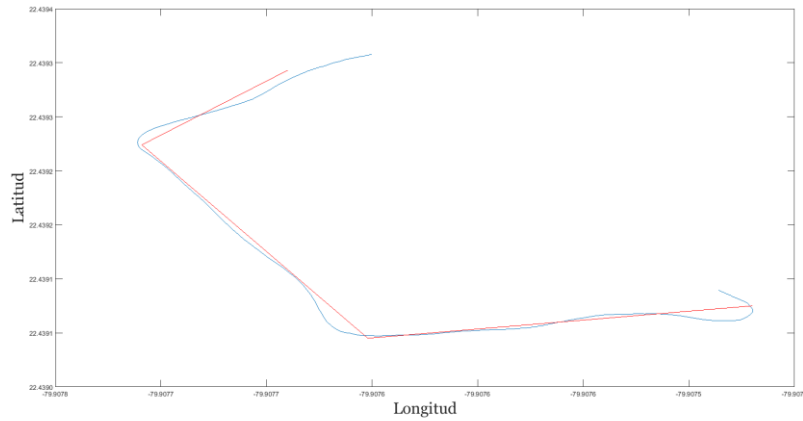


Figura 3.8. Trayectoria programada con valores modificados.

Otro parámetro que se tuvo en cuenta para el análisis del comportamiento del barco, fue el valor del giro en el eje X, en la Figura 3.9 se muestran los valores registrados del giro en X de la misión ejecutada con los valores estándares. En dicha figura se puede observar que el comportamiento de los valores que se registraban de los giros que realizaba el vehículo, eran brusco además de que en dicha misión el barco siempre se encontraba en constantes movimiento en dicho eje. En la Figura 3.10 se puede observar como mejora el comportamiento de los giros luego de realizadas la modificación al vehículo, puesto que el mismo se encuentra más estable y no realiza movimientos erróneos, según se puede observar. Los picos que se muestran en ambas figuras, son los momentos en que el *USV* realizaba los giros en los *waypoints*.

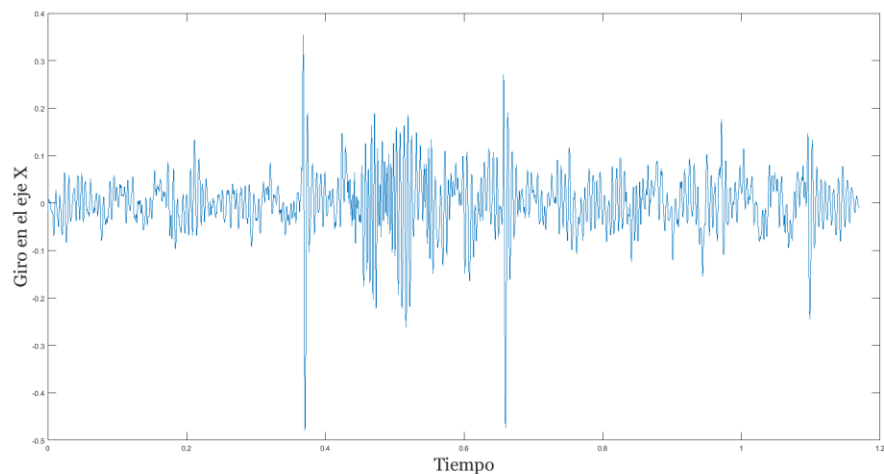


Figura 3.9. Respuesta del giro en X con valores estándares.

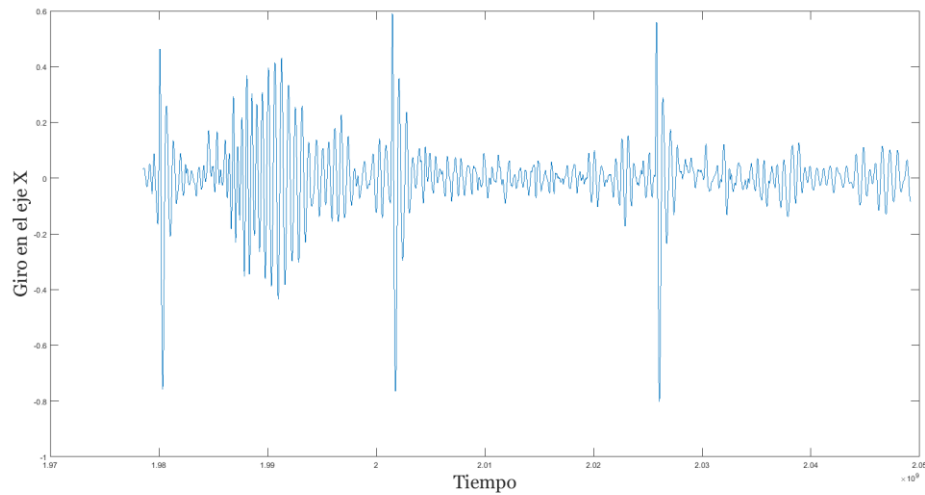


Figura 3.10. Respuesta del giro en X con valores modificados.

Uno de los parámetros elementales que están presentes en el control de rumbo del USV, son los valores de L1 Control, dichos valores están encargados de proporcionar el valor deseado de giro como se muestra en las Figuras.3.8 en color azul, y el valor registrado por el vehículo como el que se muestra en color naranja. Realizando el ploteo de este parámetro se puede ver si el vehículo está ejecutando bien la misión. Como se puede ver en la Figura 3.11, el vehículo en la primera misión ejecutada con los valores estándares no se encontraba realizando correctamente la misión debido a un problema de configuración del lazo externo de control (L1 Control), pero cuando se le realizaron las modificaciones a los valores del lazo de control, se puede ver en la Figura 3.12 el mejoramiento en el control de rumbo del vehículo.

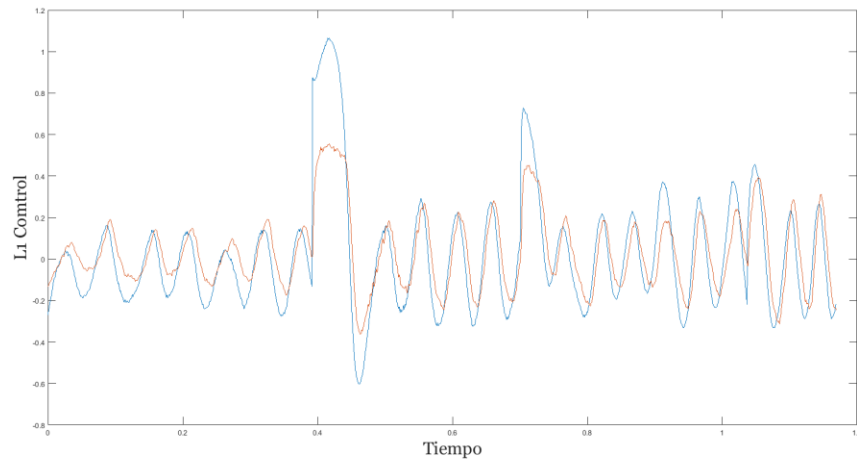


Figura 3.11. Respuesta de los valores de L1 Control deseado (Azul) y registrado (Naranja) con parámetros estándares.

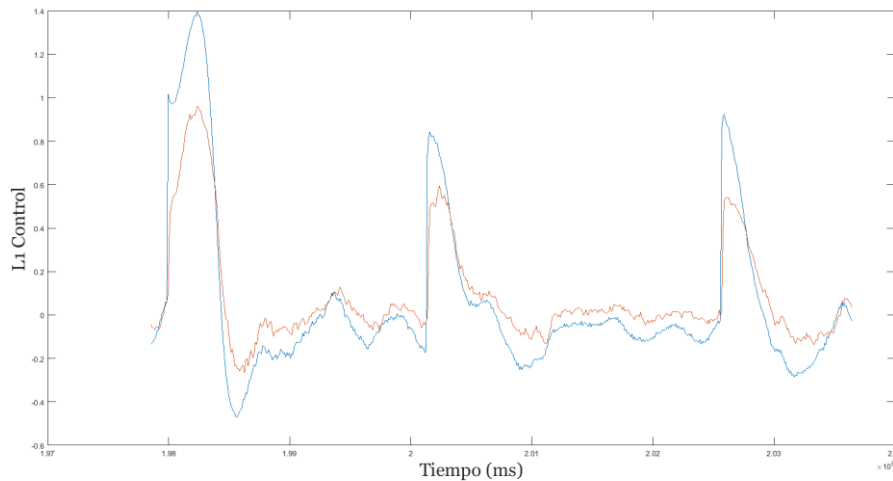


Figura 3.12. Respuesta de los valores de L1 Control deseado (Azul) y registrado (Naranja) con parámetros modificados.

3.4 Análisis Económico

El rango de aplicaciones que utilizan el planificador de misiones Mission Planner ha tenido un aumento vertiginoso durante los últimos años. Producto a esto, varias compañías a nivel mundial e instituciones han hecho de este software un producto fiable y comerciable. Además, el software Mission Planner es un software libre y de

muy fácil adquisición en internet. También su código fuente se puede adquirir en internet sin necesidad de realizar ningún gasto económico, lo cual es de gran ventaja ya que con los especialistas preparados se pueden realizar modificación al software teniendo en cuenta las necesidades que estén presentes donde se va a utilizar dicho software.

Todo el equipamiento electrónico y sensorial usado para esta aplicación fue adquirido a bajo costo en el mercado mundial. El microcontrolador *Pixhawk* utilizado para ejecutar la supervisión y control del USV, en el mercado libre, el precio de los mismos oscila entre 100 y 500 dólares americanos (USD). El *GARP* adquirió a un costo de *US \$ 105.20* al barco de pequeño porte *Krick Félix*. El *Módulo GPS + Compass* fue adquirido a un precio de 20 USD, el *Módulo de telemetría* fue adquirido a un precio de 30 USD, el *Módulo de Radio* fue adquirido a un precio de 23 USD, el *Codificador PPM* fue adquirido a un precio de 17 USD, el *Motor DC* que se utiliza se adquirió a un precio de 5 USD y el *Servo Motor Digital* fue adquirido a un precio de 45 USD.

Otro tema importante a tratar, y no menos importante, es el relacionado con el impacto medioambiental que posee esta aplicación, pues con vehículo autónomos acuáticos de superficie capaces de ejecutar misiones de forma autónoma, se incrementaría las posibilidades de la realización de investigaciones donde el ser humano, por las condiciones ambientales no pudiese acceder, y así poder investigar el ecosistema marino. Además de que esto traería también consigo de que mediante estas investigaciones se pudiese proteger la flora y fauna donde se esté realizando una investigación con este tipo de aplicación.

Finalmente se podría decir que antes de la adecuación del software Mission Planner para el correcto funcionamiento del seguimiento de vehículos autónomos acuáticos de superficie, se procedía a realizar este tipo de acciones mediante control remoto, lo cual para la realización de algunas pruebas específicas no era muy eficiente. Por ende, con la adecuación del software Mission Planner para la realización de este tipo de acción optimizaría la eficiencia del seguimiento y control a vehículos autónomos acuáticos de superficie sin la necesidad de controlarlo remotamente.

3.5 Conclusiones parciales

Con el ajuste del controlador P-D se logra estabilizar el comportamiento del vehículo para el seguimiento de caminos trazados por el software Mission Planner. El ajuste del mismo se realiza teniendo en cuenta las condiciones y los parámetros de navegación del lazo de control de rumbo que proporciona el Mission Planner.

Mediante la función SITL que brinda el software Mission Planner se pueden realizar las pruebas a vehículos sin la necesidad de estar conectado con el mismo físicamente. Esto es de gran ayuda para corregir errores de navegación antes de realizar las pruebas reales con el vehículo. Además, se puede realizar el ajuste de controladores diseñados para estabilizar el comportamiento del vehículo.

Las pruebas de validaciones son muy importantes en cualquier investigación, puesto que con las mismas se pueden comprobar los problemas que existen. En el caso de la investigación en curso, con el test de validación se pudo comprobar cuáles eran los problemas que existían, los cuales provocaban el mal comportamiento del USV a la hora de ejecutar alguna misión.

CONCLUSIONES

Como resultado final de esta investigación, se logra mediante el software Mission Planner modificar los parámetros adecuados para el seguimiento y control de USV de superficies, ajustando el controlador y los parámetros del algoritmo de guiado L1 Control en el lazo de control de rumbo, además configurando los principales parámetros de navegación del vehículo. A partir de ello se arriba a las siguientes conclusiones:

- Considerando la revisión realizada en la literatura especializada sobre el seguimiento y control de vehículos autónomos acuáticos de superficies mediante el software Mission Planner, se determina que este planificador de tareas es uno de los más fiables que existe hoy en día para realizar este tipo de acciones, además de ser un software libre.
- La amplia gama de prestaciones que proporciona el Mission Planner hacen de este software una opción fiable para el seguimiento de vehículos autónomos, además de la existencia de los parámetros fundamentales para el correcto desempeño de las acciones, son de gran vitalidad a la hora de la ejecución de alguna acción pertinente.
- Para el correcto comportamiento del USV fue necesario ajustar los principales parámetros que se proporcionan en el Capítulo 2 y además los parámetros del Algoritmo de guiado L1 Control también fueron necesarios ajustarlos puesto que los mismo están relacionados directamente con el comportamiento del USV.

- Con las configuraciones realizadas al Mission Planner se logró estabilizar el comportamiento del USV Krick Félix en la ejecución de las misiones mediante el software Mission Planner.

RECOMENDACIONES

Como continuación de este trabajo se procede a realizar las siguientes recomendaciones:

- Implementación del software Mission Planner para vehículos autónomos subacuáticos teniendo en cuenta las principales características que brinda dicho software.

REFERENCIAS BIBLIOGRÁFICAS

A.G, U. (2017). *u-blox 8 / u-blox M8 Receiver Description*. (Vol. 15–20.30).

Recuperado a partir de www.u-blox.com

Balanza, C. de la C. (2017). *Modelado dinámico del barco de pequeño porte Krick*

Felix. (Innovación Tecnológica). Universidad Central “Marta Abreu” de Las

Villas, Santa Clara. Recuperado a partir de <http://dspace.uclv.edu.cu/>

Blidberg, D. R. (2001). The development of autonomous underwater vehicles (AUV);

a brief summary. En *IEEE ICRA* (Vol. 4).

Caccia, M., Bibuli, M., Bono, R., & Bruzzone, G. (2008). Basic navigation, guidance

and control of an unmanned surface vehicle. *Autonomous Robots*, 25(4),

349–365.

Cierzo, C. (2013). *APM_Coche* (Vol. I). España. Recuperado a partir de

<http://www.diydrones.com>

Cornwall Model Boats. (2015). Recuperado el 17 de febrero de 2017, a partir de

<http://www.cornwallmodelboats.co.uk/>

de la Rosa, A. I. (2016). *Simulación en Técnicas de Software In The Loop del*

Sistema de Monitoreo de presión en Cabina de un Avión. Universidad

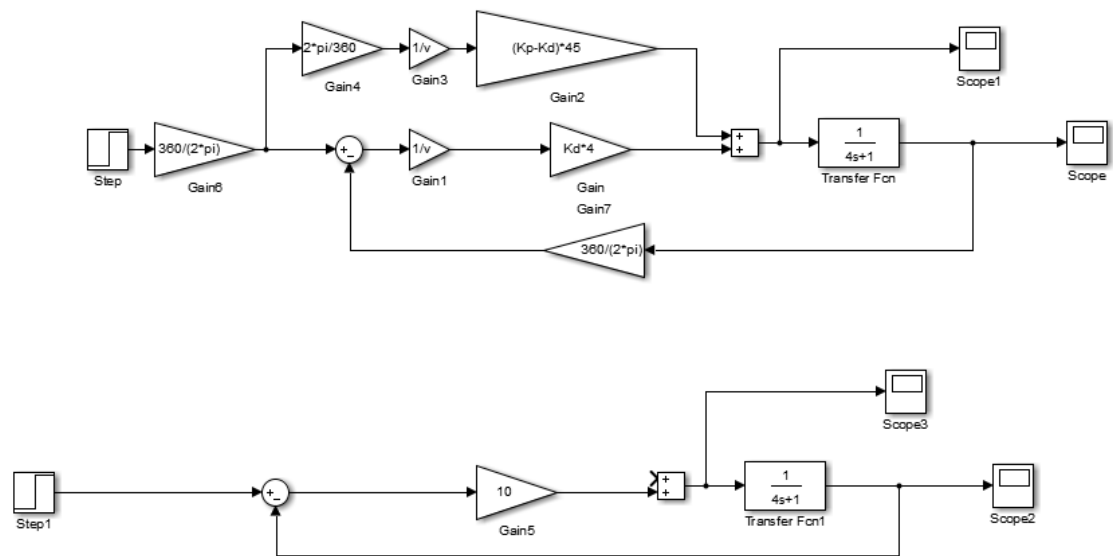
- Nacional Autónoma de México, México. Recuperado a partir de <http://www.ptolomeo.unam.mx:8080>
- DEVO F7. (2017). Recuperado el 17 de mayo de 2017, a partir de <http://www.mcmracing.com/>
- DRONES.MX. (2017). Recuperado el 17 de mayo de 2017, a partir de <https://drones.mx/>
- DronModelling. (2017). Recuperado el 22 de junio de 2017, a partir de <https://dronmodelling.es/>
- E-Radiocontrol. (2009). Recuperado el 22 de junio de 2017, a partir de <http://www.e-radiocontrol.com.ar/>
- Hernández, A. (2014). *Estrategia de control para el seguimiento de camino de un vehículo autónomo subacuático* (Innovación Tecnológica). Universidad Central "Martha Abreu" de Las Villas, Santa Clara. Recuperado a partir de <http://dspace.uclv.edu.cu/>
- Hernández, L. E. (2016). *Estudio sobre la estrategia de guiado L1 para el seguimiento de caminos rectos y curvos en vehículos autónomos* (Innovación Tecnológica). Universidad Central "Marta Abreu" de Las Villas, Santa Clara. Recuperado a partir de <http://dspace.uclv.edu.cu/>
- Larquin, O. (2016). *Implementación del algoritmo de guiado I-LOS sobre el autopiloto ArduPlane*. Universidad Central" Marta Abreu" de Las Villas. Facultad de Ingeniería Eléctrica. Departamento de Automática y Sistemas Computacionales. Recuperado a partir de <http://dspace.uclv.edu.cu/>

- Majohr, J., & Buch, T. (2006). Modelling, simulation and control of an autonomous surface marine vehicle for surveying applications measuring dolphin messin. *IEE Control Engineering Series*, 69, 329.
- Manley, J. E. (2008). Unmanned surface vehicles, 15 years of development. En *OCEANS 2008* (pp. 1–4). IEEE.
- Mission Planner. (2017). Recuperado el 6 de abril de 2017, a partir de <http://ardupilot.org/>
- Mission Planner Home. (2017). Recuperado el 5 de abril de 2017, a partir de <http://ardupilot.org/>
- Mission Planner Overview. (2017). Recuperado el 17 de mayo de 2017, a partir de <http://ardupilot.org/>
- Peláez, A. (2016). *Análisis del Filtro Extendido de Kalman presente en el Pixhawk, profundizando en cuanto a su mecanismo de ajuste*. Universidad Central “Marta Abreu” de Las Villas, Santa Clara.
- Peters, J. J. (2013). Casiopea [Caso de Estudio]. Recuperado el 22 de junio de 2017, a partir de [http://wiki.ead.pucv.cl/index.php/Funcionamiento_del_driver_CNC_DIY_para_motores_paso_a_paso_\(Marco_te%C3%B3rico\)](http://wiki.ead.pucv.cl/index.php/Funcionamiento_del_driver_CNC_DIY_para_motores_paso_a_paso_(Marco_te%C3%B3rico))
- Rabuffetti, E. J. (2016). *Diseño y contrucción de un ROV para aplicaciones de sensorización medioambiental*. Universitat Politècnica de Catalunya.

- Ratih C, N. M. R., & Suwardhi, D. (2014). Pembangunan Sistem Penentuan Posisi dan Navigasi Berbasiskan Sistem Unmanned Surface Vehicle (USV) untuk Survei Batimetri. *JURNAL ITENAS REKAYASA*, 18(1).
- Robotics, 3D. (2014). *Pixhawk Kit User Guide V8*.
- Rodríguez, W. (2015). *Estudio del algoritmo de control de rumbo L1 implementado en el autopiloto Ardupilot*. Universidad Central" Marta Abreu" de Las Villas. Recuperado a partir de <http://dspace.uclv.edu.cu/>
- Sehuveret, D. (2016). *Estrategia de guiado para el seguimiento de caminos formados por líneas rectas y segmentos de curvas por parte de un AUV*. Universidad Central "Martha Abreu" de Las Villas, Santa Clara. Recuperado a partir de <http://dspace.uclv.edu.cu/>
- Ullah, S. (2014). Autonomous Surveying Boat. *arXiv preprint arXiv:1408.6271*.
- Villar, S. A., & Acosta, G. G. (2013). *Sistema de Percepción para un Vehículo Autónomo Submarino* (Innovación Tecnológica). la Universidad nacional del centro provincia de Buenos Aires y la Universidad de las Islas Baleares, Argentina y España.
- Yan, R., Pang, S., Sun, H., & Pang, Y. (2010). Development and missions of unmanned surface vehicle. *Journal of Marine Science and Application*, 9(4), 451–457. <https://doi.org/10.1007/s11804-010-1033-2>

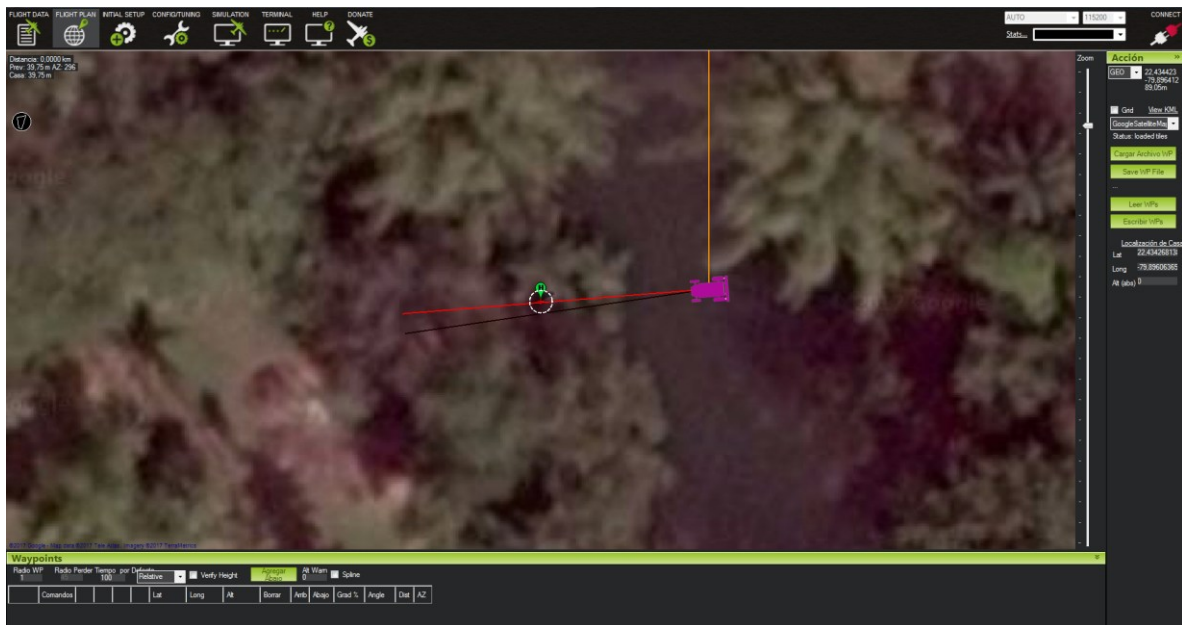
ANEXOS

Anexo A Controlador de Rumbo



Anexo A.1 Diagrama en Bloques del Código

Anexo B Funciones y pantallas del Mission Planner



Anexo B.1.Pantalla Plan de vuelo



Anexo B.2.Radio Calibración

Anexo C Código para implementación del Control de Rumbo

```
#include <AP_Math/AP_Math.h>
#include <AP_HAL/AP_HAL.h>
#include "AP_SteerController.h"

externconst AP_HAL::HAL&hal;

/*
steering rate controller. Returns servo out -4500 to 4500 given
desired yaw rate in degrees/sec. Positive yaw rate means clockwise yaw.
*/
int32_t AP_SteerController::get_steering_out_rate(floatdesired_rate)
{
    uint32_ttnow= AP_HAL::millis();
    uint32_tdt=tnow-_last_t;
    if(_last_t==0||dt>1000){
        dt=0;
    }
    _last_t=tnow;

    float speed = _ahrs.groundspeed();
    if(speed < _minspeed){
        // assume a minimum speed. This stops osciallations when first starting to move
        speed= _minspeed;
    }

    // this is a linear approximation of the inverse steering
    // equation for a ground vehicle. It returns steering as an angle from -45 to 45
    floatscaler=1.0f/ speed;

    _pid_info.desired=desired_rate;

    // Calculate the steering rate error (deg/sec) and apply gain scaler
    // We do this in earth frame to allow for rover leaning over in hard corners
    floatrate_error=(desired_rate-ToDeg(_ahrs.get_yaw_rate_earth()))*scaler;

    // Calculate equivalent gains so that values for K_P and K_I can be taken across from the old
    // PID law
    // No conversion is required for K_D
    floatki_rate= _K_I * _tau *45.0f;
    floatkp_ff= MAX((_K_P - _K_I * _tau)* _tau - _K_D ,0)*45.0f;
    floatk_ff= _K_FF *45.0f;
    floatdelta_time=(float)dt*0.001f;

    // Multiply roll rate error by _ki_rate and integrate
    // Don't integrate if in stabilise mode as the integrator will wind up against the pilots inputs
    if(ki_rate>0&& speed >= _minspeed){
        // only integrate if gain and time step are positive.
        if(dt>0){
            floatintegrator_delta=rate_error*ki_rate*delta_time*scaler;
```

```

        // prevent the integrator from increasing if steering defln demand is above the
upper limit
        if(_last_out<-45){
integrator_delta=MAX(integrator_delta,0);
}elseif(_last_out>45){
// prevent the integrator from decreasing if steering defln demand is below the lower limit
integrator_delta=MIN(integrator_delta,0);
}

        _pid_info.I+=integrator_delta;
    }
    }else{
        _pid_info.I=0;
    }

// Scale the integration limit
floatintLimScaled= _imax*0.01f;

// Constrain the integrator state
_pid_info.I=constrain_float(_pid_info.I,-intLimScaled,intLimScaled);

_pid_info.D=rate_error*_K_D*4.0f;
_pid_info.P=(ToRad(desired_rate)*kp_ff)*scaler;
_pid_info.FF=(ToRad(desired_rate)*k_ff)*scaler;

// Calculate the demanded control surface deflection
_last_out= _pid_info.D+ _pid_info.FF+ _pid_info.P+ _pid_info.I;

// Convert to centi-degrees and constrain
returnconstrain_float(_last_out*100,-4500,4500);
}

/*
lateral acceleration controller. Returns servo value -4500 to 4500
given a desired lateral acceleration
*/
int32_t AP_SteerController::get_steering_out_lat_accel(floatdesired_accel)
{
float speed = _ahrs.groundspeed();
if(speed < _minspeed){
// assume a minimum speed. This reduces osciallations when first starting to move
speed= _minspeed;
}

// Calculate the desired steering rate given desired_accel and speed
floatdesired_rate=ToDeg(desired_accel/ speed);
returnget_steering_out_rate(desired_rate);
}

/*
return a steering servo value from -4500 to 4500 given an angular
steering error in centidegrees.
*/
int32_t AP_SteerController::get_steering_out_angle_error(int32_t angle_err)
{
if(_tau <0.1f){

```

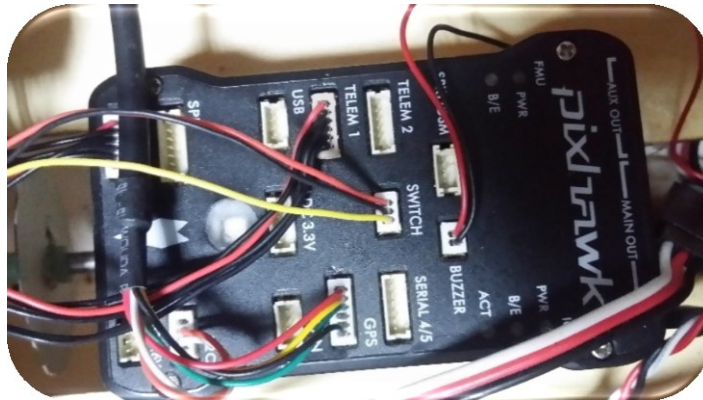
```
    _tau = 0.1f;
}

// Calculate the desired steering rate (deg/sec) from the angle error
float desired_rate = angle_err * 0.01f / _tau;

return get_steering_out_rate(desired_rate);
}

void AP_SteerController::reset_I()
{
    _pid_info.I = 0;
}
```

Anexo D Hardware del Vehículo



Anexo D.1. Conexiones con el Pixhawk



Anexo D.2. Vehículo Diseñado

Anexo E Test de validación**Anexo E.1. Estanque de pruebas****Anexo E.2. Test de validación**