

Sintonización del modelo dinámico no lineal de un robot submarino tipo ROV y estimación de sus estados mediante un filtro Kalman extendido.

Alex Daniel Barrios Durán, Brayan Buitrago Ramírez

Trabajo de Grado para optar al título de Ingeniero Electrónico

Director

Rodolfo Villamizar Mejía

PhD. Tecnologías de la información

Co-Director

David Alberto Padilla Toloza

Ingeniero Electrónico

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Bucaramanga

2020

### **Dedicatoria**

Este trabajo de grado va en memoria a mi abuelo Adolfo Durán Cristancho. A mi tío William Durán Castro y su esposa Martha Patricia Cabarcas Ayala por ayudar a forjar en mi constancia para cumplir mis sueños y lograr culminar mi carrera, su apoyo emocional y económico de principio a fin durante mi vida y mi carrera universitaria, a mis abuelos María del Carmen Castro y Adolfo Durán(QEPD) por toda una vida de apoyó como familia y ejemplo de constancia, a mi madre Yovanna Durán por darme la vida y los primeros pasos. Y a mis tíos Durán Castro por ayudar a complementar con cosas buenas lo que soy.

### ***Alex Daniel Barrios Durán***

Primeramente, a Dios pues siempre estuvo presente en todo lo que he hecho, a mis padres Rafael Buitrago Blanco y María Eugenia Ramírez Medina por su apoyo incondicional, siempre me han inspirado y motivado a lo largo de mi vida, a la memoria de mis abuelas Flor de María Blanco de Buitrago y María Emma Medina Alarcón, a todos los amigos tanto dentro como fuera de la universidad que estuvieron pendientes y contribuyeron de toda esta vida universitaria, a mi pareja Andrea Cala por acompañarme y apoyarme en esta etapa.

### ***Brayan Buitrago Ramírez***

### **Agradecimientos**

Agradecemos al profesor Rodolfo Villamizar por creer en nosotros y guiarnos en nuestro proyecto, igualmente a David padilla, la realización de este trabajo es en gran medida gracias a todo el apoyo que nos ha brindado. Sin ellos, esto no sería posible, agradecemos igualmente a todos los profesores por sus enseñanzas y a los compañeros de carrera, a nuestros amigos, los cuales contribuyeron a llegar hasta este punto.

En especial a Dios por permitirnos trabajar en este proyecto de investigación, que nos permitió crecer como personas y profesionales. Por último, a nuestros padres.

## Tabla de Contenido

<b>Introducción</b>	<b>14</b>
<b>1 Objetivos</b>	<b>16</b>
1.1 Objetivo general	16
1.2 Objetivos específicos	16
1.3 Alcances	16
<b>2 Marco conceptual</b>	<b>18</b>
2.1 Antecedentes	18
2.2 Modelo general del ROV	19
2.2.1 Modelo cinemático del ROV	19
2.2.2 Modelo dinámico del ROV	22
2.3 Estimación de modelo de caja gris.	25
2.4 Filtro de Kalman Extendido.	25
2.5 Errores que afectan los sensores tipo IMU.	28
2.6 SolidWorks	29
2.7 Mecánica de fluidos computacional (CFD)	29
2.7.1 Herramientas existentes	29
2.7.2 Ventajas consideradas	30

SISTEMA DINAMICO MODELO NO LINEAL	5
2.7.3 Inconvenientes	30
2.8 SolidWorks Flow Simulation	30
<b>3 Modelado del ROV</b>	<b>32</b>
3.1 Matrices y vectores generales del modelo VideoRay Pro-3	32
3.2 Modelado del ROV en SolidWorks	39
3.3 Obtención Anlítica de coeficientes hidrodinámicos	40
3.3.1 Movimiento axial	40
3.3.2 Movimientos rotacionales	41
3.3.3 Consideraciones	42
3.3.4 Mallado	42
3.3.5 Regresión Cuadrática	43
3.4 Método de estimación	44
3.5 Análisis de sensores inerciales IMU	46
<b>4 Pruebas</b>	<b>49</b>
4.1 Sumersión en el agua	49
4.2 Cálculo experimental de la velocidad angular de los motores	50
4.3 Sensor inercial IMU	51
4.4 Pruebas de movimiento	54
<b>5 Resultados</b>	<b>55</b>

5.1	Estimación de parámetros por medio de SolidWorks	55
5.2	Estimación de parámetros por medio de CFD	56
5.2.1	Movimiento lineal en el eje X	57
5.2.2	Movimiento lineal en el eje Y	59
5.2.3	Movimiento lineal en el eje Z	61
5.2.4	Movimiento giratorio respecto al eje X	63
5.2.5	Movimiento giratorio respecto al eje Y	65
5.2.6	Movimiento giratorio respecto al eje Z	67
5.2.7	Parámetros estimados por CFD	69
5.3	Obtención de datos de entrada.	70
5.4	Estimación de parámetros por medio de Idnlgrey	72
5.4.1	Validación del modelo no lineal	75
5.5	Matriz de covarianza	77
5.6	Comparación de estados por medio de Simulink.	80
5.6.1	Validación del Filtro de Kalman extendido	89
<b>6</b>	<b>Conclusiones</b>	<b>92</b>
<b>7</b>	<b>Trabajos futuros</b>	<b>94</b>
	Referencias Bibliográficas	95

**Lista de Figuras**

Figura 1	Imagen tomada de (9) donde se representa el modelo de predicción y actualización	27
Figura 2	Imagen tomada de (9) donde se representa los errores en sensores inerciales	28
Figura 3	Modelo elaborado en solidworks	40
Figura 4	Mallado CFD	43
Figura 5	Código regresión cuadrática	44
Figura 6	Funcion del modelo definida en matlab	45
Figura 7	Definición de datos en Matlab para la función	45
Figura 8	Definicion de caja gris en matlab	45
Figura 9	Definición del estimador y sus opciones de estimación	46
Figura 10	Sensor inercial que incluido en el ROV	47
Figura 11	Imagen del ROV desensamblado	49
Figura 12	Imagen del ROV ensamblado	50
Figura 13	Calculo experimental de la velocidad angular	51
Figura 14	Código para la extracción de datos del sensor inercial	52
Figura 15	Representación grafica del filtro Yule-walk	53
Figura 16	Gráficas del recorrido fuera del agua	54

Figura 17	Resistencia del movimiento en x	58
Figura 18	Resistencia del movimiento en Y	60
Figura 19	Resistencia del movimiento en Z	62
Figura 20	Momento en X del vehículo rotando en el eje X	64
Figura 21	Momento en y del vehículo rotando en el eje Y	66
Figura 22	Momento en y del vehículo rotando en el eje Z	68
Figura 23	Código para leer los RPM	71
Figura 24	Diagrama de bloques para la validacion del modelo	75
Figura 25	Datos de ruido de aceleraciones y giros con filtro y sin filtro para hallar matriz de covarianza	78
Figura 26	Código de promedios para hallar la covarianzas.	79
Figura 27	Montaje del filtro de Kalman extendido en simulink	80
Figura 28	Resultados gráficos de los estados del modelo y el filtro de Kalman extendido con periodo de muestreo de 0.001	81
Figura 29	Datos matrices Q,R y $T_s=0.001$	82
Figura 30	Resultados gráficos de los estados del modelo y el filtro de Kalman extendido con periodo de muestreo de 0.01	83
Figura 31	Datos matrices Q,R y $T_s=0.01$	83
Figura 32	Datos gráficos del filtro de Kalman	84
Figura 33	Datos matrices Q,R y $T_s=0.1$	84
Figura 34	Filtro simulado con los datos extraídos en el sensor instalado en el ROV	85

Figura 35	Datos del sensor instalado en el ROV con $T_s=0.001$ .	86
Figura 36	Datos matrices Q,R y $T_s=0.001$ con la IMU instalada en el ROV.	86
Figura 37	Datos del sensor instalado en el ROV con $T_s=0.01$ .	87
Figura 38	Datos matrices Q,R y $T_s=0.01$ con la IMU instalada en el ROV.	87
Figura 39	Datos del sensor instalado en el ROV con $T_s=0.1$ .	88
Figura 40	Datos matrices Q,R y $T_s=0.1$ con la IMU instalada en el ROV.	88
Figura 41	Diagrama de bloques implementando la validación del error.	89

**Lista de Tablas**

Tabla 1	Datos del sensor inercial incorporado en el ROV.	47
Tabla 2	Datos del giroscopio y acelerómetro	48
Tabla 3	Resultados de las propiedades físicas calculadas en SolidWorks	55
Tabla 4	Resultado del centro de masa calculado en SolidWorks	55
Tabla 5	Resultados de las inercias calculadas en SolidWorks	56
Tabla 6	Tabla resultados de fuerzas de resistencia debido al movimiento en x	57
Tabla 7	Tabla coeficientes debidos al movimiento en el eje X en función de la velocidad	58
Tabla 8	Tabla resultados de fuerzas de resistencia debido al movimiento en Y	59
Tabla 9	Tabla coeficientes hidrodinámicos debidos al movimiento en el eje Y en función de la velocidad	60
Tabla 10	Tabla resultados de fuerzas de resistencia debido al movimiento en Z	61
Tabla 11	Tabla coeficientes hidrodinámicos debidos al movimiento en el eje Z en función de la velocidad	62
Tabla 12	Tabla resultados de momentos en x debido al movimiento de giro en el eje X	63
Tabla 13	Tabla Coeficientes hidrodinámicos debidos al movimiento en X en función de la velocidad	64
Tabla 14	Tabla resultados de momentos debido al movimiento de giro en el eje Y	65

Tabla 15	Tabla Coeficientes hidrodámicos debidos al movimiento en Y en función de la velocidad	66
Tabla 16	Tabla resultados de momentos en Z debido al movimiento de giro en el eje X	67
Tabla 17	Tabla Coeficientes hidrodámicos debidos al movimiento en Z en función de la velocidad	68
Tabla 18	Tabla Coeficientes hidrodinámicos lineales de amortiguamiento	69
Tabla 19	Tabla Coeficientes cuadráticos	70
Tabla 20	Conversiones de RPM a velocidades angulares	72
Tabla 21	Tabla resultados de estimación de Idnlgrey	74
Tabla 22	Tabla error RMSE para el modelo no lineal	76
Tabla 23	Tabla error absoluto para el modelo no lineal	77
Tabla 24	Tabla de error RMSE	90
Tabla 25	Tabla error absoluto	91

## Resumen

**Título:** SINTONIZACIÓN DEL MODELO DINÁMICO NO LINEAL DE UN ROBOT SUBMARINO TIPO ROV Y ESTIMACIÓN DE SUS ESTADOS MEDIANTE UN FILTRO KALMAN EXTENDIDO. \*

**Autor:** Alex Daniel Barrios Durán, Brayan Buitrago Ramírez \*\*

**Palabras Clave:** Filtro Kalman extendido, Modelo no lineal, Submarino, Robot.

**Descripción:** En este documento, se propone un trabajo inspirado en el proyecto presentado por integrantes del grupo CEMOS, para el desafío numero 4 del concurso innóvate 2019 organizado por ecopetrol y Unired. Donde, este desafío planteaba el problema de: ¿Cómo desarrollar una estrategia de control de navegación autónoma para un vehículo submarino no tripulado de inspección de estructuras subacuáticas?. El grupo desarrollo un filtro de Kalman extendido para la estimación de estados del ROV que utilizaremos en este proyecto. La idea central consiste en hallar parámetros físicos de un ROV que fue donado al grupo CEMOS por un estudiante de posgrado de la Universidad Industrial de Santander del área de física, teniendo como base la tesis de maestría “Planteamiento de la arquitectura computacional e instrumentación para la navegación inercial asistida de un vehículo operado remotamente” dirigida por el Doctor Daniel Alfonso Sierra Bueno y el Doctor Rodolfo Villamizar Mejía. Sintonizaremos el modelo no lineal del prototipo donado y estimaremos sus estados mediante el filtro de Kalman extendido presentado en el concurso innóvate 2019.

---

\* Trabajo de grado.

\*\* Facultad de Ingeniería Fisicomecánicas. Escuela de Ingenierías Eléctrica, Electrónica y Telecomunicaciones. Director: Rodolfo Villamizar Mejía, Doctor en Tecnologías de la información, Co-Director: David Alberto Padilla, Ingeniero Electrónico.

### Abstract

**Title:** TUNING OF THE NON-LINEAR DYNAMIC MODEL OF A ROV-TYPE SUBMARINE ROBOT AND ESTIMATION OF ITS STATES THROUGH AN EXTENDED KALMAN FILTER. \*

**Author:** Alex Daniel Barrios Durán, Brayan Buitrago Ramírez \*\*

**Keywords:** Extended Kalman Filter, Nonlinear Model, Submarine, Robot.

**Description:** This document proposes a work inspired on an approach presented by members of the CEMOS group, for challenge number 4 of the Innovate 2019 contest organized by Ecopetrol and Unired. This challenge stated the problem of: How to develop an autonomous navigation control strategy for a Remotely Operated Vehicle (ROV) for underwater structures ?. The group proposed a hybrid control strategy, where an extended Kalman filter for estimating ROV states was included and will be validated in this project. The main idea is to find physical parameters of an ROV donated to the CEMOS group by a graduate student from the Industrial University of Santander in the physics area, based on the master's thesis "Approach to computational architecture and instrumentation for navigation inertial assistance from a remotely operated vehicle" advised by Doctors Daniel Alfonso Sierra Bueno and Rodolfo Villamizar Mejía. Then, a non-linear model of the donated prototype is tuned and its states are estimated by using the extended Kalman filter presented in the Innovate 2019 contest.

---

\* Bachelor Thesis.

\*\* Faculty of Engineering and Physical Sciences. School of Electronic and Electrical Engineering. Director: Rodolfo Villamizar Mejía, PhD in Information Technology. Co-Director: David Alberto Padilla Toloza, Electronic Engineer.

## Introducción

Las tecnologías para la exploración subacuática han sido de gran ayuda en diferentes industrias a lo largo del tiempo, cuyas aplicaciones se han enfocado principalmente en la obtención de sustancias, minerales, productos y la exploración marina, entre otros. Una de las herramientas más importantes referentes a la exploración subacuática son los vehículos remotamente operados o como se les conoce con su acrónimo en inglés ROV (Remote Operated Vehicle). Para monitorear los datos registrados por un ROV se usa una estación de control, que se comunica con el mediante un cable conocido como cordón umbilical, que se encarga de alimentarlo, transmitir comandos de control, la información de navegación y las mediciones que se registran de los sensores instalados en el vehículo.

Para lograr generar un lazo de control estable en el ROV, es necesario conocer el modelo del prototipo que se desea maniobrar y sus variables físicas más importantes. Para esto se adaptan sensores en el ROV, que a partir de sus mediciones permiten al usuario registrar variables de posición y velocidad del vehículo, mediante algoritmos de control. Generalmente se implementan sensores de medida inercial que registran variables como: velocidades angulares y aceleraciones lineales en los tres diferentes ejes espaciales. Según (9) no es posible utilizar un sistema de posicionamiento global (GPS, por sus siglas en inglés Global Positioning System) ya que el medio acuático atenúa significativamente las señales. Para estos tipos de problemas se pueden utilizar diferentes tipos de sensores que me marquen distancia y posición en ángulos para saber aproximadamente donde está

el robot.

Para aproximar el valor de los parámetros desconocidos de las ecuaciones del modelo del vehículo, sean lineales o no lineales, es frecuente usar técnicas de optimización, donde, se hacen pruebas buscando generar datos reales y con ayuda de algoritmos que puedan simular entradas y salidas, se obtiene una estimación general del modelo a implementar en la vida real, buscando el óptimo global del problema de optimización que se plantee.

## 1. Objetivos

### 1.1. Objetivo general

Estimar los estados de un robot tipo ROV mediante un Filtro Kalman Extendido usando un modelo no lineal sintonizado experimentalmente.

### 1.2. Objetivos específicos

- Sintonizar experimentalmente el modelo no lineal del ROV mediante identificación de caja gris
- Estimar los estados del ROV a partir de la sintonización del filtro de Kalman extendido.
- Validar experimentalmente el modelo no lineal identificado y el filtro Kalman sintonizado.

### 1.3. Alcances

Se realizarán pruebas del ROV en una piscina de un volumen aproximado de  $1520m^3$ , con el fin de considerar fenómenos como la fuerza de flotabilidad, empuje del agua, así como impermeabilización del ROV, con el fin de garantizar un correcto funcionamiento. Se identificarán los parámetros del modelo dinámico no lineal de un ROV previamente usado por el grupo CEMOS en el reto INNOVATE 2019, haciendo las modificaciones necesarias que se relacionen al prototipo real a partir de la identificación de un modelo de caja gris. Se estimarán los estados no medidos del ROV a partir de las mediciones obtenidas de una unidad IMU y mediante un Filtro de Kalman Extendido trabajado en el grupo de investigación CEMOS, sintonizado mediante el modelo no lineal y

determinando experimentalmente las covarianzas  $Q$  y  $R$  asociadas con el ruido de los sensores y dinámicas no modeladas.

## **2. Marco conceptual**

Este capítulo presenta los fundamentos y herramientas necesarias para dar cumplimiento a los objetivos planteados en este proyecto de investigación.

### **2.1. Antecedentes**

Para hacer un análisis referente a un prototipo robot operado desde un punto específico, se debe tener en cuenta varias preguntas como lo son: ¿cuáles son sus características?, ¿Cómo funciona?, ¿para que lo queremos?, ¿Qué profundidad queremos que recorra? Todas estas variables deben ser analizadas punto por punto, en el caso de este proyecto fue suministrado un ROV perteneciente al grupo CEMOS y del que se desconocía sus características y su electrónica. Uno de los ejemplos tomados para este tipo de trabajo es los prototipos de VideoRay Pro, son una compañía que se fundó en 1999, donde su primer prototipo de ROV fue el denominado VideoRay Pro con un peso de 8 libras con una clasificación de profundidad de 150 metros construido para el año 2000. El segundo prototipo tipo ROV se denominó VideoRay Pro II donde se aumentó su fuerza de empuje un 80%, y se añadió nueva iluminación y una cámara mejorada que mejoró el rendimiento hidrodinámico y redujo el consumo de energía en el año 2002. El tercer prototipo y en el cual se basó el trabajo de grado referenciado en (9) se le introdujo un sistema de control por software para una nueva alternativa para el panel original y abrió nuevas posibilidades para recopilar y analizar datos. El Pro-3 también contó con partes del Pro-2 pero mejoradas como cámara, propulsores, así como capacidades opcionales que se podrían agregar a través del cordón umbilical. Estos tipos de ROV son utilizados para la investigación alrededor del mundo para desarrollar y verificar estrategias de

control.

## 2.2. Modelo general del ROV

La mejor manera de describir el movimiento de un ROV es utilizando sistemas de coordenadas, también llamados marcos de referencia, según (9) existen 2 que se usan generalmente para el control de un ROV, el marco de referencia inercial o global, y el marco de referencia que se desplaza con el vehículo o local. El marco de referencia global tiene su origen en un punto arbitrario en la superficie de la tierra, a este se le considera inercial ya que la rotación de la tierra no afecta en una forma significativa el movimiento de vehículos submarinos y más si su velocidad es bastante baja. Y el marco de referencia local, que representa al vehículo o se desplaza con él, tiene su origen en el centro de gravedad del ROV.

**2.2.1. Modelo cinemático del ROV.** Para el modelado del movimiento de un ROV es necesario definir vectores que agrupen las variables de posición expresado en el sistema de coordenadas inerciales. Inicialmente,  $n_1$  agrupa las variables de posición lineal.

$$n_1 = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \quad (1)$$

También existe otro vector el cual describe los ángulos de rotación respecto a cada eje, es importante tener claro que la orientación del ROV respecto a un sistema terrestre se describe con los ángulos de Euler. Las componentes de este vector de orientación se representan con el siguiente

vector.

$$n_2 = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \quad (2)$$

Los vectores respecto a la velocidad lineal y angular en el sistema inercial se expresan de la siguiente manera:

$$v_1 = \begin{bmatrix} u \\ v \\ w \end{bmatrix} \quad (3)$$

$$v_2 = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (4)$$

El modelo cinemático del ROV obedece a la siguiente expresión:

$$\dot{n} = J(n) \cdot v \quad (5)$$

La ecuación derivada de nabra (5) permite obtener la posición y orientación del ROV a través de la

transformación de sus velocidades al sistema inercial. Donde:

$$n = \begin{bmatrix} n_1 \\ n_2 \end{bmatrix} \quad (6)$$

$$v = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (7)$$

$$J(n) = \begin{bmatrix} J_1(n) & 0_{3 \times 3} \\ 0_{3 \times 3} & J_2(n) \end{bmatrix} \quad (8)$$

$$J_1(n) = \begin{bmatrix} \cos\theta\cos\psi & (-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi) & (\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi) \\ \cos\theta\sin\psi & (\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) & (-\sin\phi\cos\psi + \cos\phi\sin\theta\cos\psi) \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad (9)$$

$$J_2(n) = \left(\frac{1}{\cos\theta}\right) \begin{bmatrix} \cos\theta\cos\psi & (-\cos\phi\sin\psi + \sin\phi\sin\theta\cos\psi) & (\sin\phi\sin\psi + \cos\phi\sin\theta\cos\psi) \\ \cos\theta\sin\psi & (\cos\phi\cos\psi + \sin\phi\sin\theta\sin\psi) & (-\sin\phi\cos\psi + \cos\phi\sin\theta\cos\psi) \\ -\sin\theta & \cos\theta\sin\phi & \cos\theta\cos\phi \end{bmatrix} \quad (10)$$

**2.2.2. Modelo dinámico del ROV.** Para representar la dinámica de un cuerpo rígido en el espacio vacío en relación a su movimiento, se aplica la siguiente expresión:

$$M_{RB}\dot{v} + C_{RB}(v)V + D(v)v + g(n) = \tau \quad (11)$$

Donde  $M_{RB}$  es la masa del vehículo,

$$M_{RB} = \begin{bmatrix} m & 0 & 0 & 0 & mz_G & -my_G \\ 0 & m & 0 & -mz_G & 0 & mx_G \\ 0 & 0 & m & my_G & -mx_G & 0 \\ 0 & -mz_G & my_G & I_{xx} & -I_{xy} & -I_{xz} \\ mz_G & 0 & -mx_G & -I_{yx} & I_{yy} & -I_{yz} \\ -my_G & mx_G & 0 & -I_{zx} & -I_{zy} & I_{zz} \end{bmatrix} \quad (12)$$

$r_{CG}$  : Está definido como el vector de distancia al centro de gravedad.

$$r_{CG} = \begin{bmatrix} x_G & y_G & z_G \end{bmatrix} \quad (13)$$

$I_o$ : Tensión de inercia, está definida por la siguiente matriz:

$$I_0 = \begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix} \quad (14)$$

La matriz  $C_{RB}$  se define como la matriz de fuerzas centrípeta y Coriolis, pero según (9) existe una matriz antisimétrica la cual es:

$$C_{RB}(v) = \begin{bmatrix} 0_{3 \times 3} & C_{12}(v) \\ -C_{12}^T(v) & C_{22}(v) \end{bmatrix} \quad (15)$$

$$C_{12}(v) = \begin{bmatrix} m(y_G q) + z_G r & -m(x_G q) + w & -m(x_G r) + v \\ -m(y_G q) + w & m(z_G r + x_G p) & -m(y_G r - u) \\ -m(z_G p - v) & -m(z_G q + u) & m(x_G p + y_G q) \end{bmatrix} \quad (16)$$

$$C_{22}(v) = \begin{bmatrix} 0 & -I_{yz}q - I_{xz}p + I_z r & I_{yz} + I_{xy}p - I_y q \\ I_{yz}q + I_{xz}p - I_z r & 0 & -I_{xz}r - I_{xy}q + I_x p \\ -I_{yz}r & I_{xz}r + I_{xy}q - I_x p & 0 \end{bmatrix} \quad (17)$$

Tao consiste en un vector de fuerzas y momentos resultantes, este vector se origina por los propulsores. Donde:

$$\tau_1 = \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} \quad (18)$$

$$\tau_2 = \begin{bmatrix} K \\ Y \\ Z \end{bmatrix} \quad (19)$$

$$\tau = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} \quad (20)$$

En el estudio de los vehículos submarinos, es necesario tener en cuenta las fuerzas hidrodinámicas generalizadas, debido al movimiento del ROV en la presencia de fluidos nacen las fuerzas y momentos causados.

- Fuerzas inducidas por radiación: La inercia del fluido nos genera una contribución de masa y amortiguamiento hidrodinámico.
- Fuerza de restauración: Consiste en la gravedad y el empuje que actúan sobre el ROV en pocas palabras flotabilidad.
- Perturbaciones ambientales: Son ruidos añadidos por condiciones climáticas.

La ecuación que describe la dinámica de un ROV teniendo en cuenta las fuerzas hidrodinámicas generalizadas y la acción del cordón umbilical es (9):

$$M\dot{v} + C(v)v + D(v)v + g(n) = \tau \quad (21)$$

### **2.3. Estimación de modelo de caja gris.**

La estimación de caja gris es una combinación entre los modelos de caja negra y caja blanca. Donde, en un modelo de caja negra no se conoce la estructura interna del sistema, únicamente sus entradas y salidas. En un modelo de caja blanca, se conoce la estructura interna del sistema, es decir, se cuenta con las ecuaciones y parámetros del modelo. Al momento de combinar estas 2 estimaciones la caja gris se encarga en ejecutar una prueba de caja negra basándose en datos de pruebas realizados por personas que conocen el sistema y que conocen parcialmente la estructura interna.

Algunos puntos a favor que tiene usar un modelo de caja gris son:

- Da beneficios de ambas pruebas, tanto de caja blanca y caja negra.
- Permite hallar variables desconocidas referentes a los modelos, ya que se conoce parte de la estructura interna y externa.

### **2.4. Filtro de Kalman Extendido.**

El filtro de Kalman es un algoritmo de estimación de estados óptimo y estocástico. Este filtro tiene muchas aplicaciones en el campo de la ingeniería. Todo es gobernado a través de una ecuación diferencial lineal que permite estimar estados de cualquier sistema, desde un punto de partida

que es son las mediciones ruidosas. Una aplicación muy común es en equipos de navegación y control de vehículos, especialmente en naves espaciales. Al hablar de equipos de navegación, cabe resaltar que los ejemplos a nivel general la mayoría son sistemas no lineales. Al tener sistemas no lineales el filtro de Kalman convencional puede no ser funcional y por tanto se habla de Filtro Kalman Extendido (EKM por sus siglas en inglés) y que resulta mejor en su estimación debido a que tiene en cuenta las no linealidades del sistema al utilizar el operador Jacobiano para realizar una aproximación lineal de las ecuaciones del sistema referente a sus estados. Una función que describe un proceso no lineal según (9) es:

$$\dot{x} = f(x, u) + w \quad (22)$$

$$z = h(x, u) + v \quad (23)$$

Para implementar un filtro de Kalman extendido existen 2 etapas: una de predicción y otra de actualización. Se maneja una simbología de + y - como exponentes de cada variable, el + es representa actualización y el - representa predicción como se muestra en la siguiente tabla:

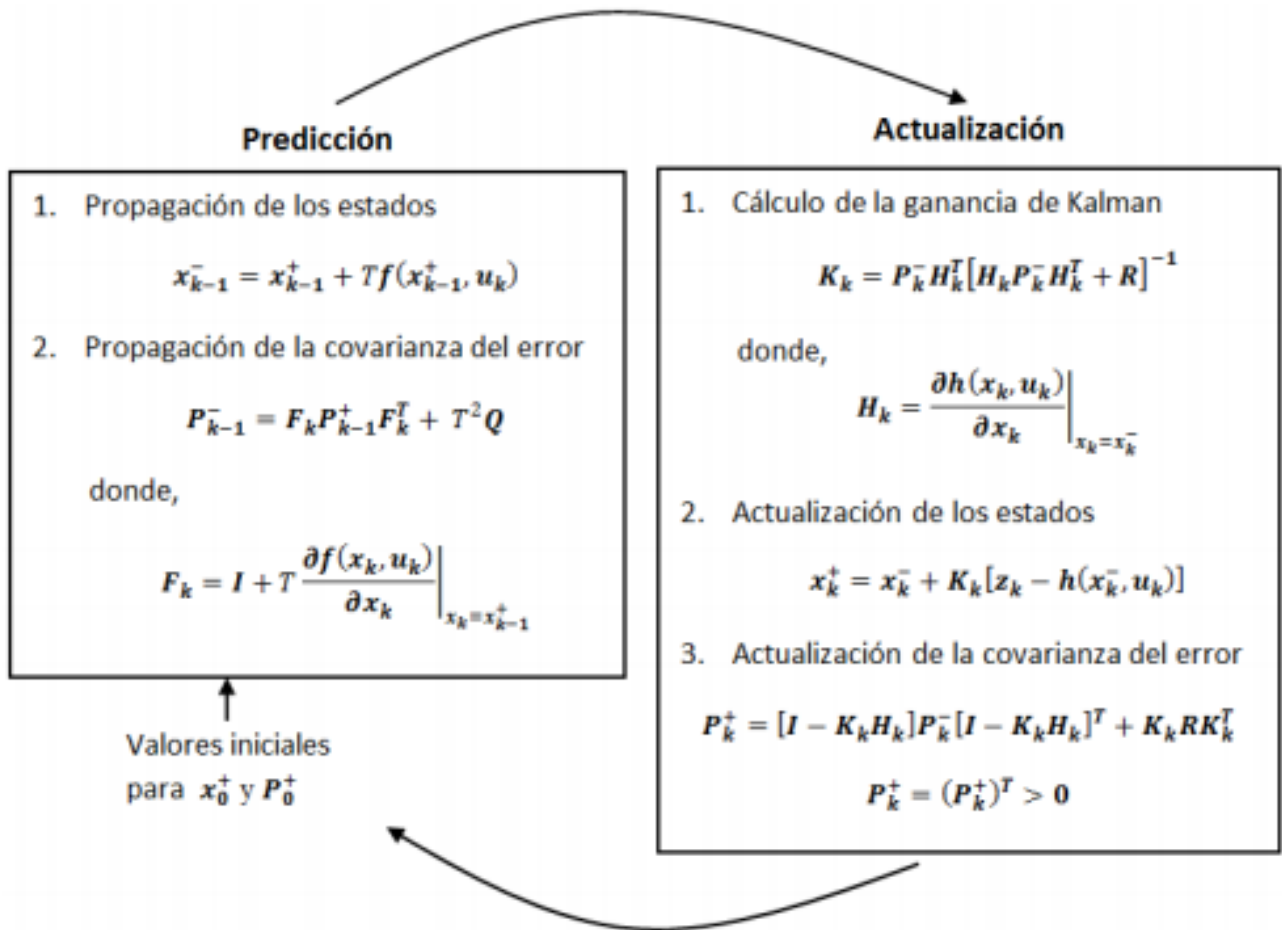


Figura 1. Imagen tomada de (9) donde se representa el modelo de predicción y actualización

Cabe resaltar que cada letra tiene su significado como lo es:

- f: Es la función no lineal.
- x: Estado depende de una entrada de control.
- u: Entrada de control.
- w: Ruido del proceso.

- $v$ : Ruido de medición.
- $Q$ : Covarianza de ruido del proceso.
- $R$ : Covarianza de ruido de medición.

## 2.5. Errores que afectan los sensores tipo IMU.

Los sensores inerciales son unidades de medida de aceleración lineal y velocidad angular, se utilizan en sistemas que estén relacionados con el análisis del movimiento tridimensional. Los sensores inerciales se componen de acelerómetros y giroscopios, también pueden tener integrados magnetómetros y/o barómetros, dependiendo de que requiera en el modelo que se esté manejando.

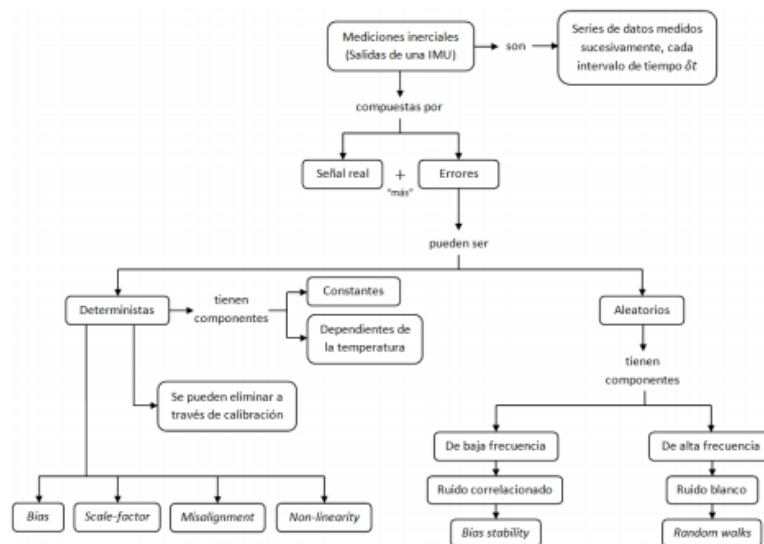


Figura 2. Imagen tomada de (9) donde se representa los errores en sensores inerciales

Como se puede observar en la figura 2 según (9) existen 2 clases que pueden afectar en cuestiones de errores como lo son deterministas y errores aleatorios, las que más afectan las medidas son las

aleatorias ya que contienen problemas respecto a la frecuencia y componentes de mala calidad que afectan la frecuencia.

## **2.6. SolidWorks**

Es un Software para el diseño asistido por computadora, en él se puede modelar piezas o ensamblajes 3D, y planos en 2D, tiene gran variedad de complementos y herramientas útiles en el diseño y desarrollo de proyectos de ingeniería.

## **2.7. Mecánica de fluidos computacional (CFD)**

Es una de las ramas de la mecánica de fluidos, utiliza métodos numéricos y algoritmos para resolver y analizar problemas relacionados con el flujo de gases y líquidos. Incluso con ecuaciones simplificadas y supercomputadoras de alto rendimiento, en muchos casos solo se pueden obtener resultados aproximados. Sin embargo, la investigación en curso permite la incorporación de software más avanzado, que reduce el tiempo de computación. La verificación de los datos obtenidos mediante CFD se suele realizar en túneles de viento u otros modelos a escala física, como los tanques de experiencia de potencia de fluidos. El método implica discretizar regiones espaciales para crear las llamadas cuadrículas espaciales y luego dividir las regiones espaciales en volúmenes de control más pequeños. Luego, la ecuación de conservación discretizada se resuelve en cada ecuación, por lo que, de hecho, la matriz algebraica se resuelve iterativamente en cada unidad hasta que el resto es lo suficientemente pequeño. (12)

**2.7.1. Herramientas existentes.** Existen gran variedad de software que cuentan con el análisis CFD dentro de sus complementos, tales como ANSYS, Rhino, y SolidWorks. Siendo el último el utilizado en el proyecto, dado que la universidad cuenta con una licencia estudiantil. Por

lo tanto, se realizó un modelo del ROV con todos los parámetros físicos reales en esta herramienta.

### **2.7.2. Ventajas consideradas.**

- Reducir significativamente el tiempo y el costo de los nuevos diseños.
- Posibilidad de analizar sistemas y condiciones difíciles de simular mediante experimentos.
- Tener un nivel de detalle ilimitado, los métodos experimentales se vuelven más costosos a medida que aumenta el número de puntos de medición, mientras que los programas CFD pueden generar mucha información y algunos pueden realizar estudios de parámetros.
- Dada la problemática del COVID-19, es una opción de realizar pruebas sin tener que utilizar instalaciones o laboratorios

### **2.7.3. Inconvenientes.**

- No siempre es posible obtener resultados suficientemente precisos, lo que puede dar lugar a errores.
- La precisión del simulador depende en gran medida a la capacidad computacional que se tenga en el momento, ya que a mayor detalle en la simulación mayor será el tiempo requerido para encontrar una solución.

## **2.8. SolidWorks Flow Simulation**

Esta herramienta es un nuevo tipo de software de análisis de dinámica de fluidos computacional (CFD), permite simular la dinámica de fluidos, transferencia del calor y las fuerzas de fluidos.

El CFD se encuentra integrada en SolidWorks como un complemento activable, que permite la simulación rápida y sencilla de flujos de líquido y gas a través de piezas ya construidas. En general es una herramienta sencilla, práctica, útil y muy concisa. La capacidad de ubicar fuerzas como la gravedad en cualquier eje, la hace muy versátil, pues no es necesario modificar la pieza para cambiar el punto de referencia de acuerdo a la posición en que se desee simular la pieza. (5)

### 3. Modelado del ROV

Para cumplir los objetivos que se plantearon para este proyecto, se empezó con un análisis general del modelo del ROV, analizando que parámetros son necesarios para lograr el objetivo general. Para esto se realizó un estudio del modelo que se tomó como referencia el trabajo realizado en (9)

#### 3.1. Matrices y vectores generales del modelo VideoRay Pro-3

Parte importante del modelo de VideoRay Pro-3 son sus matrices y vectores, para este modelo se tienen en cuenta 3 matrices que son:

- - La matriz de masa total
- - Matriz de fuerza centrípeta y Coriolis
- - Matriz de amortiguamiento hidrodinámico

Estas matrices contienen la mayoría de parámetros a estimar en el modelo de caja gris

En la matriz de masa total se analiza el movimiento del ROV, que al desplazarse en un fluido crea una masa adicional en el modelo, añadiendo valores de masa total que afectan los 6 grados de libertad que tiene el vehículo. Esta masa adicional se presenta solo cuando hay aceleración, ya que el fluido ejerce un peso respecto al ROV. Añadiendo la masa adicional la matriz de masa total queda de la siguiente manera:

$$M = M_{RB} + M_A = \begin{bmatrix} m - X_{\dot{u}} & 0 & 0 & 0 & 0 & 0 \\ 0 & m - Y_{\dot{v}} & 0 & 0 & 0 & 0 \\ 0 & 0 & m - Z_{\dot{w}} & 0 & 0 & 0 \\ 0 & 0 & 0 & I_{xx} - K_{\dot{p}} & 0 & 0 \\ 0 & 0 & 0 & 0 & I_{yy} - M_{\dot{q}} & 0 \\ 0 & 0 & 0 & 0 & 0 & I_{zz} - N_{\dot{r}} \end{bmatrix} \quad (24)$$

La matriz de fuerza centrípeta y Coriolis define que el fluido tiene una inercia mientras rodea al ROV, lo que genera una fuerza centrípeta y otra de Coriolis que experimentará el vehículo. Este tipo de fuerzas tienen representaciones matriciales como sigue:

$$C_A(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v \\ 0 & 0 & 0 & Z_{\dot{w}}w & 0 & -X_{\dot{u}}u \\ 0 & 0 & 0 & Y_{\dot{v}}v & X_{\dot{u}}u & 0 \\ 0 & -Z_{\dot{w}}w & Y_{\dot{v}}v & 0 & -N_{\dot{r}}r & M_{\dot{q}}q \\ Z_{\dot{w}} & 0 & -X_{\dot{u}}u & N_{\dot{r}}r & 0 & -K_{\dot{p}} \\ -Y_{\dot{v}}v & -X_{\dot{u}}u & 0 & -M_{\dot{q}}q & K_{\dot{p}}p & 0 \end{bmatrix} \quad (25)$$

La matriz  $C_A$  representa la fuerza centrípeta y de Coriolis, definimos la matriz  $C_{RB}$  que representa las masas añadidas, que al sumarse generan la matriz  $C(v)$

$$C_{RB}(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & mw & -mv \\ 0 & 0 & 0 & -mw & 0 & mu \\ 0 & 0 & 0 & mv & -mu & 0 \\ 0 & mw & -mv & 0 & I_{zz}r & -I_{yy}q \\ -mw & 0 & mu & -I_{zz}r & 0 & I_{xx}p \\ mv & -mu & 0 & I_{yy}q & -I_{xx}p & 0 \end{bmatrix} \quad (26)$$

$$C(v) = \begin{bmatrix} 0 & 0 & 0 & 0 & mw - Z_{\dot{w}}w & -mv + Y_{\dot{v}}v \\ 0 & 0 & 0 & -mw + Z_{\dot{w}}w & 0 & -mu - X_{\dot{u}}u \\ 0 & 0 & 0 & mv - Y_{\dot{v}}v & -mu + X_{\dot{u}}u & 0 \\ 0 & mw - Z_{\dot{w}}w & -mv + Y_{\dot{v}}v & 0 & I_{zz}r - N_{\dot{r}}r & -I_{yy}q + M_{\dot{q}}q \\ -mw + Z_{\dot{w}} & 0 & mu - X_{\dot{u}} & -I_{zz}r + N_{\dot{r}}r & 0 & I_{xx}p - K_{\dot{p}}p \\ mv - Y_{\dot{v}} & -mu + X_{\dot{u}}u & 0 & I_{yy}q - M_{\dot{q}}q & -I_{xx}p + K_{\dot{p}}p & 0 \end{bmatrix} \quad (27)$$

Una vez analizada la matriz  $C(v)$  se debe estudiar la densidad del fluido donde va a navegar el ROV. Cuando el vehículo se encuentra sumergido, presenta una fuerza de flotabilidad y fricción viscosa, que dependen de la densidad del fluido. En el modelo de VideoRay Pro-3 la matriz de amortiguamiento hidrodinámico se divide en una matriz de términos lineales y otra de términos

cuadráticos donde  $D_{lin}$ :

$$D_{lin} = - \begin{bmatrix} X_u & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_w & 0 & 0 & 0 \\ 0 & 0 & 0 & K_p & 0 & 0 \\ 0 & 0 & 0 & 0 & M_q & 0 \\ 0 & 0 & 0 & 0 & 0 & N_r \end{bmatrix} \quad (28)$$

$$D_{quad}(v) = - \begin{bmatrix} X_{u|u}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_{v|v}|v| & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_{w|w}|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & K_{p|p}|p| & 0 & 0 \\ 0 & 0 & 0 & 0 & M_{q|q}|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & N_{r|r}|r| \end{bmatrix} \quad (29)$$

$D_{lin}$  representa la matriz de términos lineales y  $D_{quad}(v)$  representa la matriz de términos cuadráticos. La suma de ambas matrices da como resultante la matriz de amortiguamiento hidrodinámico:

$$D(v) = - \begin{bmatrix} X_u + X_{u|u}|u| & 0 & 0 & 0 & 0 & 0 \\ 0 & Y_v + Y_{v|v}|v| & 0 & 0 & 0 & 0 \\ 0 & 0 & Z_w + Z_{w|w}|w| & 0 & 0 & 0 \\ 0 & 0 & 0 & K_p + K_{p|p}|p| & 0 & 0 \\ 0 & 0 & 0 & 0 & M_q + M_{q|q}|q| & 0 \\ 0 & 0 & 0 & 0 & 0 & N_{r|r}|r| \end{bmatrix} \quad (30)$$

Las fuerzas que se tienen en cuenta son la fuerza de gravedad y la flotabilidad y las fuerzas y momentos resultantes que afectan al ROV.

El centro de gravedad del vehículo es el punto debido a la resultante de la fuerza de gravedad ejercida en las diferentes partes del ROV. El centro de flotabilidad es el punto resultante de las fuerzas de flotabilidad ejercida por las diferentes partes el ROV cuando este se encuentra sumergido en el agua, se representa mediante vector:

$$r(B) = \begin{bmatrix} x_B & y_B & z_B \end{bmatrix} \quad (31)$$

Al unificar los vectores de fuerzas del vehículo por la gravedad y la flotabilidad se obtiene el siguiente vector:

$$g(n) = \begin{bmatrix} (W - B)\sin(\theta) \\ -(W - B)\cos(\theta)\sin(\phi) \\ -(W - B)\cos(\theta) \\ -(y_G W - y_B B)\cos(\theta)\cos(\phi) + (z_G W - z_B B)\cos(\theta)\sin(\phi) \\ (z_G - z_B B)\sin(\theta) + (x_G W - x_B B)\cos(\theta)\cos(\phi) \\ (x_G W - x_B B)\cos(\theta)\sin(\phi) - (y_G W - y_B B)\sin(\theta) \end{bmatrix} \quad (32)$$

Donde  $W$  es el peso del vehículo medido en Newtons y  $B$  es la fuerza de empuje del vehículo medido en Newtons.

Los estados del vector (20) se generan por el encendido de los motores o propulsores del ROV. Para calcular las fuerzas y momentos de este vector es necesario utilizar 2 modelos o ecuaciones que son:

$$\dot{n} = \beta Q - \alpha n |n| \quad (33)$$

$$T = C_T n |n| \quad (34)$$

Donde cada variable representa,

- $n$ : Velocidad angular del propulsor.

- $Q$ : Entrada de control que es igual al torque generado por el motor
- $\beta, \alpha, C_T$ : Representan las constantes de los actuadores.

Este proyecto se enfocó en el modelo de VideoRay pro, por lo tanto, se descartó la ecuación (1) ya que la dinámica que describe esta ecuación es para vehículos más rápidos que el ROV, por ejemplo un dron. De modo que solo se tiene en cuenta la ecuación (21), que se calcula teniendo en cuenta el siguiente principio:

$$\tau = \begin{cases} 5 & \text{si } x \leq 2 \\ x^2 - 6x + 10 & \text{si } 2 < x < 5 \\ 4x - 15 & \text{si } x \geq 5 \end{cases} \quad (35)$$

Donde,

- $T$ : Es la matriz de configuración de los actuadores.
- $C^+$  y  $C^-$ : Son matrices diagonales que describen la dinámica de los propulsores.
- $u$ : Un vector de velocidades angulares al cuadrado de cada actuador.

En el modelo de VideoRay Pro-3, la matriz  $T$  tiene seis filas para representar seis propulsores. El ROV usado en este proyecto cuenta con 4 propulsores instalados, por lo tanto, la matriz  $T$  se modifica de la siguiente manera:

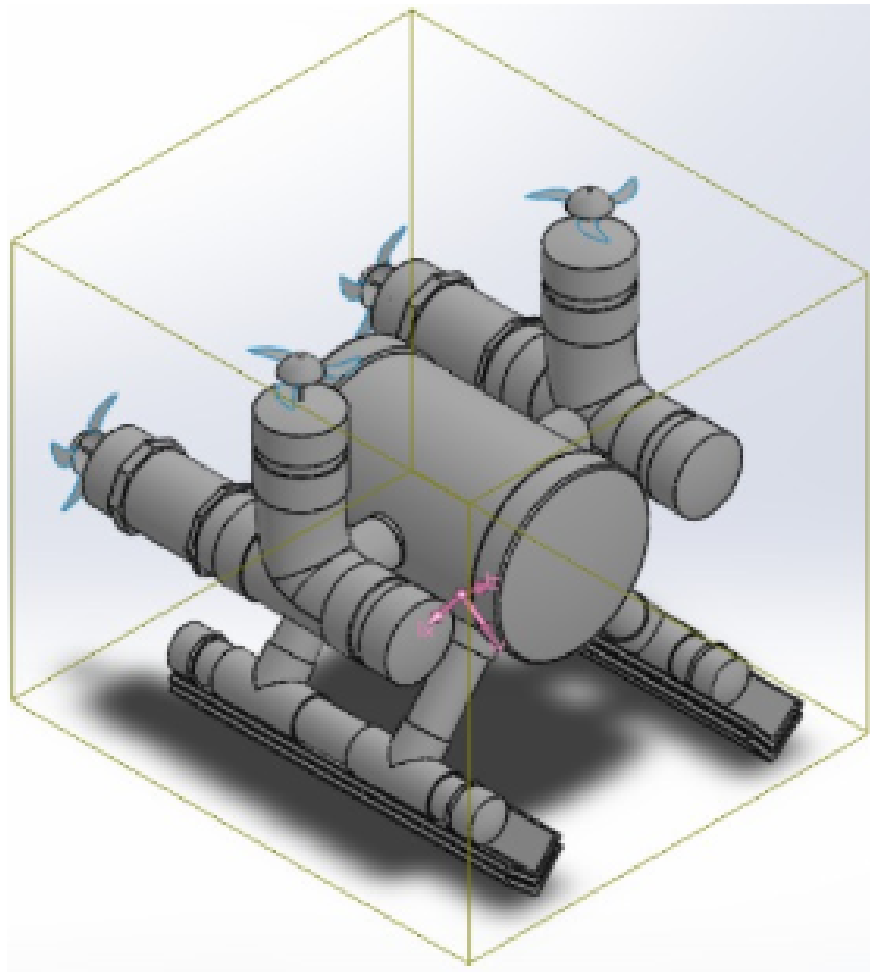
$$T = \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 \\ -ly1 & -ly2 & 0 & 0 \\ lx1 & lx2 & lz3 & lz4 \\ 0 & 0 & -ly3 & -ly4 \end{bmatrix} \quad (36)$$

### 3.2. Modelado del ROV en SolidWorks

Para el cálculo de inercias de las matrices  $M_{RB}$  e  $I_0$  se utilizó el software Solidworks. Este es ideal para para modelar sistemas físicos en 2D o 3D dependiendo la pieza o figura de interés.

Gracias a la implementación en el programa SolidWorks se estimó las inercias y volumen general del ROV, este programa permite asignarle a cada pieza el tipo de material con el que está construido el vehículo, creando así un modelo muy aproximado.

Al construir el modelo en Solidworks, se verificó que las medidas físicas sean las correspondientes al ROV, finalizando la construcción del modelo en el software, además en las propiedades físicas del ROV construido, se calcula los valores de Momentos de inercia y centro de masa, siendo estos los valores correspondientes a las matrices  $I_0$  y  $M_{RB}$ .



*Figura 3.* Modelo elaborado en solidworks

### **3.3. Obtención Análítica de coeficientes hidrodinámicos**

**3.3.1. Movimiento axial.** Cuando el vehículo se encuentra sumergido en el agua y realiza un desplazamiento lineal en la dirección de un eje, aparecen fuerzas y momentos debidos a este movimiento. Estas fuerzas de oposición al movimiento son llamadas fuerzas hidrodinámicas(12) y se expresan en función de la velocidad mediante dos coeficientes lineal y cuadrático (12). La ecuación 37 relaciona la fuerza de oposición al movimiento en x con los coeficientes hidrodinámicos

cos.

$$Fx = x_{u|u} * u|u| + x_u * u \quad (37)$$

Donde:

- $x_{u|u} * u|u|$ : Corresponde al término de amortiguamiento cuadrático y es el dominante cuando la velocidad es media o elevada.
- $x_u * u$ : Corresponde al término de amortiguamiento lineal y es el dominante cuando la velocidad es pequeña.

Esta ecuación es la relación para el movimiento lineal en el eje x, pero es de la misma manera para los otros dos ejes, Y y Z respectivamente.

$$Fy = y_{v|v} * v|v| + y_v * v \quad (38)$$

$$Fz = z_{w|w} * w|w| + z_w * w \quad (39)$$

**3.3.2. Movimientos rotacionales.** Cuando el vehículo rota respecto a un eje axial a una velocidad angular constante, se obtienen fuerzas y momentos de igual forma como sucede en los movimientos lineales. Los momentos se expresan en función de la velocidad mediante dos coeficientes cuadrático y lineal.(12)

Cuando el vehículo gira con una velocidad angular constante en el eje X aparece el siguiente

momento:

$$M_x = K_{p|p|} * p|p| + K_p * p \quad (40)$$

De igual forma se hallan momentos cuando el vehículo gira con una velocidad constante en los ejes Y y Z. En estos aparecen los siguientes momentos respectivamente.(12)

$$\text{Momento debido al giro en el eje } y = M_{q|q|} * q|q| + M_q * q \quad (41)$$

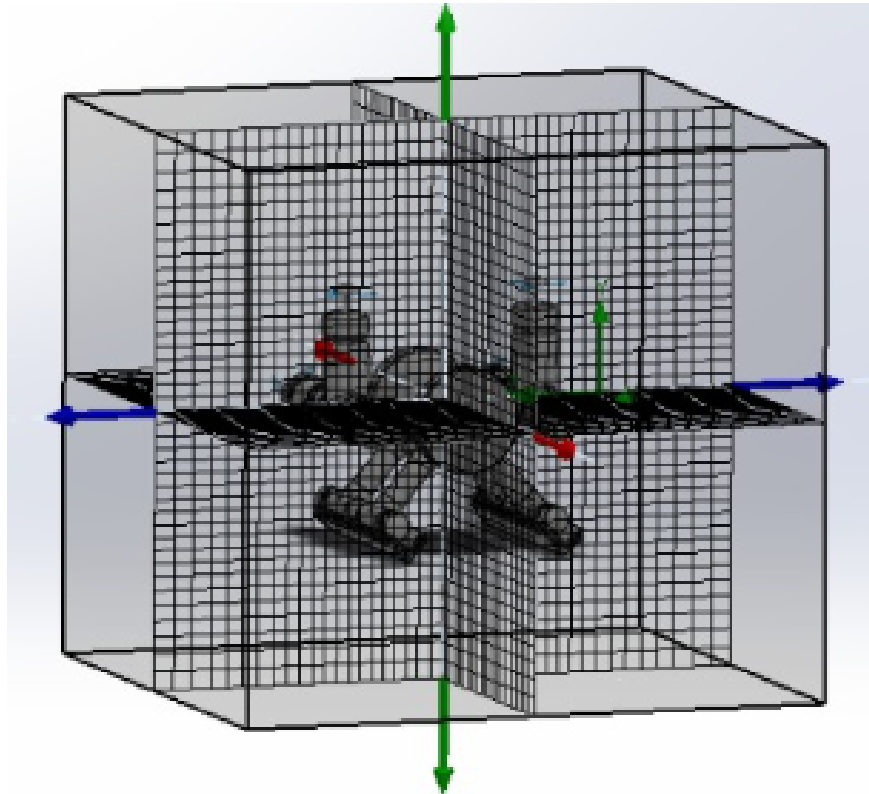
$$\text{Momento debido al giro en el eje } z = N_{r|r|} * r|r| + N_r * r \quad (42)$$

**3.3.3. Consideraciones.** Después de definir las ecuaciones que relacionan la fuerzas que serán calculadas por las simulaciones de Flow Simulation, se definen los parámetros necesarios en el software para obtener los resultados con mejor aproximación.

Se debe considerar que SolidWorks utiliza un sistema de referencia asumiendo la dirección “x” positiva hacia adelante, “y” positiva hacia arriba y la dirección “z” positiva hacia la derecha. Por lo tanto, cuando terminan las simulaciones, los resultados se deben interpretar de acuerdo con el sistema de referencia del ROV.

**3.3.4. Mallado.** El mallado es parte fundamental para la obtención de buenos resultados numéricos de simulación. La herramienta CFD permite un nivel de mallado entre 1 y 7, siendo 1 el mallado menos detallado y 7 el más detallado, lo que se traduce en la cantidad de celdas generadas

para la simulación. Entre mayor sea el número de celdas más exactos serán los resultados.



*Figura 4.* Mallado CFD

Se utilizó un mallado nivel 7, dado que el tiempo de simulación no era muy extenso y se logra una mejor estimación de la fuerza. Este mallado se traduce en el análisis de 94355 celdas, de las que 9609 celdas estaban en contacto directo con el sólido.

En comparación al mallado del trabajo (12), donde se obtuvo buenos resultados, en este proyecto hay más del doble de celdas en contacto con el ROV.

**3.3.5. Regresión Cuadrática.** Después del cálculo de las fuerzas y momentos para cada velocidad lineal y de rotación, se realiza una regresión cuadrática de los resultados para calcular

los distintos coeficientes de amortiguamiento hidrodinámico.

Para esto se realizó un algoritmo en Matlab con la función 'nlinfit', la que permite buscar los coeficientes de ajuste a una función objetivo, en nuestro caso una cuadrática con un término lineal.

```

1 -  clc
2 -  clear
3 -  %% Carga de resultados
4 -  X=[0,0.5,1,1.5,2,2.5,3,3.5];
5 -  y=[-0.000633967,-0.081259009,-0.310803556,-0.697769415,-1.175896071,-1.939618352,-2.55773068,-3.308004264];
6 -  % Punto de inicio
7 -  param0= [0,0];
8 -  %% Regresión Cuadratica
9 -  modelfun = @(a,x) (a(1).*x.^2+a(2).*x);
10 - [param,r,J,CovB,MSE,ErrorModelInfo] = nlinfit(X,y,modelfun,param0);
11
12 - %% Grafica de comparación
13 - xx= X(1):0.01:X(end);
14 - yy = param(1)*xx.^2+param(2)*xx;
15 - figure: plot (X,y'.'.',xx,yy);

```

Figura 5. Código regresión cuadratica

Los resultados del código son los dos parámetros, el primero es el valor cuadrático, donde, en las pruebas axiales en el eje x sería el  $x_{u|u}$ , mientras que el segundo valor corresponde al valor lineal, por ejemplo  $x_u$  para las pruebas ya mencionadas. Además, se calcula el error cuadrático medio que se comete en cada regresión, entre más aproximado al cero sea el resultado del MSE, mejor será la estimación de los coeficientes hidrodinámicos.

### 3.4. Método de estimación

Para la estimación de los datos desconocidos se utilizó el estimador *Non linear grey box* de la herramienta Matlab. Al principio se pretendía hallar todas las constantes hidrodinámicas no conocidas con el estimador de Matlab, pero después de varias simulaciones, se concluyó que su

tiempo de cómputo era muy grande y los resultados arrojaron un error elevado. Se decide utilizar el método de obtención de parámetros por CFD mencionado en la sección 3.4 para calcular la mayoría de los coeficientes hidrodinámicos y así reducir el número de parámetros a estimar con la herramienta.

El planteamiento del No linear grey box fue de la siguiente manera:

```
function [dx, y] = ROVfun1(t, x, u, B, Xudot, Yvdot, Zwdot, Kpdot, Mqdot, Nrdot, C_plus1, C_plus2, varargin)
```

Figura 6. Funcion del modelo definida en matlab

- Se definió la función, que contiene el modelo del ROV utilizada por el grupo CEMOS como primer paso.
- Se establecieron los parámetros ya conocidos en la función y los restantes se configuraron como desconocidos.
- Se define los datos de estimación.

donde se llama el número de entradas, salidas y tiempo de muestreo.

```
datos = iddata([ni_dot ni2], u, Ts);
```

Figura 7. Definición de datos en Matlab para la función

- Se define el modelo de caja gris, indicando la función que se va a usar, la cantidad de estados, entradas, salidas, parámetros y condiciones iniciales.

```
caja_gris = idnlgrey(odefun, order, parameters, initialstates);
```

Figura 8. Definicion de caja gris en matlab

- Por último, se llama al estimador que es `nlgreyest` y se definen diferentes tipos de opciones como un mejor gradiente, un número inicial de tolerancia, entre otras que se pueden ver en la siguiente imagen.

```
opt = nlgreyestOptions('Display','on','SearchMethod','gna');
opt.GradientOptions.Type = 'Refined';
opt.SearchOption.MaxIter = 100;
opt.SearchOption.Advanced.InitialGnaTolerance=0.0001;
opt.SearchOption.Advanced.MaxBisections=25;
opt.SearchOption.Advanced.MaxFunctionEvaluations=Inf;
opt.SearchOption.Advanced.MinParameterChange=0;
opt.SearchOption.Advanced.RelativeImprovement=0;
% opt.SearchOption.Algorithm='interior-point';

tic
estimacion=nlgreyest(datos,caja_gris,opt);
```

*Figura 9.* Definición del estimador y sus opciones de estimación

### 3.5. Análisis de sensores inerciales IMU

Para el análisis de los sensores inerciales se hizo una comparativa referente a los mismos. Se inició con el sensor MPU-6050, que estaba incorporado en el ROV inicialmente.



Figura 10. Sensor inercial que incluido en el ROV

El sensor cumple con características para la medición de aceleraciones y velocidades de giro, donde las variables principales son:

Primary Differences between MPU-6000 and MPU-6050

Part/Item	MPU-6000	MPU-6050
VDD	2.375V-3.46V	2.375V-3.46V
VLOGIC	n/a	1.71V to VDD
Serial Interfaces Supported	I <sup>2</sup> C	I <sup>2</sup> C
Pin 8	/CS	VLOGIC
Pin 9	AD0/SD0	AD0
Pin 23	SCL/SCLK	SCL
Pin 24	SDA/SDI	SDA

Tabla 1

Datos del sensor inercial incorporado en el ROV.

Tabla del giroscopio y acelerómetro

Full-scale range	+o- 250, 500, 1000 and 2000 °/sec
Resolution	16-bit ADCs
Gyroscope operating current	3.6mA
Standby current	5uA
Full-scale range	+o- 2g, 4g, 8g and 16g
Resolution	16-bit ADCs
Accelerometer normal operating current	500uA
Standby current	5uA

Tabla 2

*Datos del giroscopio y acelerómetro*

Este sensor se caracteriza por tener alta resolución en bits, su sensibilidad de aceleración es de + o – 2g. Sin embargo, las medidas resultan ruidosas, entonces cuando el vehículo realiza movimientos lentos, la aceleración representa un valor muy pequeño respecto al ruido, para arreglar esto se aplicó un filtro Yule-Walker realizado en Matlab el cual se explicará adelante.

## 4. Pruebas

### 4.1. Sumersión en el agua

El ROV del grupo de investigación CEMOS se encontraba en desuso, por esta razón, el primer paso fue su desensamble para verificar el funcionamiento de los elementos internos.



*Figura 11.* Imagen del ROV desensamblado

Luego de verificar y realizar un mantenimiento en sus partes internas se procedió a sumergir

el vehículo para comprobar su sellado. Al verificar que no fue positiva esta prueba se realizó un sello hermético con Silicona acética de curado por humedad. Este garantizó la protección contra filtraciones de agua, evitando posibles daños en la placa Rapsberry Pi 3 y demás componentes electrónicos.



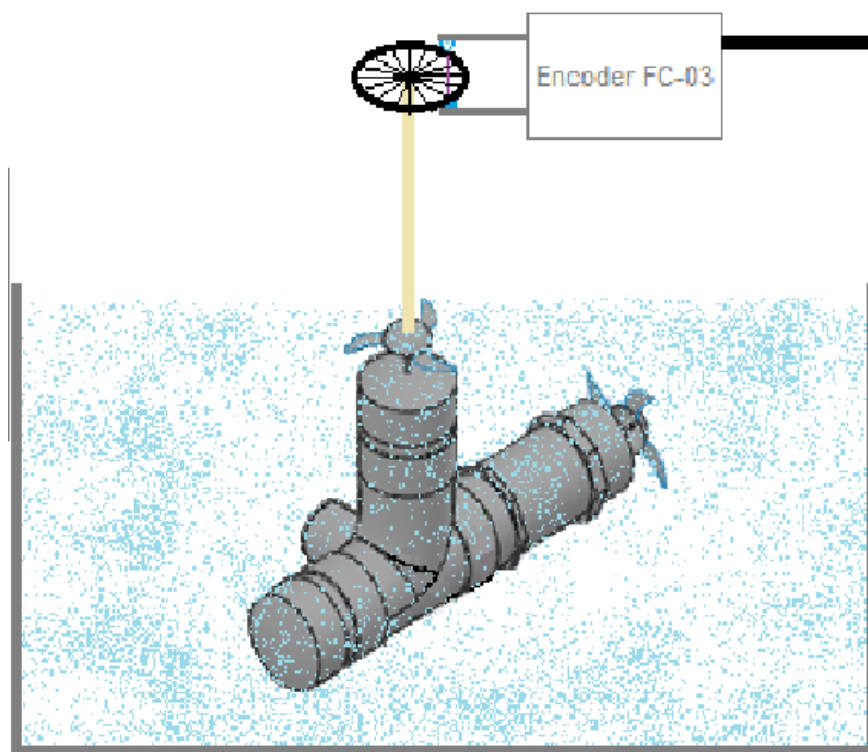
*Figura 12.* Imagen del ROV ensamblado

#### **4.2. Cálculo experimental de la velocidad angular de los motores**

La medición de velocidad de los motores son entradas del tanto para la estimación de los parámetros (modelo Idnlgrey), como para la estimación de los estados (filtro de Kalman). Por tanto, se verificaron las mediciones de velocidad angular de los 4 motores, en ambos sentidos de marcha. Para esto se utilizó un encoder optoelectrónico FC-03 conectado a una placa Arduino uno, que por medio de un algoritmo con interrupciones se calcula un valor aproximado de velocidad angular.

En esta medición se agregó un extensor del eje conectado a una rueda con muescas, que permite la medición directa de velocidad con el ROV completamente sumergido en el agua.

En la siguiente figura se muestra la toma de las velocidades mientras se encuentra sumergido el ROV a través del extensor de eje.



*Figura 13.* Calculo experimental de la velocidad angular

### 4.3. Sensor inercial IMU

Se realizaron medidas del sensor IMU MPU6050 que estaba instalado en el ROV, y se observó que dichas medidas no eran de buena calidad para bajas velocidades, ya que se perdían con el ruido del sensor. Se realizó el cambio por el sensor SEN-11028 de la marca SparkFun, que contiene internamente un sensor inercial MPU6050. Se midieron las aceleraciones y velocidades angulares

a través de un algoritmo implementado con las librerías de este módulo en la placa Rapsberry Pi 3 para ser extraídas y graficadas en la herramienta Matlab.

```
# Import the MPU6050 class from the MPU6050.py file
from mpu6050 import mpu6050
from time import time
from time import sleep
# Create a new instance of the MPU6050 class
sensor = mpu6050(0x68)
import math
import numpy as np
Ts = 0.1
t = 0
t0 = 0
n = 0
while True:
    t = time ()
    if t0 > t:
        t0 = 0
    while t - t0 >= Ts and n<=200:
        ##toma de datos del sensor, transformados en grados/segundos y metros/segundos
        accel_data = sensor.get_accel_data ()
        gyro_data = sensor.get_gyro_data ()
```

*Figura 14.* Código para la extracción de datos del sensor inercial

Primero se realizaron las medidas del sensor en reposo para capturar el ruido y de esta forma se implementaron varios tipos de filtros, tales como: Filtro de complemento y el filtro recursivo de Yule Walker, configurándose como un filtro tipo LPF y se diseñó de tal manera para que no interfiriera con las mediciones de la IMU.

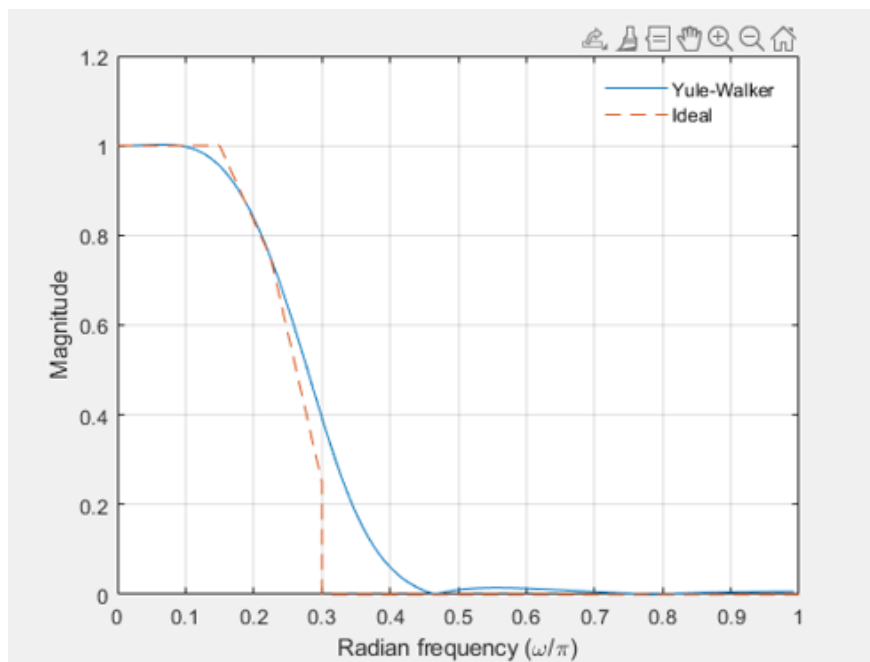


Figura 15. Representación grafica del filtro Yule-walk

Se realizaron pruebas de movimiento en el aire para comprobar el sentido de las medidas, las pruebas consistían en seguir rutas preestablecidas, para luego contrastar estas medidas con las gráficas resultantes en Matlab de la IMU.

Las primeras medidas no tuvieron resultados satisfactorios, así que se optó por comprar otro sensor que incluyera la unidad inercial MPU6050, pero de un costo mayor. Finalmente al probarlo se encontraron respuestas con resultados satisfactorios.

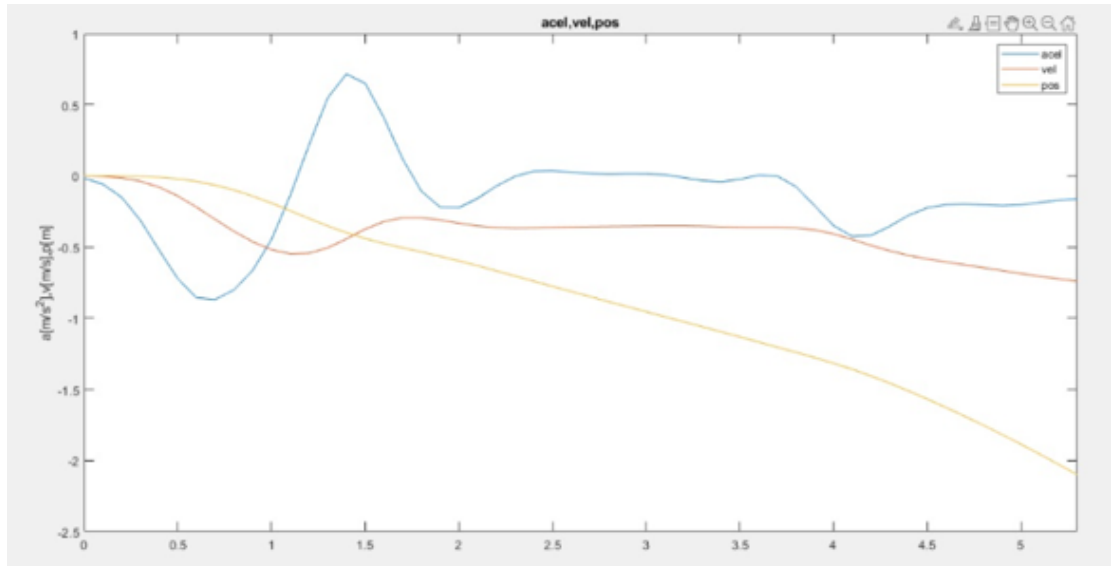


Figura 16. Gráficas del recorrido fuera del agua

#### 4.4. Pruebas de movimiento

Después de estar seguros que las medidas realizadas por el sensor inercial fuera satisfactorias, se programó un algoritmo que activara los propulsores con el fin de capturar el movimiento que realiza el ROV en el agua. Para esto se condicionó el algoritmo para que almacenara datos de la IMU por un tiempo preestablecido y luego se desactivara, pues el cordón umbilical del ROV limitaba el recorrido que se podía realizar bajo el agua.

## 5. Resultados

En este capítulo se presentarán los resultados obtenidos por SolidWorks, CFD, nlgrey, y el filtro de Kalman extendido.

### 5.1. Estimación de parámetros por medio de SolidWorks

Los parámetros físicos calculados al modelar el ROV en SolidWorks son:

Propiedades físicas		
Masa	3,675	Kg
Volumen	0,12937122	m <sup>3</sup>

Tabla 3

*Resultados de las propiedades físicas calculadas en SolidWorks*

Centro de masa		
X	0	m
Y	0	m
Z	0	m

Tabla 4

*Resultado del centro de masa calculado en SolidWorks*

Inercias		
Ix	0,03978411	kg*m <sup>2</sup>
Iy	0,04336443	kg*m <sup>2</sup>
Iz	0,05365461	kg*m <sup>2</sup>
Ixy	0,00358032	kg*m <sup>2</sup>
Ixz	0,05365461	kg*m <sup>2</sup>
Iyz	0,01029018	kg*m <sup>2</sup>

Tabla 5  
Resultados de las inercias calculadas en SolidWorks

## 5.2. Estimación de parámetros por medio de CFD

Se realizaron pruebas para los 3 ejes (tres para las velocidades lineales y tres para las velocidades angulares), teniendo así los 6 grados de libertad. Se realizaron 8 pruebas con distintas velocidades constantes en cada uno de estos grados de libertad, dado que el programa no permite realizar una simulación parametrizada en cada uno de los ejes. El intervalo escogido fue entre 0 y 3.5 m/s, pues éstas son velocidades que el ROV podría experimentar en la práctica, y son suficientes para capturar las dinámicas requeridas. En total fueron 48 simulaciones para obtener las tablas de velocidad versus fuerzas y momentos, mostradas a continuación:

**5.2.1. Movimiento lineal en el eje X.** En este movimiento la fuerza que aparece principalmente es de resistencia al movimiento en el eje x.

De esta prueba interesa la fuerza en x, que relaciona los parámetros  $X_u$  y  $X_{ulu}$ .

Velocidad(m/s)	Movimiento 'X' (N)
0	-0,00037937
0,5	-2,48872071
1	-9,9657294
1,5	-22,4001721
2	-39,7245428
2,5	-62,0015708
3	-89,2374075
3,5	-121,259062

Tabla 6

*Tabla resultados de fuerzas de resistencia debido al movimiento en x*

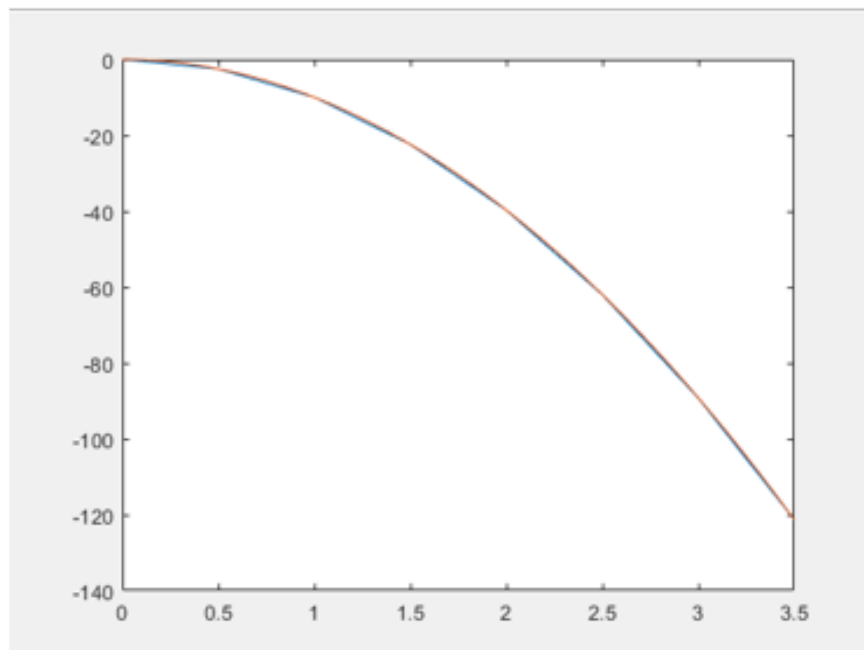


Figura 17. Resistencia del movimiento en x

La figura 17 muestra la velocidad (m/s) contra la fuerza (N) de resistencia hecha por el sólido. En esta se encuentra también la gráfica de la regresión cuadrática hecha para encontrar el valor de  $X_u$  y  $X_{u|u}$ .(12)

Con los datos de las fuerzas y velocidades se puede estimar los coeficientes hidrodinámicos, esta aproximación es hecha con la regresión cuadrática, generando los siguientes resultados:

Velocidad(m/s)	Movimiento 'X'(N)	UNIDADES	MSE
$X_u$	-4,1324	kg/s	0,0015
$X_{u u}$	-9,8648	kg/m	0,0015

Tabla 7

Tabla coeficientes debidos al movimiento en el eje X en función de la velocidad

**5.2.2. Movimiento lineal en el eje Y.** En este movimiento la fuerza que aparece principalmente es de resistencia al movimiento en el eje y.

De esta prueba nos interesa la fuerza en y, que nos relaciona los parámetros  $Y_u$  y  $Y_{ul}$ .

Velocidad(m/s)	Movimiento 'Y' (N)
0	4,648E-05
0,5	-3,7652026
1	-14,925448
1,5	-33,203234
2	-59,17235528
2,5	-92,644978
3	-133,35809
3,5	-181,23544

Tabla 8

*Tabla resultados de fuerzas de resistencia debido al movimiento en Y*

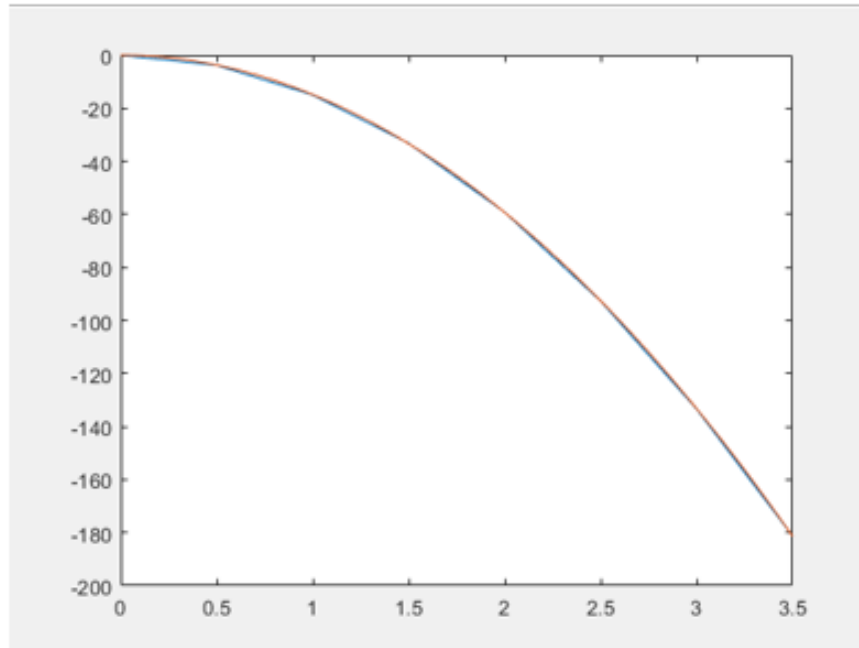


Figura 18. Resistencia del movimiento en Y

Con los datos de las fuerzas y velocidades se puede estimar los coeficientes hidrodinámicos. Esta aproximación es hecha con la regresión cuadrática, generando los siguientes resultados:

Velocidad(m/s)	Movimiento'X'(N)	UNIDADES	MSE
$Y_v$	-0,0539	kg/s	0,0113
$Y_{v v}$	-14,7866	kg/m	0,0113

Tabla 9

Tabla coeficientes hidrodinámicos debidos al movimiento en el eje Y en función de la velocidad

**5.2.3. Movimiento lineal en el eje Z.** En este movimiento la fuerza que aparece principalmente es de resistencia al movimiento en el eje z.

De esta prueba nos interesa la fuerza en z, que nos relaciona los parámetros  $Z_u$  y  $Z_{ul}$ .

Velocidad(m/s)	Movimiento 'Z' (N)
0	12,17183
0,5	8,4625925
1	-2,2664157
1,5	-19,786111
2	-44,453421
2,5	-82,88
3	-116,18026
3,5	-163,38821

Tabla 10

*Tabla resultados de fuerzas de resistencia debido al movimiento en Z*

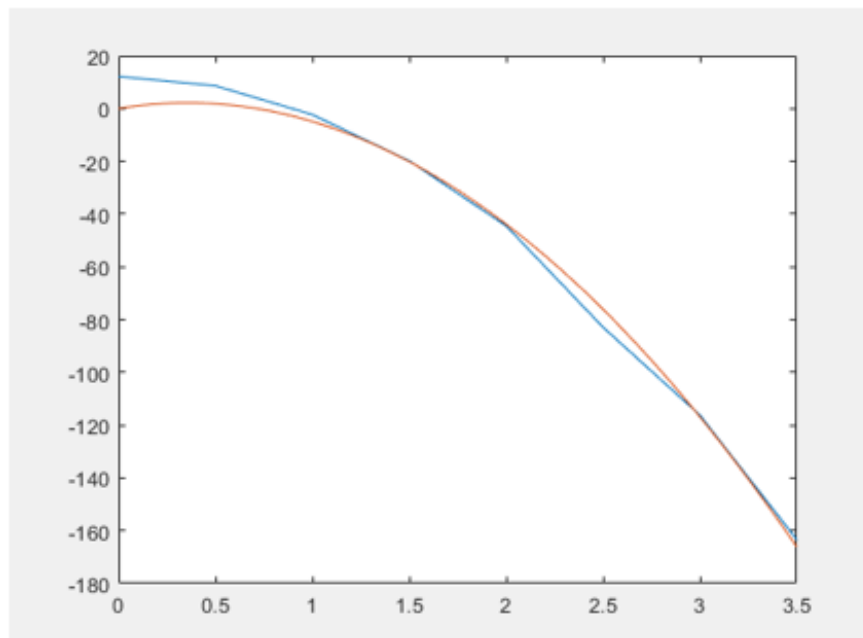


Figura 19. Resistencia del movimiento en Z

Con los datos de las fuerzas y velocidades se puede estimar los coeficientes hidrodinámicos, esta aproximación es hecha con la regresión cuadrática, generando los siguientes resultados:

Velocidad(m/s)	Movimiento' X' (N)	UNIDADES	MSE
$Z_w$	-14,1365	kg/s	14,6715
$Z_{w w}$	-17,0298	kg/m	14,6715

Tabla 11

Tabla coeficientes hidrodinámicos debidos al movimiento en el eje Z en función de la velocidad

**5.2.4. Movimiento giratorio respecto al eje X.** Valores de los momentos en el eje x debido a la velocidad angular de giro del fluido a la que fue sometida el sólido.

Velocidad de giro(m/s)	MovimientoX'K'(N*m)
0	-2,16475
0,5	-2,254633
1	-2,580207
1,5	-3,007087
2	-3,577397
2,5	-4,544108
3	-5,539047
3,5	-6,80427

Tabla 12

*Tabla resultados de momentos en x debido al movimiento de giro en el eje X*

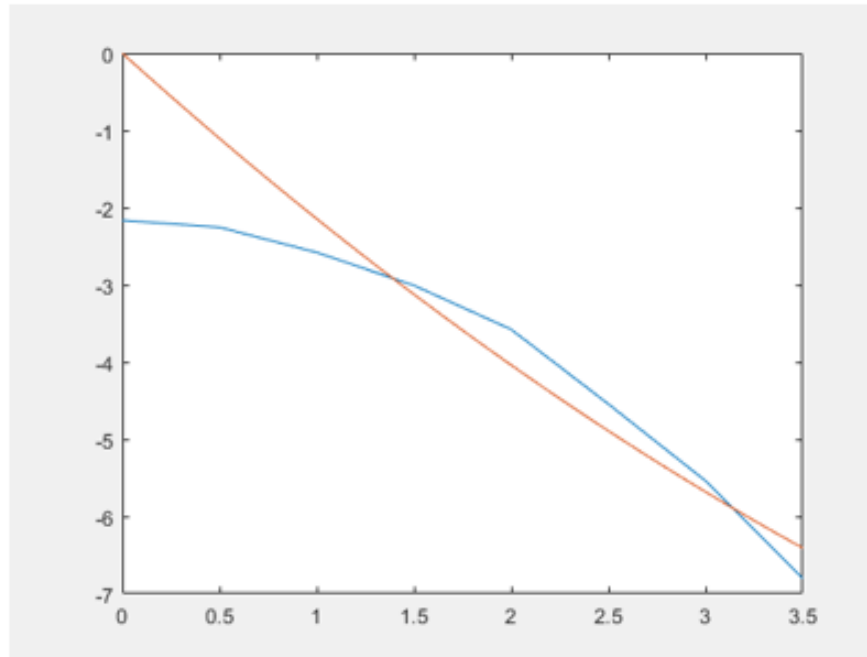


Figura 20. Momento en X del vehículo rotando en el eje X

Con los datos de los momentos y velocidades se pueden estimar los coeficientes hidrodinámicos, esta aproximación es hecha con la regresión cuadrática, generando los siguientes resultados:

CFD	VALOR	UNIDADES	MSE
$K_p$	-2,2702	kg*rad/s	1,1206
$K_{p p}$	-0,1254	kg*m <sup>2</sup> /rad <sup>2</sup>	1,1206

Tabla 13

Tabla Coeficientes hidrodámicos debidos al movimiento en X en función de la velocidad

**5.2.5. Movimiento giratorio respecto al eje Y.** Valores de los momentos en el eje y debido a la velocidad angular de giro del fluido a la que fue sometida el sólido.

Velocidad de giro(m/s)	MovimientoY'K'(N*m)
0	-0,15503
0,5	-0,15919
1	-0,17429
1,5	-0,18572
2	-0,20904
2,5	-0,23901
3	-0,27607
3,5	-0,3141

Tabla 14

*Tabla resultados de momentos debido al movimiento de giro en el eje Y*

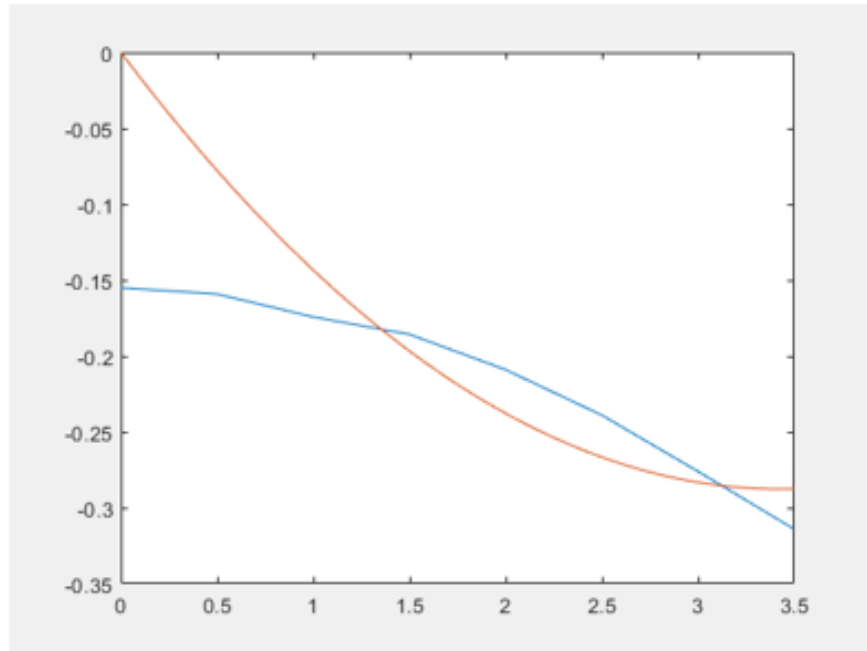


Figura 21. Momento en y del vehículo rotando en el eje Y

Con los datos de los momentos y velocidades se pueden estimar los coeficientes hidrodinámicos, esta aproximación es hecha con la regresión cuadrática, generando los siguientes resultados:

CFD	VALOR	UNIDADES	MSE
$M_p$	-0,1679	kg*rad/s	0,0962
$M_{p p}$	-0,0245	kg*m <sup>2</sup> /rad <sup>2</sup>	0,0962

Tabla 15

Tabla Coeficientes hidrodámicos debidos al movimiento en Y en función de la velocidad

**5.2.6. Movimiento giratorio respecto al eje Z.** Valores de los momentos en el eje z debido a la velocidad angular de giro del fluido a la que fue sometida el sólido.

Velocidad de giro(m/s)	MovimientoZ'N'(N*m)
0	-0,000634
0,5	-0,081259
1	-0,3108036
1,5	-0,6977694
2	-1,1758961
2,5	-1,9396184
3	-2,5577307
3,5	-3,3080043

Tabla 16

*Tabla resultados de momentos en Z debido al movimiento de giro en el eje X*

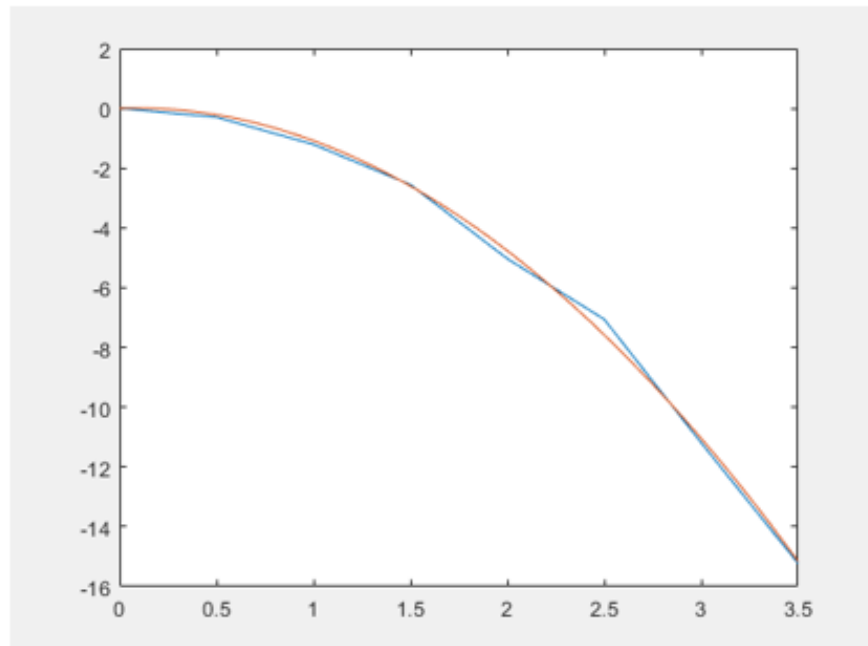


Figura 22. Momento en y del vehículo rotando en el eje Z

Con los datos de los momentos y velocidades se pueden estimar los coeficientes hidrodinámicos, esta aproximación es hecha con la regresión cuadrática, generando los siguientes resultados:

CFD	VALOR	UNIDADES	MSE
$N_r$	-0,1228	kg*rad/s	0,0047
$N_{r r}$	-0,2403	kg*m <sup>2</sup> /rad <sup>2</sup>	0,0047

Tabla 17

Tabla Coeficientes hidrodámicos debidos al movimiento en Z en función de la velocidad

**5.2.7. Parámetros estimados por CFD.** Se estimaron en total 12 parámetros, dejando 11 parámetros desconocidos para estimar con Idnlgrey.

Coeficientes lineales de amortiguamiento hidrodinámico			
CFD	VALOR	UNIDADES	MSE
Xu	-4,1324	kg/s	0,00145691
Yv	-0,0539	kg/s	0,01130712
Zw	-12,1365	kg/s	14,26715027
Kp	-2,2702	kg*rad/s	1,12061863
Mq	-0,1679	kg*rad/s	1,12061863
Nr	-0,1228	kg*rad/s	0,00468861

Tabla 18

*Tabla Coeficientes hidrodinámicos lineales de amortiguamiento*

Coeficientes hidrodinámicos cuadráticos de amortiguamiento			
CFD	VALOR	UNIDADES	MSE
$X_{u u}$	-9,8648	kg/m	0,00145691
$Y_{v v}$	-14,7866	kg/m	0,01130712
$Z_{w w}$	-17,0298	kg/m	14,26715027
$K_{p p}$	-0,1254	$\text{kg}\cdot\text{m}^2/\text{rad}^2$	1,12061863
$M_{q q}$	-0,0245	$\text{kg}\cdot\text{m}^2/\text{rad}^2$	1,12061863
$N_{r r}$	-0,2403	$\text{kg}\cdot\text{m}^2/\text{rad}^2$	0,00468861

Tabla 19

*Tabla Coeficientes cuadráticos*

### 5.3. Obtención de datos de entrada.

Las velocidades angulares de los motores se obtuvieron experimentalmente mediante el uso del sensor encoder infrarrojo FC-03 de Arduino. Se realizó un algoritmo en el IDE de Arduino, que se implementó en una placa Arduino Uno, mediante un código que contaba las interrupciones entre el led infrarrojo y el fototransistor que está permanentemente recibiendo la señal del infrarrojo, éste conteo se realiza durante 1 segundo y realiza la conversión de pulsos a RPM.

```

Cuenta_rpm$

int encoder_pin = 2;           // Pin 2, donde se conecta el encoder
unsigned int rpm = 0;         // Revoluciones por minuto calculadas.
volatile byte pulses = 0;     // Número de pulsos leídos por el Arduino en un segundo
unsigned long timeold = 0;    // Tiempo
unsigned int pulsesperturn = 20; // Número de muescas que tiene el disco del encoder.
const int wheel_diameter = 24; // Diámetro de la rueda pequeña[mm]

void setup(){
  Serial.begin(9600);         // Configuración del puerto serie
  pinMode(encoder_pin, INPUT); // Configuración del pin n°2
  attachInterrupt(0, counter, RISING); // Configuración de la interrupción 0, donde esta conectado.
  pulses = 0;
  rpm = 0;
  timeold = 0;
  Serial.print("RPM ");
}

void loop(){
  if (millis() - timeold >= 1000) // Se actualiza cada segundo
  {
    noInterrupts(); // Desconectamos la interrupción para que no actúe en esta parte del programa.
    rpm = (40 * 1000 / pulsesperturn) / (millis() - timeold) * pulses; // Calculamos las revoluciones por minuto
    timeold = millis(); // Almacenamos el tiempo actual.
    Serial.print(rpm, DEC); // Impresión de las rpm
    pulses = 0; // Inicializamos los pulsos.
    interrupts(); // Reiniciamos la interrupción
  }
}

void counter(){
  pulses++;
}

```

Figura 23. Código para leer los RPM

Después del conteo de RPM se hace la conversión a velocidad angular multiplicando el valor de RPM medido por 0,1047197551.

Se realizaron las pruebas para los cuatro motores, en ambos sentidos obteniendo lo siguientes resultados:

Motor	Sentido	RPM	Velocidad angular [rad/s]
Motor superior brazo derecho	+	200	23,0383
Motor superior brazo derecho	-	234	24,5044
Motor superior brazo izquierdo	+	222	23,2477
Motor superior brazo izquierdo	-	233	24,3997
Motor trasero derecho	+	450	47,1238
Motor trasero derecho	-	445	46,6002
Motor trasero izquierdo	+	445	46,6002
Motor trasero izquierdo	-	440	46,0766

Tabla 20  
*Conversiones de RPM a velocidades angulares*

Una vez convertidos los RPMs a velocidades angulares se creó un vector con la longitud de los datos de salida de las pruebas de movimiento, se elevó al cuadrado cada valor de ese vector con el fin de cumplir las especificaciones de entrada del modelo.

#### **5.4. Estimación de parámetros por medio de Idnlgrey**

Antes de aplicar la estimación por CFD, se intentó estimar todos los parámetros del modelo directamente con no linear grey box, ya que inicialmente habían 20 parámetros desconocidos. Cuando se hizo la estimación se establecieron unas condiciones iniciales para cada parámetro desconocido,

pero en los resultados de las simulaciones los resultados arrojados por el estimador no variaban de las condiciones iniciales. Se concluyó que el programa se estaba quedando en un mínimo local. Cada parámetro está representado en este vector respectivamente:  $[B, X_u, Y_v, Z_w, K_p, M_q, N_r, X_{u\dot{}}, Y_{v\dot{}}, Z_{w\dot{}}, K_{p\dot{}}$

Una vez calculados los parámetros estimados por CFD, el vector de incógnitas se redujo a el siguiente:  $[B, X_{u\dot{}}, Y_{v\dot{}}, Z_{w\dot{}}, K_{p\dot{}}, M_{q\dot{}}, N_{r\dot{}}, C_{plus1}, C_{plus2}, C_{plus3}, C_{plus4}]$ . Cabe resaltar que para datos atrás y abajo los Cplus se representaron como valores de Cminus. Se establece el valor de cada parámetro en el modelo del ROV, se halló aplicando los códigos que ya se explicaron en la sección de no linear grey box. Los resultados faltantes que generó el modelo de identificación son los presentado en la tabla 21. Una vez obtenidos todos los parámetros, se definió el modelo completo.

Parámetro	Idnlgrey
B	40,90175
Xu-dot	-3,8327
Yv-dot	-21,2164
Zw-dot	-30,521
Kp-dot	-0,64255
Mq-dot	-1,01728
Nr-dot	-3,025650
C1+	4,60e-04
C2+	3,90e-04
C3+	6,30e-04
C4+	6,48e-04
C1-	7,30e-04
C2-	8,38e-04
C3-	6,48e-04
C4-	6,48e-04

Tabla 21

*Tabla resultados de estimación de Idnlgrey*

**5.4.1. Validación del modelo no lineal.** Para validar el modelo no lineal se utilizó el error RMSE y el error absoluto, obtenidos del modelo implementado en Simulink como se muestra en la figura 25, donde se obtuvieron los resultados de la tabla 22.

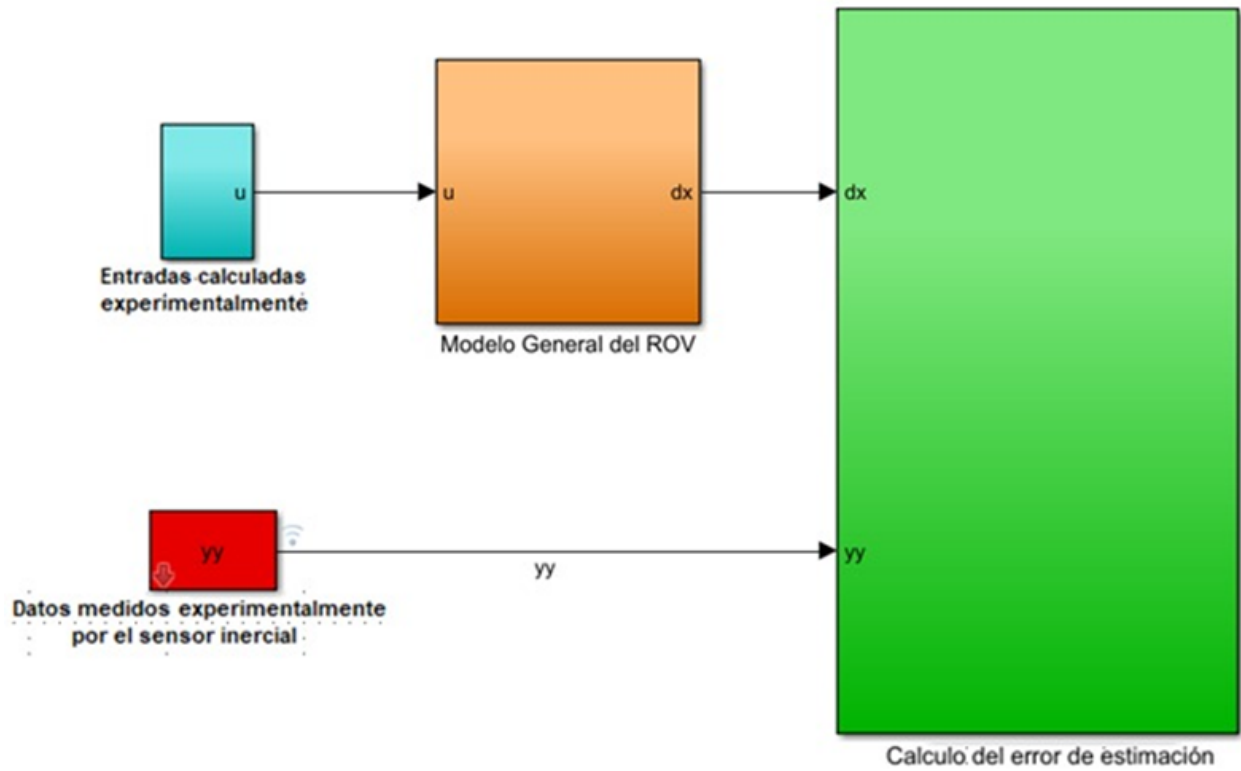


Figura 24. Diagrama de bloques para la validación del modelo

Salida	RMSE
ax	0,617008914
ay	0,897552227
az	0,582065288
gx	11,85326959
gy	9,805610639
gz	7,580237463

Tabla 22

*Tabla error RMSE para el modelo no lineal*

Salida	Error Absoluto
ax	0,4765
ay	0,6784
az	0,4268
gx	7,12
gy	5,536
gz	6,098

Tabla 23

*Tabla error absoluto para el modelo no lineal*

### 5.5. Matriz de covarianza

Para el cálculo de la matriz de covarianza se tomaron datos del sensor en reposo, con el fin de capturar el ruido agregado.

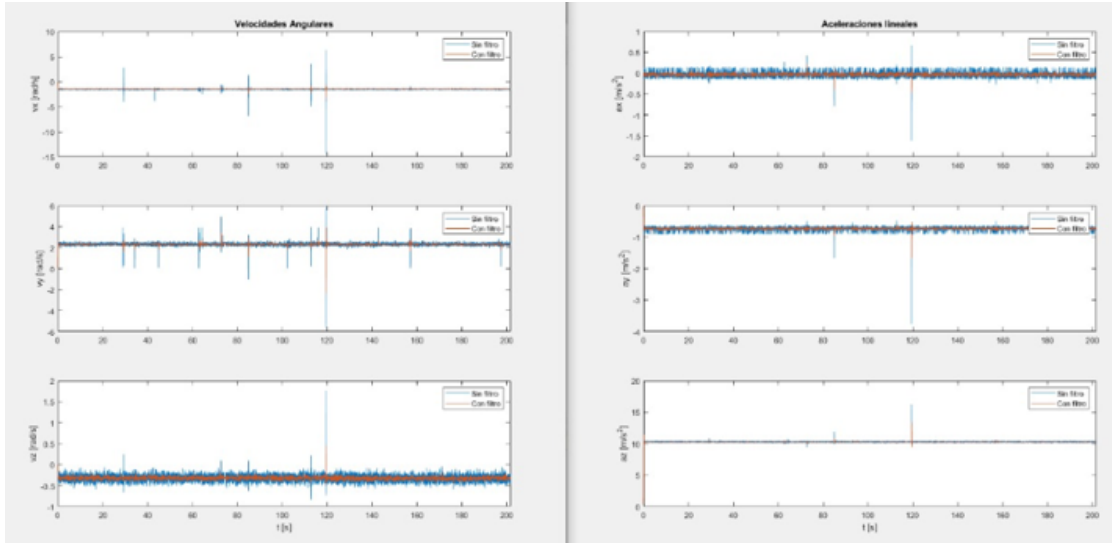


Figura 25. Datos de ruido de aceleraciones y giros con filtro y sin filtro para hallar matriz de covarianza

En la herramienta Matlab se generó un algoritmo que permite hallar el promedio de los datos de aceleración y velocidad angular de giro, que son guardados en vectores de datos. Se restó el valor promedio a cada medida tomada respectivamente para hallar vectores la varianza, y seguido de esto se realizó una multiplicación de la transpuesta por el mismo vector dividido entre el número de datos para hallar finalmente la covarianza en cada uno de los 6 grados de libertad.

```
Promedios_Co.m  X  +
4 %% Asignación de vectores de datos para los 6 ejes
5 - ax = datos6(:,8);
6 - ay = datos6(:,9);
7 - az = datos6(:,10);
8 - gx = datos6(:,5);
9 - gy = datos6(:,6);
10 - gz = datos6(:,7);
11 %% Calculo de los promedios
12 - prom_ax = mean(ax);
13 - prom_ay = mean(ay);
14 - prom_az = mean(az);
15 - prom_gx = mean(gx);
16 - prom_gy = mean(gy);
17 - prom_gz = mean(gz);
18 %% Calculo de las varianzas
19 - vax = ax - prom_ax;
20 - vay = ay - prom_ay;
21 - vaz = az - prom_az;
22 - vgx = gx - prom_gx;
23 - vgy = gy - prom_gy;
24 - vgz = gz - prom_gz;
25 %% Calculo de las covarianzas
26 - cax = (vax'*vax)/length(vax);
27 - cay = (vay'*vay)/length(vay);
28 - caz = (vaz'*vaz)/length(vaz);
29 - cgx = (vgx'*vgx)/length(vgx);
30 - cgy = (vgy'*vgy)/length(vgy);
31 - cgz = (vgz'*vgz)/length(vgz);
```

Figura 26. Código de promedios para hallar la covarianzas.

Se construyó una matriz diagonal, llamada R. Esta contiene las covarianzas de las aceleraciones lineales y de las velocidades angulares de giro, obteniendo la siguiente matriz de covarianza.

$$R = \begin{bmatrix} 0.0008 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0008 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.0007 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.0016 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0826 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0019 \end{bmatrix} \quad (43)$$

### 5.6. Comparación de estados por medio de Simulink.

Para comparar los estados estimados se hizo un montaje en simulink con el filtro de Kalman extendido elaborado por el grupo CEMOS, en él se programó el modelo del ROV con los parámetros previamente estimados.

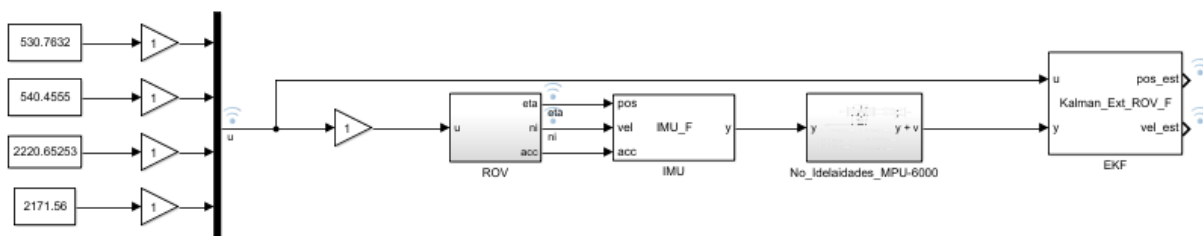


Figura 27. Montaje del filtro de Kalman extendido en simulink

Recordando que el filtro de Kalman es un observador de estados, se realizaron 2 tipos de simulaciones: la primera con un bloque IMU simulado y con los parámetros hallados del modelo tal

como se muestra en la figura 27. La segunda con los datos extraídos en las pruebas en movimiento de la IMU instalada en el ROV como se muestra en la figura 34. En ambas simulaciones se varía el periodo de muestreo del filtro de Kalman y se comparan los estados del modelo del ROV con los estados estimados. Cabe resaltar que este tipo de filtros son ampliamente utilizados en sistemas de navegación para identificar estados no medibles.

Los resultados obtenidos son:

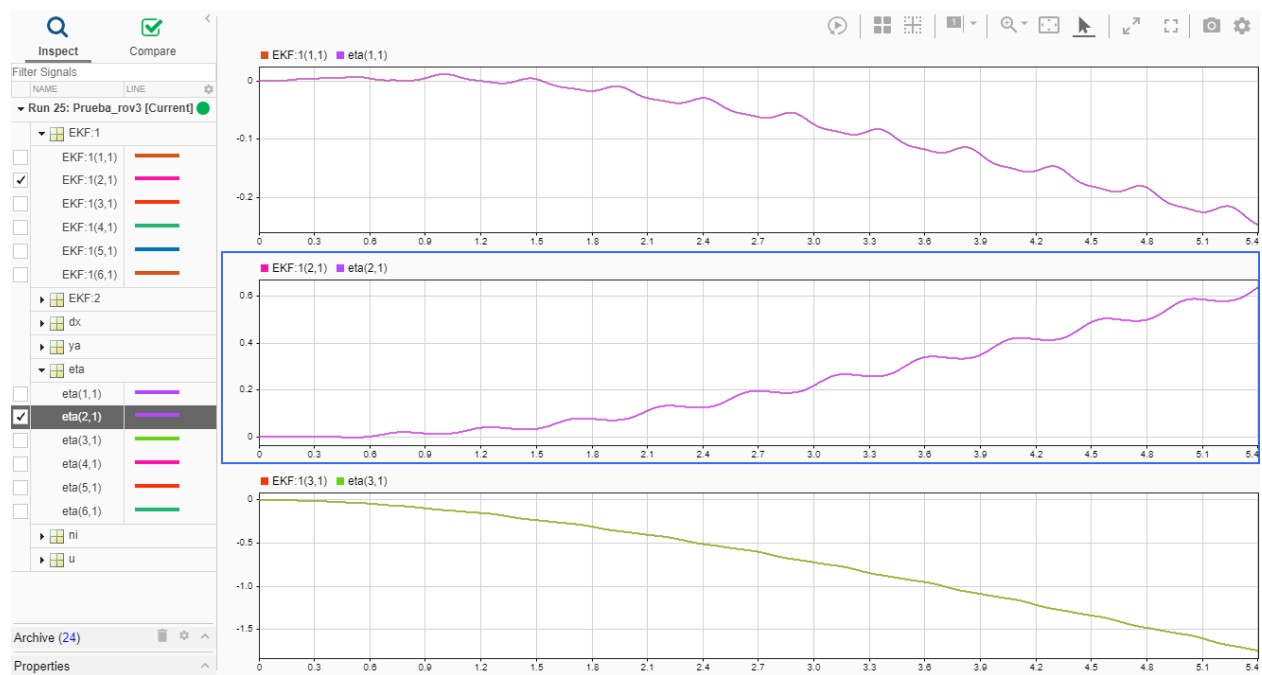


Figura 28. Resultados gráficos de los estados del modelo y el filtro de Kalman extendido con periodo de muestreo de 0.001

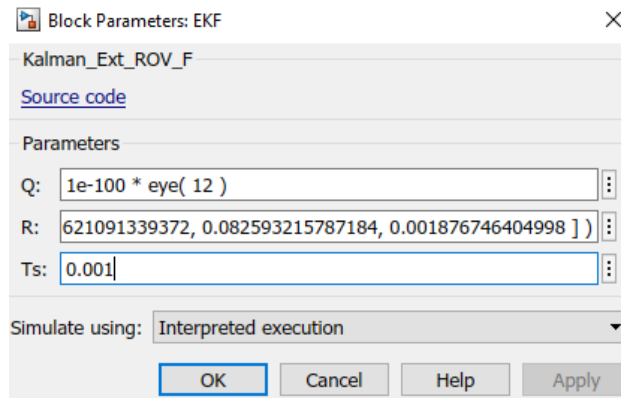


Figura 29. Datos matrices Q,R y Ts=0.001

Como se observa de las gráficas, los estados tienden a ser iguales a los estados estimados, cuando se simuló con un tiempo de muestro de 0.001 segundos. La estimación es buena, sin embargo, este filtro puede no ser implementable debido a que el periodo de muestreo es muy exigente y el modelo muy grande.



Figura 30. Resultados gráficos de los estados del modelo y el filtro de Kalman extendido con periodo de muestreo de 0.01

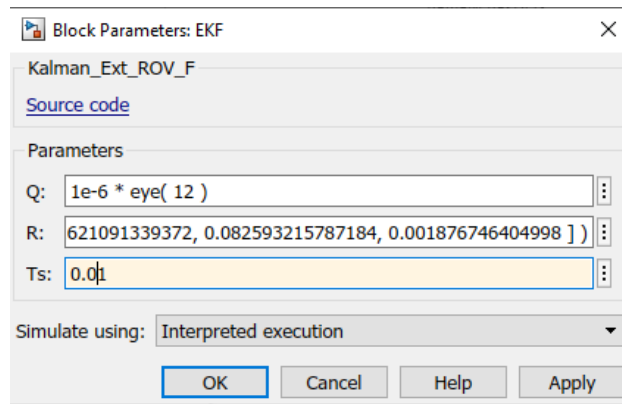


Figura 31. Datos matrices Q,R y  $T_s=0.01$

La figura 31 muestra los resultados de la estimación cuando se tiene un periodo de muestreo es 0.01 segundos.

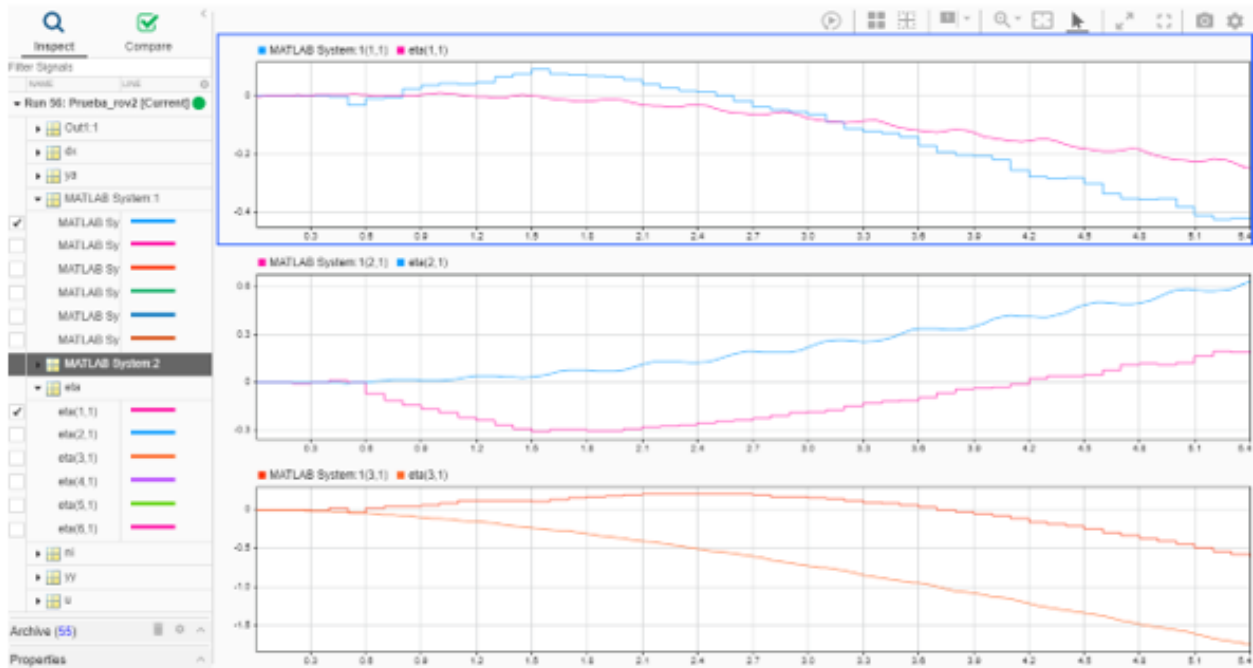


Figura 32. Datos gráficos del filtro de Kalman

La figura 32 muestra los resultados de la estimación cuando se tiene un periodo de muestreo es 0.1 segundos.

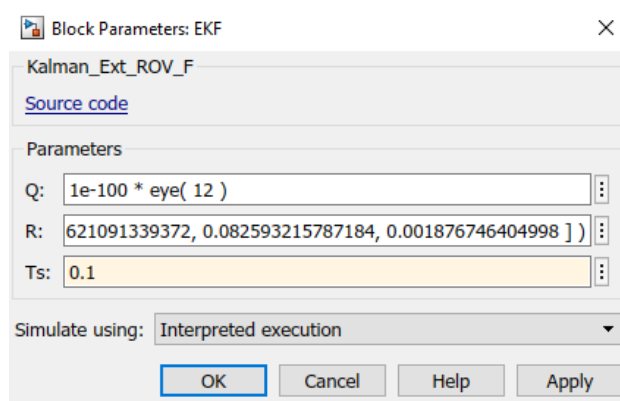


Figura 33. Datos matrices Q,R y Ts=0.1

La matriz Q representa la covarianza de ruido del proceso. Dado que no se puede saber a ciencia

cierta que tanto se aproxima el modelo a la dinámica de la planta real, se sintonizó el valor de esta matriz mediante pruebas de simulación. En la figura 34 se muestra la comparación de estados estimados en la segunda simulación, en la que se utilizó los datos de salida tomados por la IMU instalada en el ROV.

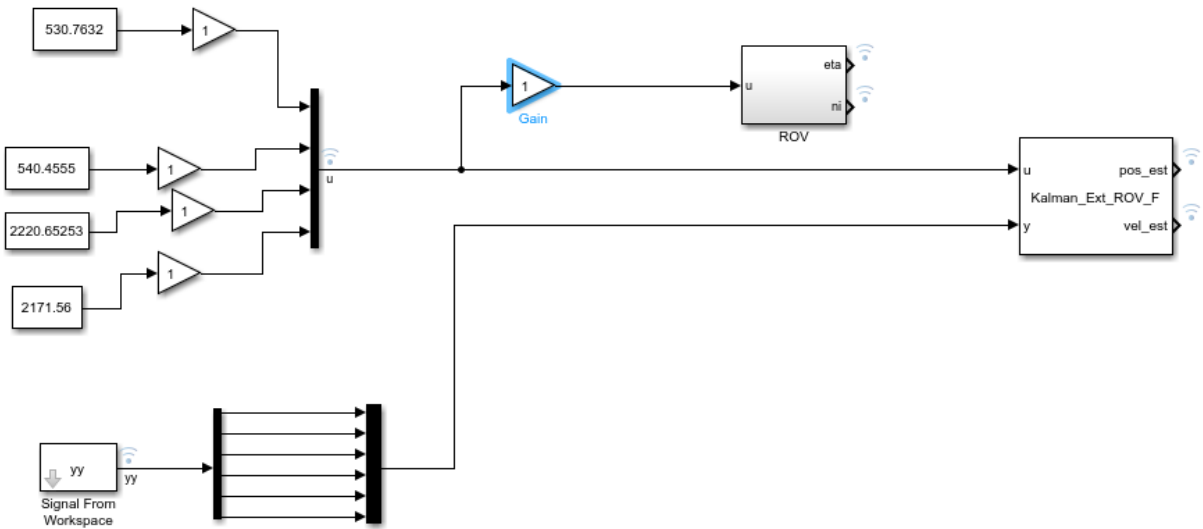


Figura 34. Filtro simulado con los datos extraídos en el sensor instalado en el ROV

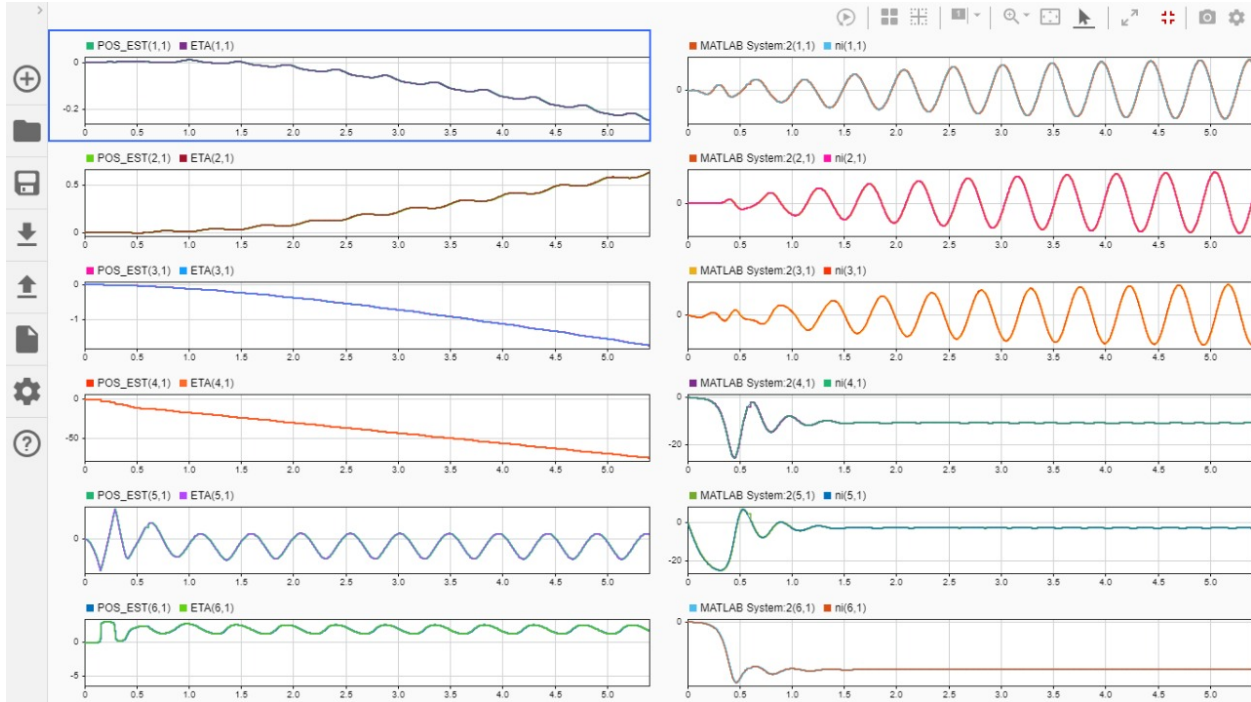


Figura 35. Datos del sensor instalado en el ROV con  $T_s=0.001$ .

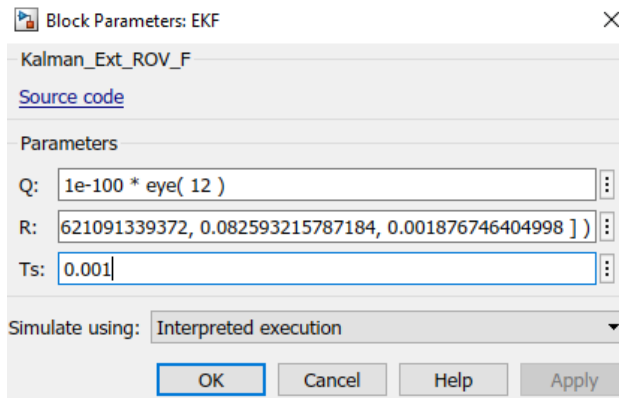


Figura 36. Datos matrices Q,R y  $T_s=0.001$  con la IMU instalada en el ROV.

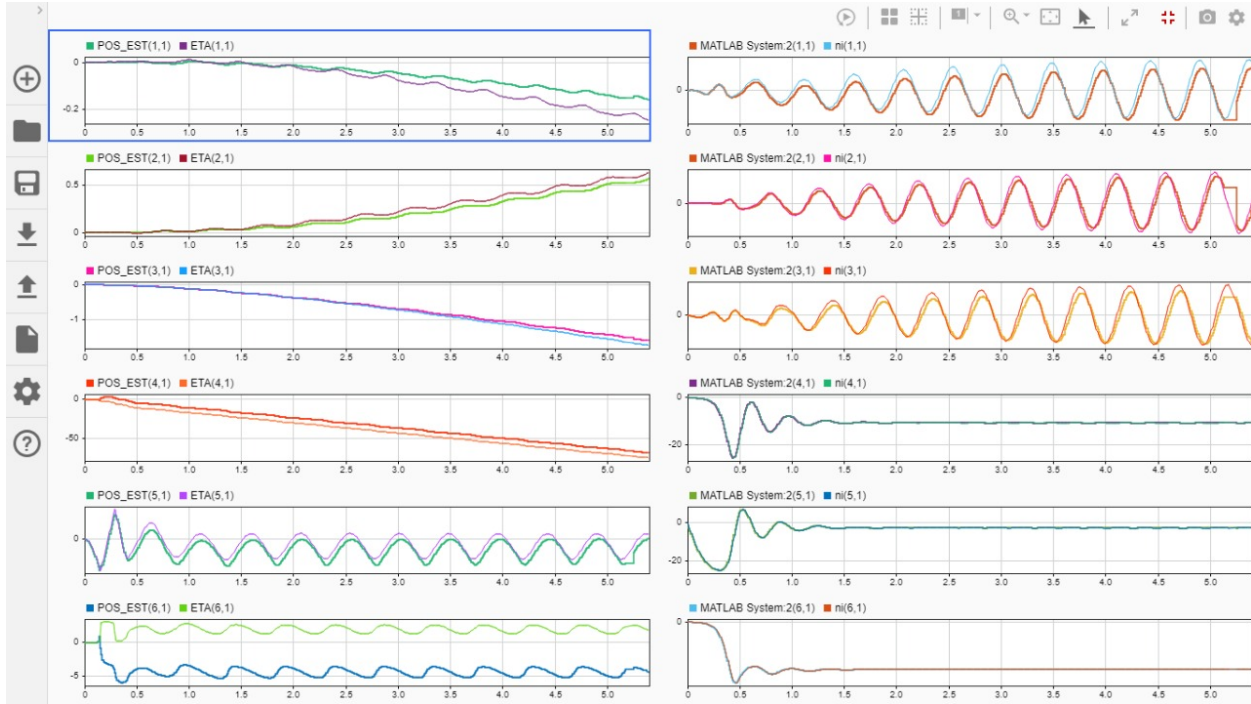


Figura 37. Datos del sensor instalado en el ROV con  $T_s=0.01$ .

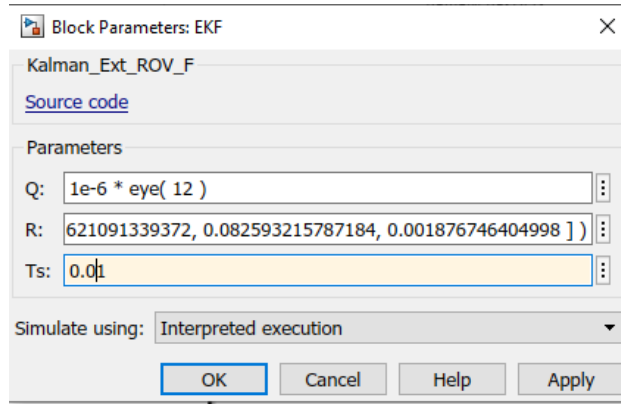


Figura 38. Datos matrices Q,R y  $T_s=0.01$  con la IMU instalada en el ROV.

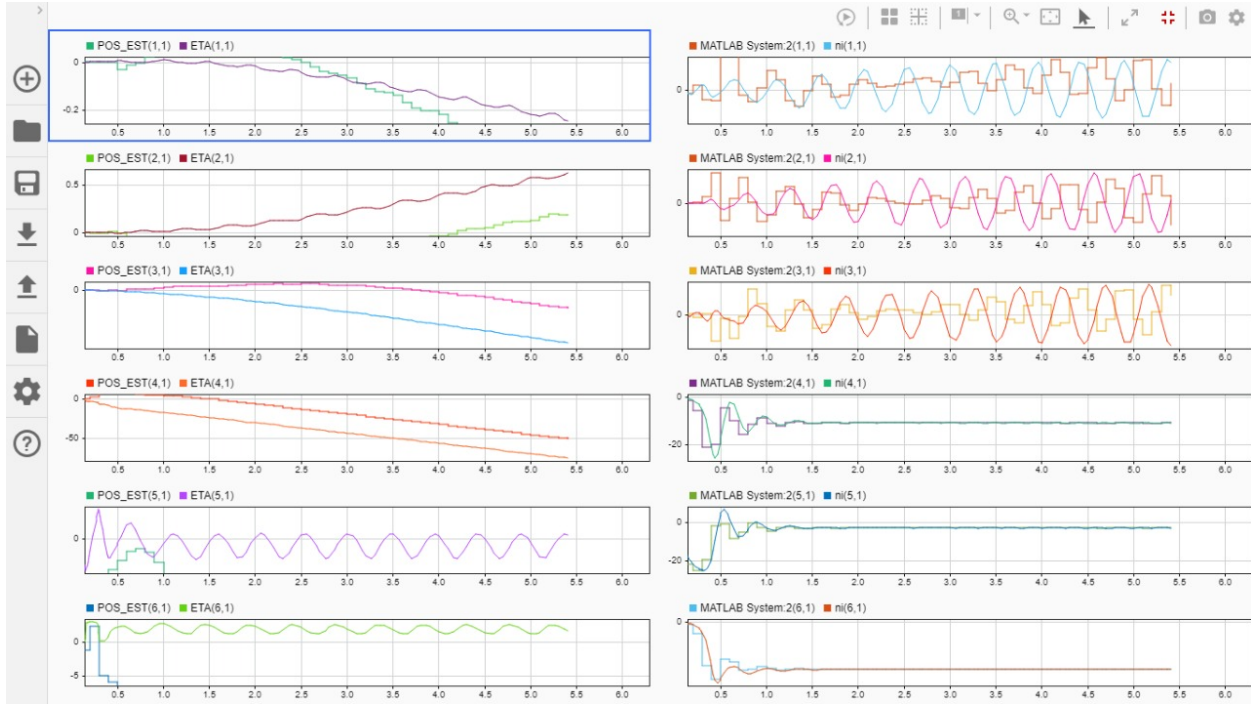


Figura 39. Datos del sensor instalado en el ROV con  $T_s=0.1$ .

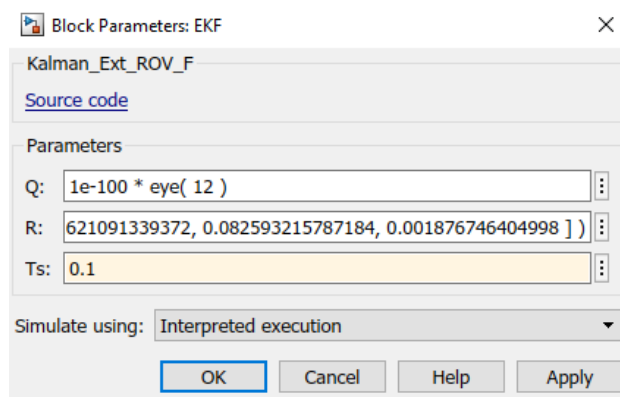


Figura 40. Datos matrices Q,R y  $T_s=0.1$  con la IMU instalada en el ROV.

**5.6.1. Validación del Filtro de Kalman extendido.** Se validaron los estados del modelo y los del filtro de Kalman con el error RMSE y el error absoluto para los diferentes periodos de muestreo como se muestra en la figura 41, cuyos resultados se presentan en las tablas 24 y ??

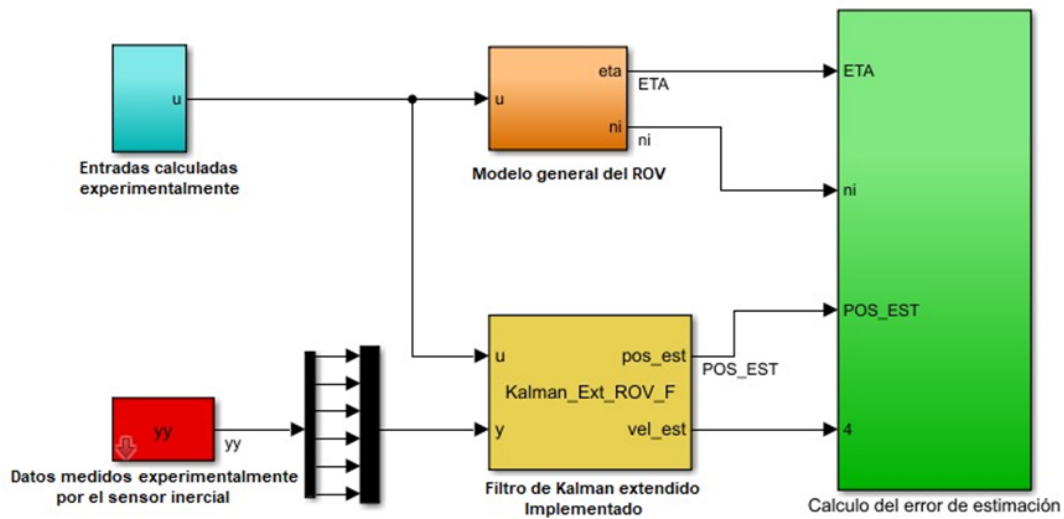


Figura 41. Diagrama de bloques implementando la validación del error.

Estado	RMSE		
	ts=0.1s	ts=0.01s	ts=0.001s
x	0.086654486	0.038392708	6.06E-05
y	0.354964787	0.042790186	1.24E-04
z	0.248977911	0.059949979	2.3E-04
$\phi$	23.075961526	3.347440429	1.22E-02
$\theta$	2.648395741	0.269685001	4.42E-03
$\psi$	14.230249476	6.243396512	1.00E-02
u	0.529905652	0.14049911	2.61E-03
v	0.309919344	0.055641711	1,65E-03
w	0.19690099	0.040224371	1.03e-03
p	1.965960325	0.237528946	2,33E-02
q	2,17324642	0,271311629	2,68E-02
r	0,570876519	0,056789083	5,55E-03

Tabla 24  
*Tabla de error RMSE*

Estado	Error absoluto		
	ts=0.1s	ts=0.01s	ts=0.001s
x	0,07088	0,0284	7,67E-09
y	0,3263	0,03487	0,00015
z	0,6777	0,04085	0,0002872
$\phi$	22,05	6,206	0,01339
$\theta$	2,546	0,241	0,007178
$\psi$	13,81	6,149	0,006304
u	0,4044	0,105	0,003777
v	0,2393	0,03624	0,002364
w	0,1521	0,02631	0,0015
p	1,17	0,1783	0,01536
q	1,477	0,2023	0,01855
r	0,2984	0,03757	0,003296

Tabla 25  
*Tabla error absoluto*

## 6. Conclusiones

- Al implementar un filtro de Kalman extendido con un periodo de muestreo pequeño se mejora la estimación, donde al usar un periodo de muestreo de 0.001 s, su estimación era casi perfecta, sin embargo, en las condiciones de operación real, el filtro y el modelo tienen una gran carga computacional, haciendo que ese periodo de muestreo no se cumpla, lo que puede resultar en una de las limitantes a la hora de la implementación.
- Los costos elevados en dispositivos electrónicos generalmente se traducen en mejores prestaciones. Con respecto a sensores, poseen mejor relación señal a ruido, lo que ocurrió con los datos provenientes de las mediciones de la IMU inicial en la se encontraban muy ruidosos, y no fue suficiente con utilizar filtros y correcciones de señal. Por lo tanto, se optó en comprar el sensor SEN 11028 de la marca SparkFun, con un costo que supera en 10 veces el valor del sensor instalado originalmente en el ROV. Es indispensable para este tipo de proyectos en los que se requiere de alta exactitud contar con sensores de alta calidad.
- El uso de estimadores de parámetros desconocidos de un modelo es bastante útil cuando se tienen puntos iniciales cercanos al valor buscado, y buenos puntos extremos para acotar la búsqueda. Dejar intervalos infinitos genera mayor gasto computacional, lo que traduce en largas jornadas de búsqueda, que pueden tomar incluso días.
- Los estimadores como idnlgrey tienden a buscar un mínimo local cercano al punto inicial de búsqueda, y por eso es tan importante asignarle un buen punto de partida. Una buena

opción es utilizar un método de varios agentes de búsqueda, tal como el de optimización por enjambre de partículas PSO por sus siglas en inglés, pero por el tamaño del modelo no era viable utilizarlo en este proyecto, dado que el gasto computacional podría tardar incluso semanas en una primera prueba.

- Para mejorar los resultados de estimación no solo se deben tener en cuenta el sensor inercial IMU. Además de éste, se debe complementar con más sensores como barómetro y magnetómetro, pues a la hora de tener una obtención de datos muy extensa se cometen errores de medida dada las integraciones en medio del proceso de conversión de medidas.

## 7. Trabajos futuros

- Se puede hacer la implementación de un controlador con el modelo estimado, además hacer uso del filtro de Kalman para utilizar controladores modernos.
- Hacer las adecuaciones para adicionar más sensores como un barómetro, magnetómetro entre otros, que ayuden a mejorar la precisión de las medidas, el sensor inercial no es suficiente para poder navegar el ROV durante un tiempo prolongado, la estimación de posición no tendría una medida adicional para comparar si se está generando errores, esto podría dañar la trayectoria del vehículo.
- Hacer modificaciones físicas en el ROV, que permitan la integración de un cable de datos a través del cordón umbilical, esto podría traer beneficios como la instalación de una cámara para poder ver en tiempo real el entorno del vehículo, así mismo efectuar nuevos comandos mientras se está en movimiento, ya que las señales de wifi y bluetooth son atenuadas por el cambio de medio haciendo casi imposible efectuar cambios en la trayectoria del ROV mientras se encuentra sumergido.

### Referencias Bibliográficas

Alfonso fuentes, E., 2014. DISEÑO Y SIMULACIÓN DE UN CONTROLADOR DE POSICIÓN PARA UN VEHÍCULO SUBMARINO OPERADO REMOTAMENTE (ROV) USANDO BACKSTEPPING CONTROL. 1st ed. Bucaramanga: Universidad Industrial De Santander, pp.(15-21).

Datasheet.es. 2020. MOCH22A Datasheet ( PDF ) - OPTO Interrupter. [online] Available at: <<http://www.datasheet.es/PDF/685501/MOCH22A-pdf.html>>[Accessed 1 October 2020].

de la Red Calvo, J., 2015. Modelado, Simulación Y Control De Un Vehículo Submarino Manipulado De Forma Remota (ROV). 1st ed. Cartagena: Universidad Politécnica de Cartagena, pp.(24-43).

digkey. 2020. MPU-6000/6050 Specification Datasheet. [online] Available at:<<https://www.digkey.co.uk/en/datasheets/tdk-invensense/tdk-invensensempu6050datasheetv3204>>[Accessed 1 October 2020].

Dr A. Sobachkin. and Dr A. Dumnov, (2014). Base Numérica De CFD Integrada En CAD. 1st ed. Rusia: SolidWorks, pp.(4-20).

García García, J., 2019. CÁLCULO DE LOS COEFICIENTES HIDRODINÁMICOS DE VARIOS VEHÍCULOS SUBMARINOS MEDIANTE ANSYS. 1st ed. Cartagena: Universidad Politécnica de Cartagena, pp.(8-69).

Garrido Pellicer, A., 2015. Estimación De Los Coeficientes Hidrodinámicos De Vehículos Autónomos Submarinos Mediante CFD. 1st ed. Cartagena: Universidad Politécnica de Cartagena, pp.(58-136).

La.mathworks.com. 2020. MATLAB Documentation - Mathworks América Latina. [online] Available at: <<https://la.mathworks.com/help/>>[Accessed 10 October 2020].

Martínez Carvajal, B., (2013). Planteamiento De La Arquitectura Computacional E Instrumentación Para La Navegación Inercial Asistida De Un Vehículo Operado Remotamente. 1st ed. Bucaramanga: Universidad Industrial de Santander, pp.(18-101).

Martínez Tellez, J., Vaquén Pulido, S. and Peña Giraldo, M., 2017. Identificación Y Control De Un Dispositivo ROV Subacuático Para La Inspección De Elementos Sumergidos. 1st ed. Bogotá: Universidad Libre, pp.(2-6).

Ruiz Reina, E., 2020. Modelado De Dinámica De Fluidos Computacional (CFD) En Procesos Multifísicos (5.1). [online] Youtube. Available at: <<https://www.youtube.com/watch?v=>

*RuTy7K9YsoUabchannel = AddlinkSoftwareCientC3ADfico2CS.L.* > [Accessed 15 September 2020].

Ruiz Ruiz, J., (2016). Cálculo De Coeficientes Hidrodinámicos De Uuvs Mediante Cfds. 1st ed. Cartagena: UNIVERSIDAD POLITÉCNICA DE CARTAGENA, pp.(22-78).

Salgado Iturrino, Á., 2016. CONSTRUCCIÓN, MODELADO Y CONTROL DE UN SUMERGI-  
BLE A ESCALA. 1st ed. Madrid: Universidad Pontificia Comillas, pp.(16-23).

Simon, D., (2006). Optimal State Estimation Kalman, H, And Nonlinear Approaches. 1st ed. New Jersey: Cleveland State University, pp.(123-144).