



UNIVERSIDAD CATÓLICA
“NUESTRA SEÑORA DE LA ASUNCIÓN”
FACULTAD DE CIENCIAS Y TECNOLOGÍA
INGENIERÍA ELECTROMECÁNICA CON ORIENTACIÓN ELECTRÓNICA

Fusión de sensores para estimar la orientación de un vehículo autónomo a escala, mediante el Filtro de Kalman

Erid Eulogio Pacheco Viana

Hernandarias, mayo 2019

UNIVERSIDAD CATÓLICA
“NUESTRA SEÑORA DE LA ASUNCIÓN”
FACULTAD DE CIENCIAS Y TECNOLOGÍA
INGENIERÍA ELECTROMECÁNICA CON ORIENTACIÓN ELECTRÓNICA

**Fusión de sensores para estimar la orientación de un vehículo autónomo a
escala, mediante el Filtro de Kalman**

Erid Eulogio Pacheco Viana

Tutor: Lic Gregorio Ariel Guerrero Moral

Hernandarias, mayo 2019

ERID EULOGIO PACHECO VIANA

Fusión de sensores para estimar la orientación de un vehículo autónomo a escala, mediante el Filtro de Kalman

Proyecto final de carrera presentado como requisito parcial para optar por el título de Ingeniero en la carrera de Ingeniería Electromecánica con Orientación Electrónica. Facultad de Ciencias y Tecnología, Universidad Católica “Nuestra Señora de la Asunción”.

Hernandarias, mayo 2019

Pacheco Viana, Erid Eulogio. (2019); Fusión de sensores para estimar la orientación de un vehículo autónomo a escala. Alto Paraná, Universidad Católica. 140 p.

Tutor: Lic. Gregorio Ariel Guerrero Moral.

Defensa de Proyecto de Fin de Carrera

Palabras clave: Fusión de sensores. Filtro de Kalman.

Dedicatoria

A mis padres Eulogio y Crisolina, por inculcarme valores, sobre todo el de la educación.

Agradecimientos

A mis padres y a mis hermanas, por el apoyo incondicional.

Al profesor tutor, Lic. Ariel Guerrero, por su orientación, sabiduría y paciencia en estos meses de trabajo.

A mis profesores, por sus enseñanzas.

Índice general

Resumen	V
Abstract	VI
Índice de tablas	VII
Índice de figuras	IX
Introducción	1
Planteamiento del problema	2
Justificación	4
Organización del documento	5
Marco teórico	7
1. Navegación	7
Resumen	7
1.1. El problema de la navegación	7
1.2. Formas básicas de navegación	7
1.3. Historia de la navegación inercial	9
1.4. Solución de la navegación	9
Conclusión	10
2. Sistemas de coordenadas	11
Resumen	11
2.1. Sistema de coordenadas Inercial I-frame	11
2.2. Sistema de coordenadas ECI i-frame	11
2.3. Sistema de coordenadas ECEF e-frame	12
2.4. Sistema de coordenadas de Navegación n-frame	12
2.5. Sistemas de coordenadas Body b-frame	13
Conclusión	13
3. Sensores utilizados para la navegación	14
Resumen	14
3.1. Transductores	14
3.2. Sensores	14
3.3. IMU	15
3.4. Acelerómetro	16
3.4.1. Modelo del error del acelerómetro	16
3.5. Giroscopio	18

3.5.1. Modelo del error del giroscopio	20
3.6. Magnetómetro	22
3.6.1. Distorsiones por hierro duro y suave	22
3.7. Calibración de sensores	24
3.7.1. Calibración del acelerómetro y del giroscopio	25
3.7.2. Calibración del magnetómetro	25
Conclusión	26
4. Fusión de sensores	27
Resumen	27
4.1. Fusión de datos	27
4.2. Filtro complementario	28
4.3. Algoritmo de Mahony	29
4.4. Algoritmo de Madwick	30
4.5. Filtro de Kalman	30
4.5.1. Ecuaciones del filtro de Kalman discreto	31
Conclusión	35
Marco metodológico	36
5. Diseño metodológico	36
Resumen	36
5.1. Contexto de la investigación	36
5.2. Alcance de la investigación	37
5.3. Diseño de la investigación	37
5.4. Enfoque de la investigación	38
5.5. Unidad de estudio	38
5.6. Técnicas e instrumentos de recolección de datos	38
Conclusión	39
6. Diseño, simulación e implementación	40
Resumen	40
6.1. Diseño de software	40
6.1.1. Diseño de la máquina de estados	41
6.1.2. Diseño de la interfaz de usuario	44
6.2. Simulación del filtro de Kalman de una variable	45
6.3. Calibración del acelerómetro	47
6.3.1. Cálculo de bias del acelerómetro	48
6.3.2. Cálculo de la matriz M	49
6.4. Calibración del giroscopio	54
6.5. Calibración del Magnetómetro	55
6.6. Fusión de datos	57
6.6.1. Filtro de Kalman para estimar los ángulos Roll φ , Pitch θ y Yaw ψ	58
Conclusión	63

7. Puebas y resultados	64
Resumen	64
7.1. Simulaciones del filtro de Kalman sobre una variable	64
7.2. Fusión Roll y Pitch	67
7.3. Fusión Yaw	70
Conclusión	74
 Consideraciones finales	 75
Conclusiones	75
Trabajos futuros	77
 Referencias	 78
 Apéndice	 81
 A. Proyecto Aguara'i	 82
A.1. Resultados del proyecto	82
A.2. Análisis de costos	83
 B. TDP presentado a la competencia	 86
 C. Anuncio de vencedores del mejor TDP de RoboCar Race 2018	 92
 D. TDP en proceso de evaluación para ser publicado en la revista JPAUT	 94
 E. Artículo presentado para el Labview Student Design Competition	 106
 F. Invitación a NIWeek 2019	 115
 G. Relación entre los sistemas de coordenadas	 117
G.1. Ángulos de Euler	117
G.2. Cambio de Body a Navegación con los ángulos de Euler	117
G.2.1. Giro en Yaw	117
G.2.2. Giro en Pitch	118
G.2.3. Giro en Roll	118
 H. Conceptos básicos de probabilidad y estadística	 120
H.1. Probabilidad de un evento	120
H.2. Variables aleatorias	120
H.3. Media, varianza y covarianza	120
H.4. Distribución normal o gaussiana	121
H.5. Covarianza	122
H.6. Ruido blanco	123
 I. LabVIEW	 124

J. Código fuente de los algoritmos implementados	126
J.1. Panel frontal	126
J.2. Diagrama de bloques	127
K. Especificaciones técnicas	133
K.1. Crius all in one pro (AIOP) v2.1	133
K.2. MPU6050	134
K.3. Magnetómetro HMC5883L	135
K.4. Myrio	137
L. Nota de agradecimiento	138

Resumen

La fusión de sensores se realizó con el fin de obtener estimaciones precisas de la orientación para la correcta navegación de un vehículo autónomo a escala. El proyecto estuvo enmarcado en el diseño experimental, con un enfoque cuantitativo a nivel comprensivo. Se emplearon como técnicas e instrumentos de recolección de datos herramientas matemáticas que permitieron determinar las propiedades del ruido de los sensores para implementarlo en el filtro de Kalman. Para la elaboración del proyecto y cumplimiento de los objetivos se utilizó como herramienta para el procesamiento de datos un myrio (dispositivo embebido de la empresa National Instruments), como software para la implementación el LabVIEW y otro de apoyo para el análisis de datos, Matlab. En efecto, se desarrollaron métodos de calibración (para el acelerómetro y giroscopio) con un enfoque práctico, para que pueda ser implementado antes del inicio de la carrera en tiempo real y para el caso del acelerómetro se utilizó el filtro de Kalman a los efectos de obtener la matriz de corrección de factor de escala y desalineamiento de ejes.

Así mismo, se desarrollaron simulaciones y se pudo determinar la influencia del cambio de los parámetros que intervienen en las ecuaciones del filtro de Kalman, y se lograron reducir los efectos indeseados como distorsiones en el campo magnético, que generan variaciones en la lectura de los sensores, que llevan a una estimación incorrecta de la orientación y por lo tanto en una navegación a su vez incorrecta.

De esta manera se pudo concluir que los resultados fueron muy efectivos y se cumplieron con los objetivos propuestos, pues con la fusión se logró mantener el error por debajo de la cota establecida para el propósito deseado (error menor a dos metros a lo largo del recorrido del auto autónomo a escala).

Palabras claves: Filtro de Kalman. Fusión de sensores.

Abstract

The fusion of sensors was carried out in order to obtain precise estimates of the orientation for the correct navigation of an autonomous vehicle at scale. The project was framed in the experimental design, with a quantitative approach at a comprehensive level. Mathematical tools were used as techniques and data collection instruments that allowed determining the noise properties of the sensors to be implemented in the Kalman filter.

For the elaboration of the project and fulfillment of the objectives, a myrio (embedded device of the National Instruments company) was used as the data processing tool, LabVIEW as software for the implementation and Matlab (another support) for the analysis of data. In fact, calibration methods were developed (for the accelerometer and gyroscope) with a practical approach, so that it can be implemented before the start of the race in real time and for the case of the accelerometer the Kalman filter was used for the purposes of obtain the scale factor correction matrix and axis misalignment.

Likewise, simulations were developed and the influence of the change of the parameters involved in the Kalman filter equations could be determined, and undesired effects such as distortions in the magnetic field, which generate variations in the reading of the sensors, were reduced, which lead to an incorrect estimation of the orientation and therefore in an incorrect navigation.

In this way it was possible to conclude that the results were very effective and the proposed objectives were met, since errors were kept within the established bounds for the desired purpose (error less than two meters along the route of the autonomous car to scale).

Keywords: Kalman filter. Sensor fusion.

Índice de tablas

6.1. Dimensiones de las matrices necesarias para aplicar el filtro de Kalman.	51
7.1. Errores RMS con $Q = 0,2$ y R variable.	66
7.2. Errores RMS con Q variable y $R = 0,5$	66
7.3. Errores RMS de la primera prueba de la fusión Roll.	68
7.4. Errores RMS de la segunda prueba de la fusión Roll.	69
7.5. Errores RMS de la tercera prueba de la fusión Roll.	70
7.6. Errores RMS de la primera prueba de la fusión Yaw.	72
7.7. Errores RMS de la segunda prueba de la fusión Yaw.	73
A.1. Costo de hardware.	83
A.2. Horas de trabajo para el proyecto Aguara'i.	84
A.3. Horas de trabajo para la fusión de sensores.	85
K.1. Características del MPU6050.	135
K.2. Características del HMC5883L.	136

Índice de figuras

1.1. Dead reckoning: Principio de funcionamiento basado en odómetro.	8
2.1. Coordenadas ECEF.	12
2.2. Coordenadas NED.	13
3.1. MPU6050.	15
3.2. Distorsión por hierro duro.	23
3.3. Distorsión por hierro duro y hierro suave.	23
4.1. Diagrama del modelo del filtro de Kalman discreto.	31
4.2. Diagrama de bloques del sistema, modelo de medición y el filtro de Kalman discreto.	35
6.1. Estructura básica de una máquina de estados.	41
6.2. Diagrama de transición de estados para este proyecto.	42
6.3. Diagrama de bloques para la simulación del filtro de Kalman de una variable. . .	47
6.4. Diagrama de bloques para hallar b_{ax} y b_{ay} de la matriz b	49
6.5. Diagrama de bloques subVI del filtro de Kalman para hallar la matriz M.	53
6.6. Diagrama de bloques del subVI KF.	53
6.7. Datos del magnetómetro sin calibrar.	55
6.8. Datos del magnetómetro sin calibrar, plano XY.	56
6.9. Datos del magnetómetro calibrados, plano XY.	56
6.10. Estructura de la fusión de datos.	57
6.11. Diagrama de bloques para el cálculo de las matrices R.	61
6.12. Filtro de Kalman implementado en LabVIEW para estimar Pitch θ	62
6.13. Diagrama de bloques del subVI EC KALMAN de la figura 6.12.	62
7.1. Panel frontal del diagrama de bloques de la figura 6.2.	65
7.2. Respuesta del filtro ante aceleraciones horizontales con $R1$	67
7.3. Respuesta del filtro ante aceleraciones horizontales con $R2$	69
7.4. Respuesta de la fusión Yaw.	71
7.5. Respuesta de la fusión Yaw ante distorsiones magnéticas con $R1$	71
7.6. Respuesta de la fusión Yaw ante distorsiones magnéticas con $R2$	73
G.1. Rotación alrededor del eje z.	118
G.2. Rotación alrededor del eje y.	118
G.3. Rotación alrededor del eje x.	119
H.1. Función de distribución de Probabilidad Normal o Gaussiana.	122

I.1.	Panel Frontal de un VI.	125
I.2.	Diagrama de bloques de una VI.	125
J.1.	Panel frontal, Kalman.	126
J.2.	Panel frontal, Matrices R.	127
J.3.	Caso: Escribir.	127
J.4.	Caso: Leer.	128
J.5.	Caso: Procesar.	128
J.6.	Caso: Calibrar Giro.	129
J.7.	Subvi del estado Calibrar Giro.	129
J.8.	Caso: Ganancias acc.	130
J.9.	Caso: Kalman.	131
J.10.	Caso: Matrices R acc.	132
K.1.	Caso: Matrices R acc.	133
K.2.	Arquitectura del MPU6050.	134
K.3.	Arquitectura del HMC5883L.	136
K.4.	Myrio.	137

Introducción

Desde hace unos años, la industria de los sensores inerciales ha ido creciendo, debido al bajo coste que estos representan, bajo consumo de energía así como al nivel de miniaturización al que están llegando (Pozo, Sotomayor, Rosero, y Morales, 2014). En la actualidad, el ámbito de uso de estos sensores es muy amplio, son utilizados en dispositivos tan comunes para las personas como los celulares, sistemas de entretenimiento, en sistemas de seguridad automotriz, etc.

Así mismo, los sistemas de medición se están integrando con más sensores, dado que han demostrado una mejora sustancial ante problemas de diversas áreas. A medida que los datos crecen, también lo hace la necesidad de combinar esos datos provenientes de distintos sensores con lo que se puede extraer la información útil que permita la comprensión del entorno.

La fusión de datos es de especial importancia en cualquier aplicación en donde los datos provenientes de distintas fuentes deben ser combinados, fusionados y agrupados para obtener la apropiada calidad de las mediciones de las variables que deben ser procesadas. Mediante la fusión de datos, la información obtenida a partir de la misma, resulta igual de confiable que la información obtenida a partir de un equipo patrón, los cuales si bien es cierto son precisos, son elevadamente costosos y por lo tanto de difícil acceso.

Por este motivo, la implementación del método de fusión de sensores es muy usado en diversos campos como la robótica, sistemas biomédicos, sistemas militares, equipos de monitoreo, sistemas de transportes, control de procesos, sistemas de información, etc., pues con ello se logra reducir costos y a la vez se logra la precisión deseada.

Ahora bien, la fusión de sensores como se ha dicho tiene innumerables usos, es así que este trabajo de investigación se desarrolla a partir de la necesidad de obtener datos precisos para la navegación de un vehículo autónomo con sensores inerciales de bajo costo pero con precisión para estimar la orientación del mismo.

Planteamiento del problema

La construcción de un vehículo autónomo es un tema en desarrollo dentro de la Ingeniería. De este modo, resulta un desafío encontrar los mecanismos para dotarlos de la mayor y mejor autonomía posible y al menor costo.

Para lograr el mayor grado de autonomía, el vehículo debe ser capaz de percibir el entorno que le rodea para evitar inconvenientes en su navegación, ahora bien, esto implica la adquisición periódica de datos, a través de distintos sensores, y el principal problema se centra en la correcta fusión de datos de estos sensores y mediante ello, lograr extraer la información relevante y más exacta posible; y conforme a ello planificar y ejecutar acciones.

Es así que, las mediciones de ese entorno, se realizan mediante sensores que cuando son de bajo costo, suelen arrojar resultados inciertos y ocasionalmente incorrectos. Otra cuestión es el error que surge de sensores que miden una variable de manera indirecta, como los giroscopios que como es de esperar, inducen errores acumulativos que crecen indefinidamente, lo cual resulta en estimaciones de orientación cada vez más desviadas (Fischer, 2014).

Otra cuestión que surge es la de determinar si el Filtro de Kalman es adecuado para minimizar los errores que surgen de la fusión de datos y que por tanto permite con la mayor fidelidad posible, la estimación de la orientación del vehículo.

Pregunta general

- ¿Cómo combinar correctamente la información proveniente de distintos sensores para estimar la orientación de un vehículo autónomo?

Preguntas específicas

- ¿Cuál es el proceso de calibración de los sensores utilizados para la navegación de un vehículo autónomo?
- ¿Cuál es la efectividad del filtro de Kalman para la fusión de distintos sensores?

- ¿Cómo diseñar simulaciones del filtro de Kalman para estudiar los parámetros que intervienen en sus ecuaciones?

Objetivo general

- Fusionar datos de sensores para estimar la orientación de un vehículo autónomo a escala, mediante el Filtro de Kalman.

Objetivos específicos

- Desarrollar procedimientos con un enfoque práctico para la calibración de sensores;
- Analizar el Filtro de Kalman en la fusión de datos de sensores iniciales;
- Evidenciar los efectos sobre el comportamiento del filtro de Kalman ante las variaciones de los parámetros que intervienen en sus ecuaciones.

Justificación

En general, un sistema de fusión puede aportar ventajas en la percepción del entorno. Para Ávila y Arancibia (2015), algunas de las ventajas son:

- Tolerancia a fallos y confiabilidad. Múltiples sensores permiten la recuperación de fallas;
- Reducción de la ambigüedad y la incertidumbre;
- Robustez contra inferencias. Utilizar más de un sensor que mida la misma variable permite fortalecer el sistema frente a interferencias ambientales;
- Mejora la resolución. Cuando se fusionan múltiples medidas independientes de la misma entidad, la resolución del resultado es mejor que las medidas por separado.

De ahí, su importancia para la utilización en la construcción de un vehículo autónomo, pues mediante ello se pueden percibir de la manera más fidedigna posible la posición y orientación del vehículo, sin el cual no podría navegar correctamente.

La fusión mediante el filtro de Kalman, que es un estimador, es utilizado en innumerables situaciones, y es el método de fusión que se usará en este trabajo. Su importancia es tal, que fue utilizado para la llegada del hombre a la luna. Fue parte del sistema de navegación del Apolo 11 (NASA, 2018). Sus aplicaciones más importantes han sido el control de sistemas dinámicos complejos, como los procesos de fabricación continua, aviones, barcos y vehículos espaciales.

Organización del documento

El presente trabajo está organizado de la siguiente manera:

Marco teórico

Capítulo 1: Navegación

Contiene los conceptos básicos sobre la navegación y una pequeña reseña histórica, es importante, debido a que es la navegación es el objetivo final del presente trabajo.

Capítulo 2: Sistemas de coordenadas

Contiene una descripción breve de los distintos sistemas de coordenadas, su importancia radica en que los datos adquiridos por los sensores, provienen de un sistema de referencia distinto al de la navegación.

Capítulo 3: Sensores utilizados para la navegación

Contiene los conceptos, los modelos de los errores y el modelo de calibración de cada sensor, que son necesarios para minimizar los errores en las estimaciones de las fusiones que son planteadas posteriormente.

Capítulo 4: Fusión de sensores

Contiene la descripción de los algoritmos de fusión más usados en la actualidad, con un énfasis en el filtro de Kalman, por ser el algoritmo de fusión implementado en este trabajo.

Marco metodológico

Capítulo 5: Diseño metodológico

Contiene la descripción, en todos sus pormenores, de la metodología usada en el transcurso del trabajo.

Capítulo 6: Diseño, simulación e implementación

Contiene el diseño del las simulaciones, del software, de la interfaz de usuario y de los algoritmos implementados de la fusión de sensores.

Capítulo 7: Pruebas y resultados

Contiene las pruebas y la interpretación de los resultados obtenidos de las simulaciones y de la fusión de datos.

Marco Teórico

Capítulo 1

Navegación

Resumen

Este capítulo contiene las definiciones básicas que son necesarias para entender el contexto de esta investigación, debido a que la fusión de sensores será implementado para estimar la orientación de un vehículo autónomo para una navegación correcta, resulta necesario estudiar previamente los tipos de navegación y los problemas que presenta, además se contiene una pequeña reseña histórica sobre la navegación inercial.

1.1. El problema de la navegación

El propósito de la navegación es dirigir el movimiento de un vehículo para llegar a un destino determinado. Una parte importante de la navegación es determinar la posición y orientación del vehículo con relación al destino (Mohinder y Angus, 2015).

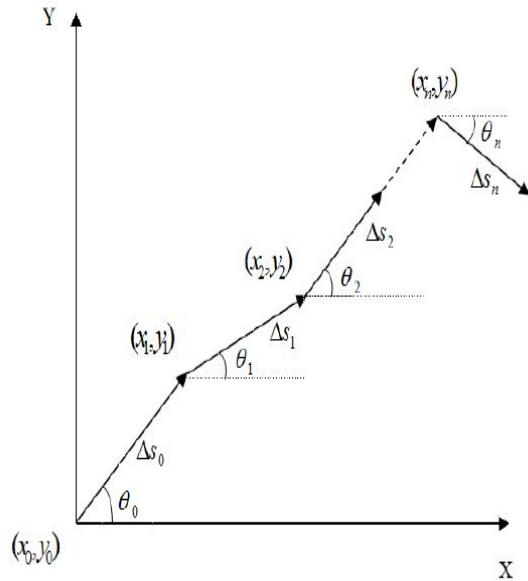
La solución a este problema, generalmente requiere observaciones o mediciones de algún tipo y poder usar esa información para determinar su posición y orientación con respecto a su destino.

1.2. Formas básicas de navegación

Según Mohinder y Angus (2015), podemos mencionar cinco formas básicas en las que un vehículo puede navegar:

1. Pilotaje, que esencialmente se basa en el reconocimiento de puntos de referencia para saber dónde y cómo está orientado.
2. Navegación por estima (Dead reckoning), que se basa en el reconocimiento de puntos de referencia para saber su posición y orientación y cierta estimación de la distancia recorrida.

Figura 1.1: Dead reckoning: Principio de funcionamiento basado en odómetro.



Fuente: Xueli, Wanli, Meng y Zhenhua (2018).

3. Navegación astronómica, usando el tiempo y los ángulos entre los objetos celestes locales verticales y conocidos (por ejemplo, el sol, la luna, los planetas, las estrellas) para estimar la orientación, la latitud y la longitud. Depende de las condiciones de visualización claras, y generalmente requiere de instrumentos especializados. La orientación astronómica supuso la posibilidad de grandes rutas de ida y vuelta que cambiaron radicalmente el panorama cultural y económico del Mediterráneo a partir del primer milenio a. C.
4. Radionavegación, que se basa en fuentes de frecuencia de radio con características de señal conocidas y ubicación conocida. Los sistemas de navegación global por satélite (GNSS por sus siglas en inglés, Global Navigation Satellite Systems) utilizan balizas en satélites para este propósito.

5. Navegación inercial, que se basa en conocer su posición, velocidad y orientación iniciales y, a partir de ese momento, medir sus velocidades, orientación y aceleraciones. Es la única forma de navegación que no se basa en referencias externas.

1.3. Historia de la navegación inercial

La navegación inercial se desarrolló a mediados del siglo XX para la guía de misiles y fue mejorado durante la Guerra Fría como una tecnología para la guía de misiles nucleares de largo alcance. Sus primeras aplicaciones fueron militares, tales como sistemas de guías para misiles balísticos autónomos, barcos militares y aeronaves militares, pero la tecnología pronto se extendió a aplicaciones no militares como la navegación de aviones comerciales (Mohinder y Angus, 2015).

Debido a su sigilo e inmunidad a las contramedidas, la navegación inercial es muy adecuada para la guía de misiles balísticos. Su tiempo de operación durante un lanzamiento de misil es del orden de varios minutos, lo que no permite mucho tiempo para una acumulación significativa de errores de navegación. Esto ha ayudado a lograr precisiones aceptables en el acierto de objetivos utilizando la tecnología de navegación inercial.

Los submarinos utilizaron giroscopios durante un largo tiempo para realizar un seguimiento de la orientación mientras estaban sumergidos, y integrar estos con un sensor de aceleración era un progreso natural para la navegación silenciosa auto-contenida. Sin embargo, los submarinos portadores de misiles nucleares requerían operaciones sumergidas durante meses, y este tiempo era demasiado largo para mantener la precisión de la navegación inercial suficiente para el lanzamiento de misiles balísticos. Algún tipo de información de navegación auxiliar sería necesaria para mantener la precisión de navegación inercial suficiente.

1.4. Solución de la navegación

Según Mohinder y Angus (2015), cualquier solución de navegación, como mínimo, debe incluir la posición actual del vehículo en coordenadas de navegación. Para la navegación inercial,

los resultados intermedios también deben incluir la velocidad y la orientación. Estas variables auxiliares también pueden ser útiles para otros aspectos de la misión, incluyendo:

- Orientación, que es útil para manejar pantallas en la cabina de mando de aeronaves para el rumbo (Compás magnético), y ángulos de balanceo y cabeceo (horizonte artificial), también es útil para determinar la velocidad del viento lateral.
- Tasa de orientación, que se puede utilizar en los bucles de control para pilotos automáticos o dirección de misiles, y para ayudar a la conmutación de antena y el bloqueo de fase para las señales de navegación por satélite en la integración GNSS / INS.
- Velocidad, que es útil para determinar el tiempo de llegada y en los bucles de control para la orientación de pilotos automáticos. También se puede utilizar para compensar el desplazamiento Doppler durante la adquisición y el seguimiento de la señal GNSS.
- Aceleración, que es útil como entrada para controlar los bucles para la guía y el control del vehículo, y como ayuda en el seguimiento de fase para las señales de satélite en la integración GNSS / INS.

Conclusión

Se ha dado una introducción a la navegación, los tipos de navegación, y cómo ha surgido la navegación inercial. Es importante recalcar que para este trabajo, el tipo de navegación usada no es totalmente una navegación inercial, esto, debido a que el magnetómetro agrega una referencia externa (norte magnético) que ayuda a corregir los errores acumulativos en la estimación de la orientación, propios de la navegación inercial.

Capítulo 2

Sistemas de coordenadas

Resumen

Es muy importante empezar hablando de los sistemas de coordenadas que serán usados, esto, debido a que los datos son tomados en un sistema de referencia distinto al que es usado en navegación, a continuación, definiremos los sistemas que son de nuestro interés, y en el Apéndice G se detalla la relación entre ellos.

2.1. Sistema de coordenadas Inercial I-frame

Es un sistema de referencia en donde las leyes de Newton tienen validez, idealmente, el sistema no se mueve ni rota respecto a nada. En la práctica resulta imposible encontrar un sistema realmente inercial. El sistema se utiliza como aproximación teórica y para visualizar otros marcos de referencia con mayor facilidad. En la práctica consiste en un conjunto de ejes perpendiculares entre sí que no rotan ni aceleran respecto al espacio inercial (Amezcua Paredes y Pineda Salgado, 2013).

2.2. Sistema de coordenadas ECI i-frame

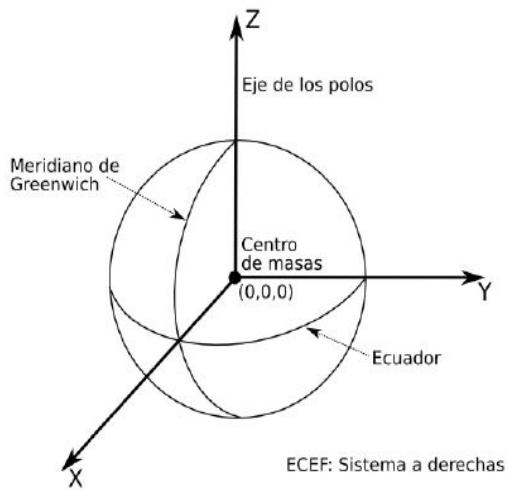
Este sistema de coordenadas ECI (por sus siglas en inglés: Earth Center Inertial) tiene su origen en el centro de masas de la Tierra y se supone que no rota respecto al espacio inercial, ya que se mueve con el planeta. Esto no es cierto debido al movimiento de los planetas, pero se

tomará como sistema inercial (Amezcua Paredes y Pineda Salgado, 2013). Sus ejes están fijados en las estrellas: el eje Z coincide con el eje polar y el plano perpendicular al eje Z coincide con el Ecuador. El eje X e Y no rotan con la tierra, apuntando X directamente al equinoccio Vernal.

2.3. Sistema de coordenadas ECEF e-frame

También conocido como sistema de coordenadas geocéntrico. Tiene su origen en el centro de masa de la tierra y sus ejes rotan con ella. El eje Z se dirige al norte a lo largo del eje polar. Los ejes X e Y están en el plano ecuatorial con X dirigido hacia el meridiano de Greenwich (0° latitud, 0° longitud) y eje Y 90° hacia el Este (Amezcua Paredes y Pineda Salgado, 2013), ver siguiente figura.

Figura 2.1: Coordenadas ECEF.



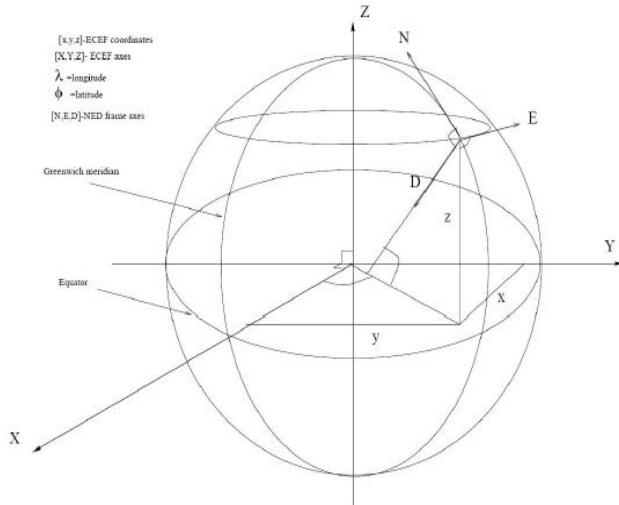
Fuente: Amezcua y Pineda (2013).

2.4. Sistema de coordenadas de Navegación n-frame

Es un sistema local con sus ejes X e Y en el plano tangente al punto de la Tierra donde está el origen. Normalmente el eje X apunta al norte, el eje Y al este y el eje Z abajo, aunque esto debe ser especificado.

Este sistema también es conocido como NED (North, East, Down) ya que sus ejes apuntan a estas direcciones. En ocasiones, se utiliza otra configuración, con el eje X apuntando al este, el eje Y al norte y el eje Z apuntando hacia arriba, también conocido como ENU (East, North, UP). El sistema ENU es el usado en este proyecto.

Figura 2.2: Coordenadas NED.



Fuente: Amezcuay Pineda (2013).

2.5. Sistemas de coordenadas Body b-frame

Este sistema tiene su origen en el centro de masas del vehículo. Típicamente usado en plataformas Strapdown, es decir, cuando los sensores están en el centro de masas del vehículo y sus ejes se mueven con él (es el caso de este proyecto).

Conclusión

Se ha dado un panorama general de los distintos sistemas de coordenadas encontrados en la literatura. Los usados en este trabajo, serán principalmente dos, coordenadas Body b-frame, es el sistema de referencia en la que se encuentra el sensor, y el sistema de coordenadas de Navegación n-frame, que es el utilizado para la navegación.

Capítulo 3

Sensores utilizados para la navegación

Resumen

En el presente capítulo se estudiará específicamente los sensores que serán utilizados para el desarrollo del trabajo de investigación (acelerómetro, giroscopio, magnetómetro), incluyendo los modelos matemáticos de sus errores y sus procedimientos de calibración.

Antes de detallar los modelos de los sensores utilizados, se definen los conceptos básicos para entender su funcionamiento.

3.1. Transductores

Se denomina transductor a todo dispositivo que convierte una señal de una forma física en una señal correspondiente de otra forma física distinta. Es, por lo tanto, un dispositivo que convierte un tipo de energía en otro (Areny, 2004).

3.2. Sensores

Un sensor es un dispositivo que, a partir de la energía del medio donde se mide, da una señal de salida transducible que es función de la variable medida.

Sensor y transductor se emplean a veces como sinónimos, pero sensor sugiere un significado más extenso: la ampliación de los sentidos para adquirir un conocimiento de cantidades físicas que, por su naturaleza o tamaño, no pueden ser percibidas directamente por los sentidos (Areny,

2004).

3.3. IMU

Una unidad de medición inercial o IMU (de sus siglas en inglés, Inertial Measurement Unit) es un dispositivo electrónico que mide e informa sobre la velocidad, posición y la orientación en el espacio usando un conjunto de sensores. Los principales sensores disponibles en una IMU son los acelerómetros y giroscopios, y a veces también se pueden usar magnetómetros o barómetros (que no son sensores iniciales, pero se pueden combinar para obtener mejores resultados) (López Montes, 2016).

Figura 3.1: MPU6050.



Fuente:SparkFun (2011).

En la Figura 3.1 se muestra la IMU utilizada para la implementación en este proyecto. El MPU6050 es una unidad de medición inercial o IMU (Inertial Measurement Units) de 6 grados de libertad (DoF) pues combina un acelerómetro de 3 ejes y un giroscopio de 3 ejes. Este sensor es muy utilizado en navegación, goniometría, estabilización, etc. En el Apéndice K se ofrece mayores datos acerca de los datos técnicos de los dispositivos utilizados para la implementación.

3.4. Acelerómetro

Un acelerómetro es un dispositivo que mide aceleraciones. Existen diferentes técnicas utilizadas para lograrlo, entre las más comunes se encuentran el uso de sistemas masa-resorte combinados con mediciones de capacitancia, sensores piezoelectricos que miden la presión ejercida por una masa inercial y hasta algunos que utilizan la convección térmica del aire (Vigouroux Cavolina, 2010).

El acelerómetro no puede distinguir entre las aceleraciones iniciales (también llamados fuerzas específicas) de la aceleración gravitacional. Un acelerómetro en caída libre (por ejemplo, en órbita) no tiene entrada detectable. Lo que miden los acelerómetros se basa en la segunda ley de Newton $F = \frac{m}{a}$ donde F es la fuerza físicamente aplicada (sin incluir la gravedad), m es la masa a la que se aplica y la fuerza específica es la relación $\frac{F}{m}$. Esto tiene un efecto profundo en la propagación del error de navegación inercial (Mohinder y Angus, 2015). Los tipos básicos de sensores de aceleración incluyen:

- PIGA (de sus siglas en inglés, Pendulous integrating gyroscopic accelerometers), en los que el centro de soporte de un giroscopio de rueda de impulso está desplazado axialmente de su centro de masa, lo que crea un torque de precesión proporcional al componente de aceleración ortogonal al desplazamiento de masa. Estos son acelerómetros de integración, porque el ángulo de precesión acumulado de salida es proporcional a la integral de la aceleración de entrada. También son los acelerómetros más precisos (y caros).
- Acelerómetros de masa de prueba, que miden la fuerza necesaria para mantener una masa alojada en el centro del instrumento. Existen diversos métodos para la medición de esta fuerza, que llevan a muchos diseños diferentes del acelerómetro.

3.4.1. Modelo del error del acelerómetro

El error de “aceleración del punto medio” para un sensor de aceleración (acelerómetro) se define como el error de aceleración efectivo en el punto medio del periodo de muestreo

(Mohinder y Angus, 2015). El modelo de error asociado para un acelerómetro en términos de parámetros desconocidos del sensor es el siguiente:

$$\Delta_{\beta_m} = \beta_m \otimes \zeta + b_A + h_A \beta_m + \beta_m^2 (FI1 - FX1) + \delta \beta, \quad (3.1)$$

$$h_A = \begin{bmatrix} S_1 & \delta_{12} & \delta_{13} \\ 0 & S_2 & \delta_{23} \\ 0 & 0 & S_3 \end{bmatrix}$$

$$\beta_m^2 = \begin{bmatrix} \beta_1^2 & 0 & 0 \\ 0 & \beta_2^2 & 0 \\ 0 & 0 & \beta_3^2 \end{bmatrix}$$

donde:

Δ_{β_m} es el error de punto medio del acelerómetro,

\otimes es producto vectorial,

ζ vector de 3×1 que representa los errores de alineación de actitud entre ejes de “plataforma” y ejes computacionales,

b_A es un vector desconocido de 3×1 con los sesgos (bias) del acelerómetro, normalizado a la magnitud de la gravedad,

S_i son los errores de los factores de escala, también desconocidos ($i = 1, 2, 3$),

δ_{ij} errores desconocidos por la no ortogonalidad de los ejes,

$\delta \beta$ son otros términos de error,

$FI1$ es un vector de 3×1 , representa una aceleración cuadrada no-lineal para la aceleración a lo largo de cada eje de entrada,

$FX1$ es un vector de 3×1 , representa una aceleración cuadrada no-lineal para la aceleración, normal cada eje de entrada, y

β_m es un vector de 3×1 ($[\beta_1 \beta_2 \beta_3]^T$) de componentes de punto medio de aceleración en coordenadas de la plataforma.

El vector de estado x^A (12×1) está compuesto por subvectores y escalares.

$$(x^A) = \left[\underbrace{b_A^T}_{1 \times 3} \quad S_1 \quad \delta_{12} \quad S_2 \quad \delta_{13} \quad \delta_{23} \quad S_3 \quad \underbrace{(FX1 - FI1)}_{1 \times 3}^T \right]^T$$

3.5. Giroscopio

Los giroscopios son sensores para medir la rotación. Los giroscopios de velocidad miden la velocidad de rotación, y los giroscopios de desplazamiento (también llamados giroscopios de ángulo completo) miden el ángulo de rotación acumulado. La navegación inercial depende de los giroscopios para mantener el conocimiento de cómo los acelerómetros están orientados en coordenadas inerciales y de navegación (Mohinder y Angus, 2015). Hay muchos mecanismos físicos básicos para detectar la rotación incluyendo:

- Los giroscopios giratorios utilizan la ley de conservación del momento angular (en dirección y magnitud) de una masa giratoria para establecer una dirección de referencia inercial en el eje de rotación de la masa giratoria.
- Giroscopios ópticos, que utilizan la velocidad constante de la luz en direcciones apropiadas alrededor de un camino cerrado para detectar rotaciones en el plano de ese camino.

Hay dos diseños básicos:

- Giroscopios láser, en los que la luz se propaga a través de unas cavidades del láser y se refleja en los espejos para completar el bucle. Las rotaciones se detectan mediante interferometría de fases relativas en las trayectorias de luz contrarrotantes. Los giroscopios láser son integradores de frecuencia.
- Giroscopios de fibra óptica (FOG, fiber-optic gyroscopes), en los cuales la luz de una fuente de láser se acopla en una fibra óptica con muchos devanados alrededor de un camino cerrado. La rotación se detecta mediante la interferometría entre la fuente del láser y la luz que sale del bucle, basado en un fenómeno denominado efecto Sagnac.
- Giroscopios de estructura vibrante, que detectan las fuerzas fuera del plano en el que un cuerpo vibra debido a la rotación de la trayectoria (efecto Coriolis). Hay muchos diseños. Los giroscopios de tipo "diapasón", por ejemplo, detectan vibraciones rotacionales de un diapasón debido al efecto Coriolis en las puntas vibrantes, los giroscopios de resonador hemisférico (también llamados Resonadores de Vaso de Vino) detectan cambios en la fase de los modos de vibración mecánica causados por el efecto Coriolis durante la rotación.

3.5.1. Modelo del error del giroscopio

El modelo de error de deriva del giroscopio de 48 estados se presenta a continuación (Mohinder y Angus, 2015):

$$\varepsilon = b_g + h_g \omega + U_g \beta + K_g \beta^1 + \text{diag}|\omega| T_g + b_{gt} t + U_{gt} t \beta, \quad (3.2)$$

$$h_g = \begin{bmatrix} S_{g1} & \Delta_{12} & \Delta_{13} \\ \Delta_{21} & S_{g2} & \Delta_{23} \\ \Delta_{31} & \Delta_{32} & S_{g3} \end{bmatrix}$$

$$U_g = \begin{bmatrix} d_{I1} & d_{01} & d_{S1} \\ d_{S1} & d_{I2} & d_{02} \\ d_{03} & d_{S3} & d_{I3} \end{bmatrix}$$

$$K_g = \begin{bmatrix} k_{II1} & k_{001} & k_{SS1} & I_{I01} & k_{IS1} & k_{S01} \\ k_{SS2} & k_{II2} & k_{002} & k_{IS2} & I_{S02} & k_{I02} \\ k_{003} & k_{SS3} & k_{II3} & k_{S03} & k_{I03} & I_{IS3} \end{bmatrix}$$

$$\beta^1 = \left[\begin{array}{cccccc} \beta_1^2 & \beta_2^2 & \beta_3^2 & \beta_1 \beta_2 & \beta_1 \beta_3 & \beta_2 \beta_3 \end{array} \right]^T$$

$$x^g(t) = \left[\begin{array}{ccccccc} 1 \times 3 & 1 \times 9 & 1 \times 9 & 1 \times 15 & 1 \times 3 & 1 \times 3 & 1 \times 3 \\ b_g^T & h_g^{1T} & U_g^{1T} & K_g^{1T} & T_g^T & b_{gt}^T & U_{gt}^{1T} \end{array} \right]^T$$

donde:

x^g es el vector de estado, del subsistema del giroscopio de 48 estados, de los parámetros de

errores de deriva,

b_g vector de los parámetros fijos de deriva desconocidos del giroscopio,

h_g matriz de 3×3 que contiene el factor de escala desconocido (S_{gi}) y los errores de alineamiento de ejes Δ_{ij} como componentes ($i, j = 1, 2, 3$),

T_g es vector de 3×1 de errores de factor de escala no lineal desconocidos del giroscopio, con elementos δS_{gi} ,

$diag|\omega|$ es una matriz diagonal de 3×3 compuesta por los valores absolutos de ω (velocidad angular inercial de la plataforma) en el elemento diagonal correspondiente,

U_g es una matriz de 3×3 de parámetros desconocidos (d_{kj}) de desequilibrio de masa del giroscopio, los índices $I, 0$, y S indican los ejes de entrada, salida y giro respectivamente, para cada giroscopio 1, 2 y 3,

K_g es una matriz de 3×6 de errores desconocidos de fiabilidad del giroscopio k_{kji} ,

b_{gt} es un vector de 3×1 de parámetros fijos desconocidos de la tendencia de deriva,

U_{gt} es un vector de 3×6 de parámetros desconocidos de la tendencia de desequilibrio de la masa del giroscopio, y

β es un vector de 3×1 de cosenos de dirección vertical $(\beta_1, \beta_2, \beta_3)^T$

β un vector de 6×1

El vector de estado $x^A(12 \times 1)$ está compuesto por subvectores y escalares.

$$(x^A) = \left[\underbrace{b_A^T}_{1 \times 3} \quad S_1 \quad \delta_{12} \quad S_2 \quad \delta_{13} \quad \delta_{23} \quad S_3 \quad \underbrace{(FX1 - FI1)^T}_{1 \times 3} \right]^T$$

3.6. Magnetómetro

El magnetómetro es un tipo de sensor que mide la fuerza y la dirección del campo magnético local. El campo magnético medido será una combinación del campo magnético terrestre y cualquier campo magnético creado por objetos cercanos. El campo magnético se mide en el marco de referencia del sensor (VectorNav Technologies, 2019).

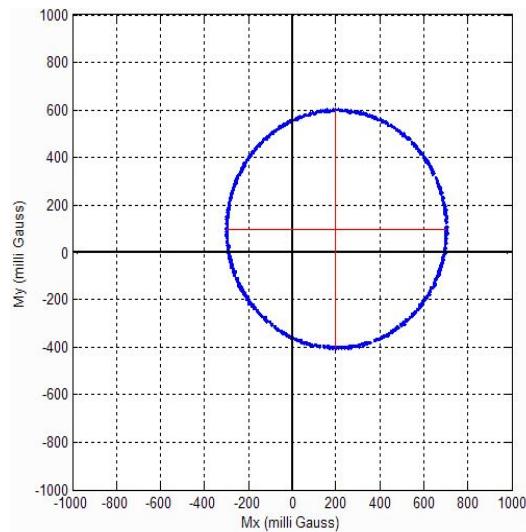
Estos sensores son susceptibles a perturbaciones producidas por campos magnéticos externos debido a minerales en torno a la tierra y los componentes electrónicos que rodean al sensor (Bogado y Benítez, 2015). Es por esto, que se hace necesario realizar ajustes para rechazar estas perturbaciones.

3.6.1. Distorsiones por hierro duro y suave

Las medidas magnéticas serán sometidas a distorsión. Se considera que estas distorsiones caen en una de dos categorías: hierro duro o suave. Las distorsiones de hierro duro son creadas por objetos que producen un campo magnético. Un altavoz o pieza de hierro magnetizado, por ejemplo, causará una distorsión de hierro duro. Si la pieza de material magnético está físicamente unida al mismo marco de referencia que el sensor causará una polarización permanente en la salida del sensor. Las distorsiones por hierro suave se consideran desviaciones o alteraciones en el campo magnético existente. Estas distorsiones alterarán el campo magnético dependiendo de la dirección en la cual actúa el campo en relación al sensor. Este tipo de distorsión es comúnmente causada por metales como el níquel y el hierro. En la mayoría de los casos, las distorsiones de hierro duro tendrán una contribución mucho mayor al error total no corregido que el hierro blando (VectorNav Technologies, 2019).

Una forma común de visualizar y corregir distorsiones de hierro duro y suave es trazar la salida del magnetómetro en un gráfico 2D. En las siguientes figuras se muestran las medidas tomadas por el magnetómetro a medida que el dispositivo gira lentamente alrededor del eje z (solo son figuras demostrativas).

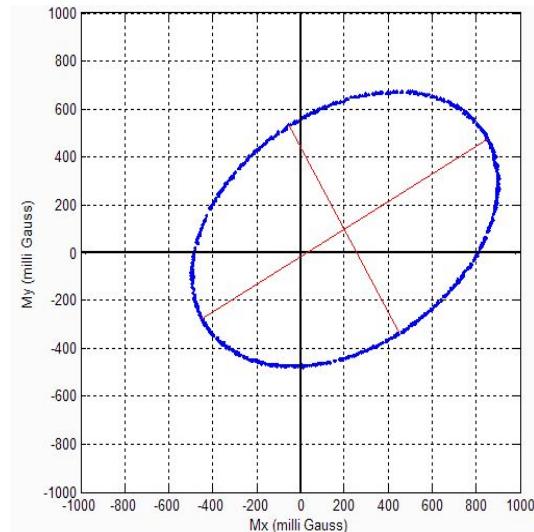
Figura 3.2: Distorsión por hierro duro.



Fuente: VectorNav Technologies (2019).

En la Figura 3.2 se puede observar el efecto de las distorsiones por hierro duro, que causa un desplazamiento del origen, pero no lo deforman, debido a un campo magnético constante, distinto del campo magnético terrestre, que actúa sobre el sensor (VectorNav Technologies, 2019).

Figura 3.3: Distorsión por hierro duro y hierro suave.



Fuente: VectorNav Technologies (2019).

En la Figura 3.3 se observa el efecto de las distorsiones por hierro duro y suave. Las distorsiones por hierro suave deforman los campos magnéticos existentes, esto se ve reflejado en una forma elíptica (VectorNav Technologies, 2019).

3.7. Calibración de sensores

Según Mohinder y Angus (2008), la calibración de sensores es el proceso de estimar la relación entre las entradas y las salidas de los sensores. Tiene como propósito convertir las salidas del sensor a entradas equivalentes, de modo que un valor de salida (ya sea analógico o digital) se pueda volver a convertir en unidades de aceleración de entrada (por ejemplo, metros por segundo al cuadrado) o unidades de velocidad de rotación de entrada (por ejemplo, radianes por segundo) para uso en navegación.

Entre los errores de sensores más comunes que requieren calibración y compensación se encuentran:

1. Sesgo del sensor, que equivale a la salida del sensor cuando no hay entrada.
2. Factor de escala del sensor, que es la relación entre la variación de salida y la variación de entrada.
3. Errores de desalineación del eje de entrada del sensor.

Las salidas de los sensores inerciales son integradas para su utilización en la navegación, lo que los hace relativamente insensibles al ruido blanco de media cero del sensor. Sin embargo, son muy sensibles a los errores que no tengan media cero en un periodo relativamente corto de tiempo (Mohinder y Angus, 2008).

Para muchas aplicaciones prácticas, es suficiente usar un factor de escala y sesgo para cada eje del sensor y quizás dos componentes de los ángulos de desalineación del eje de entrada (Mohinder y Angus, 2015).

3.7.1. Calibración del acelerómetro y del giroscopio

Para los sensores INS con acelerómetro y giroscopio de tres ejes, los modelos de compensación resultante tienen la forma:

$$z_{output} = Mz_{input} + b, \quad (3.3)$$

donde M es una matriz de 3×3 y b es una vector columna de dimensión 3. Los valores de M y b pueden ser determinados por un ajuste de mínimos cuadrados o filtro de Kalman de los pares de entrada-salida. Una vez que se han determinado M y b , se puede invertir fácilmente la ecuación para obtener la fórmula de compensación de error como se muestra en la siguiente ecuación:

$$z_{input} = Nz_{output} + d \quad (3.4)$$

$$N = M^{-1}$$

$$d = -Nb,$$

los valores de N y d también pueden calcularse directamente por ajuste de mínimos cuadrados con los pares de entrada-salida.

3.7.2. Calibración del magnetómetro

Es posible eliminar los efectos de las distorsiones de hierro duro y suave en las salidas del magnetómetro. Para ello, se utiliza el siguiente modelo de calibración (VectorNav Technologies, 2019):

$$M = C(H - b), \quad (3.5)$$

$$\begin{bmatrix} mx \\ my \\ mz \end{bmatrix} = \begin{bmatrix} c_1 & c_2 & c_3 \\ c_4 & c_5 & c_6 \\ c_7 & c_8 & c_9 \end{bmatrix} \begin{bmatrix} Hx - b_{Hx} \\ Hy - b_{Hy} \\ Hz - b_{Hz} \end{bmatrix}, \quad (3.6)$$

El modelo consta de 12 parámetros de compensación de hierro duro y blando. Los primeros 9 parámetros (c_1, c_2, \dots, c_9) son para la corrección por hierro suave, mientras que los últimos 3

$(b_{Hx}, b_{Hy} b_{Hz})$ son para el hierro duro.

Conclusión

Se ha desarrollado los conceptos básicos relacionados a los sensores, luego se presentaron los modelos de los errores y los métodos de calibración de cada sensor, que son indispensables para un correcto funcionamiento del algoritmo de fusión de datos que se presenta en el siguiente capítulo.

Capítulo 4

Fusión de sensores

Resumen

En el presente capítulo se introducen a los conceptos relacionados a la fusión de sensores, se presentan los algoritmos más usados actualmente para fusionar sensores, con un enfoque especial en el filtro de Kalman, debido a que es el algoritmo implementado en este trabajo. En el Apéndice H se dan algunos conceptos básicos de probabilidad y estadística que ayudarán a entender los conceptos relacionados al filtro de Kalman.

4.1. Fusión de datos

La fusión de datos es un tema amplio, por tanto, han surgido muchas definiciones que se usan de manera indistinta para referirse a ella. Una de las definiciones más citadas es la que corresponde a la propuesta por la Junta de Directores de Laboratorios del grupo de fusión de datos *Joint Directors Of Laboratories (JDL)* que la define de la siguiente manera: “un proceso de múltiples niveles y fases de detección automática, que tratan con la asociación, correlación, estimación, y combinación de datos e información de una o múltiples fuentes para lograr posiciones de refinado, estimaciones de identidad, evaluaciones completas, situaciones de amenazas oportunas a su significado.” (Hall y Llinas, 1997).

Teniendo en cuenta la definición anterior, podemos decir que la fusión de sensores combina lecturas de diferentes sensores para obtener información con menos incertidumbre que la información obtenida por cada sensor individual. Esto permite aumentar la supresión de ruido

y la precisión del sistema de medición.

La variedad de algoritmos para fusión y filtrado de datos de sensores inerciales es amplia, algunos de ellos se describen brevemente en este capítulo. Se pondrá especial énfasis al filtro de Kalman, por ser el algoritmo implementado en este trabajo.

4.2. Filtro complementario

El filtro complementario aprovecha las características complementarias de dos señales de la misma variable para contrarrestar sus respectivos ruidos. Se comienza dividiendo la variable observada en dos, a partir de alguna frecuencia de corte. El filtro complementario busca combinar las partes estables de cada señal y descartar las inestables (Fischer, 2014). La separación del espectro se realiza a partir de filtros pasa-bajo y pasa-alto.

Sean y_{x_1} , y_{x_2} dos señales de la variable x .

$$y_{x_1} = x + \mu_x, \quad (4.1)$$

$$y_{x_2} = x + b_0, \quad (4.2)$$

donde μ_x es ruido de alta frecuencia y b_0 es una perturbación dominada por contenido de baja frecuencia. Sean $L(s)$ y $H(s)$ filtros pasa-bajo y pasa-alto respectivamente, donde s es la transformada de Laplace de la señal, respecto de una misma frecuencia de corte f_0 , de tal manera que $L(s) + H(s) = 1$. Luego se tiene:

$$L(s)x + H(s)x = x \quad (4.3)$$

Al aplicar los filtros sobre las señales mencionadas, eligiendo f_0 , de tal manera que la frecuencia

de la señal μ_x sea predominantemente mayor y la de b_o menor.

$$L(s)y_{x_1} \approx L(s)x \quad (4.4)$$

$$L(s)y_{x_1} \approx H(s)x \quad (4.5)$$

Luego podemos asumir que:

$$L(s)y_{x_1} + H(s)y_{x_2} \approx x \quad (4.6)$$

4.3. Algoritmo de Mahony

El algoritmo de Mahony (Mahony, Hamel, y Pflimlin, 2005) es un filtro complementario, aplica un filtro pasa-bajo a las estimaciones provenientes de los acelerómetros y magnetómetros, y un filtro pasa-alto a las estimaciones obtenidas del giroscopio, luego, se fusionan ambas estimaciones.

El filtro de Mahony se basa en una representación en forma de cuaterniones. Así, la derivada del vector de orientación se expresa como:

$$\dot{q}_\omega = \frac{1}{2} \otimes (s_\omega + \delta), \quad (4.7)$$

en donde δ es un término de corrección de las velocidades angulares generado por un bloque proporcional-integral (PI), que actúa sobre el error de las estimaciones obtenidas de los acelerómetros/magnetómetros y el giroscopio, el cual tiene la siguiente forma:

$$\delta = k_p e + k_i \int e dt, \quad (4.8)$$

en donde e representa el error en las estimaciones.

4.4. Algoritmo de Madwick

El algoritmo desarrollado por Sebastian Madgwick (Madgwick, Harrison, y Vaidyanathan, 2011) emplea una representación de la orientación por medio de cuaterniones, está basado en el algoritmo del gradiente descendente para calcular la dirección del error de medición a partir de los datos del giroscopio.

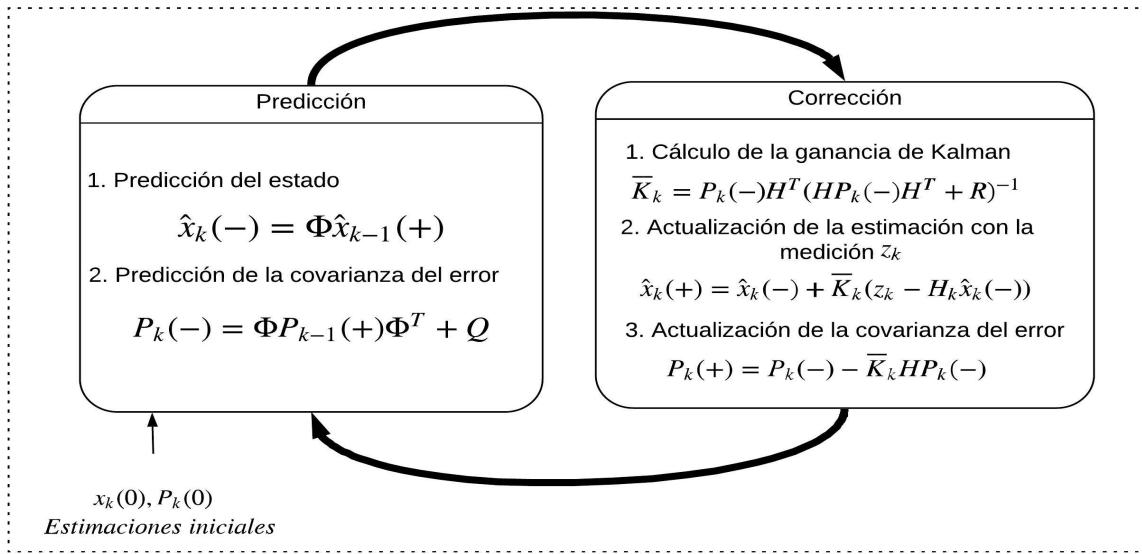
El algoritmo está dividido en cuatro partes principales que son: cálculo de orientación a partir de las velocidades angulares medidas por el giroscopio; el cálculo de las orientaciones a partir de los vectores medidos del campo gravitacional y del campo magnético; la fusión de las dos estimaciones anteriores y por último la normalización del cuaternion de la medición.

4.5. Filtro de Kalman

El filtro de Kalman es un algoritmo desarrollado por Rudolf E. Kalman en 1960 (Kalman, 1960). Según Mohinder y Angus (2008), el filtro de Kalman es un estimador de lo que se llama “un problema cuadrático lineal”, que es un problema de estimar el “estado” instantáneo de un sistema dinámico lineal perturbado por ruido blanco gaussiano, mediante el uso de medidas relacionadas linealmente con el estado, pero corrompidas por el ruido blanco. El estimador resultante es estadísticamente óptimo con respecto a cualquier función cuadrática de error de estimación.

En la práctica, el filtro de Kalman ha sido uno de los más grandes descubrimientos en la historia de la estimación estadística. Ha permitido a la humanidad hacer muchas cosas que no se podrían haber hecho sin él. Sus aplicaciones más inmediatas han sido para el control de sistemas dinámicos complejos, aviones, barcos o naves espaciales. En un sistema dinámico, el filtro de Kalman proporciona un medio para inferir la información faltante a partir de mediciones indirectas y ruidosas. También es usado para predecir los probables cursos de ríos durante las inundaciones, las trayectorias de los cuerpos celestes o los precios de los productos comercializados (Mohinder y Angus, 2008).

Figura 4.1: Diagrama del modelo del filtro de Kalman discreto.



Fuente: Elaboración propia.

La Figura 4.1 muestra la operación del filtro de Kalman, que requiere los valores iniciales de las estimaciones de los estados y de la matriz de covarianza del error para el inicio de su funcionamiento. Es un algoritmo recursivo, puede correr en tiempo real usando únicamente las mediciones de entrada actuales, el estado calculado previamente y su matriz de incertidumbre, y no requiere ninguna otra información adicional.

4.5.1. Ecuaciones del filtro de Kalman discreto

El filtro de Kalman se ocupa del problema general de estimar el estado $x_k \in \mathbb{R}^n$ de un proceso controlado en tiempo discreto que es gobernado por una ecuación estocástica lineal en diferencia. Opera por medio de un mecanismo de predicción y corrección. En esencia, el algoritmo pronostica el nuevo estado a partir de su estimación previa, añadiendo un término de corrección proporcional a la diferencia entre la medición (z_k) y la predicción ($\hat{x}_k(-)$), de tal forma que la varianza del error es minimizado estadísticamente.

Se parte del modelo del sistema, que describe la evolución en el tiempo de la cantidad que se quiere estimar, esta cantidad es expresada por el vector de estados $x_k \in \mathbb{R}^n$. El paso de un estado

a otro se realiza mediante la matriz de transición de estados Φ_k , y la adición de un ruido w_k , este ruido se caracteriza por ser un ruido blanco gaussiano de media cero y covarianza Q_k . Esto se representa mediante una ecuación en diferencias lineal estocástica de la siguiente forma:

$$x_k = \Phi_{k-1} x_{k-1} + \Gamma u_{k-1} + w_{k-1} \quad (4.9)$$

$$x_{k+1} = \Phi_k x_k + \Gamma u_k + w_k,$$

donde la matriz Φ_k relaciona el estado previo con el estado actual. En la práctica, esta matriz podría cambiar con cada paso de tiempo, pero de aquí se asume que es constante. u_k se define como la entrada del sistema.

El modelo de la medición relaciona el vector de medida $z_k \in \Re^l$ con el vector de estados del sistema x_k a través de la matriz de observación (o sensitividad) de las mediciones H_k y la adición de un ruido v_k , este ruido se caracteriza por ser un ruido blanco gaussiano de media cero y covarianza R . Esto se representa mediante la siguiente ecuación:

$$z_k = H_k x_k + v_k \quad (4.10)$$

En la práctica, la matriz H_k podría variar con el tiempo, pero en este trabajo se asume como constante.

Además, se asume que los ruidos w_k y v_k son independientes entre ellas, que son independientes entre ellas, y que tienen distribución de probabilidad normal, y por lo tanto que

cumplen:

$$E\{w_k\} = E\{v_k\} = 0,$$

$$E\{w_k v_k^T\} = E\{v_k w_k^T\} = 0,$$

$$E\{w_k w_k^T\} = Q,$$

$$E\{v_k v_k^T\} = R,$$

$$E\{w_k w_j^T\} = 0 \quad \forall k \neq j \quad y \quad E\{v_k v_j^T\} = 0 \quad \forall k \neq j,$$

en donde $E\{\}$ es la media o valor esperado de la variable aleatoria.

Como se mencionó al comienzo de la sección, el algoritmo puede separarse en dos etapas; la primera llamada predicción (o estimación *a priori*), y la segunda es una etapa de corrección (o estimación *a posteriori*). El $(-)$ (ecuaciones (4.11) y (4.12)) indica los valores a priori de las variables (antes de que se use la información de la medición) y $(+)$ (ecuaciones (4.13) al (4.15)) indica los valores a posteriori de las variables (después de que se usa la información de la medición). La variable \bar{K} es la ganancia de Kalman. Estas etapas se describen a continuación:

4.5.1.1. Ecuaciones de predicción

Además de considerar que Φ es constante, se considera que $u_k = 0$. Consecuentemente, las ecuaciones que describen la estimación *a priori* quedan de la siguiente forma:

$$\hat{x}_k(-) = \Phi \hat{x}_{k-1}(+) \tag{4.11}$$

$$P_k(-) = \Phi P_{k-1}(+) \Phi^T + Q_{k-1} \tag{4.12}$$

En esta etapa, se proyectan los estados de un instante de tiempo anterior utilizando la matriz de transición de estado (Φ), además, se propaga la incertidumbre de la estimación previa al estado actual utilizando el modelo del sistema y la covarianza de la perturbación aleatoria del proceso(Q).

4.5.1.2. Ecuaciones de corrección

Las ecuaciones de corrección o de actualización de la medida para el filtro están dadas por:

$$\bar{K}_k = P_k(-) H^T (H P_k(-) H^T + R)^{-1} \quad (4.13)$$

$$\hat{x}_k(+) = \hat{x}_k(-) + \bar{K}_k(z_k - H_k \hat{x}_k(-)) \quad (4.14)$$

$$P_k(+) = P_k(-) - \bar{K}_k H P_k(-) \quad (4.15)$$

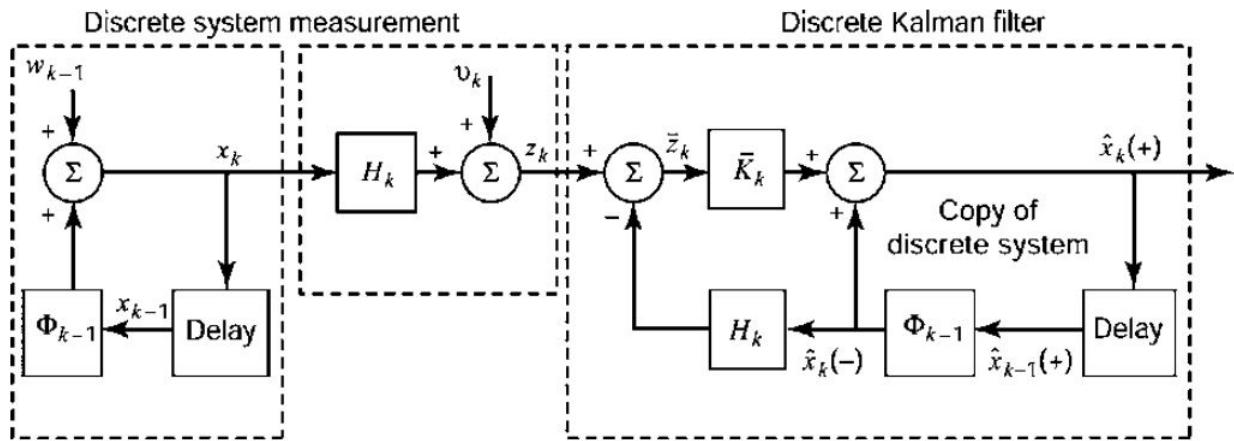
Esta etapa del filtro es la encargada de incorporar las nuevas mediciones del proceso, para producir una estimación con menor varianza; primeramente se calcula la ganancia de Kalman (4.13), a partir de la incertidumbre de la predicción de los estados ($P_k(-)$) y de la covarianza de la perturbación aleatoria de las mediciones (R), luego se hace una estimación a *posteriori* (4.14), en donde se corrige la estimación a *priori* usando el valor de la medición (z_k) con la ganancia de Kalman (\bar{K}_k); finalmente se actualiza la incertidumbre asociada a la estimación a *posteriori* (4.15).

Para tener obtener detalles acerca de la deducción de estas ecuaciones, se puede consultar (Mohinder y Angus, 2008) y (Prado O., 2005).

Las ecuaciones de actualización de vector de estado, ganancia de Kalman y covarianza de error representan un sistema asintóticamente estable, y, por lo tanto, la estimación del estado x_k se vuelve independiente de la estimación inicial x_0 y P_0 a medida que k aumenta (Mohinder y Angus, 2008).

La Figura 4.2 clarifica el modelo de medición y la operación del Filtro de Kalman.

Figura 4.2: Diagrama de bloques del sistema, modelo de medición y el filtro de Kalman discreto.



Fuente: Mohinder y Angus (2008).

Conclusión

Se han presentado los algoritmos de fusión más utilizados en la actualidad, con un enfoque especial en el filtro de Kalman, se ha detallado los algoritmos que la componen que servirán para su implementación.

Marco metodológico

Capítulo 5

Diseño metodológico

Resumen

En este capítulo se describe la metodología usada en el transcurso del trabajo, el contexto en el que se desarrolla, su alcance, enfoque y las técnicas e instrumentos de recolección de datos.

5.1. Contexto de la investigación

El presente trabajo de investigación se realiza en el contexto de la invitación a la competencia internacional de vehículos autónomos a escala RobocarRace (2018) llevada a cabo en la ciudad de San Pablo - Brasil, organizada por la Facultad de Tecnología de Santo André, Universidad Federal ABC y Robótica Paula Souza (RobocarRace, 2018) en la cual se ha participado en representación de la Universidad Católica Nuestra Señora de la Asunción, sede Regional Alto Paraná, Facultad de Ciencias y Tecnología, y se ha adoptado como nombre del equipo “A2G” (Autonomous Arandu Group) y el vehículo presentado fué el “Aguara‘i”.

Para este propósito, un auto eléctrico a escala 1:8 remotamente controlado es modificado, y se lo ha dotado de sensores (encoders, acelerómetro, giroscopio y magnetómetro) y un dispositivo (myRIO) para realizar la tarea de adquisición, procesamiento de los algoritmos de fusión, control y navegación.

5.2. Alcance de la investigación

En este trabajo se realizará el diseño y la implementación de la fusión de sensores utilizando como herramienta el filtro de Kalman, para estimar la orientación de un vehículo autónomo a escala. Los sensores utilizados percibir el entorno serán: un acelerómetro, un giroscopio y un magnetómetro. Como herramienta de procesamiento se utilizará LabVIEW (software, ver Apéndice I) y un myrio (hardware, ver Apéndice K.4).

De este modo, el alcance de esta investigación es descriptiva y explicativa, pues busca especificar con bastante precisión el problema de la fusión de sensores con su correspondiente resultado, así como explicar de manera teórica el fenómeno en estudio y mediante su implementación la solución de problemas futuros.

Con frecuencia en las investigaciones cuantitativas, la meta del investigador consiste en describir fenómenos, situaciones, contextos y sucesos; esto es, detallar cómo son y se manifiestan. Con los estudios descriptivos se busca especificar las propiedades, las características y los perfiles de personas, grupos, comunidades, procesos, objetos o cualquier otro fenómeno que se someta a un análisis (Hernández et al., 2014).

Los estudios explicativos van más allá de la descripción de conceptos o fenómenos o del establecimiento de relaciones entre conceptos; es decir, están dirigidos a responder por las causas de los eventos y fenómenos físicos o sociales. Como su nombre lo indica, su interés se centra en explicar por qué ocurre un fenómeno y en qué condiciones se manifiesta o por qué se relacionan dos o más variables (Hernández et al., 2014).

5.3. Diseño de la investigación

El diseño es experimental, los experimentos manipulan tratamientos, estímulos, influencias o intervenciones (denominadas variables independientes) para observar sus efectos sobre otras variables (las dependientes) en una situación de control. Es decir, los diseños experimentales se utilizan cuando el investigador pretende establecer el posible efecto de una causa que se

manipula (Hernández et al., 2014).

En este caso, se manipula intencionalmente las variables que intervienen en las ecuaciones del filtro de Kalman (matrices de covarianza del proceso y del ruido; condiciones iniciales) para determinar los efectos de las mismas sobre la estimación de la orientación de un vehículo autónomo.

5.4. Enfoque de la investigación

Este trabajo de investigación tiene un enfoque cuantitativo. El diseño e implementación de la plataforma es cuantitativa porque en el proceso de elaboración se utilizarán cálculos matemáticos para el desarrollo del algoritmo de fusión. Utiliza la recolección de datos para probar hipótesis con base en la medición numérica y el análisis estadístico, con el fin establecer pautas de comportamiento y probar (Hernández et al., 2014).

En efecto, se utilizarán cálculos matemáticos y su posterior análisis para establecer las pautas de comportamiento y sus efectos para determinar la orientación de un vehículo autónomo.

5.5. Unidad de estudio

Los datos provenientes de los sensores que perciben variables físicas del entorno del vehículo autónomo (aceleración, velocidad angular y campo magnético). Los sensores utilizados son un acelerómetro, un giroscopio y un magnetómetro, cuyas características se detallan en el Apéndice K.

5.6. Técnicas e instrumentos de recolección de datos

Se utilizan sensores (acelerómetro, giroscopio y magnetómetro) para percibir variables físicas del entorno (aceleración, velocidad angular y campo magnético), los datos técnicos de los sensores usados se detallan en el Apéndice K.

Conclusión

En este capítulo se describió, con todos sus pormenores, la metodología utilizado en el desarrollo de la investigación.

Capítulo 6

Diseño, simulación e implementación

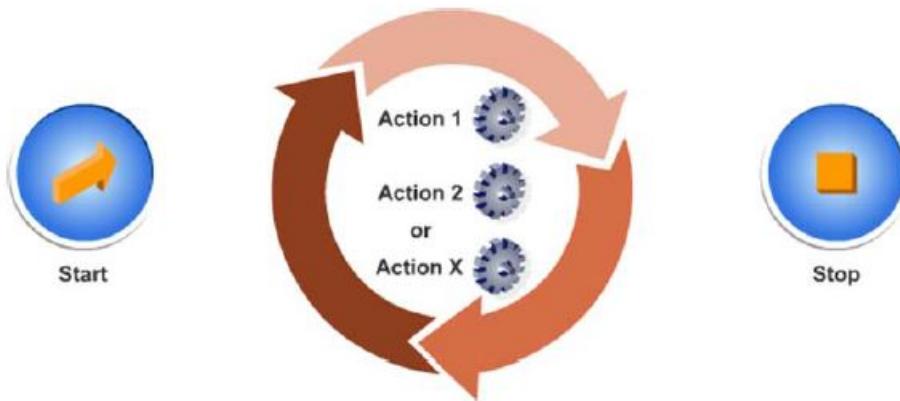
Resumen

En este capítulo se diseña simulaciones del filtro de Kalman para estimar el valor de una variable con ruido, se diseña el software que será utilizado, la interfaz de usuario, también se diseñan los algoritmos a ser implementados, empezando por la calibración de sensores y luego la implementación del filtro de Kalman para estimar los ángulos $Pitch (\theta)$, $Roll (\varphi)$ y $Yaw (\psi)$.

6.1. Diseño de software

EL diseño del programa está basado en un patrón de diseño denominado “máquina de estados” el cual consiste en una estructura de casos embebida en un bucle. Esta arquitectura permite ejecutar código diferente cada vez que se ejecuta el bucle, en función de alguna condición. Esto se detalla mejor en la Figuara 6.1.

Figura 6.1: Estructura básica de una máquina de estados.



Fuente: National Instruments (2014b)

La estructura de la Figura 6.1 resulta ser muy útil para nuestra aplicación, debido a que tenemos estados que se deben ejecutar antes de fusionar la información de sensores, como la calibración de los mismos, que luego ya no se ejecutarán.

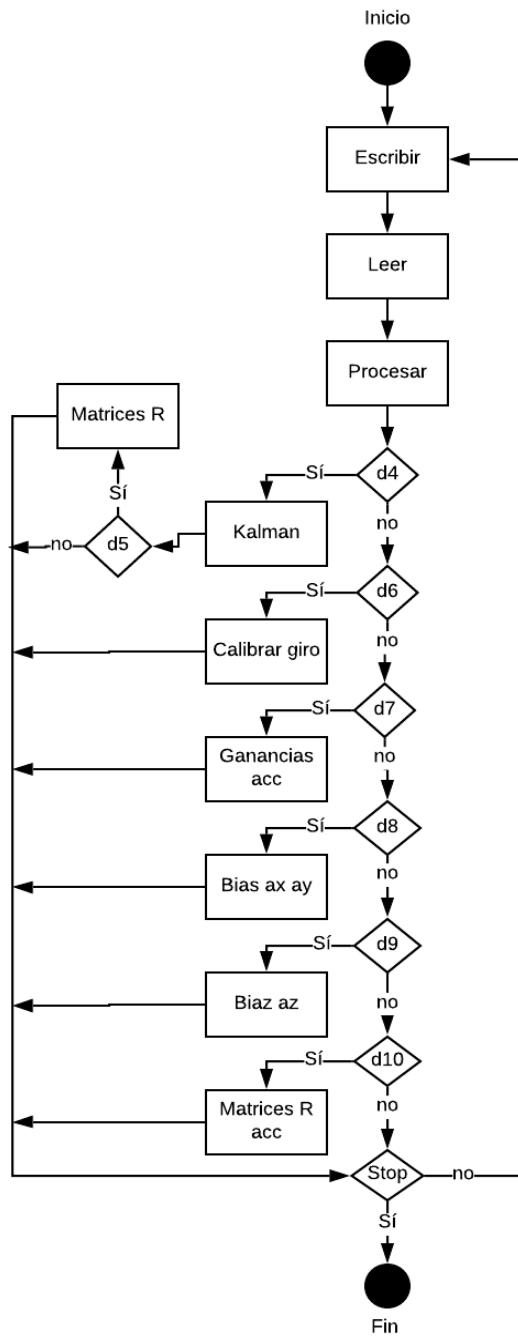
6.1.1. Diseño de la máquina de estados

Cabe resaltar que el lenguaje de programación seleccionado es LabVIEW. En el Apéndice I se da una introducción al lenguaje de programación y a las partes que la componen.

Una máquina de estados en LabVIEW consta de un bucle While, una estructura Case y un registro de desplazamiento. Cada máquina de estados es un caso distinto en la estructura Case (National, 2014b).

El diagrama de transición de estados de la máquina de estados implementada, se muestra siguiente figura.

Figura 6.2: Diagrama de transición de estados para este proyecto.



Fuente: Elaboración propia.

Consta de diez estados y de ocho entradas digitales (d4 al d10 y Stop), las entradas digitales d4 al d10 son para ejecutar cada estado de manera individual, la última entrada digital (Stop), es

para finalizar la ejecución del programa, cada estado se describe brevemente a continuación.

6.1.1.1. Escribir

Estado que se encarga de solicitar información al sensor.

6.1.1.2. Leer

Estado que se encarga de leer los datos del sensor.

6.1.1.3. Procesar

Estado que recibe y corrige los datos del sensor, utilizando variables locales de las matrices que se calculan en los estados de calibración respectivos.

6.1.1.4. Kalman (d4)

Estado que se encarga de la fusión de sensores, genera los gráficos mostrados en el panel frontal en el que se muestran los resultados de la fusión y los ángulos calculados a partir de las lecturas de cada sensor.

6.1.1.5. Matrices R (d5)

Estado que toma conjuntos de datos de las variables que intervienen en la fusión de sensores y calcula su varianza (componentes de las matrices de covarianza de las mediciones).

6.1.1.6. Calibrar Giro (d6)

Estado que se encarga del cálculo de bias de cada eje del giroscopio con el método de ajuste de mínimos cuadrados.

6.1.1.7. Ganancias acc (d7)

Estado que se encarga del cálculo de la matriz M requerida para la calibración del acelerómetro, el proceso requiere poner cada eje en dirección de la gravedad, de manera que cada columna de la matriz son calculados de manera independiente.

6.1.1.8. Bias ax ay (d8)

Estado que halla los sesgos de los ejes x e y del acelerómetro mediante un ajuste de mínimos cuadrados, aprovechando que LabVIEW ofrece VIS que realizan este cálculo. Antes de ejecutar este caso, se debe colocar el sensor en posición adecuada, con los ejes x y y en posición horizontal, de manera que no estén afectados por la fuerza de gravedad.

6.1.1.9. Bias az (d9)

Estado que halla el sesgo de la componente z del acelerómetro, se encuentra en un caso distinto, debido que se necesita poner el acelerómetro en una posición distinta, con el eje z del acelerómetro en posición horizontal.

6.1.1.10. Matrices R acc d10

Estado que toma un conjunto de datos de cada eje del acelerómetro y calcula su varianza, es necesario para el filtro de Kalman que estima las componentes de la matriz M necesario para la calibración del acelerómetro.

Los códigos implementados en LabVIEW se agregan en el Apéndice J.

6.1.2. Diseño de la interfaz de usuario

La interfaz del usuario está dividida en tres partes o pestañas, cada parte realizan tareas distintas que se describen a continuación.

6.1.2.1. Calibrar

En esta pestaña se pueden realizar las calibraciones de los sensores, a excepción de la calibración del magnetómetro, que se realiza offline con ayuda de Matlab.

Los procedimientos para la calibración se describen en las secciones 6.3 y 6.4 de este capítulo.

6.1.2.2. Kalman

En esta pestaña, se inicia la fusión de datos luego de la calibración y se pueden visualizar los resultados de la fusión y los ángulos calculados a partir de cada sensor de manera individual de manera a poder compararlos.

6.1.2.3. Matrices R

Una vez iniciada la fusión, se puede acceder a esta pestaña para modificar manualmente o calcular las matrices de covarianza de las mediciones de los ángulos y las velocidades angulares.

6.2. Simulación del filtro de Kalman de una variable

Resulta útil el estudio del filtro de Kalman para estimar el valor de una sola variable bajo la influencia de ruido blanco Gaussiano. Puede ayudar a entender mejor la influencia de los parámetros al evitar trabajar con matrices. También podemos analizar su eficiencia, tomando como parámetro de rendimiento el error cuadrático medio (error RMS) de la medición y de la estimación, al contar con el valor real de la variable de entrada.

Empezamos definiendo el vector de estados como:

$$x_k = x_c, \quad (6.1)$$

en donde x_c es una constante establecida arbitrariamente.

El vector de medición será el siguiente:

$$z_k = xc \quad (6.2)$$

El modelo del proceso y de la medición quedan de la siguiente manera:

$$x_k = x_{k-1} + w_k, \quad (6.3)$$

$$z_k = xc_k + v_k, \quad (6.4)$$

en (6.3) establecemos $\Phi = 1$, w_k tiene media cero varianza $Q = \sigma_p^2$, la ecuación (6.4) indica que se está midiendo la variable directamente, lo cual implica que $H = 1$, pero afectado por el ruido v_k con media cero y varianza $R = \sigma_{x_c}^2$.

Planteamos las ecuaciones de predicción como sigue:

$$\hat{xc}(-) = xc_{k-1}(+), \quad (6.5)$$

$$P_k(-) = P_{k-1}(+) + Q \quad (6.6)$$

Las ecuaciones de corrección son las siguientes:

$$\bar{K}_k = \frac{P_k(-)}{P_k(-) + R}, \quad (6.7)$$

$$\hat{xc}(+) = \hat{xc}(-) + \bar{K}_k(z_k - \hat{xc}(-)), \quad (6.8)$$

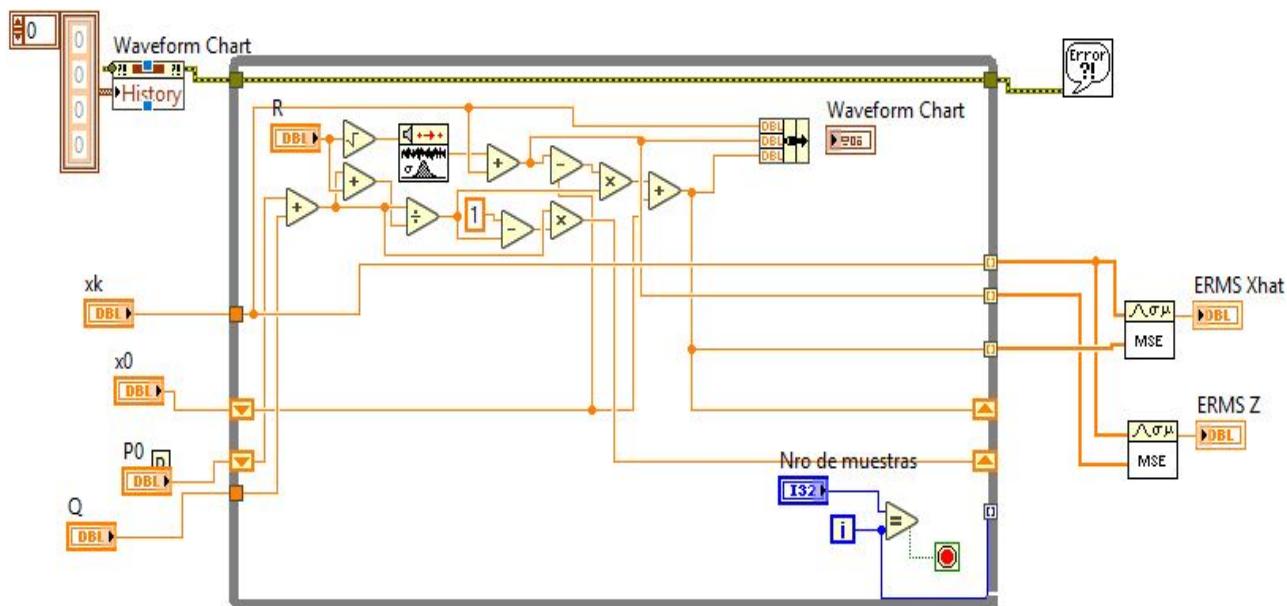
$$P_k(+) = (1 - \bar{K}_k)P_k(-), \quad (6.9)$$

como se puede ver en la ecuación (6.8), el valor de la ganancia de Kalman (\bar{K}_k) determina si la predicción o la medición tiene mayor peso sobre la estimación actual, con una ganancia que tiende a cero, tiene mayor peso la predicción, en cambio; con una ganancia cercana a uno, la medición actual es más influyente en la estimación del tiempo actual. La ganancia de Kalman \bar{K}_k a su vez, depende de, primeramente de R , si R tiene un valor elevado (que indica poca fiabilidad

en la medición), la ganancia de Kalman disminuye, y a medida que R disminuye, el valor de \bar{K}_k aumenta. También se puede ver que el valor de Q influye de manera inversa a R , aumentando la ganancia de Kalman si sus valores son grandes, y disminuyendo la ganancia con valores altos (Q indica la fiabilidad del proceso) (Huerta, 2015).

El diagrama de bloques de la simulación, se muestra en la Figura 6.3. Se aprovechan herramientas que ofrece LabVIEW para la visualización y el análisis de datos, como el cálculo de error RMS.

Figura 6.3: Diagrama de bloques para la simulación del filtro de Kalman de una variable.



Fuente: Elaboración propia.

Los resultados de esta simulación se muestran en el siguiente capítulo, sección 7.1.

6.3. Calibración del acelerómetro

Para la calibración del acelerómetro, utilizaremos un método propuesto en (Mohinder y Angus, 2015), con la ecuación (3.3). Este método requiere excitar la entrada del sensor con valores conocidos para hallar los parámetros de calibración, esto es conveniente para el acelerómetro debido a la estabilidad del valor de la gravedad. Para efectos prácticos (calibración en tiempo

real antes de una competencia), se ha hecho una pequeña modificación algebraica, de manera a poder hallar la matriz d y la matriz N^{-1} en la ecuación (3.4). Las modificaciones son las siguientes:

De la ecuación (3.3):

$$z_{output} = Mz_{input} + b$$

$$z_{output} - b = Mz_{input} \quad (6.10)$$

$$z_{input} = M^{-1}(z_{output} - b), \quad (6.11)$$

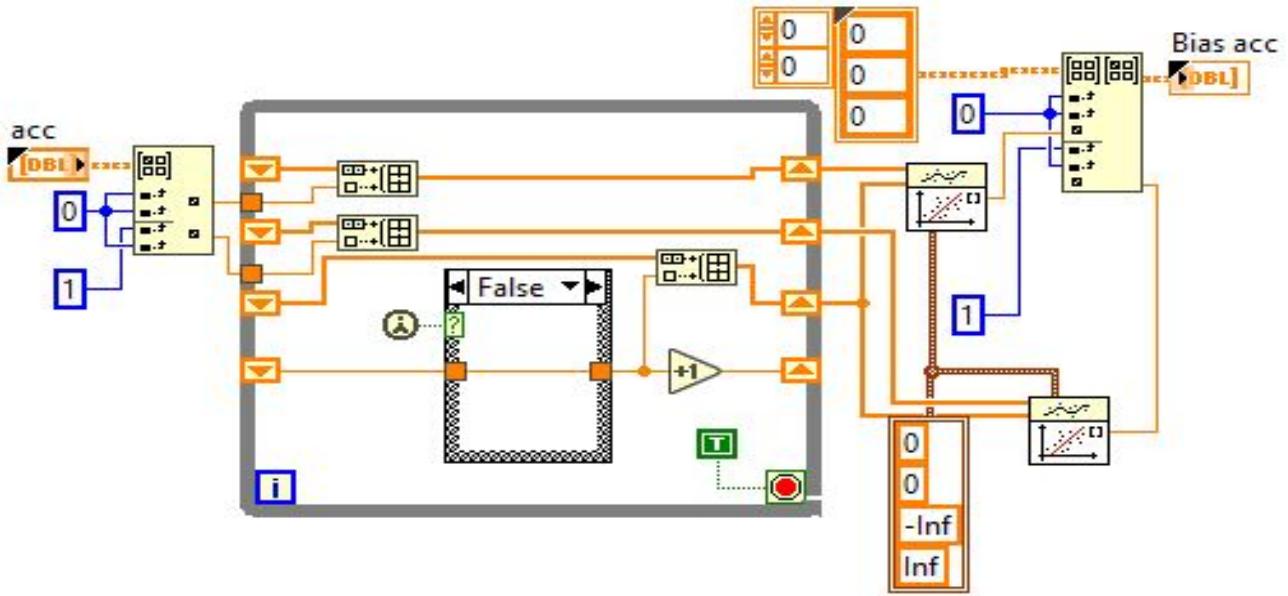
esto es conveniente, para hallar la matriz b (que representa el bias) cuando el sensor no está sometido a ninguna aceleración mediante un ajuste de mínimos cuadrados, y luego estimar el valor de la matriz M utilizando el filtro de Kalman teniendo como entrada conocida z_{input} , como vector de estado a los elementos de la matriz M , y como vector de medición $z_{output} - b$ (z_k), la salida del sensor con corrección de bias (offset).

Teniendo los valores de M , se calcula su inversa para obtener la matriz que representa la corrección de los factores de escala y desalineamiento de los ejes.

6.3.1. Cálculo de bias del acelerómetro

Teniendo en cuenta que el bias es la salida del sensor cuando este no se encuentra sometido a ninguna entrada, colocamos el sensor con dos ejes paralelo al suelo y tomamos una serie de datos, luego realizamos un ajuste de curva mediante el método de mínimos cuadrados. Por último, se repite esta operación para el tercer eje. LabVIEW ofrece un instrumento virtual que facilita este cálculo. El código implementado para hallar el bias de los ejes x e y puede verse a en la siguiente figura:

Figura 6.4: Diagrama de bloques para hallar b_{ax} y b_{ay} de la matriz b .



Fuente: Elaboración propia

b_{az} se halla de manera similar, pero colocando el eje z del acelerómetro en posición horizontal. Luego, la matriz b queda:

$$b = \begin{bmatrix} b_{ax} & b_{ay} & b_{az} \end{bmatrix}^T \quad (6.12)$$

6.3.2. Cálculo de la matriz M

Para el cálculo de esta matriz, sometemos a cada eje a un valor conocido de aceleración (la gravedad) y utilizaremos el filtro de Kalman para estimar los valores de las componentes de M . Primeramente definimos el vector de estado, que corresponden a los 9 elementos de la matriz M .

Sea:

$$M_{acc} = \begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix}$$

el vector de estados queda de la siguiente manera:

$$M_k = \begin{bmatrix} M_{00} & M_{01} & M_{02} & M_{10} & M_{11} & M_{12} & M_{20} & M_{21} & M_{22} \end{bmatrix}^T$$

El modelo del proceso (6.13) y de la medición (6.14) se definen en las siguientes ecuaciones:

$$M_k = M_{K-1} + w_k, \quad (6.13)$$

$$z_k = HM_k + v_k, \quad (6.14)$$

en donde z_k es $z_{output} - b$ (lectura de los sensores menos el bias) y H es una matriz de 3×9 que relaciona los vectores de estado con z_k .

Según (6.10) Mz_{input} también relaciona el las componentes del vector de estados con el vector de medición, es decir:

$$H_{3 \times 9} M_{9 \times 1} = M_{3 \times 3} z_{input}^{3 \times 1}, \quad (6.15)$$

en donde

$$z_{input} = \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T$$

a_1, a_2, a_3 corresponden a valores conocidos de vector aceleración.

El producto de Mz_{input} queda de la siguiente manera:

$$Mz_{input} = \begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} a_1 & a_2 & a_3 \end{bmatrix}^T,$$

luego:

$$Mz_{input} = \begin{bmatrix} M_{00}a_1 & M_{01}a_2 & M_{02}a_3 \\ M_{10}a_1 & M_{11}a_2 & M_{12}a_3 \\ M_{20}a_1 & M_{21}a_2 & M_{22}a_3 \end{bmatrix} \quad (6.16)$$

De esta manera, para que se cumpla (6.15), H debe ser:

$$H = \begin{bmatrix} a_1 & a_2 & a_3 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_1 & a_2 & a_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & a_1 & a_2 & a_3 \end{bmatrix}$$

que al multiplicar por la matriz de estados $M_{k9 \times 1}$, queda igual que (6.16).

En siguiente tabla, se presentan las dimensiones de las matrices que se utilizaron en el filtro de Kalman Lineal para hallar los valores de la matriz M .

Cuadro 6.1: Dimensiones de las matrices necesarias para aplicar el filtro de Kalman.

Matrices/Vectores	Símbolo	Filas	Columnas
Vector de estados	x_0	9	1
Matriz de transición de estados	Φ	9	9
Matriz de covarianza del proceso	Q	9	9
Vector de mediciones	z_k	3	1
Matriz de covarianza de la medición	R	3	3
Matriz de sensitividad de las mediciones	H	3	9
Covarianza del error	P_0	9	9

Fuente: Elaboración propia.

El vector de estados x_0 y la matriz de covarianza de error P_0 son las estimaciones iniciales y en 4.5.1.2 vimos que la estimación del estado x_k es independiente a estas estimaciones iniciales a medida que k aumenta. Entonces elegimos valores arbitrarios para x_0 y P_0 . Los valores iniciales se muestran a continuación junto con las demás matrices de la tabla 6.1a excepción de la matriz H cuyo valor ya se presentó anteriormente.

$$M_0 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T,$$

$$P0 = I_{9 \times 9},$$

$$\Phi = I_{9 \times 9},$$

$$Q = 1 \times 10^{-5} I_{9 \times 9},$$

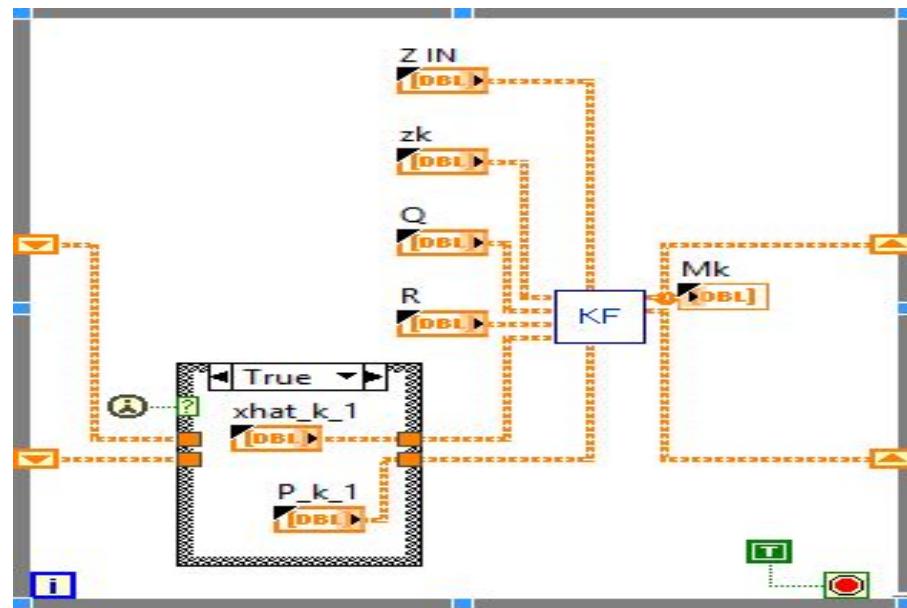
$$z_k = [z_{output} - b],$$

$$R = \begin{bmatrix} \sigma_{ax}^2 & 0 & 0 \\ 0 & \sigma_{ay}^2 & 0 \\ 0 & 0 & \sigma_{az}^2 \end{bmatrix}$$

en donde $I_{m \times m}$ es una matriz identidad de m filas y m columnas y σ_{ax}^2 , σ_{ay}^2 y σ_{az}^2 son las varianzas de un conjunto de mediciones tomadas de cada eje del acelerómetro, estando el vehículo en reposo.

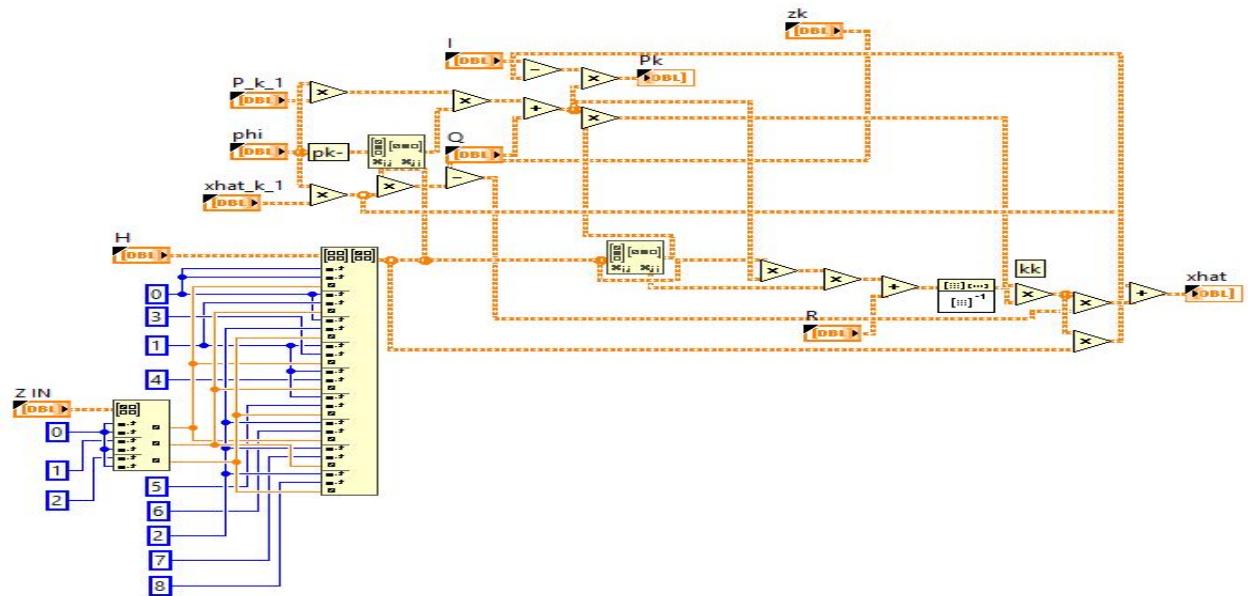
Teniendo el valor de todas las matrices necesarias, se implementó el filtro de Kalman, el diagrama de bloques del código se muestra en la Figura 6.5 y el subVI correspondiente en la Figura 6.6.

Figura 6.5: Diagrama de bloques subVI del filtro de Kalman para hallar la matriz M.



Fuente: Elaboración propia.

Figura 6.6: Diagrama de bloques del subVI KF.



Fuente: Elaboración propia.

Una vez obtenido los valores de la matriz M , se halla la inversa y se utiliza la ecuación (6.11)

para hallar los valores del acelerómetro calibrado y queda de la siguiente manera:

$$\begin{bmatrix} acx \\ acy \\ acz \end{bmatrix} = \begin{bmatrix} M_{00} & M_{01} & M_{02} \\ M_{10} & M_{11} & M_{12} \\ M_{20} & M_{21} & M_{22} \end{bmatrix}^{-1} \left(\begin{bmatrix} ax \\ ay \\ az \end{bmatrix} - \begin{bmatrix} b_{ax} \\ b_{ay} \\ b_{az} \end{bmatrix} \right) \quad (6.17)$$

en donde acx , acy y acz son los valores calibrados del acelerómetro y ax , ay y az son los valores sin calibrar.

6.4. Calibración del giroscopio

Para la calibración del giroscopio no podemos proceder de la misma manera que la calibración del acelerómetro, debido a que no disponemos de una plataforma que genere entradas conocidas para el sensor, por lo que se decide utilizar la (6.11), pero usando los datos de ganancia del fabricante en la matriz M^{-1} , sin hacer la corrección en el desalineamiento de los ejes. Luego, se procede a hallar la matriz b de la misma manera que en el acelerómetro, utilizando un código similar.

La calibración queda de la siguiente manera:

$$z_{input} = M^{-1}(z_{output} - b)$$

$$\begin{bmatrix} wx \\ wy \\ wz \end{bmatrix} = \left(\frac{1}{Gg} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} - \begin{bmatrix} b_{wx} \\ b_{wy} \\ b_{wz} \end{bmatrix} \right) \quad (6.18)$$

en donde wx , wy y wz son los valores calibrados del giroscopio, $\frac{1}{4,096}$ (en unidades $^0/s$) por la matriz identidad representa la ganancia de cada eje del giroscopio, y, b_{wx} , b_{wy} y b_{wz} son las salidas del sensor cuando no está sometido a rotación.

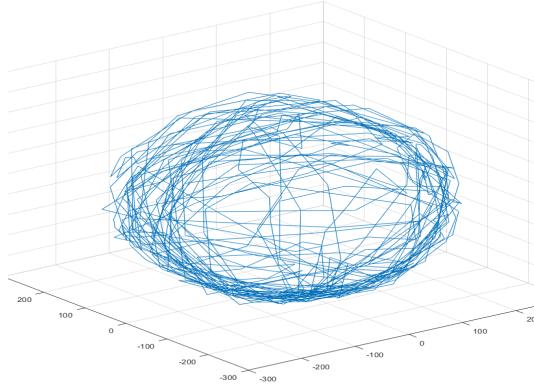
El código implementado es similar al de la figura 6.4, con la excepción de que se calculan

para los tres ejes simultáneamente.

6.5. Calibración del Magnetómetro

A diferencia de la calibración del acelerómetro y giroscopio, esta calibración se realiza offline, es decir, se toma un conjunto de datos, y se hace los cálculos de las matrices C y b de la ecuación (3.6) fuera de línea. Utilizando un algoritmo proporcionado por Yuri Petrov implementado en MATLAB, obtenemos las matrices deseadas.

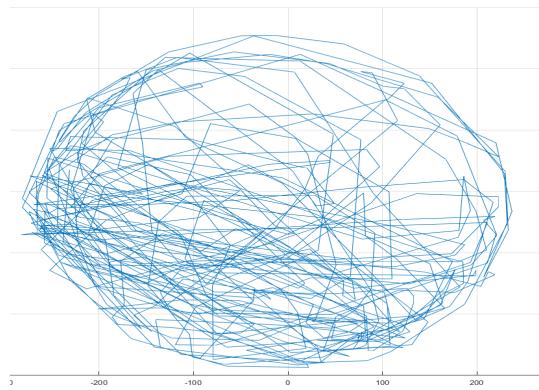
Figura 6.7: Datos del magnetómetro sin calibrar.



Fuente: Elaboración propia.

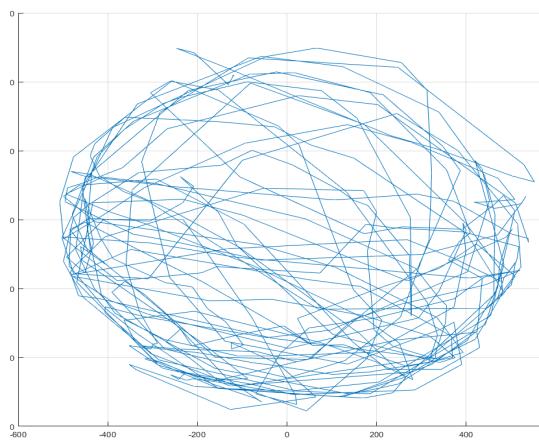
En la Figura 6.7 se observa los datos recogidos del magnetómetro sin calibrar, esta figura, se proyecta en el plano x-y (Figura 6.8) en donde se puede apreciar que la figura se asemeja a una elipse, y que el centro está ligeramente desplazado a la izquierda del origen. Luego de la calibración, se muestran los resultados en la Figura 6.9 en donde se corrige el despazamiento horizontal, aunque no se logra un circulo perfecto, se mejora la forma de la figura, respecto a la Figura 6.8.

Figura 6.8: Datos del magnetómetro sin calibrar, plano XY.



Fuente: Elaboración propia

Figura 6.9: Datos del magnetómetro calibrados, plano XY.

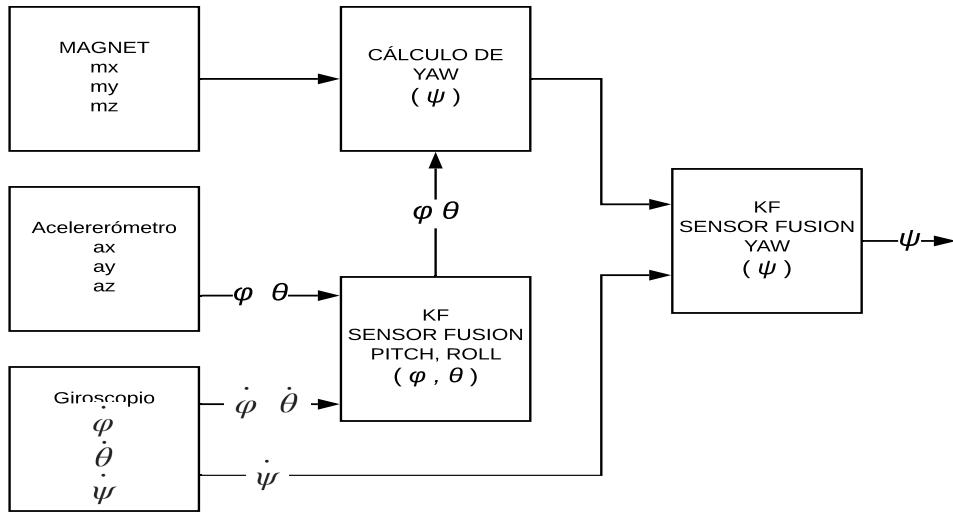


Fuente: Elaboración propia

6.6. Fusión de datos

La estructura básica de la fusión implementada es la siguiente:

Figura 6.10: Estructura de la fusión de datos.



Fuente: Elaboración propia

Para el cálculo de los ángulos Pitch (θ) y Roll (φ), se fusionan los datos provenientes del acelerómetro y del giroscopio. Primeramente, se calculan los ángulo φ y θ a partir de la lectura del acelerómetro, utilizando las relaciones trigonométricas entre los ejes y el vector gravedad, mediante la siguientes fórmulas:

$$\varphi = \arctan\left(\frac{ay}{az}\right) \quad (6.19)$$

$$\theta = -\arctan\left(\frac{ax}{\sqrt{ay^2 + az^2}}\right) \times \text{sign}(az), \quad (6.20)$$

Estas relaciones son válidas cuando el vehículo está en reposo o con velocidad constante, cuando el vehículo sufre aceleraciones, las lecturas no son válidas, es por esto que debemos fusionar estos ángulos con los datos del giroscopio.

El cálculo del ángulo Yaw (ψ) es resultado de la fusión del ángulo calculado con el magnetómetro y $\dot{\psi}$ proveniente del giroscopio, calculado a partir de (6.6).

Para el cálculo del ángulo Yaw (ψ), con el magnetómetro, proyectamos sus componentes x e

y sobre el plano horizontal utilizando las matrices de rotación presentadas en el Apéndice G, luego el ángulo Yaw (ψ) está dado por:

$$\psi = \arctan2\left(\frac{-m_{yh}}{m_{xh}}\right), \quad (6.21)$$

donde,

$$m_{yh} = m_y \cos \varphi_{KF} - m_z \sin \varphi_{KF},$$

$$m_{xh} = m_x \cos \theta_{KF} + m_y \sin \varphi_{KF} \sin \theta_{KF} + m_z \cos \varphi_{KF} \sin \theta_{KF},$$

los ángulos θ_{KF} y φ_{KF} , son los ángulos calculados a partir de la fusión del acelerómetro y el giroscopio, como se indica en la figura 6.10.

Las lecturas del giroscopio vienen dadas en el sistema de referencia del vehículo. Para su utilización en el filtro, debemos tener las lecturas referidas al sistema de referencia de navegación, para ello, utilizamos las relaciones entre los ejes descripta en el Apéndice G, en el que utilizamos los ángulos calculados en (6.19) y (6.20).

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & \sin(\varphi) \tan(\theta) & \cos(\varphi) \tan(\theta) \\ 0 & \cos(\varphi) & -\sin(\varphi) \\ 0 & \sin(\varphi) \sec(\theta) & \cos(\varphi) \sec(\theta) \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \quad (6.22)$$

siendo, p , q y r las lecturas adquiridas del giroscopio, que están referidas al marco de referencia del vehículo.

6.6.1. Filtro de Kalman para estimar los ángulos Roll φ , Pitch θ y Yaw ψ

Debido a que el algoritmo para la fusión de datos es similar para los tres ángulos, a excepción de las matrices Q y R , que serán diferentes para cada caso, se detallará de manera general la fusión, sobre una variable que es elegida arbitrariamente (α) y su derivada ($\dot{\alpha}$) (que representan el ángulo y la velocidad angular para cada caso).

Entonces, el vector de estados será el siguiente:

$$x_k = \begin{bmatrix} \alpha \\ \dot{\alpha} \end{bmatrix} \quad (6.23)$$

El modelo del proceso se representa por la siguiente ecuación en diferencias lineal estocástica:

$$\underbrace{\begin{bmatrix} x_k \\ \alpha_k \\ \dot{\alpha}_k \end{bmatrix}}_{\tilde{x}_k} = \underbrace{\begin{bmatrix} \Phi \\ 1 & \Delta T \\ 0 & 1 \end{bmatrix}}_{\tilde{A}} \underbrace{\begin{bmatrix} x_{k-1} \\ \alpha_{k-1} \\ \dot{\alpha}_{k-1} \end{bmatrix}}_{\tilde{x}_{k-1}} + w_{k-1} \quad (6.24)$$

en donde w representa el ruido del proceso con media cero y matriz de covarianza Q . Según (Mohinder y Angus, 2015), para la matriz Φ dada en (6.24), le corresponde la siguiente matriz Q :

$$Q_k = q \begin{bmatrix} \frac{(\Delta t)^3}{3} & \frac{(\Delta t)^2}{2} \\ \frac{(\Delta t)^2}{2} & \Delta t \end{bmatrix} \quad (6.25)$$

en donde q es la covarianza del ruido del proceso en tiempo continuo y Δt es el tiempo de iteración del algoritmo. El valor de q lo ajustamos manualmente mediante prueba y error (Huerta, 2015).

El vector de medición está dado por:

$$\underbrace{\begin{bmatrix} z_k \\ \alpha \\ \dot{\alpha} \end{bmatrix}}_{\tilde{z}_k} = \underbrace{\begin{bmatrix} H \\ 1 & 0 \\ 0 & 1 \end{bmatrix}}_{\tilde{C}} \underbrace{\begin{bmatrix} x_k \\ \alpha_k \\ \dot{\alpha}_k \end{bmatrix}}_{\tilde{x}_k} + v_k \quad (6.26)$$

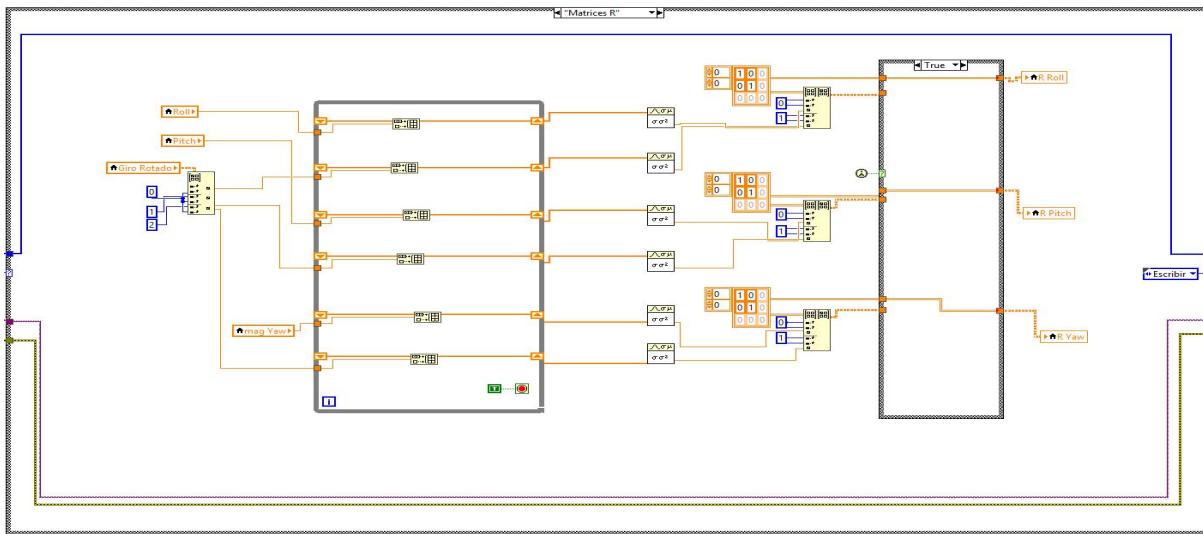
en donde v es el ruido de las mediciones con media cero y matriz de covarianza R .

La matriz de covarianza R de las mediciones está dada por:

$$R = \begin{bmatrix} \sigma_\alpha^2 & \sigma_\alpha \sigma_{\dot{\alpha}} \\ \sigma_{\dot{\alpha}} \sigma_\alpha & \sigma_{\dot{\alpha}}^2 \end{bmatrix} \quad (6.27)$$

en donde σ_α^2 y $\sigma_{\dot{\alpha}}^2$ son las varianzas de los ruidos de las mediciones α y $\dot{\alpha}$ respectivamente, $\sigma_\alpha \sigma_{\dot{\alpha}} = \sigma_{\dot{\alpha}} \sigma_\alpha$ son las covarianzas de las mediciones. En la siguiente figura, se muestra la implementación en LabVIEW del cálculo de las matrices de covarianza de las mediciones.

Figura 6.11: Diagrama de bloques para el cálculo de las matrices R.

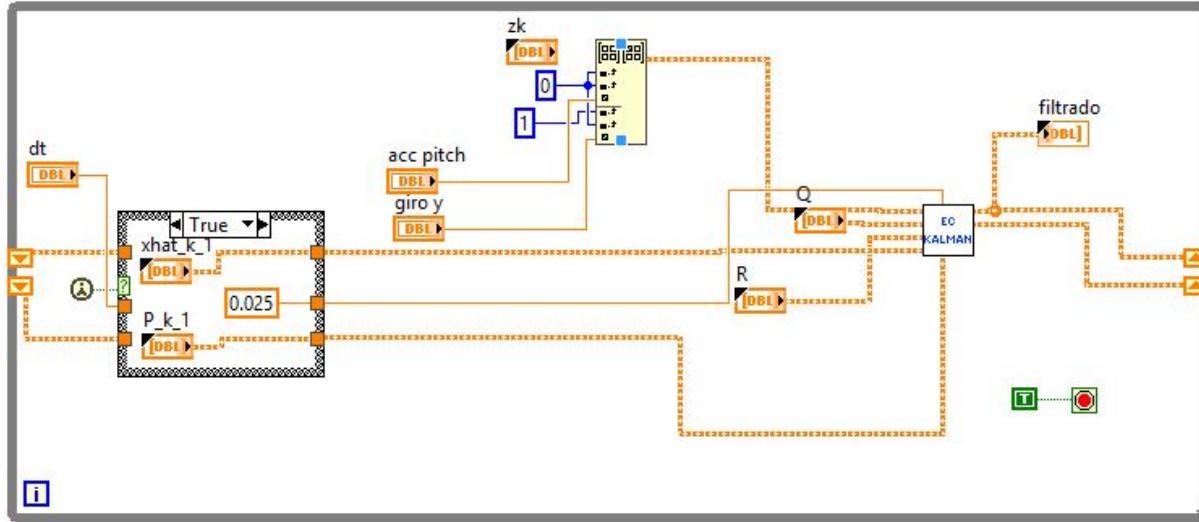


Fuente: Elaboración propia.

El cálculo de las matrices de covarianza de las mediciones se realizó con el vehículo en reposo y con el vehículo sometido a aceleraciones lineales, los efectos de esto se exponen en el siguiente capítulo.

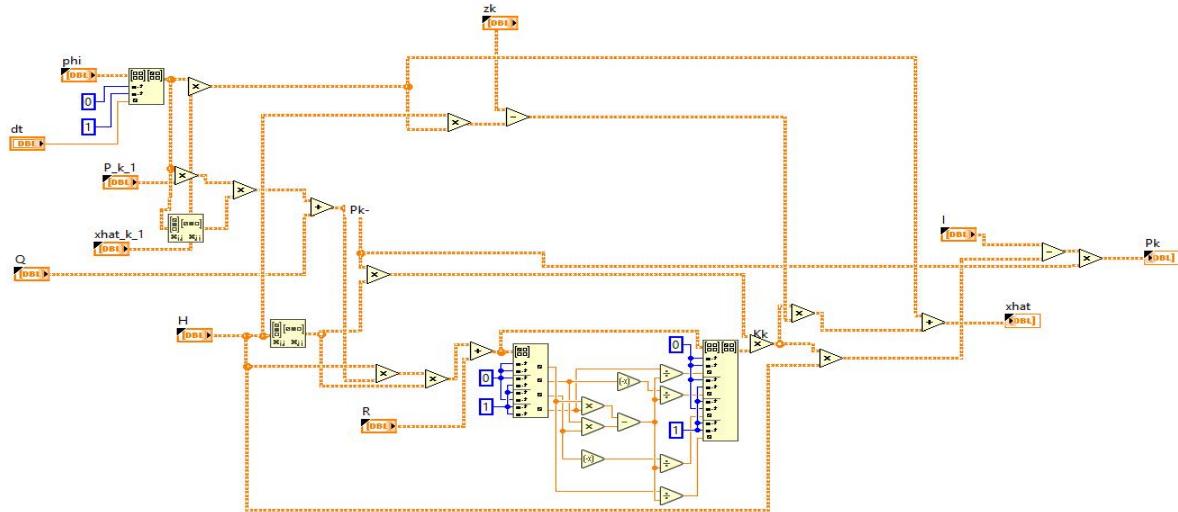
Con estas matrices definidas, ya se puede implementar el filtro de Kalman utilizando las ecuaciones (4.11) al (4.15). La siguiente figura se muestra el diagrama de bloques de la fusión entre el acelerómetro y giroscopio para el cálculo del ángulo Pitch (θ).

Figura 6.12: Filtro de Kalman implementado en LabVIEW para estimar Pitch θ .



Fuente: Elaboración propia.

Figura 6.13: Diagrama de bloques del subVI EC KALMAN de la figura 6.12.



Fuente: Elaboración propia.

En el Apéndice J se encuentran los códigos implementados para la estimación de los ángulos.

Conclusión

Se ha diseñado una simulación para estimar el valor de una variable, cuya medición está efectuada por ruido blanco gaussiano, esto, para analizar y estudiar el comportamiento del algoritmo, y la influencia de la variación de parámetros como las matrices de covarianza del ruido del proceso (Q) y de la medición (R), los resultados se exponen en el siguiente capítulo. Se realizó el diseño de la estructura general del programa, que está basada en una máquina de estados, también, el diseño de la interfaz de usuario.

Se ha mostrado los diseños de los algoritmos que fueron implementados, tanto para la calibración como para la fusión de datos.

Capítulo 7

Puebas y resultados

Resumen

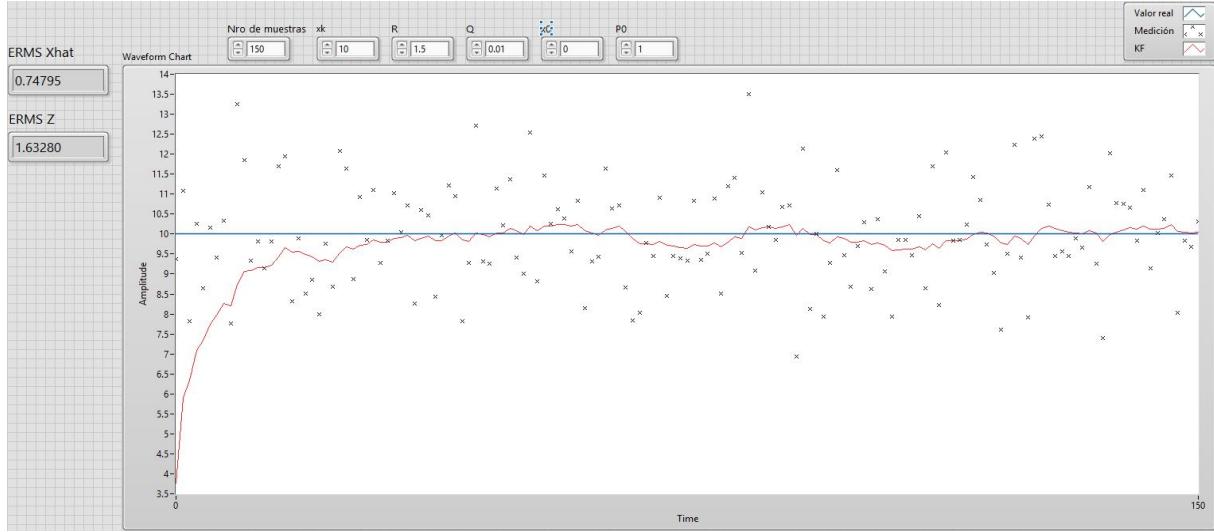
En este capítulo presentaremos los resultados de las implementaciones realizadas, verificaremos el funcionamiento del filtro de Kalman, primeramente con la simulación, y luego veremos el comportamiento del filtro ante ruidos que se introducen al sistema, como aceleraciones lineales (caso del acelerómetro) y distorsiones magnéticas (caso del magnetómetro).

7.1. Simulaciones del filtro de Kalman sobre una variable

Se muestran los resultados de la simulación diseñada en la sección 6.2. El diseño de la interfaz permite que podamos modificar los parámetros Q , R , x_0 , P_0 y se puede visualizar el error RMS, tanto del filtro, como de las mediciones sin el filtro, de manera a comprobar su efectividad.

El panel frontal del diagrama de bloques de la figura 6.3 es el siguiente:

Figura 7.1: Panel frontal del diagrama de bloques de la figura 6.2.



Fuente: Elaboración propia.

En la Figura 6.2: en azul, se muestra el valor real de la constante que se quiere estimar; los puntos en negro son los valores de las mediciones, que es el valor real, sumado con una variable aleatoria varianza establecida arbitrariamente por el usuario y en rojo se muestran las estimaciones de la variable real. En este caso, el valor de la variable real es 10, y el valor inicial de la estimación es 0, se puede ver la convergencia del filtro después de cierto número de iteraciones.

En la siguientes tablas, podemos ver la influencia de los parámetros Q y R sobre la eficiencia del filtro, manteniendo constantes las demás variables ($x_k = 10$, $x_0 = 0$, $P_0 = 1$), el número de muestras es 150.

Cuadro 7.1: Errores RMS con $Q = 0,2$ y R variable.

Q	R	ERMS KF	ERMS MEDICIÓN
0,2	0,25	0,14186	0,24519
0,2	0,5	0,21237	0,51645
0,2	1	0,39957	1,09777
0,2	1,25	0,42984	1,29836
0,2	1,5	0,59537	1,80550

Fuente: Elaboración propia.

Luego, se deja constante R y se varía Q .

Cuadro 7.2: Errores RMS con Q variable y $R = 0,5$.

Q	R	ERMS KF	ERMS MEDICIÓN
0,1	0,5	0,09052	0,42115
0,05	0,5	0,07717	0,51715
0,001	0,5	0,00875	0,49620
0,0005	0,5	0,00701	0,49013
0,0001	0,5	0,00403	0,48012

Fuente: Elaboración propia.

A partir de estos datos, podemos ver la eficiencia del filtro de Kalman, mediante el cual se reduce en todos los casos, el error RMS, como Q representa la varianza de la incertidumbre del proceso, y sabemos que estamos midiendo una constante, el filtro resulta más efectivo a medida que $Q \rightarrow 0$, como se observa en la Tabla 6.1, también vemos que al ir aumentando la incertidumbre de la medición, la predicción gana más peso, reduciendo significativamente el error (Tabla 7.1), como se había mencionado en la sección 6.2.

7.2. Fusión Roll y Pitch

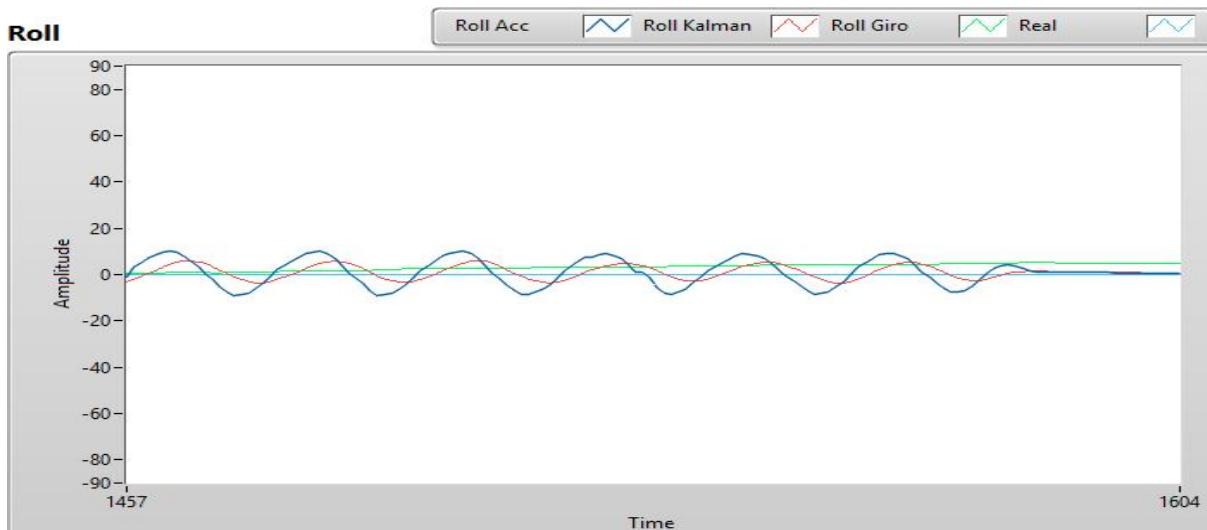
En esta sección se presentan los resultados de la fusión para estimar el ángulo Roll (φ) utilizando las ecuaciones planteadas en la sección 6.6.1. No se presenta los resultados de la fusión para estimar el ángulo Pitch (θ), debido a que es similar a los resultados de la fusión Roll (φ). La estimación de estos ángulos es muy importante, para luego estimar la orientación utilizando la ecuación (6.21).

Primeramente se ha hecho las pruebas con el cálculo de la matriz de covarianza de las mediciones R con el vehículo en reposo, y se obtuvo una buena lectura de las inclinaciones, pero una mala respuesta cuando el vehículo se somete a aceleraciones distintas al de la gravedad, debido a que las ecuaciones (6.19) y (6.20) solo son válidas para el vehículo en reposo. Podemos ver el resultado de la implementación en la Figura 7.2.

R calculada con el vehículo en reposo:

$$R1 = \begin{bmatrix} 0,00270492 & 2,315 \times 10^{-5} \\ 2,315 \times 10^{-5} & 0,0033176 \end{bmatrix}$$

Figura 7.2: Respuesta del filtro ante aceleraciones horizontales con $R1$.



Fuente: Elaboración propia.

En la Figura 7.2 se muestra el ángulo Roll calculado a partir del acelerómetro (en azul), del giroscopio (en verde) y la estimación a partir de la fusión de sensores (en rojo), la línea celeste corresponde a una línea de referencia (0 grados).

Se observa el error que genera la lectura del ángulo del acelerómetro cuando el vehículo es sometido a aceleraciones horizontales, al tener una varianza pequeña del error, el filtro da fiabilidad a estas lecturas, y no puede ignorar esas aceleraciones, que es lo que queremos conseguir en este proyecto. Tomando el plano horizontal como referencia, esta prueba arroja los resultados mostrados en la siguiente tabla:

Cuadro 7.3: Errores RMS de la primera prueba de la fusión Roll.

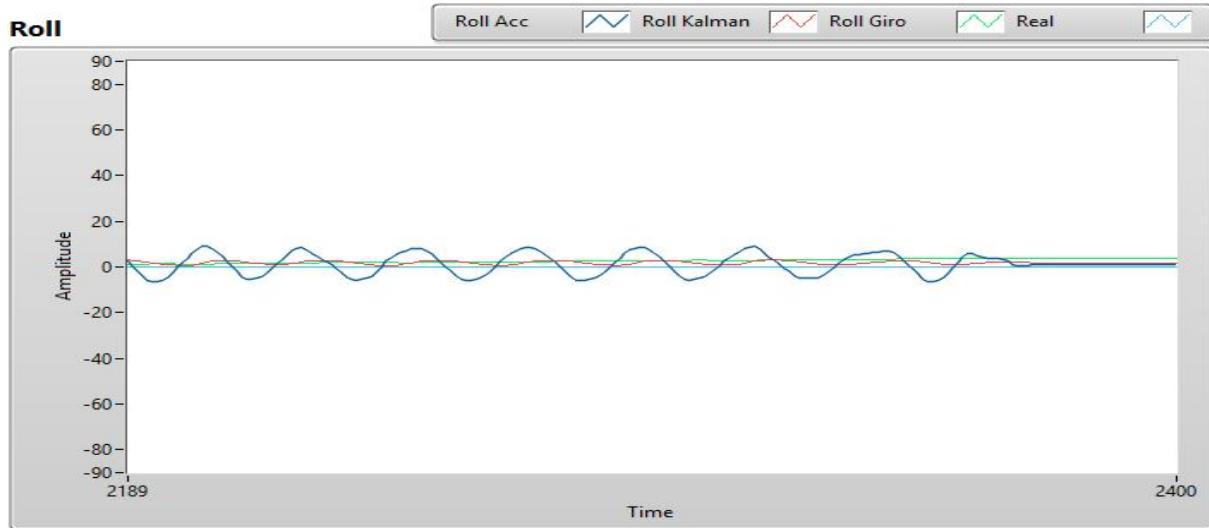
φ	Error RMS
Acelerómetro	38,41379
Giroscopio	14,515079
Fusión	10,45079

Fuente: Elaboración propia.

Los resultados obtenidos, no son satisfactorios para el objetivo deseado, debido a esto, se calcula la matriz de covarianza de las mediciones (R) con el vehículo sometido a aceleraciones horizontales. La R calculada se muestra a continuación con el resultado de la implementación:

$$R2 = \begin{bmatrix} 15,5496 & 1,78962 \\ 1,78962 & 3,14495 \end{bmatrix}$$

Figura 7.3: Respuesta del filtro ante aceleraciones horizontales con $R2$.



Fuente: Elaboración propia.

El uso de esta R se acerca a lo deseado, aunque no ignora completamente, reduce en gran medida la oscilación de ángulo Roll (θ).

Podemos ver el rendimiento del filtro en la siguiente tabla:

Cuadro 7.4: Errores RMS de la segunda prueba de la fusión Roll.

φ	Error RMS
Acelerómetro	35,50311
Giroscopio	12,60833
Fusión	2,34261

Fuente: Elaboración propia.

Mediante el método de ajuste de prueba y error, llegamos a un valor de R que se acerca más a lo deseado para este proyecto:

$$R = \begin{bmatrix} 8 & 0,6 \\ 0,6 & 1,3 \end{bmatrix}$$

con el siguiente rendimiento:

Cuadro 7.5: Errores RMS de la tercera prueba de la fusión Roll.

φ	Error RMS
Acelerómetro	36,88941
Giroscopio	11,0113
Fusión	1,96216

Fuente: Elaboración propia.

7.3. Fusión Yaw

Una vez obtenidos los ángulos Pitch y Roll, se puede calcular Yaw a partir de la lectura del magnetómetro utilizando la ecuación (6.21), este ángulo se fusiona con la lectura del giroscopio Ψ , obtenida de la ecuación (6.6).

Primeramente, se calcula la matriz R tomando una cierta cantidad de datos, y calculando la varianza de las mediciones, luego sometemos el sensor a disturbios magnéticos (causado con un pequeño imán permanente) para ver el comportamiento de la fusión, finalmente, se realiza un ajuste manual, hasta que alcance el comportamiento deseado.

Con R :

$$R1 = \begin{bmatrix} 0,0503868 & 0,000178227 \\ 0,000178227 & 0,00113254 \end{bmatrix} \quad (7.1)$$

Se tienen los resultados mostrados en las siguientes figuras:

Figura 7.4: Respuesta de la fusión Yaw.

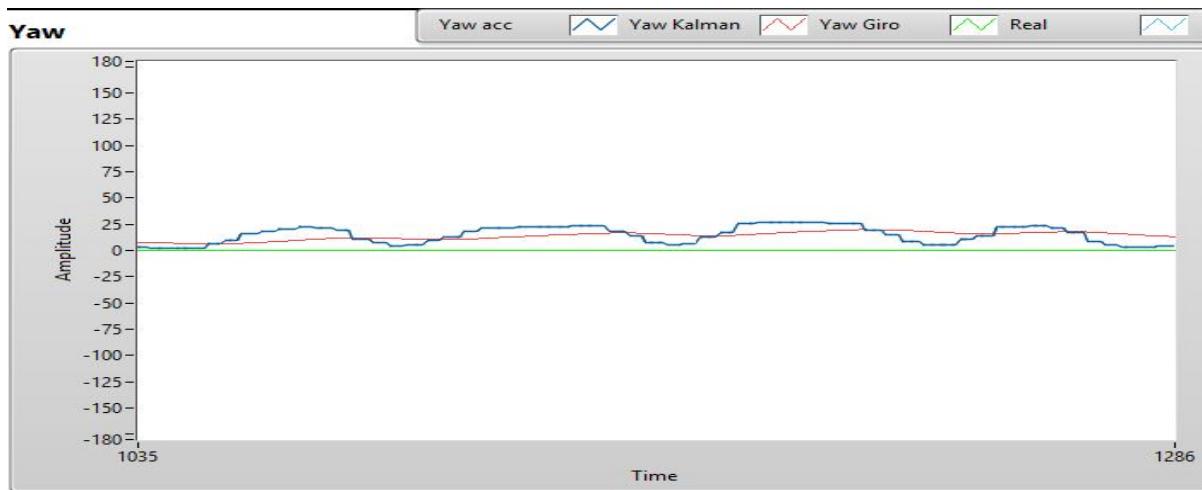


Fuente: Elaboración propia.

En la Figura 7.5 se muestra el ángulo Yaw calculado a partir del magnetómetro (en azul), del giroscopio (en verde) y la estimación a partir de la fusión de sensores (en rojo).

Se puede notar la mejoría que introduce el giroscopio, suavizando la curva, y la componente del magnetómetro contribuye a que no haya errores acumulativos (propias del giroscopio). Pero presenta problemas ante la presencia de distorsiones magnéticas como se puede ver en la siguiente figura:

Figura 7.5: Respuesta de la fusión Yaw ante distorsiones magnéticas con R1.



Fuente: Elaboración propia.

En la Figura 7.5 queda claro que la distorsión magnética no afecta al giroscopio, en cambio, afecta en gran medida a la lectura del magnetómetro y a salida del filtro de Kalman. Se busca que el filtro pueda ignorar estos disturbios. Tomando como referencia la orientación antes de la presencia de distorsión, se puede estimar rendimiento del filtro tomando los errores RMS de las lecturas de los ángulos. Esto se puede ver en la siguiente tabla:

Cuadro 7.6: Errores RMS de la primera prueba de la fusión Yaw.

ψ	Error RMS
Magnetómetro	260,36643
Giroscopio	0,0170
Fusión	190,5

Fuente: Elaboración propia.

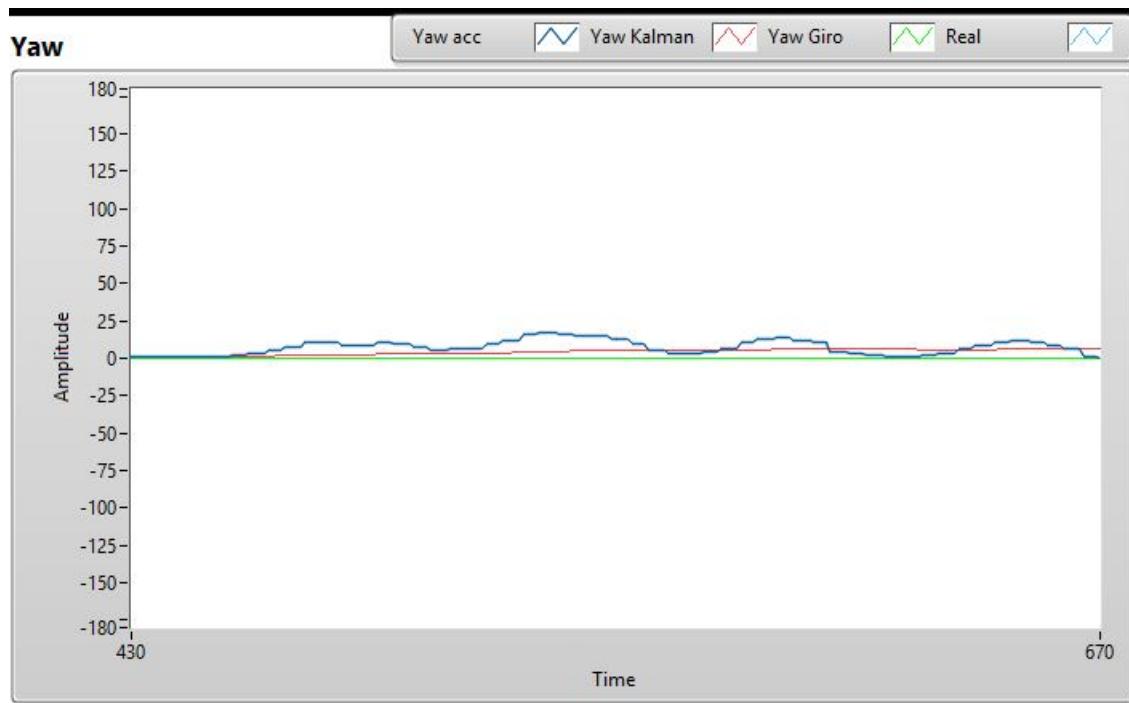
Para mejorar estos datos, hacemos un ajuste manual, de manera a darle mayor peso la lectura del giroscopio, aumentando la varianza de la medición del magnetómetro, de manera a que sea más robusta en ambientes con perturbaciones magnéticas.

Con varios ensayos de prueba y error, se llega al siguiente valor de R .

$$R2 = \begin{bmatrix} 29 & 0 \\ 0 & 0,000798572 \end{bmatrix} \quad (7.2)$$

Los resultados se pueden ver en la siguiente figura:

Figura 7.6: Respuesta de la fusión Yaw ante distorsiones magnéticas con R2.



Fuente: Elaboración propia.

En la Figura 7.6 vemos un mejor comportamiento del filtro ante perturbaciones, de esta manera, se consigue que las lecturas provenientes del giroscopio tenga más peso en la estimación del ángulo Yaw. Los errores RMS se ven en la siguiente tabla:

Cuadro 7.7: Errores RMS de la segunda prueba de la fusión Yaw.

ψ	Error RMS
Magnetómetro	250,822902
Giroscopio	0,02036
Fusión	4,065

Fuente: Elaboración propia.

Con esto se consigue un comportamiento adecuado para el entorno de aplicación de la fusión.

Dependiendo del entorno de aplicación, se pueden escoger los valores de R adecuados.

Conclusión

En este capítulo, se ha visto los resultados de los algoritmos diseñados en el capítulo anterior, con la simulación, se pudo estudiar el comportamiento del algoritmo ante las variaciones de las variables que intervienen en sus ecuaciones, de esta manera, se puede seleccionar constantes adecuadas para el tipo de aplicación que necesitemos, dependiendo del entorno de uso.

Consideraciones finales

Conclusiones

Resulta indiscutible la importancia de la fusión de datos de distintos sensores para obtener la calidad adecuada de las mediciones de las variables que deben ser procesadas, conforme a lo que se ha expuesto en este trabajo. Y en efecto, mediante este procedimiento se logra la utilización de sensores a bajo costo para usarlas mediante la combinación y de este modo funcionan como los dispositivos de alta calidad pero a un costo mucho más bajo.

Como resultado de la presente investigación se pudo fusionar los sensores (acelerómetro, giroscopio y magnetómetro) de manera correcta y se obtuvo datos más precisos que permitió extraer la información más relevante y exacta posible, a partir de esto se pudo planificar y ejecutar acciones para hacer posible la navegación de un vehículo autónomo.

Con el diseño y la implementación del filtro de Kalman se pudo minimizar los errores en la fusión de los datos y de este modo se pudo estimar la orientación del vehículo y por tanto la navegación del mismo de manera autónoma. El análisis del comportamiento del algoritmo mediante el diseño de simulaciones sirvió para determinar cómo influyen distintos parámetros en la calidad de las estimaciones.

EL filtro de Kalman, también fue utilizado para la calibración del acelerómetro y mostró muy buenos resultados. Por tanto, se ha logrado responder a todas las preguntas de investigación y en consecuencia se ha logrados todos los objetivos en este trabajo. De este modo: Se desarrollaron los procedimientos laboratoriales para la calibración de sensores, se analizó la aplicación del Filtro de Kalman en la fusión de datos de sensores, se determinó el proceso de calibración de los sensores utilizados para la navegación de un vehículo autónomo, se pudo establecer la efectividad del filtro de Kalman para la fusión de datos de distintos sensores, se diseñaron las simulaciones del filtro de Kalman para el estudio de los parámetros que intervienen en sus ecuaciones (como las matrices de covarianza del ruido del proceso y medición) y se diseñaron los algoritmos en la calibración de sensores.

Entonces, mediante el logro de los objetivos específicos se ha alcanzado el objetivo general y por tanto se ha fusionado los datos de sensores, lo que hizo posible estimar la orientación de un vehículo autónomo a escala mediante la aplicación del Filtro de Kalman.

Trabajos futuros

Este trabajo puede constituir la base de investigaciones futuras tales como:

- **Matrices de covarianza del ruido variable:** A lo largo del proyecto hemos visto la influencia de esta matriz sobre el comportamiento de la fusión, pero se asumen constantes una vez iniciado el proceso. Esto se puede modificar en función de las varianzas de las lecturas de los sensores para detectar períodos estáticos y dinámicos (caso del acelerómetro), de manera a mejorar el rendimiento del filtro dependiendo de las características del entorno.
- **Filtro extendido de Kalman:** Hemos aplicado el filtro de Kalman a partir del cálculo de los ángulos usando las relaciones trigonométricas entre el vector gravedad y las componentes de aceleración del acelerómetro. Estas relaciones se pueden incluir en un filtro extendido de Kalman (debido a que no son relaciones lineales) e implementar la fusión con los datos calibrados de cada sensor. Además, se puede incluir la estimación de la posición del vehículo autónomo a partir de las lecturas del acelerómetro.
- **Añadir el encoder a la fusión para el cálculo de posición:** Aunque para el desarrollo del vehículo se haya usado un encoder para estimar la distancia recorrida, no fué parte de la fusión de sensores, se lo puede incluir para minimizar los errores de estimación de la posición.
- **Añadir un módulo GPS (Global Positioning System):** para realizar una navegación teniendo un marco de referencia absoluta y no relativa, e incluir en la fusión de sensores mediante un filtro extendido de Kalman.

Referencias

- Amezcua Paredes, R., y Pineda Salgado, A. G. (2013). *Sistema de referencia inercial: análisis de funcionamiento, fundamentos y evolución* (Tesis Doctoral no publicada).
- Areny, R. P. (2004). *Sensores y acondicionadores de señal*. Marcombo.
- Ávila, M. A. O., y Arancibia, J. A. G. (2015). Sistema de localización autónoma para robots móviles basado en fusión de sensores propioceptivos. *Revista Politécnica*, 11(21), 75–84.
- Bogado, Y., y Benitez, W. (2015). Desarrollo de un prototipo de vant (vehículo aéreo no tripulado) para inspección visual de líneas eléctricas aéreas. *Universidad Católica Nuestra Señora de la Asunción, Hernandarias*.
- Bresciani, T. (2008). Modelling, identification and control of a quadrotor helicopter. *MSc Theses*.
- Fischer, T. (2014). Fusión de encoders de cuadratura, sensores inerciales y magnéticos para la localización de robots móviles. *UBA, Buenos Aires*.
- Hall, D. L., y Llinas, J. (1997). An introduction to multisensor data fusion. *Proceedings of the IEEE*, 85(1), 6–23.
- Hernández Sampieri, R., Fernández Collado, C., Baptista Lucio, P., y cols. (2014). *Metodología de la investigación* (Vol. 6). México: McGraw-Hill.
- Honeywell. (2012). *Datasheet hmc5883l*, 2012.
- Huerta, M. (2015). Implementación de un sistema de determinación de orientación mediante diseño sopc en una fpga para vehículo aéreo no tripulado del tipo quadrotor.
- INVENSENSE, I. (2013). *Datasheet mpu6050, revision 3.4*, 2013.
- Kalman, R. (1960). A new aproach to linear filter and prediction theory. *J. Basic Eng. D*, 82, 35–45.
- López Montes, D. (2016). Fusión sensorial para la estimación del estado de un vehículo autónomo.

Universidad de Cantabria, Santander.

- Madgwick, S. O., Harrison, A. J., y Vaidyanathan, R. (2011). Estimation of imu and marg orientation using a gradient descent algorithm. En *2011 ieee international conference on rehabilitation robotics* (pp. 1–7).
- Mahony, R., Hamel, T., y Pflimlin, J. M. (2005). Complementary filter design on the special orthogonal group so (3). En *Proceedings of the 44th ieee conference on decision and control* (pp. 1477–1484).
- Mohinder, G., y Angus, A. (2008). *Kalman filtering: Theory and practice using matlab* (3.^a ed.). Hoboken, New Jersey: JOHN WILEY & SONS, INC.
- Mohinder, G., y Angus, A. (2015). *Kalman filtering: Theory and practice using matlab* (4.^a ed.). Hoboken, New Jersey: JOHN WILEY & SONS, INC.
- NASA. (2018). *Nasa ames research center, moffett field, calif., history related to the apollo moon program and lunar prospector mission.* https://www.nasa.gov/centers/ames/news/releases/2004/moon/apollo_ames_atmos.html.
- National, I. (2014a). *Labview core 1, participant guide*. Austin, Texas: National Instruments.
- National, I. (2014b). *Labview core 2, participant guide*. Austin, Texas: National Instruments.
- Pozo, D., Sotomayor, N., Rosero, J., y Morales, L. (2014). Medición de ángulos de inclinación por medio de fusión sensorial aplicando filtro de kalman. *Revista Politécnica*, 33(1).
- Prado O., G. (2005). *Técnicas recursivas para estimación dinámica: una introducción matemática al filtro de kalman* (Tesis Doctoral no publicada). Universidad Nacional de Colombia.
- SparkFun. (2011). *Ivensense 3-axis-gyro-accelerometer-ic mpu-6050.* https://upload.wikimedia.org/wikipedia/commons/8/87/IvenSense_3-Axis-Gyro-Accelerometer-IC_MPU-6050_10937-01.jpg.
- VectorNav Technologies, L. (2019). *Magnetometer*. <https://www.vectornav.com/support/library/magnetometer>.
- Vigouroux Cavolina, D. (2010). Implementación de unidad de mediciones iniciales (imu) para robótica utilizando filtro de kalman. *Sartenejas, Venezuela*.

Xueli, C., Wanli, L., Meng, G., y Zhenhua, Z. (2018). Mobile robot self-localization based on multi-sensor fusion using limited memory kalman filter with exponential fading factor. *Journal of Engineering Science Technology Review*, 11(6), 187 - 196. Descargado de <http://search.ebscohost.com/login.aspx?direct=true&AuthType=ip,sso&db=egs&AN=135119753&lang=es&site=eds-live>

Apéndices

Apéndice A

Proyecto Aguara'i

A.1. Resultados del proyecto

En efecto, en la competencia internacional de vehículos autónomos a escala RobocarRace (2018) llevada a cabo en la ciudad de San Pablo - Brasil, organizada por la Facultad de Tecnología de Santo André, Universidad Federal ABC y Robótica Paula Souza (RobocarRace, 2018) en la cual se ha participado en representación de la Universidad Católica Nuestra Señora de la Asunción, sede Regional Alto Paraná, Facultad de Ciencias y Tecnología, la participación del equipo ha sido exitosa, pues se logró:

- Segundo puesto en la calidad del TDP (Technical Description Paper), el artículo de descripción técnica será publicado en la prestigiosa revista JPAUT (Journal of Production and Automation) de Brasil. Véase Apéndice B y Apéndice C.
- Cuarto lugar en la competencia (categoría libre).

Además, se ha participado en la competencia Labview Student Design Competition, y como resultado, la National Instruments ha invitado al equipo principal de A2G al evento NIWeek 2019, que se desarrollará entre los días 20 al 23 de mayo de 2019, véase invitación en el Apéndice F. Dicho evento tendrá una duración de cuatro días, en ese tiempo serán desarrollados sesiones técnicas, conferencias, podrán ser observados proyectos desarrollados por ingenieros de la empresa o asociados a mismo, entre otros. El paper presentado para la competencia se adjunta en el Apéndice E.

A.2. Análisis de costos

En esta sección se expondrá el costo total del proyecto Aguara'i, así como las horas de trabajo que ha sido empleado para su elaboración completa. El costo de los equipos utilizados se muestran en la siguiente tabla:

Cuadro A.1: Costo de hardware.

Partes	Especificaciones	Cantidad	Precio en Gs.
Auto eléctrico*	Hobao Hyper VSE 1/8 Buggy	1	3.850.000
Controlador de vuelto**	Crius AIOP V2.1	1	292.000
Cargador*	iMAX B6	1	418.000
Batería LiPo*	SMC 4S 5400mAh	1	440.000
Bateria LiPo***	VEX 2S 3000mAh	1	200.000
Encoder*	FC-03	1	40.000
Controlador****	myRIO 1900	1	5.000.000
Otros*****	Impresión 3D, tornillos, cables, etc.	-	1.000.000
Total			11.260.000

Fuente: Team A2G (2018).

Es importante aclarar que el costo antecede fue financiado de la siguiente manera:

* Universidad Católica Nuestra Señora de la Asunción, sede regional Alto Paraná.

** Ing. Walter Benitez-Ing. Yessica Bogado

*** Lic. Ariel Guerrero.

**** National Instruments.

***** Universidad Católica, junto con alumnas de la carrera de Arquitectura de dicha institución.

El tiempo de desarrollo del proyecto Aguara'i se muestra en la siguiente tabla:

Cuadro A.2: Horas de trabajo para el proyecto Aguara‘i.

Actividad	Detalles	Horas
Planificación	Investigación, especificaciones de hardware y software, planificación del desarrollo del proyecto.	50
Identificación de sistema	Modelado del auto eléctrico a escala, para el diseño de los algoritmo de control.	40
Sistema de control	Diseño e implementación de los algoritmos de control (PID) para la velocidad y el control de dirección del vehículo autónomo a escala.	50
Fusión de datos	Diseño e implementación de los algoritmos de fusión de sensores para estimar la posición y orientación del vehículo autónomo a escala.	50
Desarrollo hardware	Modificaciones necesarias para dotar al vehículo eléctrico de los dispositivos necesarios para una conducción autónoma.	40
Algoritmo de navegación	Diseño e implementación del algoritmo de navegación (Pure Pursuit).	60
Comprobación del sistema	Se llevan a cabo las pruebas de la fusión de los algoritmos desarrollados de manera independiente en las etapas previas.	50
Cierre	Elaboración de la documentación correspondiente (TDP), organización del viaje y otras actividades relacionadas al la participación en la competencia.	20
Total		360

Fuente: Team A2G (2018).

A su vez, el tiempo de desarrollo de la fusión de sensores se muestra en la siguiente tabla:

Cuadro A.3: Horas de trabajo para la fusión de sensores.

Semana	Actividad	Hs/semana
1	Investigación	10
2	Investigación	10
3	Diseño del algoritmo y programación	15
4	Pruebas	15
Total		50

Fuente: Elaboración propia.

Apéndice B

TDP presentado a la competencia

Desarrollo de un vehículo autónomo a escala 1:8

Gregorio Ariel Guerrero Moral
Centro de Tecnologías de la
Información y Comunicación
Parque Tecnológico de Itaipu -
Paraguay
Hernandarias, Paraguay
ariel.guerrero@pti.org.py

Ariel David Bogado Arce
Centro de Investigación en Ciencias,
Tecnología e Innovación Avanzada
Universidad Católica “Nuestra Señora
de la Asunción”
Hernandarias, Paraguay
ariel.bogado@uc.edu.py

Micaela Carolina Jara Ten Katheren
Centro de Investigación en Ciencias,
Tecnología e Innovación Avanzada
Universidad Católica “Nuestra Señora
de la Asunción”
Hernandarias, Paraguay
micaela.jara@uc.edu.py

Jesús María Franco Santacruz
Centro de Investigación en Ciencias,
Tecnología e Innovación Avanzada
Universidad Católica “Nuestra Señora
de la Asunción”
Hernandarias, Paraguay
jesus.franco@uc.edu.py

Erid Eulogio Pacheco Viana
Centro de Investigación en Ciencias,
Tecnología e Innovación Avanzada
Universidad Católica “Nuestra Señora
de la Asunción”
Hernandarias, Paraguay
erid.pacheco@uc.edu.py

Resumen— El desarrollo de un vehículo autónomo es objeto de amplio estudio por muchos investigadores [1]. Uno de los enfoques de estudio consiste en el empleo de modelos físicos a escala cuya construcción es el objeto de este artículo. Para este propósito un auto eléctrico a escala 1:8 remotamente controlado es modificado. Se utiliza un myRIO para las tareas de adquisición de datos (odómetro, acelerómetro, magnetómetro, giroscopio, gps), y las de actuación. Otra tarea que realiza es fusionar los datos de los sensores de manera tal a que la información de posición y orientación pueda ser utilizada para la toma de decisión de navegación. Estos datos pueden ser remitidos a una notebook ejecutando una aplicación en LabVIEW. Esta estación de trabajo provee una interfaz hombre máquina al operador para visualizar los parámetros de navegación.

Palabras clave—vehículo autónomo, IMU, GPS, algoritmo de navegación, kalman

I. INTRODUCCIÓN

Los vehículos autónomos han atraído una gran cantidad de interés en investigación en los últimos años, así como importantes esfuerzos de desarrollo de la industria. En 2007, DARPA ha ejecutado Urban Grand Challenge [2], con las entradas de varias universidades documentadas en muchas publicaciones diferentes, por ejemplo, [3, 4].

Muchas empresas del sector automotriz poseen su propia división dedicadas a la investigación de vehículos autónomos y recientemente otras empresas que no son del sector tales como NVIDIA, APPLE, GOOGLE, YANDEX, BIADU han comenzado a desarrollar su propio vehículo autónomo. Proyectos que han sido ampliamente reportado en las noticias [5, 6]. Si bien se han realizado importantes esfuerzos en este campo, aún quedan muchos problemas por resolver, entre ellos, los problemas de detección, los diferentes tipos y niveles de control y la interacción de los vehículos autónomos con su entorno.

Un vehículo autónomo, es un vehículo capaz de imitar las capacidades humanas de manejo y control. El conductor

podrá elegir el destino, pero no se le requiere para activar ninguna operación mecánica del vehículo.

Los vehículos autónomos perciben el entorno mediante sensores tales como láser, radar, lidar, sistema de posicionamiento global y visión computarizada. Los sistemas avanzados de control interpretan la información para identificar la ruta apropiada, así como los obstáculos y la señalización relevante. Los vehículos autónomos generalmente son capaces de recorrer carreteras previamente programadas y requieren una reproducción cartográfica del terreno, con lo cual si una ruta no está recogida por el sistema se puede dar el caso que no pueda avanzar de forma coherente y normal.

II. ARQUITECTURA HARDWARE DEL SISTEMA

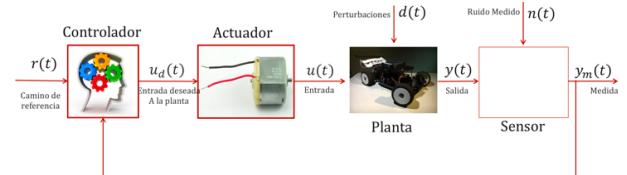


Figura 1 - Modelo matemático de un vehículo autónomo

En la literatura consultada [7] se ha verificado la necesidad de comprender cómo el modelo de la planta, así como sus restricciones, así como la compresión de los marcos referenciales que utilizados por un observador para medir la posición y otras medidas físicas de un sistema físico. Se ha realizado la identificación de la planta utilizando una operación *off-line* (acoplamiento indirecto), en el cual se almacenan los datos adquiridos y posteriormente se transfieren al ordenador para ser evaluados y procesados.

El sistema a ser implementado consta de:

- a. Vehículo Autónomo a escala 1:8: Chasis modificado de un auto eléctrico miniaturizado en el cual se adapta un myRIO 1900 así como los sensores y actuadores requeridos para navegación

- inercial (IMU+GPS). En esta plataforma se ejecutarán los algoritmos que permitan determinar la información de posición y orientación con base a los sensores. Así mismo permitirá la adquisición de datos de los sensores, ejecutar el algoritmo de navegación seleccionado y determinar las señales de control a los actuadores.
- b. Estación Base: Para configuración de los parámetros de navegación (waypoints) y visualización de estado de los sensores.

La plataforma hardware consta de los siguientes elementos:

A. Crius Crius AIOP v.2.1

Esta placa electrónica tipo MARG (Magnetic, Angular rate and Gravity) tiene incorporado varios sensores como un giroscopio/accelerómetro MPU6050 de 6 ejes, un altímetro de alta precisión MS5611-01BA01, y un magnetómetro HMC5883L de 3 ejes. El microcontrolador integrado es un ATMEGA 2560 de 8 bits, 16 MHz y se comunica con los dispositivos externos mediante los pines y puertos seriales.

B. Sensor: MPU6050 : Acelerómetro y giroscopio

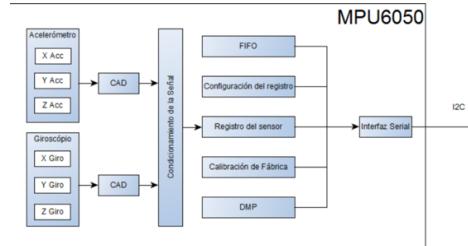


Figura 2 - Arquitectura del MPU 6050 [8]

En la Figura 2 se presenta la arquitectura del MPU6050 [9], el sensor posee conversores analógicos digitales para cada uno de los ejes y así obtener los valores en simultáneo con un rango de hasta $\pm 2000^\circ$ por segundo en el caso del giroscopio y de $\pm 16g$ para el acelerómetro. A continuación, los datos se filtran de acuerdo a la configuración preestablecida y la calibración de fábrica, para pasar luego al registro del sensor, los datos pueden ser accedidos por el DMP o por el usuario. El DMP actualiza los datos del FIFO leídos a una frecuencia determinada con el fin de evitar el desbordamiento. La interfaz serial de comunicación del MPU6050 es el protocolo de comunicación I2C.

C. Sensor: HMCL5883L: Magnetómetro integrado

El magnetómetro integrado en el controlador es el HMC5883L de Honeywell, el sensor triaxial tiene un campo de operación de -8 a +8 gauss lo que de esta manera se obtiene la dirección real del norte geográfico considerando la inclinación de 15 grados que hay en la zona. La interfaz serial del HMC5883L cumple el protocolo de comunicación I2C a 400 kHz.

D. Sensor de velocidad: Encoder FC03

Voltaje de Operación: 3.3V - 5V DC

Salidas: Analogica y Digital TTL

Sensor: MOCH22A

Modelo Placa: FC-03/FZ0888

Tipo de emisor: Fotodiodo IR

Tipo de detector: Fototransistor
 Longitud de onda del emisor: 950 nm (infrarrojo)
 Peso: 8 g
 Dimensiones: 32*14*7 mm
 Ranura de 5 mm
 Comparador Opamp: LMS393
 Led indicador de alimentación
 Led indicador de pulso
 Salida TTL ON: sensor bloqueado
 Salida TTL OFF: sensor desbloqueado

Conocer la posición o velocidad de un motor es muy importante en robótica, para lo cual existen diversas alternativas, siendo una de las más comunes el uso de encoders de tipo óptico. Los encoders incrementales ópticos realizan la medición de movimiento con el uso de un haz de luz infrarrojo que se ve interrumpido por las ranuras de un disco acoplado al eje. La cantidad de ranuras por vuelta determinará la precisión del encoder, en este caso de 4 pulsos por vuelta.

E. Actuador: Servo digital para dirección HB-5514 14kg



Figura 3 - Servo digital para dirección

Es un dispositivo actuador que tiene la capacidad de ubicarse en cualquier posición dentro de su rango de operación, y de mantenerse estable en dicha posición.

F. Actuador: Motor BLDC 2200KV (rmp/V)

Son motores síncronos alimentados por CC travéz de un inversor o fuente de alimentación de conmutación que produce una corriente eléctrica de CA para controlar cada fase del motor a través de un controlador de circuito cerrado. El controlador proporciona pulsos de corriente a los devanados del motor que controlan la velocidad y el par del motor.

G. Actuador: Control de velocidad ESC 100A

Un control de velocidad electrónico o ESC es un circuito electrónico que controla y regula la velocidad de un motor eléctrico. También puede proporcionar la inversión de giro del motor y el frenado dinámico. Los controles electrónicos de velocidad en miniatura se utilizan en modelos controlados por radio y con alimentación eléctrica. Los vehículos eléctricos de tamaño completo también tienen sistemas para controlar la velocidad de sus motores de accionamiento.

H. Planta: Chasis Haboo Hyper VS 1/8



Figura 4 - Chasis

Dimensión: 460 mm. x 306 mm. x 140 mm.

Distancia entre ejes: 322 mm.

Peso: 4720 g

Batería para myRIO: LI-PO 2S 3000 mAh 7.4 v

Batería para motor: LI-PO 4S 5400 mAh 14.8 v

Chasis de aluminio anodizado.

Torre de aluminio delantera 4mm y trasera 3mm

Soporte de suspensión reforzado

Soporte de baterías con velcro

Amortiguadores Big Bore de 17mm

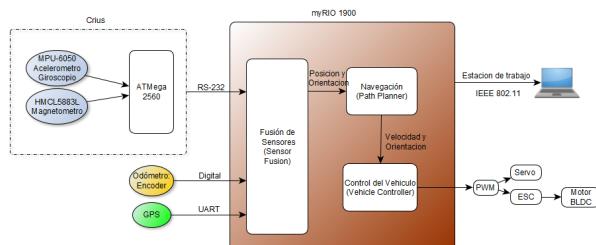


Figura 5 - Arquitectura del Hardware

III. ARQUITECTURA SOFTWARE

En esta sección se enfocará más en el tratamiento de los datos de los sensores para obtener los datos de posición y orientación, y la utilización de los mismos para el cálculo de la trayectoria de recorrido.

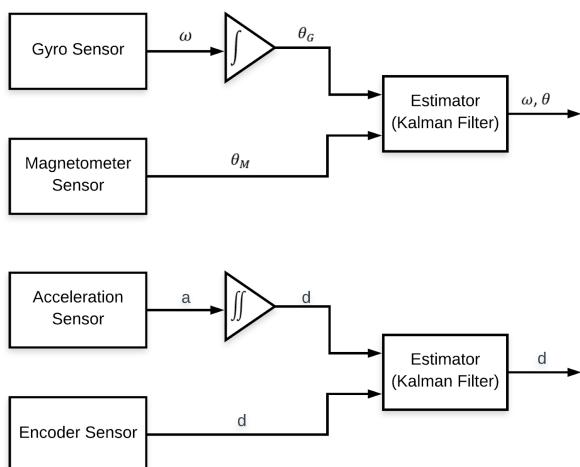


Figura 6 - Visión general del Sistema de fusión

A. Sensor Fusión

Se utilizaron señales de los sensores para mejorar y corregir la medición de la posición propia del vehículo

autónomo de 4 ruedas para obtener una estimación de posición más confiable. A partir de esto, calculamos la estimación de la posición y redujimos los errores sistemáticos y no sistemáticos durante las pruebas y tuvimos éxito en estimar la desviación del sesgo del giro. La herramienta básica aquí es un filtro de Kalman.

En la Figura 6, ω es el dato de la velocidad angular proveniente del giroscopio, a es la aceleración del acelerómetro y θ es el ángulo referente al norte magnético obtenido del magnetómetro.

De los estimadores, se obtienen la distancia recorrida y la orientación, para luego obtener las coordenadas (x,y). Se previo la corrección de la posición absoluta a partir de un GPS (Global Position Systems), pero no fue implementado.

B. Path Planner

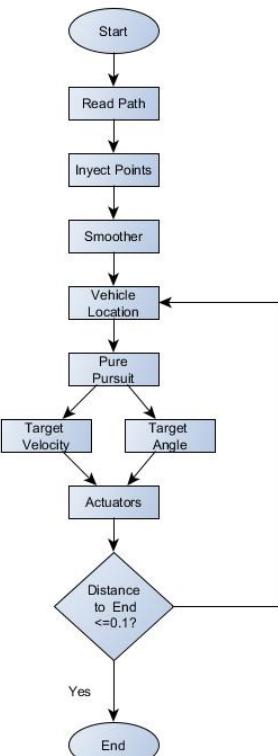


Figura 7 - Planificador de ruta

Este módulo se encarga de llevar al vehículo autónomo desde una posición inicial a una final siguiendo una trayectoria. El algoritmo utilizado es el denominado de persecución pura (pure pursuit algorithm [14]). A efectos prácticos se ha recurrido a la implementación del “Team 1712” [15] y modificado según el requerimiento.

Con este algoritmo se consigue determinar la velocidad objetivo del vehículo autónomo dependiendo de la curvatura del segmento de la trayectoria en el cual se encuentra el vehículo autónomo, como también, establecer la dirección a la cual debe dirigirse conociendo su posición actual y un punto objetivo llamado “Look Ahead Point”.

Para obtener mayor precisión en el cálculo de la velocidad y la curvatura, se inyectaron puntos a la trayectoria original obteniendo de esta manera puntos más

cercanos, para luego pasarlos por una etapa de suavizado y lograr la continuidad en el trayecto.

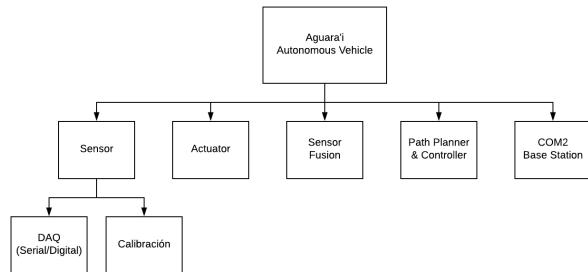


Figura 8 - Arquitectura del Sistema

IV. PRUEBAS REALIZADAS

A. FUSIÓN DE SENSORES

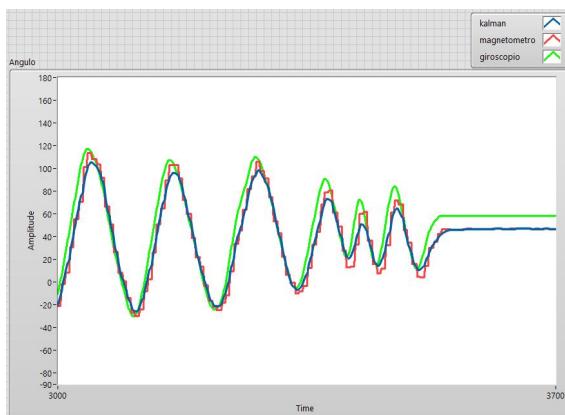


Figura 9 - Filtro de Kalman para la orientación

En la Figura 9, se muestra el resultado de la fusión del giroscopio y el magnetómetro. En ella, se puede observar que el magnetómetro no posee buena respuesta a frecuencias altas, en cambio, a baja frecuencia, la respuesta es buena. Por otro lado, el comportamiento de la curva del giroscopio es suave, pero el error acumulativo aumenta y no hay manera de corregirlo sin realizar la fusión de los sensores.

B. PATH PLANNER

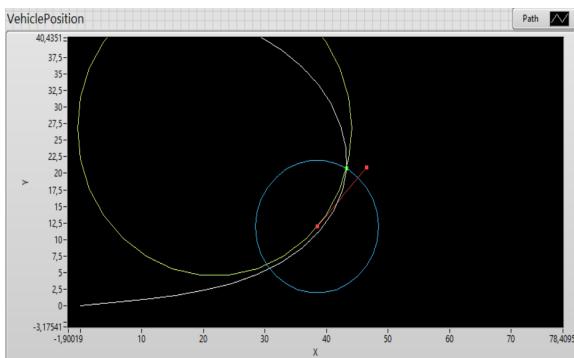


Figura 10 - Simulación del algoritmo persecución pura

Durante las pruebas se evidenció que el algoritmo de persecución pura arroja buenos resultados en la determinación de seguimiento de la trayectoria

establecida. Debido a su robustez, esta permite que se tengan algunos errores en la adquisición de los datos de los sensores o en la sintonización del control de velocidad y dirección, sin embargo como tiene en cuenta una realimentación de un estado anterior, permite actualizar los actuadores con valores coherentes por lo cual se van atenuando los pequeños errores de orientación y posición.

CONCLUSIONES

Se ha modificado el chasis de un vehículo eléctrico a escala 1:8 e implementado un algoritmo de navegación basado en la estrategia denominada de persecución pura. La plataforma utilizada para estos trabajos a sido el myRIO 1900 con el entorno de desarrollo de LabVIEW. El trabajo se ha implementado en dos meses y medio, utilizando ejemplos de aplicaciones de National Instruments en conjunto con librerías del “Team 1712”, lo cual nos ha permitido realizar las primeras pruebas operativas del prototipo, con un error aceptable.

El proyecto recalca el valor pedagógico de la enseñanza basada en retos, permitiendo a los alumnos adquirir la experiencia de trabajo en equipo, en un entorno multidisciplinario así como la experiencia de reutilización de código de terceros. También ha permitido que los alumnos comparen los conceptos desarrollados en distintas materias de la universidad al calibrar los distintos sensores, a mejorar la lectura minimizando los errores mediante la implementación del filtro de kalman.

La plataforma de trabajo seleccionada también permitió que alumnos de diferentes años puedan acceder al mismo conocimiento, desde el punto de vista de implementación de un algoritmo, al permitirles obtener la certificación CLAD, con lo cual no solo han aprendido el lenguaje de programación, también se han enfocado en la metodología de trabajo en un proyecto específico.

RECONOCIMIENTOS

Se agradece al apoyo de las siguientes instituciones: Universidad Católica “Nuestra Señora de la Asunción”, Centro de Investigación en Ciencias, Tecnología e Innovación Avanzada, Fundación Parque Tecnológico Itaipu – Paraguay.

REFERENCIAS

- [1] Iyengar, D., & Peters, D. L. (2015, October). Development of a miniaturized autonomous vehicle: Modification of a 1: 18 scale rc car for autonomous operation. In ASME 2015 Dynamic Systems and Control Conference (pp. V003T50A008-V003T50A008). American Society of Mechanical Engineers.
- [2] McBride, J. (2007). Darpa urban challenge.
- [3] Urmsom, C., Bagnell, J. A., Baker, C. R., Hebert, M., Kelly, A., Rajkumar, R., & Team, D. U. C. (2007). Tartan racing: A multi-modal approach to the DARPA
- [4] Montemerlo, M., Becker, J., Bhat, S., Dahlkamp, H., Dolgov, D., Etinger, S., & Thrun, S. (2008). Junior: The Stanford entry in the Urban Challenge. Journal of field Robotics, 25(9), 569-597
- [5] The Economist. (2013). Look, no hands. Extraído de <http://www.economist.com/news/special-report/21576224-one-day-every-car-may-come-invisible-chauffeur-look-no-hands>

- [6] Dockterman, E. (2015). Google's self-driving car may come with airbags on the outside. Time Magazine. Extraído de <http://time.com/3758446/googles-self-driving-car-may-come-with-airbags-on-the-outside/>
- [7] The DuckieTown project. (2017). Extraido de <https://www.duckietown.org/>
- [8] Benítez, W., & Bogado, Y. (2015). Desarrollo de un prototipo de VANT (Vehículo Aéreo No Tripulado) para inspección visual de líneas eléctricas aéreas (Tesis de Grado). Universidad Católica "Nuestra Señora de la Asunción" Campus Alto Paraná. Paraguay.
- [9] MPU-6000 and MPU-6050 Product Specification Revision 3.4. Sunnyvale, California, Estados Unidos. Extraído de https://store.invensense.com/datasheets/invensense/MPU-6050_Datasheet_V3%204.pdf
- [10] EureCar, KAIST Self-Driving car. Extraido de <https://forums.ni.com/t5/Projects-Products/EureCar-KAIST-Self-Driving-car/ta-p/3517884>
- [11] Kok, M., Hol, J. D., & Schön, T. B. (2017). Using inertial sensors for position and orientation estimation. *arXiv preprint arXiv:1704.06053*.
- [12] Ozyagciar, T. (2012). Implementing a tilt-compensated eCompass using accelerometer and magnetometer sensors. *Freescale semiconductor, AN, 4248*.
- [13] Zunaidi, I., Kato, N., Nomura, Y., & Matsui, H. (2006). Positioning system for 4-wheel mobile robot: encoder, gyro and accelerometer data fusion with error model method. *CMU. Journal*, 5(1).
- [14] Conlter, R. C. (1992). Implementation of the Pure Pursuit Path Tracking Algorithm.
- [15] Implementation of adaptative pure persuit controller. Extraído de <https://www.chiefdelphi.com/media/papers/3488>.

Apéndice C

Anuncio de vencedores del mejor TDP de RoboCar Race 2018



erid pacheco <eridpacheco@gmail.com>

Anuncio dos vencedores de melhor TDP do RoboCar Race 2018

1 mensaje

RoboCar Race <robocar.race@gmail.com>
Cco: eridpacheco@gmail.com

6 de diciembre de 2018, 22:09

Boa noite a Todos.

Desculpem-nos pela longa demora no retorno a vocês após o evento. Estamos entrando em contato para anunciar os vencedores do melhor TDP.

1a - Equipe Semear - USP São Carlos

2a - Equipe A2G - Universidad Católica Nuestra Señora de la Asunción

3a - Equipe Car-not-found - UFABC

Os vencedores terão o TDP publicado na revista JPAUT - <https://jpaut.com.br> na 3a Edição do 1a trimestre de 2019.

Solicitamos que os vencedores entrem em contatos conosco para receber maiores informações sobre o formato, correções e extensão do paper.

Agradecemos a participação de todos e contamos com a vossa presença no próximo evento de 2019. Manteremos vocês informados sobre as novidades.

Abraços

--

Prof. Dr. Edson Kitani (FATEC Santo André)

Prof. Dr. Luiz Celiberto Jr. (UFABC)

Coordenadores

Apéndice D

TDP en proceso de evaluación para ser publicado en la revista JPAUT

Development of an autonomous vehicle at a 1:8 scale

Ariel Guerrero¹, Micaela Jara², Erid Pacheco², Ariel Bogado², Jesús Franco²

¹Parque Tecnológico de Itaipu – Paraguay, ²Universidad Católica “Nuestra Señora de la Asunción”

ABSTRACT

The development of an autonomous vehicle is the subject of extensive study by many researchers [1]. One of the approaches of study consists of the use of physical models at scale whose construction is the object of this article. For this purpose, a RC (Remotely Controlled) electric car with a 1: 8 scale is modified. A myRIO is used for data acquisition (odometer, accelerometer, magnetometer, gyroscope) and operation tasks. Another task is to merge the data of the sensors in such a way that the position and orientation information can be used for the navigation decision making. This data can be sent to a workstation, running an application in LabVIEW. This workstation provides a human-machine interface for the operator to display the navigation parameters.

Keywords: Autonomous Vehicle, IMU, Navigation Algorithm, Kalman.

INTRODUCTION

Autonomous vehicles have attracted a great deal of interest in research in recent years, as well as important industry development efforts. In 2007, DARPA has executed the Urban Grand Challenge [2], with entries from several universities documented in many different publications, for example, [3, 4].

Many companies in the automotive sector have their own division dedicated to the investigation of autonomous vehicles and recently other companies that are not of the sector such as NVIDIA, APPLE, GOOGLE, YANDEX, BIADU have started to develop their own autonomous vehicle. Projects that have been widely reported in the news [5, 6]. Although important efforts have been made in this field, there are still many problems to be solved, among them, the problems of detection, the different types and levels of control and the interaction of autonomous vehicles with their environment.

An autonomous vehicle is a vehicle capable of imitating the human capacities of management and control. The driver may choose the destination but is not required to activate any mechanical operation of the vehicle. Autonomous vehicles perceive the environment through sensors such as laser, radar, lidar, global positioning system and computer vision. Advanced control systems interpret information to identify the appropriate route, as well as obstacles and relevant signage. Autonomous vehicles are generally capable to travel previously programmed roads and require a cartographic reproduction of the terrain, so if a route is not picked up by the system it is possible that it cannot advance coherently and normally.

METHODOLOGY

The methodology used in this project was the traditional design. In Figure 1, the flow diagram of the tasks performed is presented. The main problem was how to develop, in a short period of time, an autonomous scaled electric car. Based on this approach, possible solutions were analyzed, carrying out the necessary studies and evaluations. Once the objectives to be met were set, the hardware was prepared, in this case the adaptation of the components in the scaled electric car, and the design of the software. The software and hardware implementations were carried out. Subsequently, the necessary tests were carried out in order to verify problems, and if necessary, implement modifications and improvements to the system, whether they were in the software or hardware.

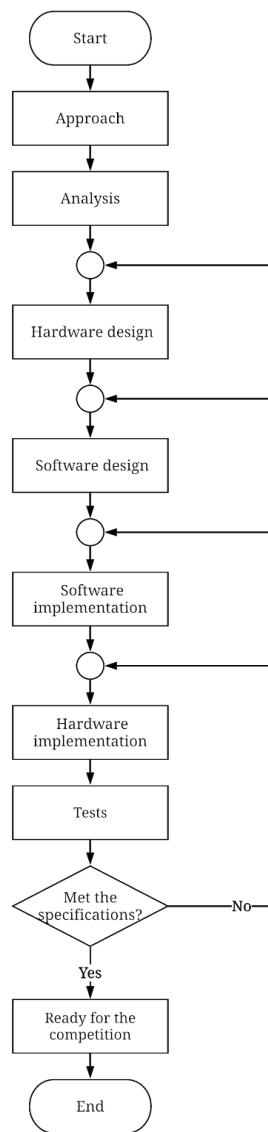


Figure 1 - Project methodology flowchart

DEVELOPMENT

I. SYSTEM HARDWARE ARCHITECTURE

Following the consulted literature [7] we needed to have a mathematical model that describes the behavior of the plant. As it would take more time to reach an accurate model taking into account the short time available to complete the project and the fact that we needed to have the control of both the traction and the direction of the plant, we proceeded to obtain the mathematical model through a transfer function using the "black box" method, which consists in the study of an element from its output behavior for a given entry without covering its internal functioning. The diagram can be seen in figure 1. Knowing the input and output, the transfer function of the "black box" was identified using the Matlab System Identification tool.

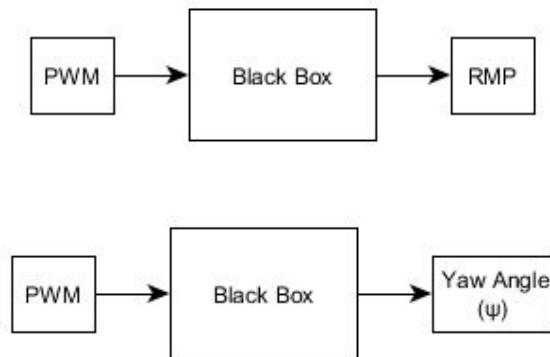


Figure 2 -Powertrain and direction schematic

In the "black box" of the powertrain, there is a brushless motor (BLDC) powered by an electronic speed controller (ESC), which receives a PWM signal to control the speed of the motor. On the other hand, in the "black box" of the steering, there is a servo motor, which receives another PWM signal to control the Yaw Angle of the front wheels, and consequently control the direction of the car.

The result obtained through this method had an approximation of 70% for the powertrain and 87% for the direction, which was enough to be able to control the plant.

The system to be implemented consists of:

- Autonomous Vehicle at 1: 8 scale: Modified chassis of a miniaturized electric car in which a myRIO 1900 is adapted as well as the sensors and actuators required for inertial navigation (IMU + GPS). In this platform the algorithms that allow determining the position and orientation information based on the sensors will be executed. It will also allow the acquisition of data from the sensors, execute the selected navigation algorithm and determine the control signals to the actuators.
- Base Station: For configuration of the navigation parameters (waypoints) and visualization of the status of the sensors.

The hardware platform consists of the following elements:

A. Crius AIOP v2.1

This electronic board type MARG (Magnetic, Angular rate and Gravity) has several built-in sensors such as a gyroscope /accelerometer MPU6050 6-axis, a high-precision altimeter MS5611-01BA01, and a magnetometer HMC5883L 3-axis. The integrated microcontroller is an ATMEGA 2560 8-bit, 16 MHz and communicates with external devices through pins and serial ports.

1) Sensor: MPU6050 : Accelerometer and gyroscope

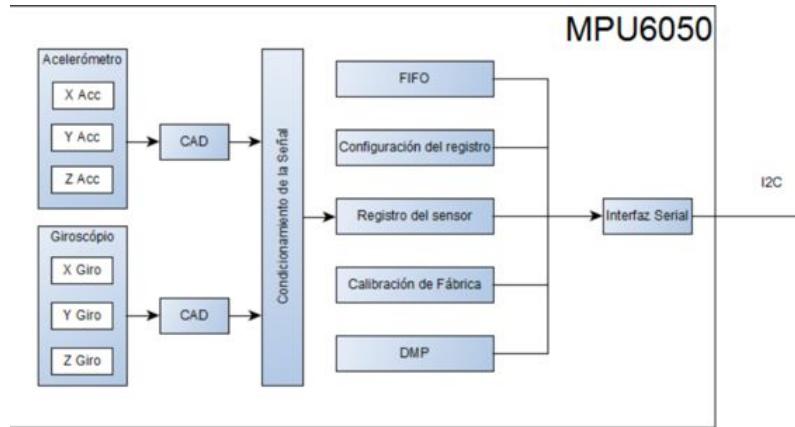


Figure 3 - Architecture of the MPU 6050 [8]

Figure 3 shows the architecture of the MPU6050 [9], the sensor has digital analog converters for each of the axes and thus obtains the values simultaneously with a range of up to ± 2000 per second in the case of the gyroscope and $\pm 16g$ for the accelerometer. Then, the data is filtered according to the preset configuration and the factory calibration, then it goes to the sensor register, the data can be accessed by the DMP or by the user. The DMP updates the FIFO data read at a certain frequency in order to avoid overflow. The communication serial interface of the MPU6050 is the I2C communication protocol.

2) Sensor: HMCL5883L: Integrated magnetometer

The magnetometer integrated in the controller is the Honeywell HMC5883L, the triaxial sensor has an operation field of -8 to +8 gauss, which in this way gives the real direction of the geographic north considering the 15 degree inclination that exists in the zone. The serial interface of the HMC5883L complies with the I2C communication protocol at 400 kHz.

B. Odometer and Speed Sensor Encoder FC03

Operating voltage: 3.3V - 5V DC

Outputs: Analogica y Digital TTL

Sensor: MOCH22A

Board model: FC-03/FZ0888

Type of emitter: Photodiode IR

Detector type: Phototransistor

Wavelength of the emitter: 950 nm (infrared)

Weight: 8 g

Dimensions: 32*14*7 mm

Slot: 5 mm

Opamp comparator: LMS393

Power indicator LED

Pulse indicator LED

Output TTL ON: blocked sensor

Output TTL OFF: unlocked sensor

Knowing the position or speed of an engine is very important in robotics, for which there are several alternatives, one of the most common being the use of optical type encoders. The incremental optical encoders perform the measurement of movement with the use of an infrared beam that is interrupted by the slots of a disk coupled to the shaft. The number of slots per revolution will determine the encoder's accuracy, in this case 4 pulses per revolution.

C. Actuator: Digital Servo HB-5514 14kg



Figure 4 - Digital servo for direction

It is an actuator device that has the ability to be located in any position within its operating range, and to remain stable in that position.

D. Actuator: Motor BLDC 2200KV (rmp/V)

They are synchronous motors fed by DC through an inverter or switching power supply that produces an AC electric current to control each phase of the motor through a closed circuit controller. The controller provides pulses of current to the motor windings that control the speed and torque of the motor.

E. Actuator: Electronic Speed Controller ESC WP-8BL100, 100A

An electronic speed controller or ESC is an electronic circuit that controls and regulates the speed of an electric motor. It can also provide reversing of the motor and dynamic braking. ESCs are often used on motors essentially providing an electronically-generated three-phase electric power low voltage source of energy for the motor.

F. Mechanical Plant: Chassis Haboo Hyper VS 1/8



Figure 5 - Chassis

Dimension: 460 mm. x 306 mm. x 140 mm.

Distance between axis: 322 mm.

Weight: 4720 g

Battery for myRIO: LI-PO 2S 3000 mAh 7.4 v

Motor battery: LI-PO 4S 5400 mAh 14.8 v

Anodized aluminum chassis.

Front aluminum turret 4mm and rear 3mm

Reinforced suspension support

Battery holder with velcro

Big Bore 17mm shock absorbers

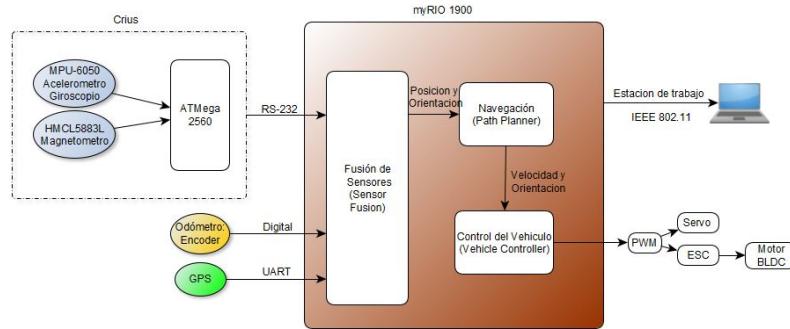


Figure 6 - Hardware Architecture

II. SOFTWARE ARCHITECTURE

In this section, we will focus more on the processing of sensor data to obtain the position and orientation of the vehicle, and the use of them for the calculation of the path of travel. Before the fusion of sensors, the collected data go through a calibration process, where offset and gain errors are eliminated, this process can be found in [8], then the data goes through a process of changing the reference, passing from a fixed frame of reference to the autonomous vehicle, to which the inertial sensors belong, to a frame of reference fixed to the ground, known as the navigation reference frame.

A. Sensor Fusion

Sensor signals were used to improve and correct the position measurement of the 4-wheel autonomous vehicle to obtain a more reliable position estimate. From this, we calculated the estimation of the position and reduced the systematic and non-systematic errors during the tests and we succeeded in estimating the deviation of the turn bias. The basic tool here is a Kalman filter.

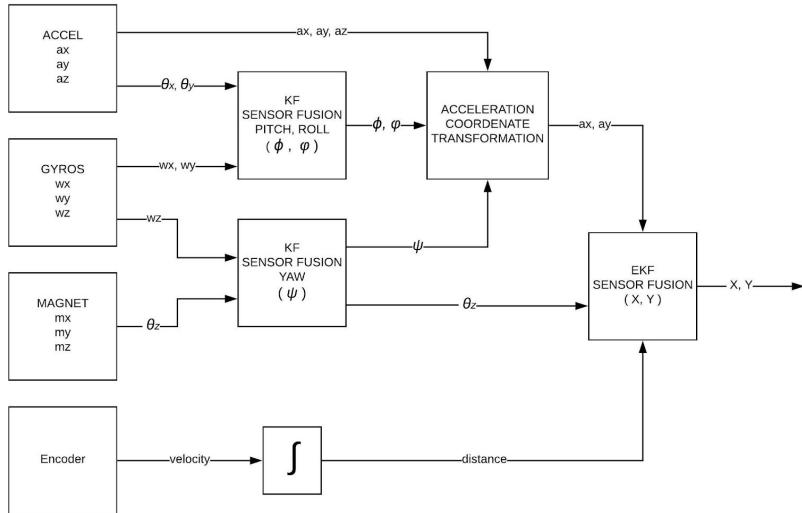


Figure 7 - Overview of the fusion system

As shown in figure 7; initially, a linear Kalman filter (KF) is used to merge the data of the accelerometer and the gyroscope, with this we obtain the pitch (φ) and roll (θ) angles, the yaw angle (ψ) is obtained from the fusion of the gyroscope with the magnetometer, also through a linear Kalman filter (KF) [14].

In Figure 8, the result of the fusion of the gyroscope and the magnetometer is shown. In the figure, it can be seen that the magnetometer does not have good response at high frequencies, however, at low frequency, the response is good. On the other hand, the behavior of the gyroscope curve is smooth, but the cumulative error increases and there is no way to correct it without performing the fusion of the sensors.

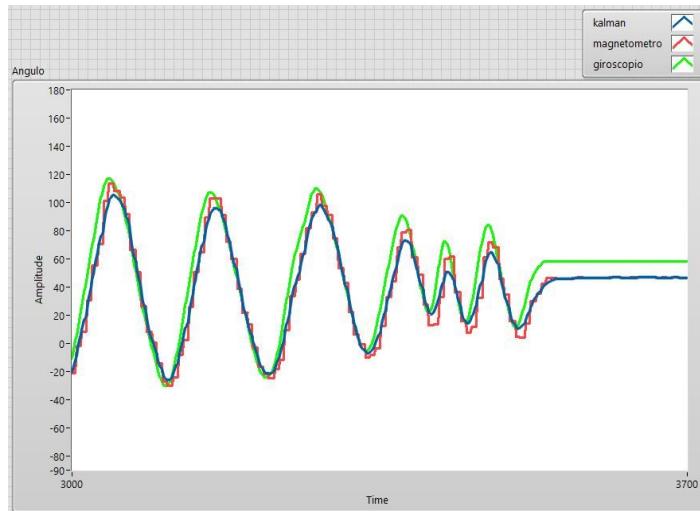


Figure 8 - Kalman filter for orientation

Finally, the encoder is added to the system, in order to have a better estimate of the distance traveled, an extended Kalman filter (EKF) is used, taking into account the non-linearity of the estimation by means of the odometry model [15].

This model is subject to cumulative errors that increase with time, because there is no external reference. These errors can be minimized by integrating a GPS (Global Position Systems) into the system, but this is beyond the scope of this work [16].

B. Path Planner

This module is responsible for taking the autonomous vehicle from an initial position to a final, following a trajectory. The algorithm used is the so-called pure pursuit algorithm [17]. For practical purposes, the implementation of "Team 1712" [18] has been used and modified according to the requirement.

With this algorithm it is possible to determine the target speed of the autonomous vehicle depending on the curvature of the segment of the trajectory in which the autonomous vehicle is located, as well as to establish the direction to which it should go knowing its current position and a target point called "Look Ahead Point" (Figure 10).

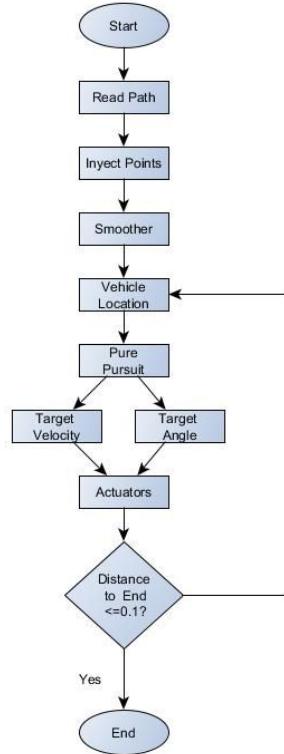


Figure 9 - Route planner

The "Look Ahead Point" is a fundamental parameter in the application of this algorithm, since by varying its value it is possible to vary: the response of the car to deviations from the wanted trajectory, and the stability in which the car follows its trajectory preventing oscillations . Its value can be static or dynamic, that is, static if its value is predetermined by the programmer before the car starts its trajectory, or dynamic when its value depending on characteristics such as the speed of the car and the curvature of the trajectory make its value to get the best response. In this project the static was applied due to its simplicity and rapid implementation.

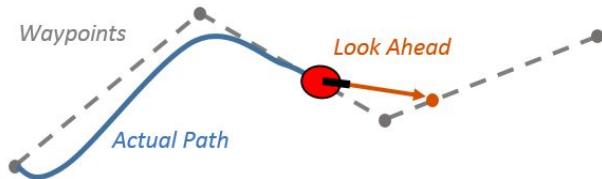


Figure 10 - Look Ahead Distance [19]

With regard to its value, the choice of a small number will cause the vehicle to quickly seek to approach the desired trajectory, however, as a consequence, the car begins to oscillate in search of the trajectory as shown in Figure 11.

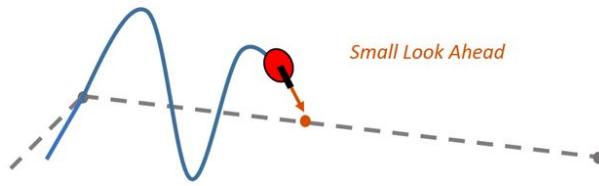


Figure 11 - Small Look Ahead [19]

On the other hand, a choice of a large value will cause the car to stop oscillating, however the response to sudden variations in the trajectory becomes very slow, as does the curvature in which the car follows its trajectory (Figure 12).

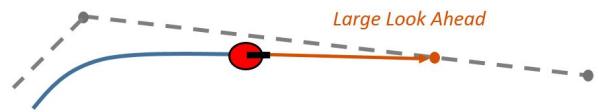


Figure 12 - Large Look Ahead [19]

To obtain greater precision in the calculation of the speed and curvature, points were injected into the original trajectory obtaining in this way closer points, then, to pass them through a smoothing stage and achieve continuity in the path.

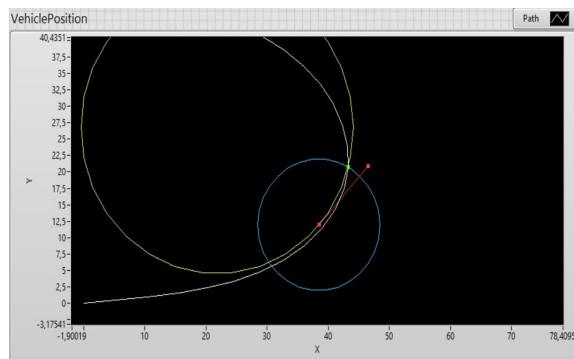


Figure 13 - Simulation of Pure Pursuit Algorithm

FINDINGS

During the tests it was evidenced that the pure pursuit algorithm gives good results in the determination of follow-up of the established trajectory. Due to its robustness, this allows some errors in the acquisition of sensor data or in the tuning of the speed and direction control, however, as it takes into account feedback from a previous state, it allows to update the actuators with consistent values so that small errors of orientation and position are attenuated.

The choice of the best value for the Look Ahead Point was made through experimentation finding that for the speed 2 m/s, a value of Look Ahead Point equal to 1.8m, prevents oscillations and a good response is obtained for more curves closed.

CONCLUSIONS

Autonomous navigation was validated in the tests, demonstrating its effectiveness in tracking a defined trajectory. A set of tests was performed varying the initial position and the orientation to verify that, despite the different initial conditions, in the same way, the car manages to follow a predefined trajectory, as expected by the results of the simulations. Tests in a controlled environment have obtained satisfactory results for the desired purposes, taking into account that the scaled vehicle was used in a race of autonomous scaled cars. For this purpose, the cumulative errors inherent in the system were reduced so that the influence on the result was negligible. In addition, the inertial navigation algorithm turned out to be very effective compared to other types of navigation used for this purpose, achieving a higher response speed due to its low computational requirement.

However, in uncontrolled environments, it has not been very effective, due to the inability of the linear Kalman filter to eliminate electromagnetic distortions that affect the readings of the magnetometer, these readings with distortion considerably affect the calculation of the yaw angle, and consequently, they produce errors in the calculation of the x and y coordinates.

Bearing in mind that the duration of the race does not generate significant cumulative errors, the use of a magnetometer could be eliminated, and thus make the system less sensitive to disturbances in the magnetic field (with the cost that this entails in the absence of an absolute orientation, which is what the magnetometer offered).

ACKNOWLEDGEMENTS

We thank the support of the following institutions: Universidad Católica "Nuestra Señora de la Asunción", Centro de Investigación en Ciencias, Tecnología e Innovación Avanzada (CICTIA), Fundación Parque Tecnológico Itaipu – Paraguay and National Instruments Brazil.

REFERENCES

- [1] IYENGAR, D., & PETERS, D. L. (2015, October). Development of a miniaturized autonomous vehicle: Modification of a 1: 18 scale rc car for autonomous operation. In ASME 2015 Dynamic Systems and Control Conference (pp. V003T50A008-V003T50A008). American Society of Mechanical Engineers.
- [2] MCBRIDE, J. (2007). Darpa urban challenge.
- [3] URMSON, C., BAGNELL, J. A., BAKER, C. R., HEBERT, M., KELLY, A., RAJKUMAR, R., & TEAM, D. U. C. (2007). Tartan racing: A multi-modal approach to the DARPA
- [4] MONTEMERLO, M., BECKER, J., BHAT, S., DAHLKAMP, H., DOLGOV, D., ETTINGER, S., & THRUN, S. (2008). Junior: The Stanford entry in the Urban Challenge. Journal of field Robotics, 25(9), 569-597
- [5] THE ECONOMIST. (2013). Look, no hands. Recovered from The economist: <http://www.economist.com/news/special-report/21576224-one-day-every-car-may-come-invisible-chauffeur-look-no-hands>
- [6] DOCKTERMAN, E. (2015). Google's self-driving car may come with airbags on the outside. Time Magazine. Recovered from Time: <http://time.com/3758446/googles-self-driving-car-may-come-with-airbags-on-the-outside/>
- [7] THE DUCKIETOWN FOUNDATION. (2017). The Duckietown Project. Recovered from Duckietown: <https://www.duckietown.org/>

- [8] BENÍTEZ, W., & BOGADO, Y. (2015). Desarrollo de un prototipo de VANT (Vehículo Aéreo No Tripulado) para inspección visual de líneas eléctricas aéreas (Tesis de Grado). Universidad Católica "Nuestra Señora de la Asunción" Campus Alto Paraná. Paraguay.
- [9] INVENSENSE. (2013). MPU-6000 and MPU-6050 Product Specification Revision 3.4. Sunnyvale, California, United States of America. Recovered from Invensense: https://store.invensense.com/datasheets/invensense/MPU-6050_DataSheet_V3%204.pdf
- [10] LEE, U., OH, J., SHIN, S., SHIM, I., CHOI, J., JUNG, Y., PARK, K., KIM, M., & JUNG, J. (2014). EureCar, KAIST Self-Driving car. Recovered from National Instruments: <https://forums.ni.com/t5/Projects-Products/EureCar-KAIST-Self-Driving-car/ta-p/3517884>
- [11] KOK, M., HOL, J. D., & SCHÖN, T. B. (2017). Using inertial sensors for position and orientation estimation. *arXiv preprint arXiv:1704.06053*.
- [12] OZYAGCILAR, T. (2012). Implementing a tilt-compensated eCompass using accelerometer and magnetometer sensors. *Freescale semiconductor, AN, 4248*.
- [13] ZUNAIDI, I., KATO, N., NOMURA, Y., & MATSUI, H. (2006). Positioning system for 4-wheel mobile robot: encoder, gyro and accelerometer data fusion with error model method. *CMU. Journal*, 5(1).
- [14] VIGOUROUX CAVOLINA, D. P. (2010). Implementación de unidad de mediciones iniciales (IMU) para robótica utilizando filtro de Kalman. Sartenejas, Venezuela.
- [15] FISCHER, T., NITSCHE, M. A., & PEDRE, S. (2014). Fusión de encoders de cuadratura, sensores iniciales y magnéticos para la localización de robots móviles. Facultad de Ciencias Exactas y Naturales - UBA, Buenos Aires.
- [16] MOHINDER S., G., & ANGUS P., A. (2008). Kalman Filtering: Theory and Practice Using MATLAB (Third ed.). Hoboken, New Jersey: JOHN WILEY & SONS, INC.
- [17] Coulter, R. C. (1992). Implementation of the pure pursuit path tracking algorithm (No. CMU-RI-TR-92-01). Carnegie-Mellon UNIV Pittsburgh PA Robotics INST.
- [18] FRC TEAM 1712 (2018). Implementation of adaptive pure pursuit controller. Recovered from chiefdelphi: <https://www.chiefdelphi.com/media/papers/3488>.
- [19] MATHWORKS (w/d). Pure Pursuit Controller. Recovered from MathWorks: <https://www.mathworks.com/help/robotics/ug/pure-pursuit-controller.html>

CONTACT INFORMATION

Gregorio Ariel Guerrero Moral (corresponding author)
ariel.guerrero@pti.org.py

Micaela Carolina Jara Ten Kathen
micaela.jara@uc.edu.py

Erid Eulogio Pacheco Viana
erid.pacheco@uc.edu.py

Ariel David Bogado Arce
ariel.bogado@uc.edu.py

Jesús María Franco Santacruz
jesus.franco@uc.edu.py

Apéndice E

Artículo presentado para el Labview Student Design Competition

Contact Information:

Country: Paraguay

Year Submitted: 2019

University: Universidad Católica "Nuestra Señora de la Asunción" Campus Alto Paraná

List of Team Members (with year of graduation):

- Micaela Carolina Jara Ten Kathon (2019)
- Ariel David Bogado Arce (2019)
- Erid Eulogio Pacheco Viana (2019)
- Jesús María Franco Santacruz (2020)

Faculty Advisers: Prof. Gregorio Ariel Guerrero Moral

Main Contact Email Address: ariel.guerrero@uc.edu.py

Project Information:

Title: Development of an autonomous vehicle at a 1:8 scale

Description:

The challenge of the Robocar Race 2018 competition held in San Paulo-Brazil is to design an autonomous scale vehicle that can compete in 2023 against a radio-controlled car piloted by a man. Our team of students of the Universidad Católica Nuestra Señora de la Asunción" accepted the challenge and presented a solution based on inertial navigation using the NI myRIO and LabVIEW, being the first students of Paraguay to accept the challenge of building an autonomous car.

Products:

NI LabVIEW 2017, NI LabVIEW myRIO 2017, NI LabVIEW Real-Time Module, NI LabVIEW FPGA Module and myRIO-1900.

The Challenge:

Brazilian branch of National Instruments challenged LabVIEW Student Ambassadors (LSAs) to design and build an autonomous scale vehicle to compete at the Robocar Race, in return, they would give them a NI myRIO controller. Our ambassador for PTI-UCA accepted the challenge and assembled the first team from Paraguay to develop an

autonomous scaled vehicle. Our goal was to win the competition that consisted in running the track in the shortest time. We entered the race with our car called Aguara'i (little fox in Guarani), a Radio Controlled RC car that we modified to be autonomous. Although we finished in 4th place, our work inspired a lot of engineering students from our country to take part of this exciting and promising field.



Members of the team with Robocar Race 2018 juices.

The Solution:



Project Overview - The resulting performance of Aguara'i

System Configuration

Autonomous Vehicle at 1: 8 scale: Modified chassis of a miniaturized electric RC car in which a myRIO 1900 is adapted as well as the sensors and actuators required for inertial navigation (IMU). In this platform the algorithms that allow determining the position and orientation information based on the sensors will be executed. It will also allow the acquisition of data from the sensors, execute the selected navigation algorithm and determine the control signals to the actuators.

Base Station: For configuration of the navigation parameters (waypoints) and visualization of the status of the sensors. Human-machine interface with LabVIEW.



"Aguara'i" Autonomous Car

Using LabVIEW and myRIO

The first step is to acquire the signals of the sensors. The encoder was connected to one of the FPGA's (Field-Programmable Gate Array) inputs, and the IMU sensors were connected through a serial connection from an arduino microcontroller (ATMega2560) to the myRIO CPU.

In order to control the speed and direction of the car, we made use of the parallelism of the loops in the FPGA, these PWMs (Pulse Width Modulation) values to control the direction and the speed came from constantly updated values in the myRIO. To communicate this data from the RT-target to FPGA-target we use global variables, since we needed that several processes running in parallel in labview, can communicate their data and in this way obtain the processing and control of the state variables of the car in real time.

Next, the configuration of the system can be seen in Figure 1.

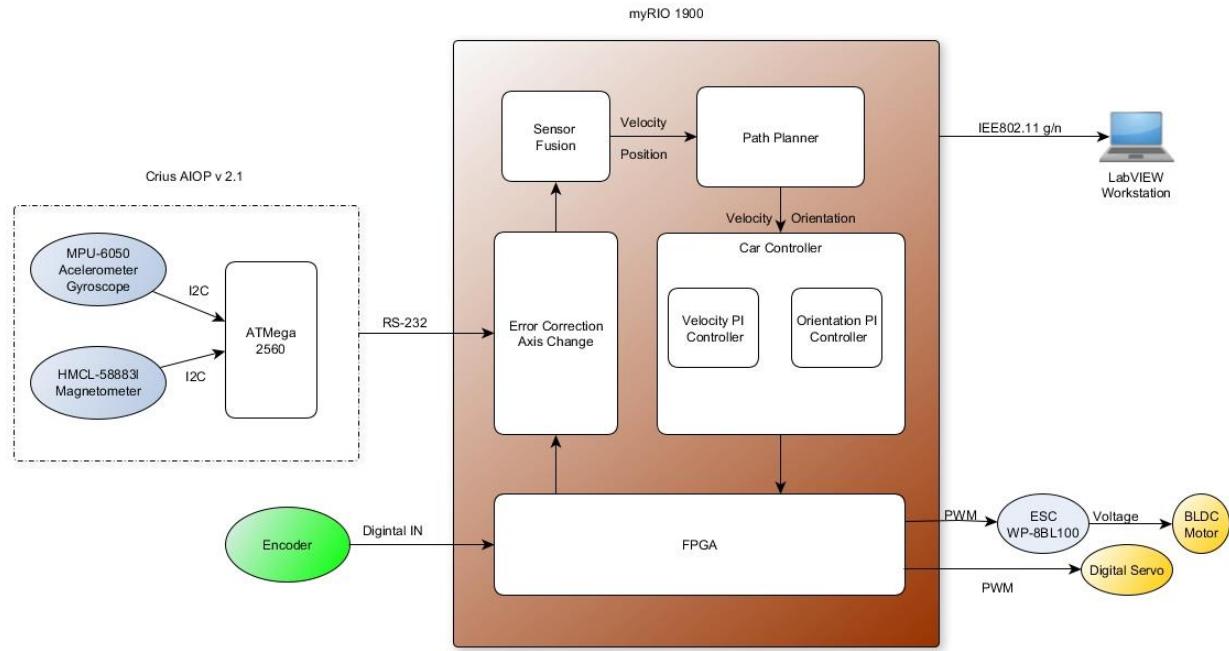


Figure 1 - Aguara'i Architecture

Sensor Fusion

Sensor signals were used to improve and correct the position measurement of the 4-wheel autonomous vehicle to obtain a more reliable position estimate. From this, we calculated the estimation of the position and reduced the systematic and non-systematic errors during the tests and we succeeded in estimating the deviation of the turn bias. The basic tool here is a Kalman filter.

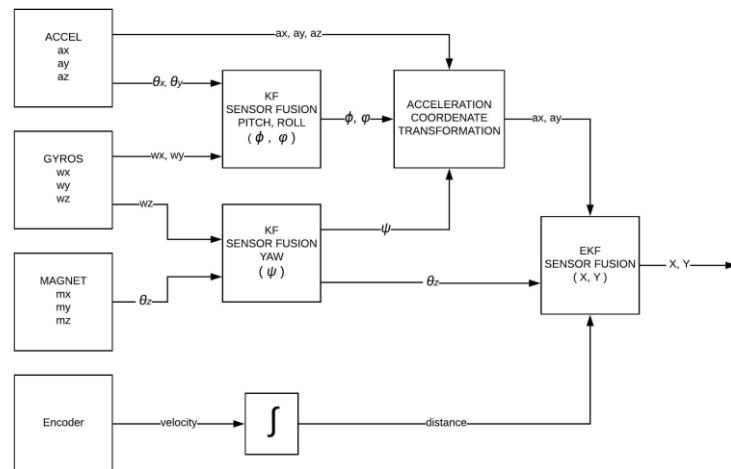


Figure 2 - Overview of the Fusion System

As shown in Figure 2; Initially, a linear Kalman filter (KF) is used to merge the data of the accelerometer and the gyroscope, with this we obtain the pitch (ϕ) and roll (ψ) angles, the yaw angle (ψ) is obtained from the fusion of the gyroscope with the magnetometer, also through a linear Kalman filter (KF).

In Figure 3, the result of the fusion of the gyroscope and the magnetometer is shown. In it, it can be seen that the magnetometer does not have good response at high frequencies, however, at low frequency, the response is good. On the other hand, the behavior of the gyroscope curve is smooth, but the cumulative error increases and there is no way to correct it without performing the fusion of the sensors.

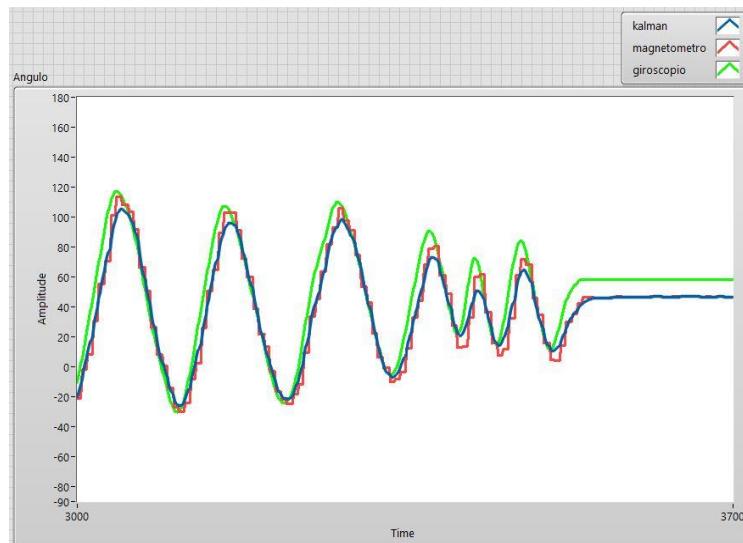


Figure 3 - Kalman filter for orientation

Finally, the encoder is added to the system, in order to have a better estimate of the distance traveled, an extended Kalman filter (EKF) is used, taking into account the non-linearity of the estimation by means of the odometry model.

This model is subject to cumulative errors that increase with time, because there is no external reference. These errors can be minimized by integrating a GPS (Global Position Systems) into the system, but this is beyond the reach of this work.

Path Planner

This module is responsible for taking the autonomous vehicle from an initial position to a final, following a trajectory. The algorithm used is the so-called pure pursuit algorithm. For practical purposes, the implementation of "Team 1712" has been used and modified according to the requirement.

With this algorithm it is possible to determine the target speed of the autonomous vehicle depending on the curvature of the segment of the trajectory in which the autonomous vehicle is located, as well as to establish the direction to which it should go knowing its current position and a target point called "Look Ahead Point".

To obtain greater precision in the calculation of the speed and curvature, points were injected into the original trajectory obtaining in this way closer points, to then passing through smoothing stage and achieve continuity in the path.

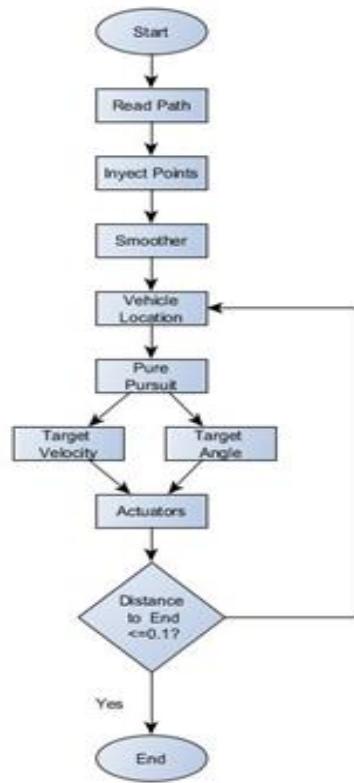


Figure 4 - Path Planner

The Conclusion

NI hardware and software allowed us to develop an autonomous scale car in a short time and with little previous knowledge.

If it wasn't for the NI platform, which made us achieve a very high level of development without entering into hardware programming and allowed us to make use of modularity even using third-party code, we wouldn't have been able to develop the car in time for the competition.

This project was not carried out only by five people, this project involved the entire university community, students of electromechanical engineering, architecture and teachers of the university. In the aguara'i, the effort of each one of them, colleagues who worked with the members of the team during the morning, afternoon and night is put. This experience not only brought intellectual growth but also growth in teamwork. We learned that there are always people willing to help for the simple fact of being necessary. With the project it was possible to inspire the students of the inferior courses of the career and even to high school students , in the exciting area of the autonomous cars.

The Technical Description Paper (TDP), presented in Robocar Race 2018 competition, was selected, along with two others TDP, to be published in the Journal of Production and Automation of the city of São Paulo, besides, the paper achieved the first place in the Research Day organized by the Center for Research in Sciences, Technology and Advanced Innovation (CICTIA) belonging to our institution. This project allowed to start a new line of research at CICTIA.

Time to Build

This project was carried out in two months.The period in which it was made is September 2018 - October 2018

Apéndice F

Invitación a NIWeek 2019



Dear Erid Pacheco,

It is with great satisfaction we would like to invite you for attending **NIWeek 2019**, our annual user conference in Austin, Texas from May 20-23. Given your engagement and accomplishments using our technology along with the fact you are enrolled in an academic institution makes you a valuable participant and contributor for the conference, it will be a pleasure to have your presence participating and contributing to the event through discussions with peers from other universities and industry from all parts of the world.

A Full Conference Pass for NI Week 2019 costs US\$ 1,095.00 if purchased before May 19th, due to your commitment, outstanding accomplishments and positive impact at the Robocar Race in 2018, National Instruments is glad to offer you a free Full Conference Pass for NI Week this year.

Overall, the NIWeek conference provides:

- More than 200 hours of technical, case study, and panel sessions
- 40+ hands-on product sessions
- 50+ training, certifications, and badging opportunities
- Over 100 exhibitors featuring the latest test and measurement advancements
- Keynote presentations from industry thought leaders

I thought you might find these sessions in the Academic track valuable:

- Using Record Players and LabVIEW to Teach PID Control
- Industry-University Partnerships
- Increasing Undergraduate Exposure to Interdisciplinary Design

See the attached flyer for an overview of all sessions and presenters.

At the NIWeek Expo, I think you should visit our Academic pavilion to see these demos:

- Oklahoma State University's Endeavor Labs
- New Solutions for Teaching Circuits and Measurements
- 16x1 Massive MIMO

If needed I would be happy to set up a quick 10 min follow-up call with a NI team member or even an in-person meeting at an appropriated time if you have any questions on the event.

Looking forward to your participation in this year NI Week edition.

Best Regards,


Andre Oliveira
Regional Sales Manager
Brazil and Cono Sur

Apéndice G

Relación entre los sistemas de coordenadas

Antes de definir la relación entre los sistemas de coordenadas de navegación y body, necesitamos definir los ángulos de Euler, mediante el cual, estableceremos la relación entre ellos.

G.1. Ángulos de Euler

Son tres ángulos, que mediante una sucesión ordenada de giros, definen el cambio de un sistema de coordenadas a otro. Estos ángulos son: Roll (φ), Pitch (θ) y Yaw (ψ). El ángulo Roll (φ), es el giro alrededor del eje x en sentido antihorario (ver fig. , el ángulo Pitch (θ), es la rotación alrededor del eje y en sentido antihorario (ver fig. y Yaw (ψ) es la rotación del vehículo alrededor del eje z en sentido antihorario (ver fig. , tomando como referencia el norte.

Los ejes x , y y z , mencionados en el párrafo anterior corresponden a las coordenadas de navegación NED.

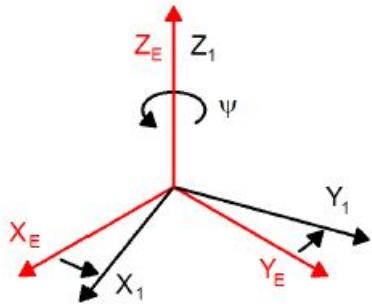
G.2. Cambio de Body a Navegación con los ángulos de Euler

Resulta más conveniente y fácil hacer un análisis de cambio de coordenadas de navegación (x, y, z) a body (x''', y''', z''') , tomando de manera independiente la rotación de cada eje.

G.2.1. Giro en Yaw

Es el primer giro que se hará, con esta matriz pasamos de coordenadas (x, y, z) a (x', y', z') , la matriz de rotación es la siguiente:

Figura G.1: Rotación alrededor del eje Z.



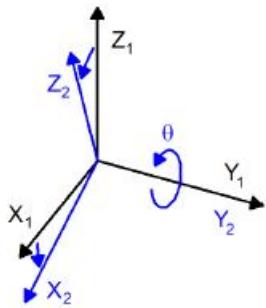
$$C_{\Psi} = \begin{bmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (\text{G.1})$$

Fuente: Bresciani (2008).

G.2.2. Giro en Pitch

Luego giramos el sistema (x' , y' , z') para conseguir (x'' , y'' , z'') con la siguiente matriz de transformación:

Figura G.2: Rotación alrededor del eje y.



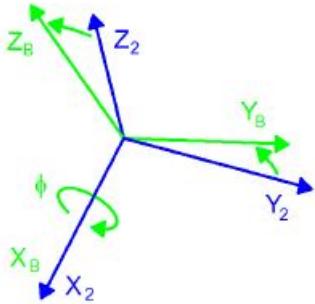
$$C_{\theta} = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \quad (\text{G.2})$$

Fuente: Bresciani (2008).

G.2.3. Giro en Roll

Finalmente, giramos el sistema (x'' , y'' , z'') para conseguir (x''' , y''' , z''') con la siguiente matriz de transformación:

Figura G.3: Rotación alrededor del eje x.



$$C_\varphi = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\varphi & \sin\varphi \\ 0 & -\sin\varphi & \cos\varphi \end{bmatrix} \quad (\text{G.3})$$

Fuente: Bresciani (2008).

Entonces, definimos la matriz de cambio de coordenadas de navegación a body como C_n^b como:

$$C_n^b = C_\psi C_\theta C_\varphi \quad (\text{G.4})$$

$$C_n^b = \begin{bmatrix} \cos\theta\cos\psi & \cos\theta\sin\psi & -\sin\theta \\ \cos\psi\sin\theta\sin\varphi - \sin\psi\cos\varphi & \cos\psi\cos\theta\sin\varphi + \sin\psi\sin\theta\sin\varphi & \cos\theta\sin\varphi \\ \cos\psi\sin\theta\cos\varphi + \sin\psi\sin\varphi & \sin\psi\sin\theta\cos\varphi - \cos\psi\sin\varphi & \cos\theta\cos\varphi \end{bmatrix} \quad (\text{G.5})$$

La ecuación (G.5) corresponde la matriz de cambio de coordenadas de navegación a body. Sin embargo la transformación que nos interesa es justo la inversa, buscamos un cambio de coordenadas de body a navegación. Aplicando la inversa a la ecuación (G.5), se obtiene:

$$C_n^b = (C_b^n)^T \quad (\text{G.6})$$

Apéndice H

Conceptos básicos de probabilidad y estadística

Estos conceptos básicos son extraídos de la tesis de grado de Prado O. (2005).

H.1. Probabilidad de un evento

La probabilidad que el resultado de un evento discreto favorezca un evento particular A , está definida por;

$$p(A) = \frac{\text{Posibles resultados favorables del evento } A}{\text{Total de posibles resultados}} \quad (\text{H.1})$$

H.2. Variables aleatorias

Una variable aleatoria es una función que mapea todos los puntos en el espacio de muestra a los números reales. Las variables aleatorias pueden ser discretas, cuando se puede contar el conjunto de sus resultados posibles, o continuas, cuando toma valores en una escala continua. En el caso de las variables aleatorias continuas, la probabilidad de cualquier evento simple discreto A es de hecho 0. Esto es. $P(A) = 0$.

H.3. Media, varianza y covarianza

Es muy familiar el concepto de promedio de una secuencia de números para algún espacio muestral N (se define espacio muestral como todos los resultados posibles de un experimento estadístico) de una variable discreta X . En lo sucesivo se hará referencia a este valor como la

media de la variable aleatoria X , y se expresa como μ_X o simplemente μ cuando se sabe a qué variable aleatoria se refiere. Es común referirse a esta media como el valor esperado de la variable aleatoria X , y se expresa como $E(X)$. El promedio o media está dada por:

$$\bar{X} = \frac{X_1 + X_2 + \cdots + X_N}{N} \quad (\text{H.2})$$

si X es una variable aleatoria discreta con distribución de probabilidad $f(x)$ la media o valor esperado de X es:

$$\mu = E(X) = \sum_x x f(x) \quad (\text{H.3})$$

si X es una variable aleatoria discreta con distribución de probabilidad $f(x)$ y media μ , la varianza de X se expresa como σ^2 y se define de la siguiente forma:

$$\sigma^2 = E[(X - \mu)^2] = \sum_x (x - \mu)^2 f(x) \quad (\text{H.4})$$

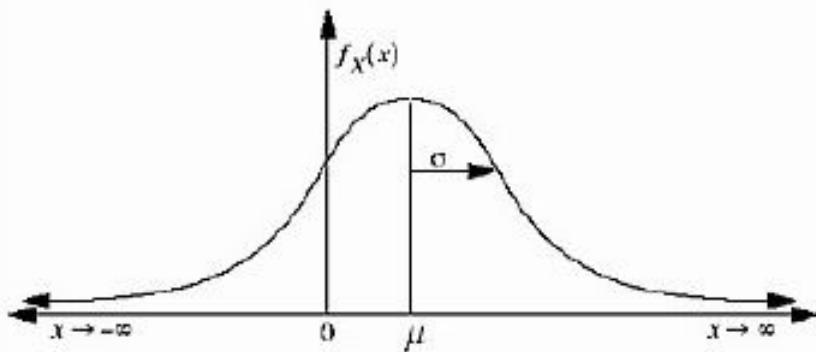
La varianza es una propiedad estadística muy usada en las señales aleatorias, es una medida de dispersión definida como la esperanza del cuadrado de la desviación de dicha variable respecto a su media. También nos da una ponderación de cuanto ruido hay en una señal.

Si X es continua, la raíz cuadrada de la varianza $\sqrt{\sigma^2} = \sigma$, se llama desviación estandar de X .

H.4. Distribución normal o gaussiana

Esta distribución de probabilidad ha tenido un uso muy extendido en modelamiento de sistemas aleatorios. Muchos procesos de la naturaleza parecen tener una distribución normal o muy cercana a esta. En efecto, bajo ciertas condiciones, se puede probar que una suma de variables aleatorias con cualquier distribución tiende hacia una distribución normal. El teorema que define esta propiedad se llama el teorema de límite central.

Figura H.1: Función de distribución de Probabilidad Normal o Gaussiana.



Fuente: Prado O. (2005)

Una variable aleatoria continua X que tiene la distribución en forma de campana de la figura (ref figura) se llama variable aleatoria normal. La ecuación matemática para la distribución de probabilidad de la variable normal depende de los parámetros μ y σ . Por lo tanto se representan los valores de densidad de X por $n(x; \mu, \sigma)$.

La función de densidad de probabilidad para una variable aleatoria normal X , con media μ y varianza σ^2 está dada por:

$$n(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{1}{2}\frac{(x-\mu)^2}{\sigma^2}} \text{ para } -\infty < x < \infty \quad (\text{H.5})$$

H.5. Covarianza

La covarianza es un valor que indica el grado de variación conjunta de dos variables aleatorias respecto a sus medias. Es el dato básico para determinar si existe dependencia entre ambas variables. Su signo, expresa la tendencia en la relación lineal entre las variables. La covarianza entre dos variables x e y viene dada por:

$$\sigma_{(xy)} = \frac{1}{n} \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}) \quad (\text{H.6})$$

H.6. Ruido blanco

Es una señal aleatoria (proceso estocástico) que se caracteriza por el hecho de que sus valores de señal en dos tiempos diferentes no guardan correlación estadística. como consecuencia de ello, su densidad espectral de potencia es una constante. Esto significa que la señal contiene todas las frecuencias y todas ellas muestran la misma potencia.

Apéndice I

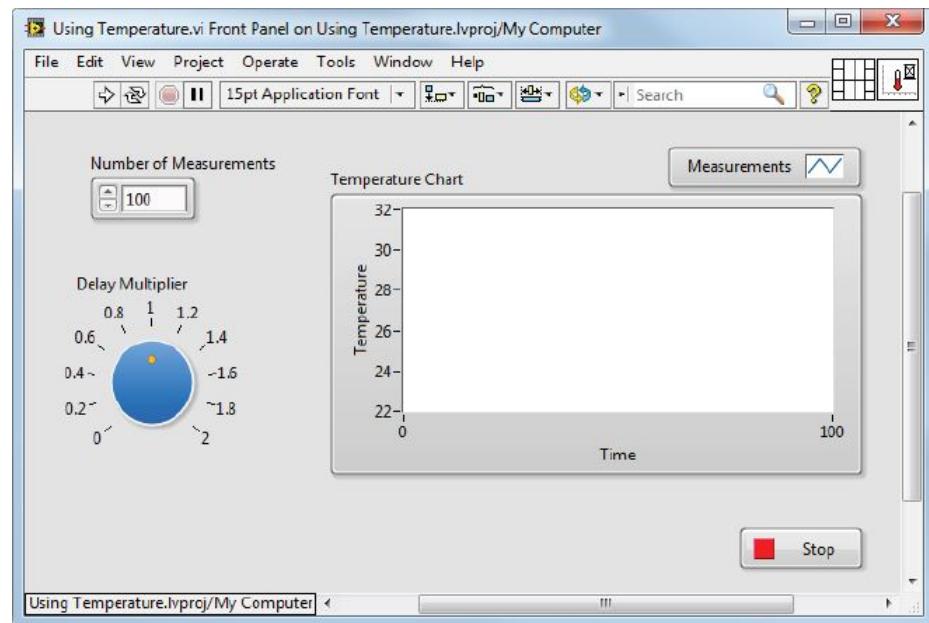
LabVIEW

Esta sección fue extraído del manual de participante del curso de LabVIEW Core 1 (2014) (National, 2014a).

Labview es un entorno de programación gráfico. Ofrece integración con miles de dispositivos y facilita cientos de librerías integradas para análisis avanzados y visualización de datos para crear instrumentos virtuales.

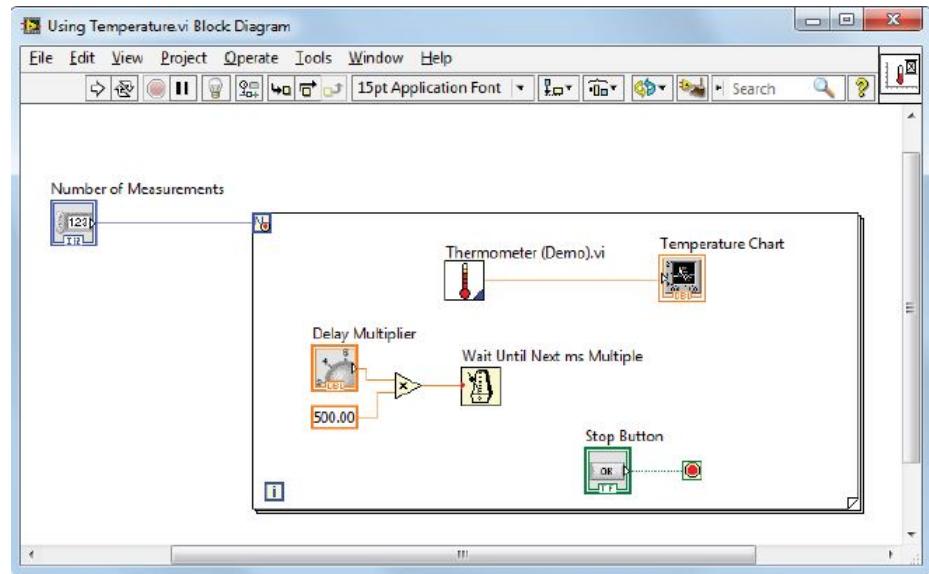
Ya que LabVIEW imita la apariencia y operación de instrumentos físicos, como osciloscopios y multímetros, los porgramas en LabVIEW son llamados instrumentos virtuales o VIs. Los VIs tienen panel frontal y diagrama de bloques. El panel frontal es la interfaz de usuario. El diagrama de bloques es el programa detrás de la interfaz de usuario. El código en el diagrama de bloques es código gráfico, también conocido como código G o código de diagrama de bloques.

Figura I.1: Panel Frontal de un VI.



Fuente: National Instruments (2014a).

Figura I.2: Diagrama de bloques de una VI.



Fuente: National Instruments (2014a).

Apéndice J

Código fuente de los algoritmos implementados

Se muestran los códigos implementados en LabVIEW descriptos en 6.1. El código completo está disponible en el siguiente repositorio:

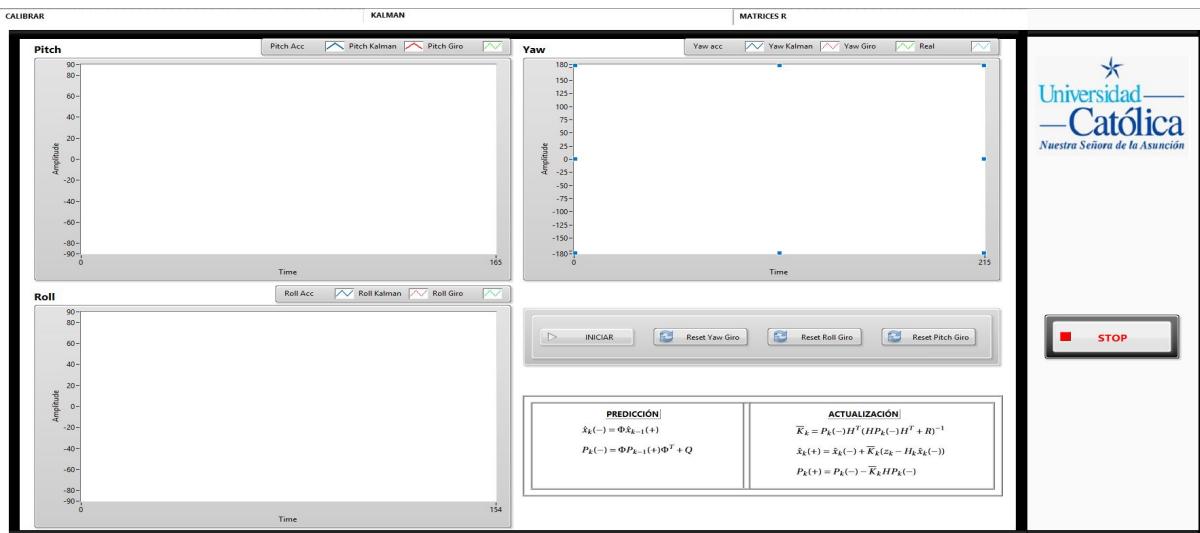
J.1. Panel frontal

Calibrar

Véase figura 6.2.

Kalman

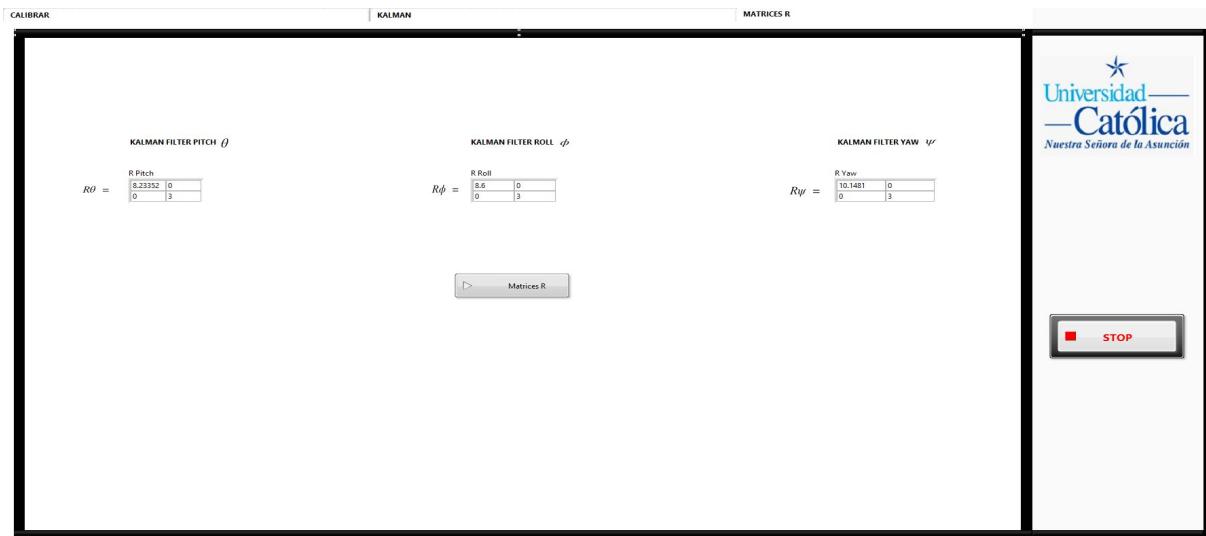
Figura J.1: Panel frontal, Kalman.



Fuente: Elaboración propia.

Matrices R

Figura J.2: Panel frontal, Matrices R.

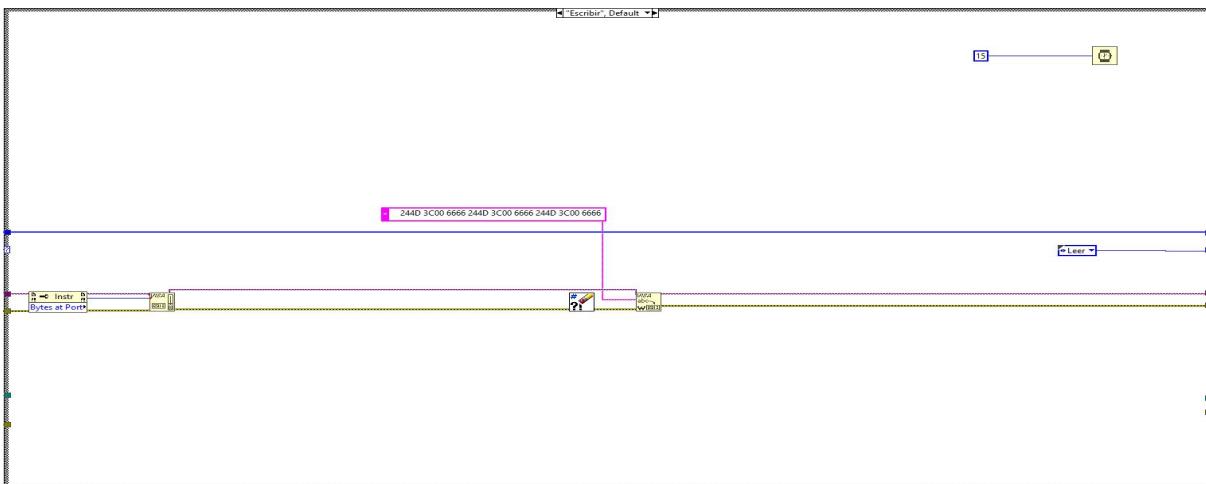


Fuente: Elaboración propia.

J.2. Diagrama de bloques

Escribir

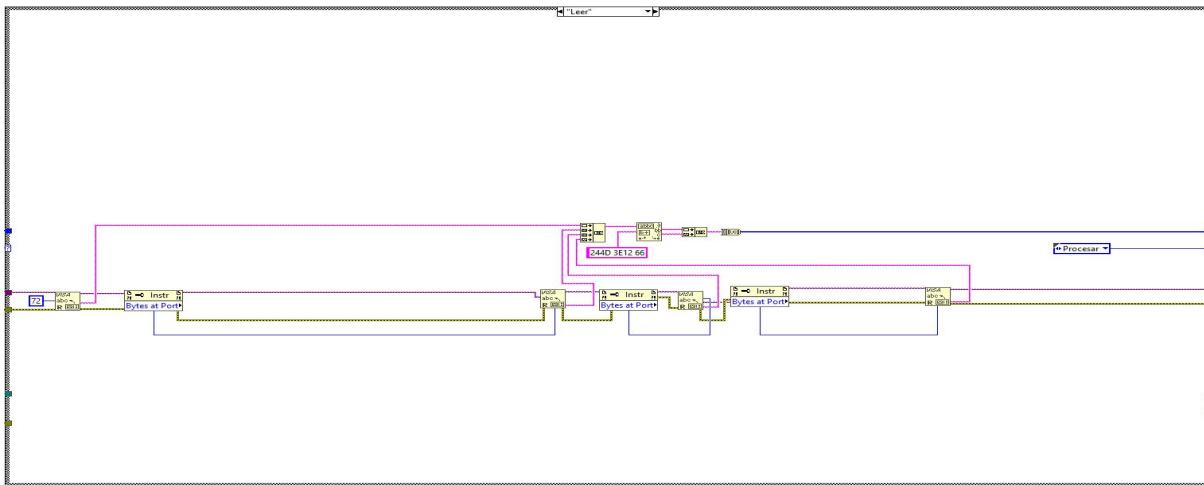
Figura J.3: Caso: Escribir.



Fuente: Elaboración propia.

Leer

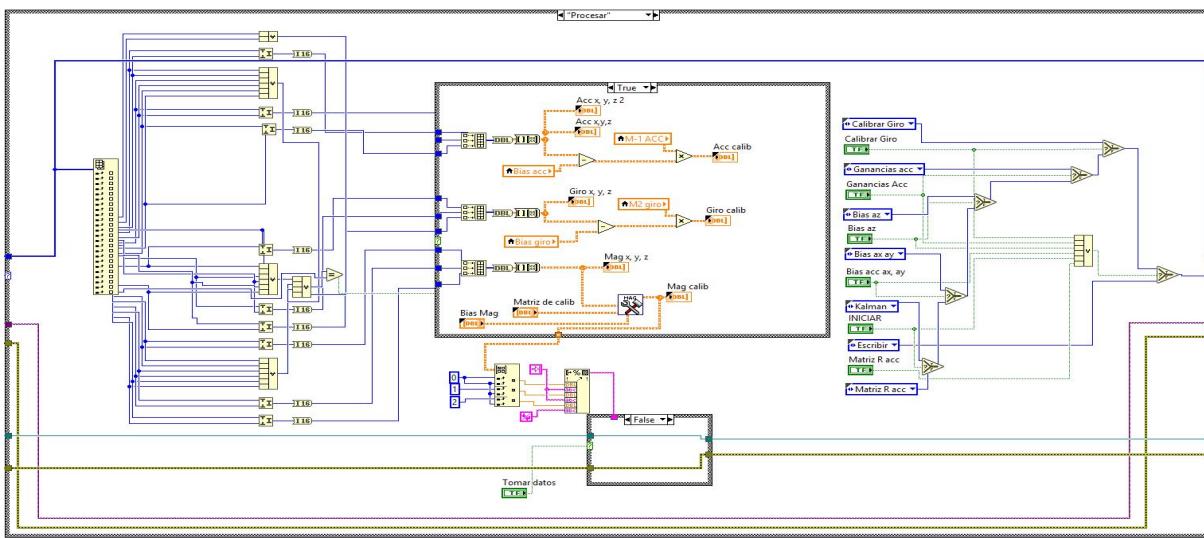
Figura J.4: Caso: Leer.



Fuente: Elaboración propia.

Procesar

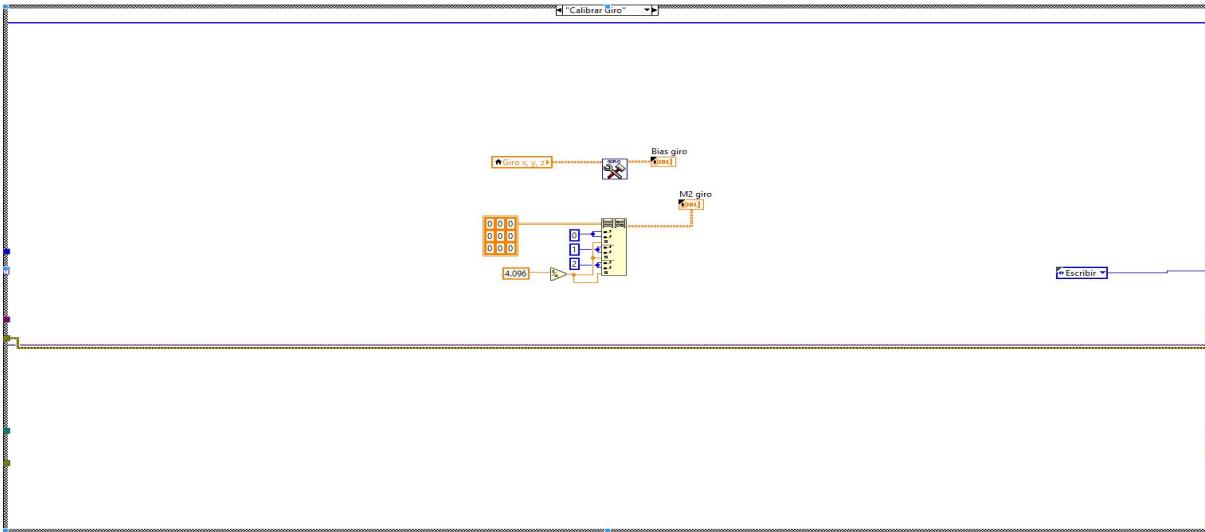
Figura J.5: Caso: Procesar.



Fuente: Elaboración propia.

Calibrar Giro

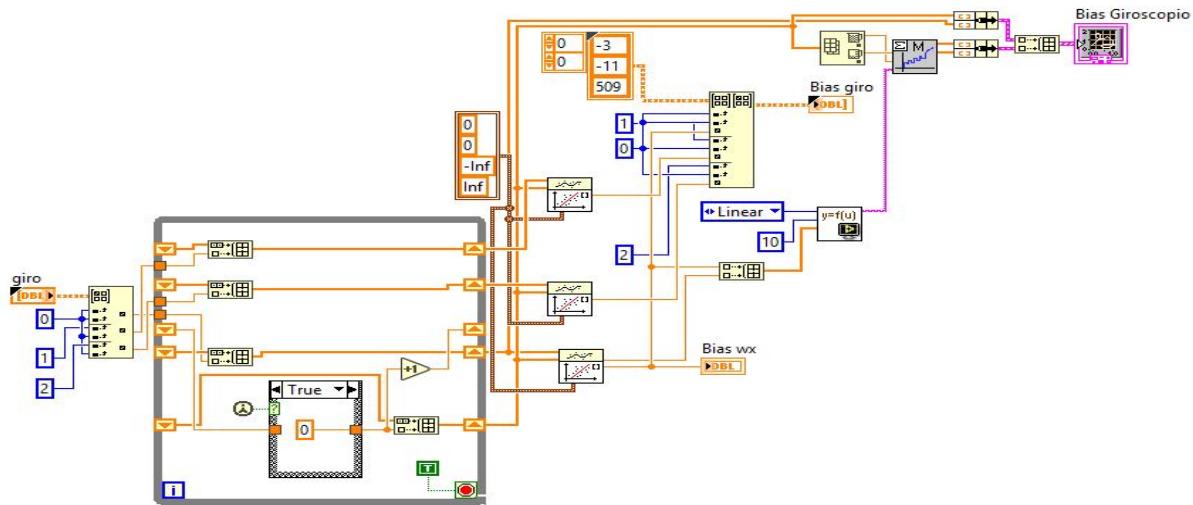
Figura J.6: Caso: Calibrar Giro.



Fuente: Elaboración propia.

El diagrama de bloques del subVI para calibrar el giroscopio es el siguiente:

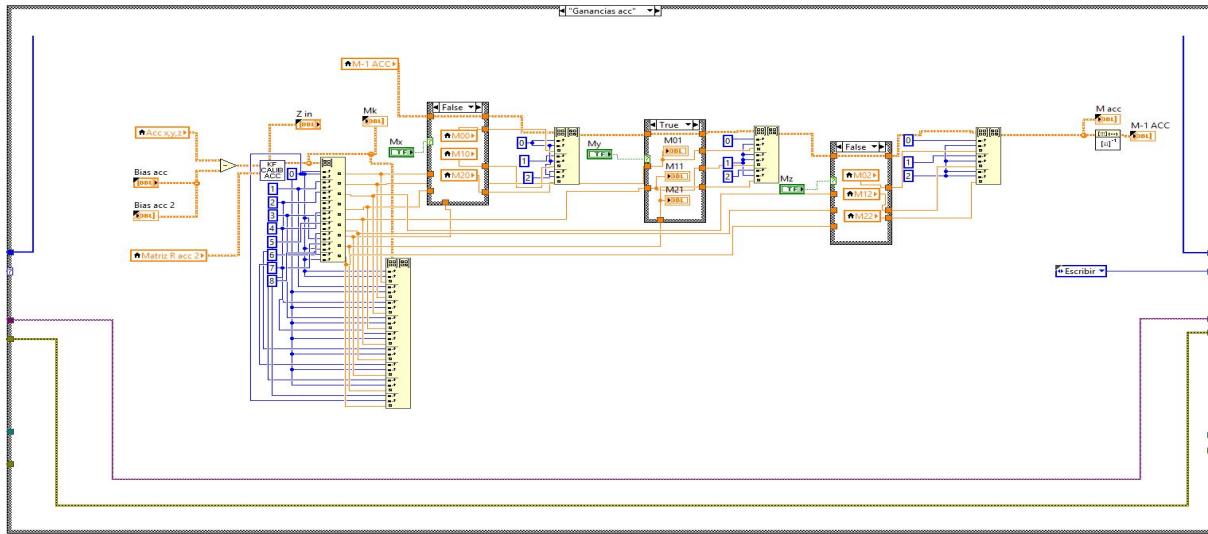
Figura J.7: Subvi del estado Calibrar Giro.



Fuente: Elaboración propia.

Ganancias acc

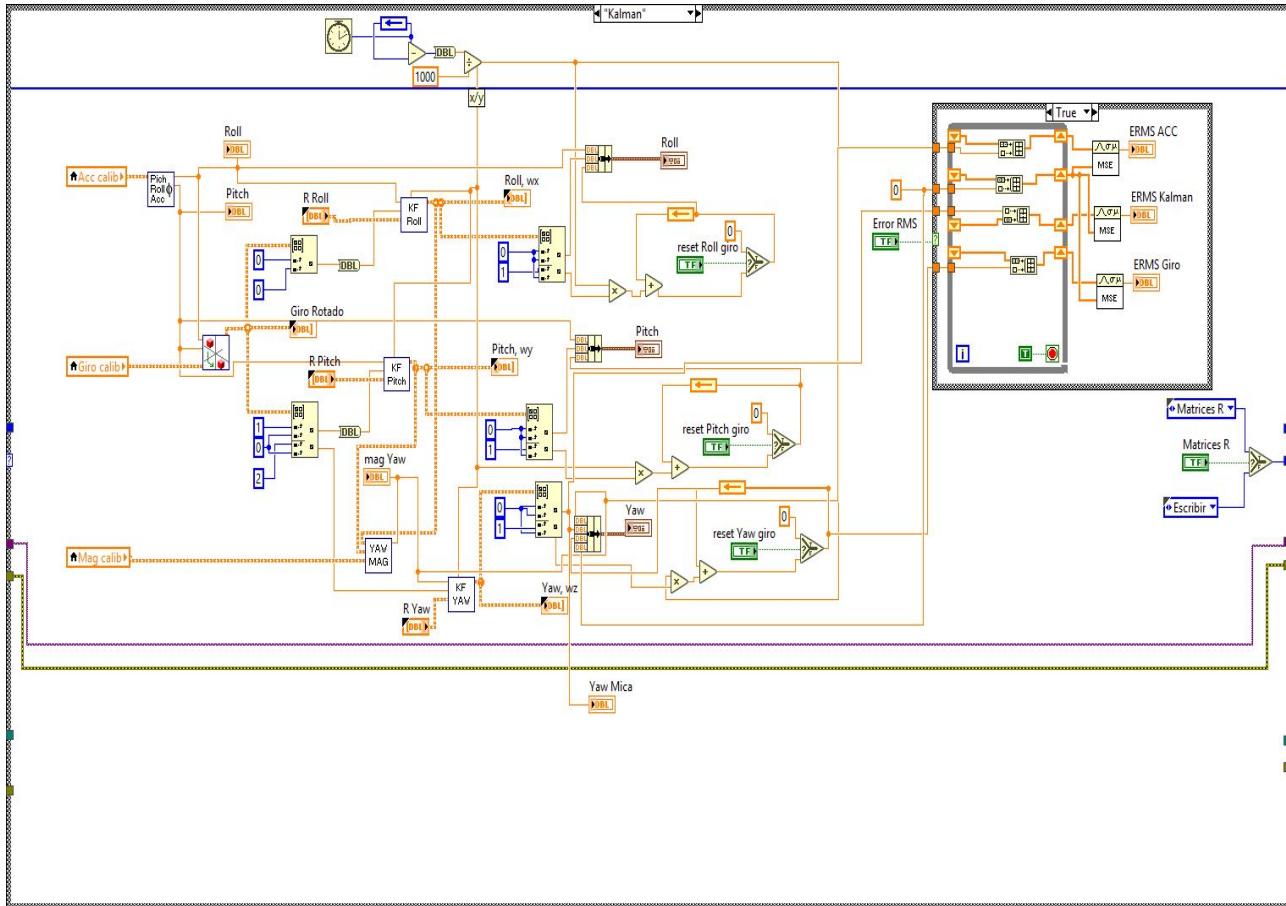
Figura J.8: Caso: Ganancias acc.



Fuente: Elaboración propia.

Kalman

Figura J.9: Caso: Kalman.



Fuente: Elaboración propia.

Los diagramas de bloques de los subvis que se muestran en la figura anterior, se muestran en las figuras 6.12 y 6.13.

Matrices R

Véase figura 6.11.

Bias ax ay

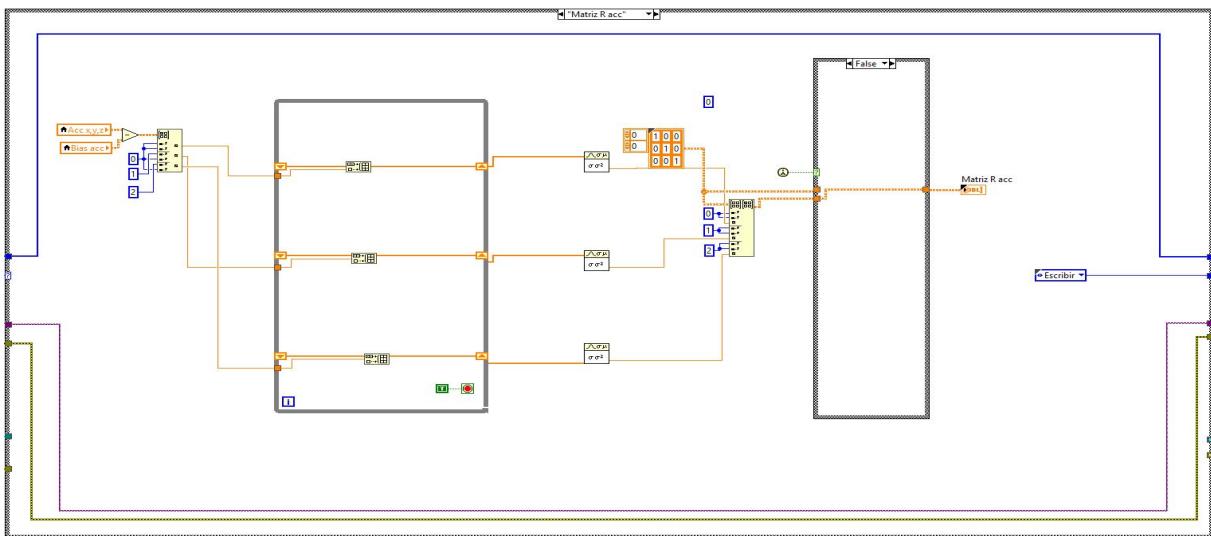
Véase figura 6.4.

Bias az

Véase figura 6.4.

Matrices R acc

Figura J.10: Caso: Matrices R acc.



Fuente: Elaboración propia.

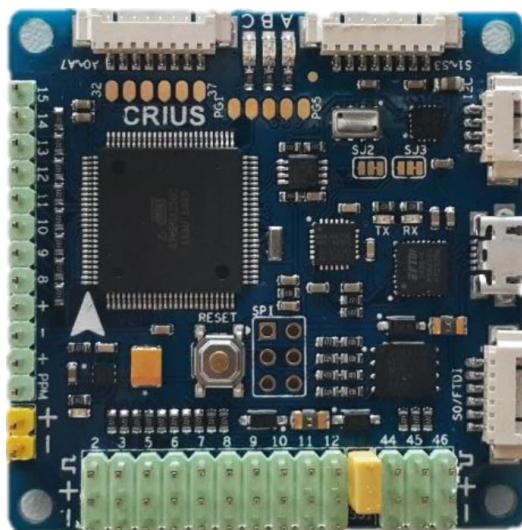
Apéndice K

Especificaciones técnicas

K.1. Crius all in one pro (AIOP) v2.1

Es un controlador de vuelo. Tiene incorporado varios sensores como un giroscopio/acelerómetro MPU6050 de 6 ejes, un altímetro de alta precisión MS5611-01BA01, y un magnetómetro HMC5883L de 3 ejes. El microcontrolador integrado es un ATMEGA 2560 de 8 bits. 16MHz y se comunica con los dispositivos externos mediante los pines y puertos seriales.

Figura K.1: Caso: Matrices R acc.

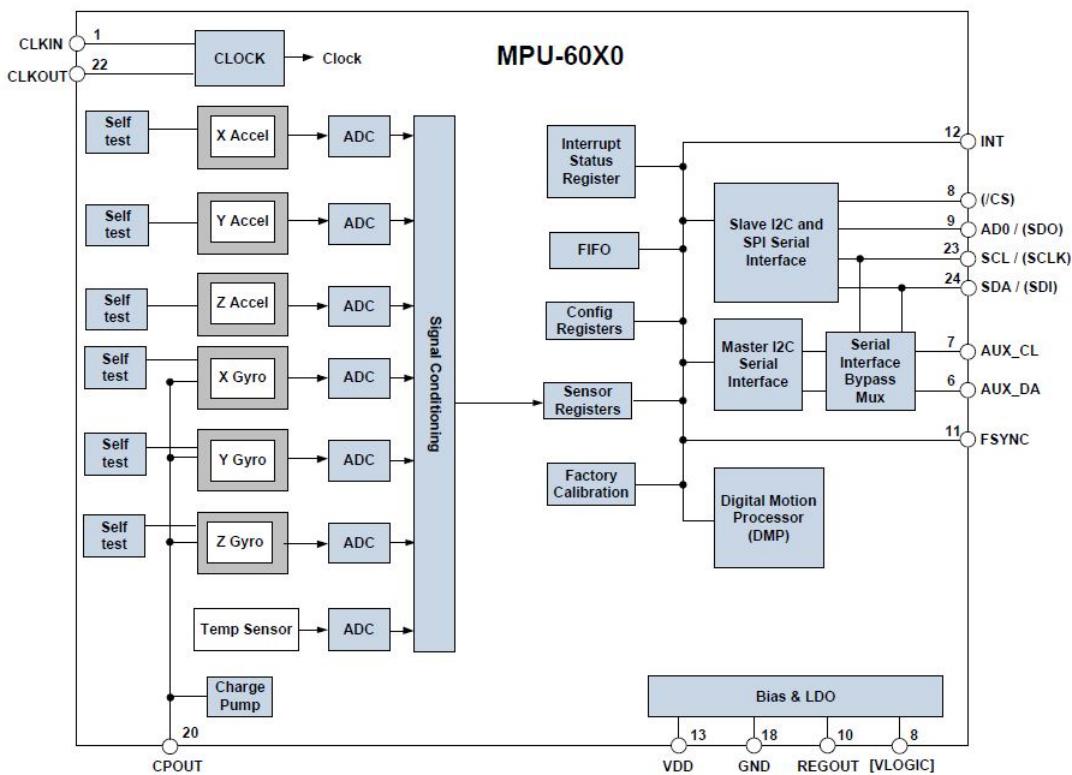


Fuente: Bogado y Benitez (2015).

K.2. MPU6050

Fabricado por InvenSense, combina un acelerómetro y un giroscopio, ambos de 3 ejes, junto con un DMP y un filtro pasa bajo programable. Cuenta con un rango de $\pm 2000^{\circ}/s$ en el caso del giroscopio y de $\pm 16g$ para el caso del acelerómetro. La interfaz serial de comunicación del MPU6050 es el protocolo I2C. Su arquitectura, se muestra en la siguiente imagen.

Figura K.2: Arquitectura del MPU6050.



Fuente: INVENSENSE (2013)

Sus características eléctricas se muestran en la siguiente tabla.

Cuadro K.1: Características del MPU6050.

Características	MPU6050
VDD	2.375V-3.46V
VLOGIC	1.71V to VDD
Interfaz serial soportada	I^2C
Temperatura ideal	25°C

Fuente: INVENSENSE (2013)

Se puede encontrar más información en la hoja técnica de datos que se encuantra en el siguiente enlace: https://store.invensense.com/datasheets/invenSense/MPU-6050_DataSheet_V3%204.pdf.

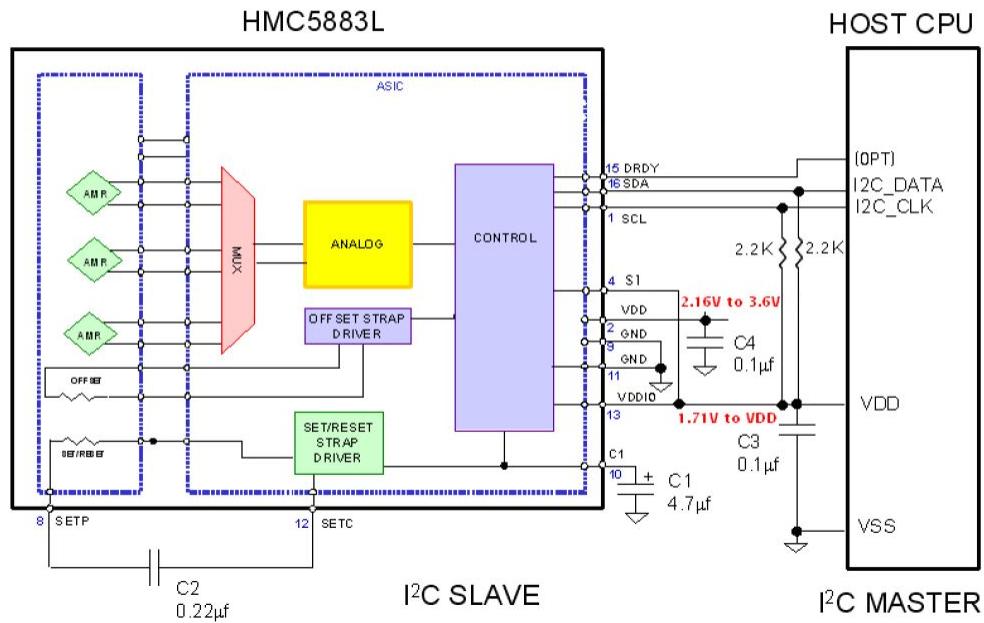
K.3. Magnetómetro HMC5883L

Fabricado por Honeywell, tiene un campo de operación de -8 a +8 gauss. La interfaz serial del HMC5883L cumple el protocolo de comunicación I2C.

Presenta la siguiente arquitectura:

Figura K.3: Arquitectura del HMC5883L.

HMC5883L



Fuente: Honeywell (2012).

Sus características eléctricas se muestran en la siguiente tabla:

Cuadro K.2: Características del HMC5883L.

Características	MPU6050
VDD	2.16V-3.6V
VLOGIC	1.71V to VDD+0.1
Interfaz serial soportada	I^2C
Temperatura ideal	25°C

Fuente: Honeywell (2012).

Se puede encontrar más información en la hoja técnica de datos que se encuentra en el siguiente enlace: <http://www.farnell.com/datasheets/1683374.pdf>.

K.4. Myrio

Es el dispositivo que usaremos para procesar la información. Es un dispositivo enbebido para estudiantes, cuenta con E/S en ambos lados del dispositivo en forma de conectores MXP y SMP. Incluye entradas analógicas, salidas analógicas, líneas de E/S digitales, LEDs, un push-button, un acelerómetro interno, un FPGA Xilinx y un procesador dual-core ARM Cortex-A9. Algunos modelos también incluyen soporte para Wi-Fi (como en nuestro caso). Se puede programar el myRIO con LabVIEW o C.

Figura K.4: Myrio.



Fuente: www.ni.com.

Apéndice L

Nota de agradecimiento



UNIVERSIDAD CATÓLICA “NUESTRA SEÑORA DE LA ASUNCIÓN”
CAMPUS UNIVERSITARIO ALTO PARANÁ
FACULTAD DE CIENCIAS Y TECNOLOGÍA
Teléfono: 595 61 572 467/9 Interno: 113 Web: www.ucap.edu.py
Paraná Country Club - Hernandarias - Paraguay

Hernandarias, 23 de noviembre de 2018

Ing. Juan Carlos Ocampos Núñez

Decano

Facultad de Ciencias y Tecnología

Universidad Católica Nuestra Señora de la Asunción

Campus Alto Paraná

MICHAELA JARA, ERID PACHECO, ARIEL BOGADO y JESÚS FRANCO, alumnos de la Facultad de Ciencias y Tecnología, de la carrera Ingeniería Electromecánica, nos dirigimos a usted y por su intermedio a donde corresponda para manifestar cuanto sigue:

Antes que todo, nuestro más profundo agradecimiento por haber apoyado en todos los aspectos la participación en la corrida de autos autónomos a escala denominado ROBOCAR RACE en la cual participamos como representantes de esta casa de estudios y que fue llevado a cabo en la ciudad de Sao Paulo – Brasil.

Nuestra participación no hubiera sido posible sin el apoyo de la Universidad por su intermedio, así como del estamento estudiantil y de la sociedad en general inclusive, quienes nos ayudaron a cubrir los gastos que generó nuestra participación en dicho evento, y que fue motivado principalmente por el respaldo de la Universidad.

Agradecemos igualmente a la Universidad Católica “Nuestra Señora de la Asunción” campus Alto Paraná y al director general de campus, el Ing. Ladislao Aranda, por su intermedio, pues como ya hemos manifestado, no hubiera sido posible nuestra participación en dicha competencia, sin su apoyo.

Le comentamos, que igualmente hemos tenido el apoyo del Centro de Estudiantes de la Facultad de Ciencias y Tecnología (CEFACYT), quienes nos apoyaron de forma económica y logística en la organización de una fiesta con el fin de recaudar fondos para el viaje. Esto demuestra el compañerismo y el compromiso con esta casa de estudios, lo que nos ha sorprendido gratamente y queremos poner a su conocimiento.

Además, agradecemos el apoyo incondicional del Lic. Ariel Guerrero, profesor de la Universidad quien nos asesoró durante todo el proyecto. Por otra parte, agradecemos de igual manera el apoyo moral de los profesores y alumnos de nuestra facultad, en especial de un grupo de alumnos del tercer año de nuestra carrera, quienes incansablemente nos han ayudado en todo lo que refiere a la parte técnica de la construcción del auto autónomo al que le llamamos AGUARA’I. También tuvimos la ayuda de una alumna la carrera de Arquitectura y de dos ingenieros egresados de nuestra carrera a quienes agradecemos.



UNIVERSIDAD CATÓLICA “NUESTRA SEÑORA DE LA ASUNCIÓN”
CAMPUS UNIVERSITARIO ALTO PARANÁ
FACULTAD DE CIENCIAS Y TECNOLOGÍA
Teléfono: 595 61 572 467/9 Interno: 113 Web: www.ucap.edu.py
Paraná Country Club - Hernandarias - Paraguay

Adjuntamos a la presente nota, los detalles técnicos de la construcción del auto autónomo y con el cual participamos en la competencia mencionada, de lo cual podemos referir que ha sido un éxito, a pesar de que no pudimos traer los primeros lugares de la competencia, pudimos demostrar que nuestra Universidad tiene el mismo nivel que Universidades del extranjero y los alumnos la misma capacidad.

En la página del Facebook del evento (<https://www.facebook.com/robocarrace/>), se pueden observar fotografías y videos de la corrida, donde estuvimos en representación del Paraguay, por la Universidad Católica Nuestra Señora de la Asunción.

Le comentamos, que deseamos realizar una demostración del auto autónomo que hemos construido para todos los integrantes de esta comunidad educativa y para el efecto le solicitamos nos indique una fecha y se puedan cursar las invitaciones correspondientes para que la sociedad en general incluso pueda conocer sobre el trabajo realizado en esta Universidad.

Nos despedimos nuevamente agradeciendo y con este pensamiento: *¡Con el apoyo de todos los integrantes de esta casa de estudios y de la sociedad, pudimos!*

Micaela Jara
Mat.: 45.581

Erid Pacheco
Mat.: 45.990

Ariel Bogado
Mar.: 45.302

Jesús Franco
Mat.: 46.204