

TECNOLOGICO NACIONAL DE MEXICO
INSTITUTO TECNOLÓGICO DE AGUASCALIENTES



TECNOLOGICO
NACIONAL DE MEXICO

**Ciencia y
Tecnología**

Secretaría de Ciencia, Humanidades,
Tecnología e Innovación



Instituto **Tecnológico**
de Aguascalientes

DOCTORADO EN CIENCIAS DE LA INGENIERÍA

**Navegación de un vehículo de guiado
autónomo (AGV) asistido por fusión de
sensores**

Tesis de Doctorado
Para obtener el título de:
Doctor en Ciencias de la Ingeniería

Presenta:

M.C.I. Victor Manuel Mora Romo

Director

Dr. Julio Cesar Martínez Romo

Co-Director

Dr. Francisco Javier Luna Rosas

Asesores

Dr. Miguel Mora González

Dr. Marco Antonio Hernández Vargas

Dr. Mario Alberto Rodríguez Díaz

Aguascalientes, Aguascalientes, Enero de 2025





Educación
Secretaría de Educación Pública



TECNOLÓGICO
NACIONAL DE MÉXICO



Instituto Tecnológico de Aguascalientes
División de Estudios de Posgrado e Investigación

Aguascalientes, Ags., 21/enero/2025

No. De Oficio: 011/2025

Asunto: **Autorización de Tesis de Grado Para su Impresión**

M.C. VICTOR MANUEL MORA ROMO
PRESENTE

Los abajo firmantes, Miembros del Jurado para Examen de Grado de Doctorado, hacemos **CONSTAR** que, habiendo revisado el Trabajo de Tesis desarrollado por Usted, bajo el título: "**NAVEGACIÓN DE UN VEHÍCULO DE GUIADO AUTOMÁTICO (AGV) BASADO EN FUSIÓN DE SENSORES**" hemos dictaminado que éste es de aceptarse, por lo que se autoriza su impresión.

ATENTAMENTE

Excelencia en Educación Tecnológica®
Ingenio, Cultura y Saber, que conducen a la Excelencia®

PRESIDENTE


DR. JULIO CESAR MARTÍNEZ ROMO

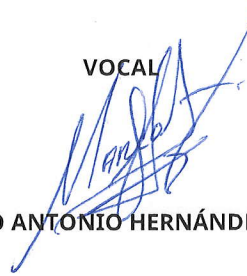
SECRETARIO


DR. FRANCISCO JAVIER LUNA ROSAS

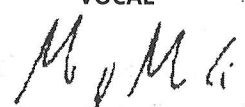
VOCAL

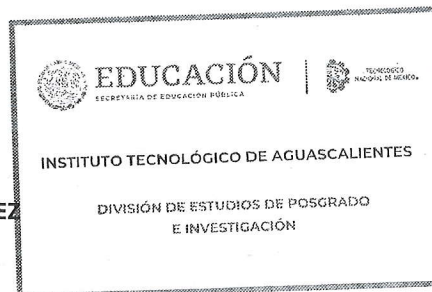

DR. MARIO ALBERTO RODRÍGUEZ DÍAZ

VOCAL


DR. MARCO ANTONIO HERNÁNDEZ VARGAS

VOCAL


DR. MIGUEL MORA GONZÁLEZ



2025
Año de
La Mujer
Indígena

Av. Adolfo López Mateos #1801. Fracc. Bona Gens

C.P. 20256 Aguascalientes, Ags. Tel. 449 910 5002 ext. 127

e-mail: jefatura.depi@aguascalientes.tecnm.mx

tecnm.mx | www.aguascalientes.tecnm.mx



ABET



Instituto Tecnológico de Aguascalientes
División de Estudios de Posgrado e Investigación

Aguascalientes, Ags., **21/enero/2025**
Asunto: Autorización de uso de derechos

MORA ROMO VICTOR MANUEL, mayor de edad, con domicilio ubicado en AV Canal Interceptor 101 interior 8, San Cayetano, Aguascalientes, Aguascalientes en mi calidad de titular y autor de la tesis denominada **"NAVEGACIÓN DE UN VEHÍCULO DE GUIADO AUTOMÁTICO (AGV) BASADO EN FUSIÓN DE SENSORES"** quien para todos los fines del presente documento se denominará **EL AUTOR Y/O TITULAR**, suscribo el presente documento de autorización de uso de derechos patrimoniales de autor a favor del Tecnológico Nacional de México/Instituto Tecnológico de Aguascalientes el cual se registrá por las cláusulas siguientes:

PRIMERA – AUTORIZACIÓN: **EL AUTOR Y/O TITULAR**, mediante el presente documento autoriza la utilización de los derechos patrimoniales de autor al Tecnológico Nacional de México/Instituto Tecnológico de Aguascalientes, de la tesis denominada **"NAVEGACIÓN DE UN VEHÍCULO DE GUIADO AUTOMÁTICO (AGV) BASADO EN FUSIÓN DE SENSORES"**, a través del Repositorio Institucional del Tecnológico Nacional de México (en lo sucesivo TecNM) y en el Repositorio Nacional, que puede ser consultado en la liga electrónica: (www.repositorionacionalcti.mx/).

SEGUNDA – OBJETO: Por medio del presente escrito, **EL AUTOR Y/O TITULAR** autoriza al Tecnológico Nacional de México/Instituto Tecnológico de Aguascalientes, a través del Repositorio Institucional del TecNM y en el Repositorio Nacional para que de conformidad con la Ley Federal del Derecho de Autor y la Ley de la Propiedad Industrial, use los derechos del documento antes referido, con fines exclusivamente académicos.

TERCERA – TERRITORIO: Los derechos aquí autorizados se dan sin limitación geográfica o territorial alguna.

CUARTA – ALCANCE: La presente autorización se da tanto para formato o soporte material, y se extiende a la utilización en medio óptico, magnético, electrónico, en red, mensajes de datos o similar conocido o por conocer, del ejemplar o número respectivo de la publicación.

QUINTA – EXCLUSIVIDAD: La autorización de uso aquí establecida no implica exclusividad en favor del Tecnológico Nacional de México/Instituto Tecnológico de Aguascalientes. Por lo tanto, **EL AUTOR Y/O TITULAR** en su carácter de autor de la obra objeto del presente documento se reserva el derecho de publicar directamente, u otorgar a cualquier tercero, autorizaciones de uso similares o en los mismos términos aquí acordados.



Instituto Tecnológico de Aguascalientes
División de Estudios de Posgrado e Investigación

SEXTA – DERECHOS MORALES (Créditos y mención): La Autorización de los derechos antes mencionados no implica la cesión de los derechos morales sobre los mismos por cuanto en conformidad con lo establecido en los artículos 18, 19, 20, 21, 22 y 23 de la Ley Federal de Derechos de Autor, dada la cuenta que estos derechos son inalienables, imprescriptibles, irrenunciables e inembargables. Por lo tanto, los mencionados derechos seguirán radicados en cabeza de EL AUTOR Y/O TITULAR, y siempre deberá mencionarse su nombre cuando se utilice la obra.

SÉPTIMA – AUTORÍA: EL AUTOR Y/O TITULAR, declara y ratifica que el material objeto de la presente fue realizada por él o por ella sin violar o usurpar derechos de Propiedad Intelectual de terceros.

ATENTAMENTE

Excelencia en Educación Tecnológica®

Ingenio, Cultura y Saber, que conducen a la Excelencia

MORA ROMO VICTOR MANUEL
AUTOR DE LA TESIS

Vo.Bo.

DR. JULIO CESAR MARTÍNEZ ROMO
DIRECTOR DE LA TESIS

ccp. Archivo



2025
Año de
**La Mujer
Indígena**

Av. Adolfo López Mateos #1801. Fracc. Bona Gens
C.P. 20256 Aguascalientes, Ags. Tel. 449 910 5002 ext. 127
e-mail: jefatura.depi@aguascalientes.tecnm.mx
tecnm.mx | www.aguascalientes.tecnm.mx



Instituto Tecnológico
de Aguascalientes





Agradecimientos

En primer lugar, deseo expresar mi agradecimiento a mi asesor de esta tesis doctoral, Dr. Julio Cesar Martínez Romo, por la dedicación y apoyo que ha brindado a este trabajo, por la dirección y el rigor que ha facilitado a las mismas. Gracias por la confianza ofrecida desde que llegué a esta institución, así como su apoyo incondicional y su calidez tanto como persona como profesional en todo momento, de igual manera a los doctores: Dr. Francisco Javier Luna Rosas, Dr. Miguel Mora González, Dr. Marco Antonio Hernández Vargas, Dr. Mario Alberto Rodríguez Díaz, que con su retroalimentación han enriquecido este trabajo de manera asertiva para su realización.

A mis padres Manuel Mora Tiscareño y María Delia Romo Márquez, quienes con su amor, paciencia y esfuerzo me han permitido llegar a cumplir hoy un sueño más, gracias por inculcar en mí el ejemplo de esfuerzo y valentía, de no temer las adversidades.

A mi amada esposa por ser el motor que impulsa mis proyectos, ser esa chispa especial que se ha convertido en mí compañera y mi confidente y que con su paciencia ha encaminado que nuestro amor crezca cada día en nuestra familia, así mismo a mi amado hijo que me ha enseñado más cosas que las que yo podría haber aprendido en tan poco tiempo y que con su forma de ver la vida simplifica mis procesos haciéndolos ver más maravillosos.

Mi profundo agradecimiento a todas las autoridades y personal del Instituto Tecnológico de Aguascalientes, por confiar en mí, abrirme las puertas y permitirme realizar todo el proceso de investigación dentro de su establecimiento educativo.

Así mismo a todos mis compañeros y amigos míos que se han aventurado a preverme de calidez humana y voto de confianza en todo momento.

Resumen

La presente tesis aborda el desarrollo de un sistema de navegación avanzado para Vehículos de Guiado Autónomo (AGV), centrándose en la mejora de la estimación de pose (x , y , θ) mediante la integración de sensores y el uso de un filtro de Kalman. La investigación se enfoca en resolver las limitaciones inherentes a la odometría convencional, como los errores acumulativos y la sensibilidad al ruido, proponiendo una metodología basada en la fusión de datos para garantizar una navegación más precisa y robusta.

Se diseñó un prototipo de AGV tipo triciclo equipado con un conjunto de sensores, incluyendo giroscopios MEMS y encoders, para recopilar datos relacionados con su posición y orientación. Estos datos fueron integrados a través de un filtro de Kalman, permitiendo reducir significativamente los errores de estimación y mejorar la confiabilidad del sistema en entornos estructurados y no estructurados. Además, se llevaron a cabo pruebas experimentales que validaron la capacidad del sistema para seguir trayectorias planeadas con alta precisión.

Los resultados obtenidos demuestran que la propuesta supera las soluciones tradicionales basadas exclusivamente en odometría. En particular, se logró una reducción promedio del 12.34% en el error absoluto de posición final y una mejora del 31.33% en el seguimiento de trayectorias. Sin embargo, la implementación del filtro de Kalman también implicó mayores requerimientos computacionales, lo que plantea la necesidad de optimizar su implementación para sistemas de hardware más limitados.

Entre las principales contribuciones de la tesis se encuentra la validación de la fusión de sensores como una herramienta clave para la navegación autónoma, así como la identificación de líneas de trabajo futuro. Estas incluyen la incorporación de sensores adicionales como láseres o cámaras, el desarrollo de métodos automáticos de calibración y la exploración de variantes avanzadas del filtro de Kalman, como el filtro extendido (EKF) o el filtro no lineal (UKF).

Índice General

Agradecimientos	2
Resumen	3
Índice General.....	4
Índice de Figuras.....	8
Índice de Tablas	11
Capítulo 1. Introducción.....	12
1.1 Antecedentes.....	12
1.1.1 Estrategias de Navegación de robots móviles.....	14
1.1.2 Algoritmos de navegación basados en visión artificial.....	23
1.1.3 Topologías.....	25
1.1.4 Herramientas de control y sector de aplicación	25
1.1.5 El ángulo θ para la navegación de vehículos autónomos.....	25
1.2 Planteamiento del problema.....	27
1.3 Justificación.....	28
1.4 Hipótesis.....	29
1.5 Objetivos.....	30
1.5.1 Objetivo general.....	30
1.5.2 Objetivos Particulares.....	30
1.6 Conclusiones.....	31
Capítulo 2. Marco teórico.....	33
2.1 Robótica Móvil.....	33
2.1.1 Entornos en que opera un robot móvil.....	34
2.1.2 Sistemas de locomoción.....	35
2.1.3 Tipos de ruedas.....	36
2.1.4 Disposición de las ruedas.....	38
2.1.5 Robot omnidireccional	39
2.1.6 Uniciclo	41
2.1.7 Triciclo	41

2.1.8	Cuatriciclo	47
2.1.9	Tracción y dirección	48
2.1.10	Configuraciones especiales.....	51
2.1.11	Tipos de sensores en robótica móvil	52
2.1.12	Sensores en robots móviles	55
2.2	Arquitectura de un AGV.	76
2.3	Topología tipo triciclo.	77
2.3.1	Diseño de un sistema AGV.....	78
2.3.2	Objetivos de un sistema AGV	79
2.3.3	Interacción de un AGV con subsistemas	80
2.3.4	Ventajas de los sistemas AGV	80
2.3.5	Componentes de un sistema AGV.....	81
2.3.6	Sistemas de guía	81
2.4	Odometría de terreno	83
2.5	Navegación Inercial.....	83
2.6	Fusión de Sensores	83
2.6.1	Sistema autónomo.....	84
2.6.2	4 formas en que la fusión de sensores puede ayudar	85
2.6.3	Acelerómetro	86
2.6.4	Problemas del acelerómetro	87
2.6.5	Magnetómetro.....	87
2.6.6	Resolviendo el otro problema de las aceleraciones lineales corruptas..	87
2.7	Filtro de Kalman	89
2.8	Varianza de Allan	95
2.9	Sistemas Embebidos.....	97
2.9.1	3.7.1 Características de un sistema embebido	99
2.9.2	3.7.2 Hardware de sistemas embebidos.....	100
2.9.3	3.7.3 Principios básicos de un sistema embebido	103
2.9.4	3.7.4 Aplicaciones de un sistema embebido.....	105
2.10	IMU: Inertial Measurement Unit.	105

2.10.1	MEMS.....	106
2.10.2	El MPU9250	128
2.11	Motores de DC (Corriente continua)	130
2.12	Tipos de sensores.....	131
2.13	Sistemas de alimentación de baterías	132
2.13.1	Técnicas de carga de las baterías.....	133
2.13.2	Batería.....	133
2.14	Sistemas de monitoreo Continuo	134
2.15	Sistema de Monitoreo Periódico	134
2.16	Etapa de Control	135
2.17	Conclusiones	135
Capítulo 3. Desarrollo.....		137
3.1	Desarrollo físico del prototipo de AGV.	137
3.1.1	Requerimientos del AGV	137
3.1.2	Diseño propuesto.....	138
3.1.3	Diseño de la estructura del AGV.....	139
3.1.4	Elementos físicos del sistema	140
3.1.5	Integración.....	141
3.2	Desarrollo electrónico y plataforma de software del AGV.	142
3.2.1	Arquitectura de software del AGV.....	142
3.2.2	Elección placa de desarrollo	143
3.2.3	Sistema electrónico	146
3.3	Desarrollo e implementación del algoritmo.	152
3.3.1	Función de procesamiento y lectura de 2 sensores IMU (Inertial Measurement Unit) MPU6050	154
3.3.2	Función de estimación del robot en el Plano	157
3.3.3	Función de actualización del modelo cinemático de topología triciclo	162
3.3.4	Función de Actualización de encoders	165
3.3.5	Función de Configuración para el Motor y PWM	166
3.3.6	Función de Servomotor de dirección	167

3.3.7	Bloque de ejecución del programa	168
3.4	Navegación punto a punto	170
3.4.1	Navegación basada en el modelo cinemático	170
3.4.2	Varianza de Allan.....	176
3.5	Fusión de sensores con el Filtro de Kalman	179
3.5.1	Encuadre	179
3.5.2	Diseño del filtro de Kalman.....	180
3.6	Conclusiones.....	185
Capítulo 4.	Resultados	187
4.1	Calibración del modelo del AGV.	187
4.1.1	Calibración Odométrica	190
4.1.2	Distancia d entre el eje de la llanta motriz y el eje trasero.....	197
4.2	Filtrado de Kalman	198
4.2.1	Covarianza del ruido de los sensores giroscópicos	200
4.2.2	Covarianza del ruido de odometría.....	201
4.2.3	Integración del filtro de Kalman	201
4.2.4	Construcción del filtro de Kalman en Python.	203
4.3	Navegación	206
4.3.1	Trayectoria Propuesta	207
4.3.2	Resultados de la Navegación	208
4.4	Filtro complementario.....	219
4.5	Conclusiones.....	220
Capítulo 5.	Conclusiones	222
5.1	Introducción.....	222
5.2	Análisis de Resultados	222
5.3	Implicaciones y Trabajo Futuro	223
5.4	Conclusión General.....	223
Bibliografía.....		225
Anexo		234

Figura 2-1. robots de interior (amigobot) y exterior (seekur) de activemedia.	34
Figura 2-2. Entorno estructurado (biblioteca) y no estructurado (almacén en línea de producción).....	35
Figura 2-3. Robots terrestres. (a) hermes (robosoft). (b) journey (smu). (c) scour (usf).	35
Figura 2-4. Robot acuático (nsf) y robot aéreo (tu Berlín)	36
Figura 2-5. Tipos de ruedas. (a) rueda fija. (b) rueda orientable centrada. (c) rueda loca.....	37
Figura 2-6. Detalle de una rueda sueca y su disposición sobre una estructura mecánica.	38
Figura 2-7. Robot omnidireccional con ruedas suecas. (a) maniobrabilidad. (b) robot uranus (universidad de Michigan).	39
Figura 2-8. Robot omnidireccional con ruedas orientables centradas. (a) disposición sobre una estructura mecánica. (b) robot seekur (activemedia).	40
Figura 2-9. Sincronismo entre el sistema de tracción y dirección con ruedas omnidireccionales (synchro drive). (a) mecánico. (b) electrónico.....	40
Figura 2-10. (a) Estructura. (b) Robot pioneer (activemedia).	41
Figura 2-11. Triciclo. (a) Estructura. (b) Robot neptune (universidad carnegie mellon).	41
Figura 2-12. Robot Móvil de 2 ruedas.	42
Figura 2-13. Localización en el plano cartesiano del robot.	43
Figura 2-14. Relaciones geométricas para el ángulo de giro.	44
Figura 2-15. Modelo de la estructura cinemática del robot.....	45
Figura 2-16. Elementos de la cinemática de la configuración tipo triciclo. (Martínez-Romo, 2017).....	46

Figura 2-17. Sistema de dirección ackerman.	48
Figura 2-18. Sistema de tracción y dirección en ejes independientes.....	49
Figura 2-19. Sistema de tracción y dirección sobre un mismo eje.	50
Figura 2-20. Sistema de tracción y dirección sobre todos los ejes.....	50
Figura 2-21. Robot de múltiples grados de libertad desarrollado en la universidad de michigan.	51
Figura 2-22. Módulos cooperativos (universidad de hamburgo y beihang).	51
Figura 2-23. robot de exploración espacial pathfinder (jpl).....	52
Figura 2-24. Descriptores dinámicos de un sensor.	54
Figura 2-25. Sistema de referencias en un robot móvil.	56
Figura 2-26. Odometría en un robot móvil.....	59
Figura 2-27. Sensor doppler.....	60
Figura 2-28. Codificador óptico. (a) principio de funcionamiento (b) dispositivo comercial.	60
Figura 2-29. Acelerómetro. (a) Principio de funcionamiento. (b) Acelerómetro electrónico.	61
Figura 2-30. Compás electrónico hm55b basado en el efecto hall.....	62
Figura 2-31. Girocompás desarrollado por herman anschütz-kaempfe en 1903...	62
Figura 2-32. Para cada punto sobre una línea hiperbólica, la distancia $(abc - ac) = k$, siendo k constante.....	64
Figura 2-33 Cobertura del sistema loran.	65
Figura 2-34. Esquema del funcionamiento de un sistema de trilaterización.....	66
Figura 2-35. El sistema de posicionamiento global (gps) consiste de tres segmentos fundamentales: espacio, control y usuario.	68
Figura 2-36 Procedimiento general para el posicionamiento mediante marcas. ...	70
Figura 2-37. Diferentes posibilidades de triangulación. A) ángulos absolutos de referencia. B) ángulos observados entre marcas. C) distancias observadas a las marcas. D) ángulo y distancia a una marca.	70
Figura 2-38. Posicionamiento basado en mapas.	71
Figura 2-39. Distintos tipos de mapas. (a) geométrico. (b) topológico.	75
Figura 2-40. Arquitectura del AGV.	76

Figura 2-41. Elementos de la cinemática de la configuración tipo triciclo. (Martínez-Romo, 2017).....	77
Figura 2-42. Vehículo de guiado automático AGV (Automatic Guided Vehicle)...	77
Figura 2-43. Fusión de sensores en sistemas autónomos	85
Figura 2-44. Orientación descrita en una rotación.....	86
Figura 2-45. Diferencia de tiempo en desplazamiento.	95
Figura 2-46. Campana de gauss en varianza de Allan.....	97
Figura 2-47. Dependencia temporal de O_y en el tiempo.....	97
Figura 2-48. Competencia de métrica de diseño.....	105
Figura 2-49. Representación de un acelerómetro	108
Figura 2-50. Movimiento de la representación de un acelerómetro.....	108
Figura 2-51. Representación de la fuerza en el acelerómetro.....	109
Figura 2-52. Representación de el comportamiento de un acelerómetro triaxial.	110
Figura 2-53. Sistema de coordenadas a los ejes del acelerómetro	111
Figura 2-54. Vector de fuerzas del sistema de coordenadas a los ejes del acelerómetro	114
Figura 2-55. sistema de coordenadas a los ejes del giroscopio	115
Figura 2-56. Placa genérica de un IMU.....	119
Figura 2-57. vector de un componente a la vez. RxGyro.	124
Figura 3-1 Imagen de prototipo 3	139
Figura 3-2 Dimensiones del AGV	139
Figura 3-3 Vista lateral izquierda, B) Vista lateral derecha.....	141
Figura 3-4 Configuración del sistema.....	142
Figura 3-5 Raspberry 4	145
Figura 3-6 Pinout raspberry 4.....	145
Figura 3-7 Diagrama de bloques del sistema en general.	148
Figura 3-8 Diagrama Electrónico del sistema.....	152
Figura 3-9 La pose del AGV. Distancia en metros.	176
Figura 3-10 Desviación de Allan.....	178
Figura 3-11 Diagrama de flujo fusión de sensores	184

Figura 4-1 Separaciones entre las llantas y el eje. a) Modelo Real b)	
Representación del modelo.....	190
Figura 4-2 Trayectoria propuesta	207
Figura 4-3 Comparación de Errores Absolutos	217

Indice de Tablas

Tabla 2-1. Clasificación de los sensores en robótica.	52
Tabla 2-2. Posicionamiento de un robot móvil (sensores y técnicas)	58
Tabla 3-1 Material del AGV	140
Tabla 3-2 Dimensiones de las matrices del Filtro de Kalman.....	183
Tabla 4-1 Pose final de las 25 corridas utilizadas para la calibración odométrica.	191
Tabla 4-2 Trayectorias de seguimiento.	207
Tabla 4-3 Resultados de la navegación en 5 ejecuciones de trayectoria de la tabla 4-2.	208
Tabla 4-4 Medición del error entre las coordenadas propuestas (target) y las reales	209
Tabla 4-5 Medición del error entre datos reales y obtenidas sin fusión.....	211
Tabla 4-6 Medición del error entre datos target y sin fusión.....	212
Tabla 4-7 Medición del error entre datos real y obtenidas con fusión	213
Tabla 4-8 Medición del error entre datos target y obtenidas con fusión	214
Tabla 4-9 Medición del error entre datos fusionados y sin fusión.....	215
Tabla 4-10 Tabla odometría vs fusión de datos, respecto a la posición objetivo (target).....	216

Capítulo 1. Introducción.

En este capítulo se abordan los antecedentes y el estado del arte relacionados con los vehículos de guiado autónomo, sentando las bases para comprender el contexto en el que se desarrolla el proyecto, así como sus problemáticas; a partir de estas últimas, se plantea el problema a resolver, como punto de partida para definir los demás elementos metodológicos de esta investigación. Al final de este capítulo, se delinea el contenido de esta.

1.1 Antecedentes.

El uso de robots en la industria manufacturera moderna es amplio y diversificado, destacándose la aplicación de la robótica móvil como uno de los campos más relevantes. La robótica móvil es un tema de investigación fascinante por varias razones fundamentales. En primer lugar, su implementación requiere una integración sinérgica de conocimientos y técnicas provenientes de diversas disciplinas, tales como la física, matemáticas, mecánica, electrónica, control automático y computación. Esto se debe a que la robótica móvil exige un alto grado de inteligencia en su funcionamiento. En segundo lugar, los robots móviles representan una aproximación a la creación de agentes inteligentes (Russell & Norvig, 2003) capaces de procesar información ambiental y tomar decisiones similares a las de un ser humano. En tercer lugar, las aplicaciones de los robots móviles son innumerables, abarcando campos como la exploración planetaria, la minería, la vigilancia, la asistencia médica, el entretenimiento, la investigación y desarrollo, la industria militar, la agricultura, el transporte, entre otros. Además, estos robots se utilizan como plataformas móviles en aplicaciones que requieren la integración con otros tipos de robots, como aquellos que incorporan brazos manipuladores para tareas de seguridad, neutralización de dispositivos explosivos y manipulación de sustancias peligrosas.

Un robot móvil se define como un sistema electromecánico capaz de moverse de manera autónoma en un espacio de trabajo determinado, tomando decisiones basadas en su percepción del entorno, en lugar de seguir una secuencia predefinida

de instrucciones. La mayoría de las veces, estos robots operan en un lazo cerrado, lo que significa que ajustan su movimiento en tiempo real en respuesta a su entorno.

Los robots móviles se clasifican en tres categorías según su locomoción: robots de ruedas (Alexander & Maddocks, 1989), (Guy Champion, 1996), (Patrick et al., 1986), robots de patas (Raibert et al., 1983) y robots de orugas (Grzegorz Granosik & Johann Borenstein, 2005). La mayoría de los robots móviles utilizan ruedas debido a su eficiencia energética en superficies lisas y firmes (Muir, 1986), así como a su menor complejidad en términos de diseño y fabricación (Bekke, 1969). La importancia de los robots móviles radica en su capacidad de desplazamiento en un espacio de trabajo ilimitado, en contraste con los robots manipuladores fijos, que están restringidos a un espacio definido por sus dimensiones físicas. Esto aumenta significativamente su movilidad y capacidad de trabajo.

Los robots móviles cambian constantemente su posición y orientación en función de su movimiento, lo que requiere que reconozcan continuamente su entorno y se adapten en consecuencia. Para lograr esto, necesitan una sólida capacidad de percepción y un grado de autonomía para ejecutar movimientos planificados y estabilizar el vehículo frente a perturbaciones.

Uno de los aspectos clave para que un robot móvil logre desplazarse y adaptarse a su entorno de manera eficiente es el concepto de pose, el cual describe su posición y orientación en el espacio. La pose de un robot en un entorno bidimensional se representa mediante tres variables: x, y, θ . Aquí, x, y corresponden a las coordenadas espaciales que definen la ubicación del robot en un plano, mientras que θ es el ángulo que describe su orientación con respecto a un marco de referencia, como el eje x del sistema de coordenadas.

El ángulo θ es fundamental porque determina hacia dónde está dirigido el robot. No solo es importante conocer en qué punto del espacio se encuentra el robot, sino también hacia dónde se orienta, lo que le permite planificar su dirección de movimiento. Esta capacidad de conocer y actualizar continuamente su pose es crítica para que el robot ajuste su trayectoria en función de las condiciones del

entorno, evite obstáculos y mantenga su estabilidad frente a perturbaciones externas.

A medida que el robot se desplaza, su pose cambia continuamente, lo que significa que debe ser capaz de percibir su entorno de manera precisa para ajustar tanto su posición como su orientación de manera autónoma.

Un tipo específico de robot móvil es el Vehículo de Guiado Automático (AGV), que se utiliza comúnmente en la industria manufacturera para transportar piezas desde el almacén al punto de ensamblaje. Estos AGVs suelen seguir rutas marcadas por cintas magnéticas en el suelo y su sistema de navegación está diseñado para realizar tareas específicas. A medida que evoluciona la tecnología, se buscan soluciones más flexibles y adaptables que permitan a los AGVs responder a cambios inesperados en su entorno de trabajo.

1.1.1 Estrategias de Navegación de robots móviles

Existen trabajos a través de su propuesta pretenden el diseño por completo de un vehículo terrestre que puede conducirse por sí solo a través de las diversas observaciones que va haciendo a su paso (Ahsan, 2014), el cual cuenta con una computadora que procesa las imágenes al frente del vehículo en tiempo real y guía al sistema de conducción indicándole cómo y cuándo debe de moverse; de manera similar, trabajos donde se enfocan de manera total en la reducción significativa de sensores a través de un sistema de navegación basado en visión resultando una posible manipulación de una cantidad de AGV's simultáneamente y de manera adecuada (Ben Kotze & G. J., 214d. C.).

Como antecedentes, existen investigaciones que abordan temas relacionados, como la visión estéreo y la fusión de sensores, pero no se ha explorado suficientemente la combinación de ambas técnicas para reducir el error de medición causado por la deriva, como se menciona en (Natalia et al., 2009). Utilizando la fusión de sensores de posición de efecto Hall y una cámara de video en (Bostanci et al., 2018), se desarrolló un procedimiento para implementar algoritmos de

navegación guiada y autónoma, donde uno de los problemas principales es que se observa un error entre la trayectoria ya que se utilizó un GPS comercial con una exactitud del orden de dos metros. Por ese motivo las tareas de navegación en el laboratorio no son válidas debido al alto error. Para implementar el algoritmo en el GPS real es necesario agrandar la escala de la imagen de 80 m2 a 2500 m2.

En (Jiménez et al., 1995) se presenta un sistema de modelado de entornos tridimensionales para tareas de teleoperación. Se incide en los aspectos de reconstrucción tridimensional del entorno mediante la integración de las técnicas de visión estéreo y el control activo de los parámetros del cabezal estereoscópico con control de convergencia y enfoque, a fin de conseguir un sistema robusto que permita la generación de mapas visuales integrables dentro del conjunto robot-planificador de trayectorias desempeñando tareas en entornos complejos y no estructurados. Se ha acometido el problema de la integración de diferentes técnicas sensoriales para una eficiente reconstrucción tridimensional de la escena, que permita generar mapas visuales integrables dentro del conjunto robot-planificador de trayectorias, desempeñando tareas en entornos complejos y no estructurados más aún todas las técnicas de fusión son entorno de visión más no de ningún de otro sensor auxiliar ajeno al mismo.

En (Kumar et al., 2010)(Kumar et al., 2010) se ha presentado un método de fusión de sensores de Cámara LRF y estéreo para generar cuadrículas de ocupación 2D para la detección y evitación de obstáculos.

Tanto los datos de láser como de cámara son necesario para la navegación exitosa de la ruta. La precisión del láser se utiliza para complementar las propiedades de imágenes en 3D del cámara de estereovisión del cual los únicos dos utilizados.

En (Villarreal-O et al., 2022)(Villarreal-O et al., 2022) se presenta el desarrollo del robot móvil "LuboBot" diseñado para entornos educativos universitarios en el ámbito de la "Robótica Móvil". Utiliza un sensor Kinect 360 y una IMU MPU6050 para la generación de mapas y la navegación autónoma. Se emplea el paquete Gmapping de ROS para la creación de mapas utilizando el sensor Kinect y el nodo

depthimage_to_laserscan para convertir las lecturas en 3D a 2D. Además, se destaca el uso de paquetes como robot_localization para la fusión de sensores y el paquete de navegación de ROS para controlar el movimiento del robot. La validación del estimador de odometría con encoders e IMU se realiza mediante un Filtro de Kalman Extendido, y la localización se lleva a cabo con el paquete AMCL de ROS.

En (Guido & Sánchez, 2011), el enfoque central es en el desarrollo de un sistema de navegación autónoma para robots móviles en entornos desconocidos. El objetivo principal es permitir que el robot se desplace desde un punto inicial hasta un punto final de manera autónoma, sin intervención humana, utilizando únicamente datos provenientes de sensores ultrasónicos. El ambiente completamente desconocido presenta obstáculos cuyas posiciones y características son desconocidas. El robot, equipado con sensores ultrasónicos, utiliza métodos de odometría para calcular su posición en tiempo real y su orientación. La navegación se divide en tres componentes principales: el seguimiento del objetivo, la evasión de obstáculos y la determinación de direcciones posibles. En el seguimiento del objetivo, se calcula un ángulo de giro mínimo necesario para alinear el robot con el punto final. Este ángulo se traduce en direcciones deseadas utilizando reglas difusas, permitiendo que el robot se oriente de manera efectiva hacia el objetivo. El componente de evasión de obstáculos evalúa la cercanía de obstáculos utilizando datos de sensores ultrasónicos y ajusta la transitabilidad de direcciones específicas en función de la proximidad de los obstáculos. Se destaca la implementación de simulaciones en MATLAB que permiten evaluar el rendimiento del sistema de navegación en entornos desconocidos, marcando el trayecto del robot y midiendo el tiempo de llegada al objetivo.

En el trabajo de (López Orozco, 2003), se presenta un método de integración y fusión de medidas destinado a robots móviles terrestres autónomos, con un enfoque en la simplificación de la prueba y desarrollo de algoritmos sin grandes modificaciones. El sistema de control y sensorización para robots móviles permite

la utilización de diversos sensores conectados como una caja negra, facilitando la realización de pruebas con diferentes grupos de sensores, ya sean complementarios o redundantes. El documento aborda la diferencia entre integración y fusión multisensorial, destacando que la integración se refiere al uso de información de distintos dispositivos para ayudar en una tarea, mientras que la fusión implica la combinación de diferentes fuentes de información sensorial en un formato común. Además, se clasifica la fusión según el tipo de información recibida, ya sea de señal, píxel, características o simbólico. Se enfatiza la importancia de la elección correcta de la arquitectura de fusión, considerando la eficacia, restricciones operativas, recursos eficientes, flexibilidad operacional y crecimiento funcional. Se discuten diversas teorías de fusión multisensorial, como la teoría de la detección y la teoría de la estimación.

En (Li et al., 2024)El trabajo se centra en la fusión de información de estado e información de entorno, utilizando un filtro de Kalman distribuido modificado. Se realizan simulaciones para evaluar el sistema en distintas situaciones, destacando el comportamiento en reconocimiento de entorno, construcción de mapas, navegación reactiva y detección de objetos en movimiento. Se implementa un sistema de control orientado a objetos en tiempo real utilizando el lenguaje de especificación gráfica ROOM y PVM como herramienta de implementación.

En las pruebas con un robot real, se incorporan sistemas de seguridad para reaccionar ante obstáculos, acelerando el tiempo de respuesta, reduciendo la velocidad ante la proximidad de objetos y calculando un nuevo camino alternativo. Se destaca la adaptabilidad del sistema de fusión y la arquitectura de control a un entorno real, resaltando propiedades como la robustez frente a fallos de sensores.

El trabajo concluye que, a pesar de utilizar algoritmos de control de ruedas y planificadores de caminos simples en las pruebas reales, se ha logrado un enfoque exitoso en la integración y fusión de múltiples sensores, creando un sistema fácilmente paralelizable, adaptable y independiente del número de sensores utilizados.

Respecto a la resolución del problema de control de seguimiento de trayectoria podemos mencionar a manera de investigación trabajos como en (Altamirano, 2010), donde el movimiento del robot es guiado por el gradiente del campo y su característica principal es que al mismo tiempo se realiza la generación de la trayectoria y puede efectuar el movimiento del robot.

En cuanto al análisis y diseño de control en (Bañó Azcón, 2003) se calcula un controlador digital de posición, con estabilidad demostrada, el cual permite que el movimiento del móvil tenga la precisión necesaria, y que siga la trayectoria deseada.

Por otra parte, en (Hernández Millán et al., 2012), dado que no siempre es posible conocer con exactitud la función de transferencia de todo sistema, se resalta la necesidad de utilizar herramientas alternativas a las técnicas de control tradicionales. Una de estas herramientas clave son las redes neuronales artificiales (RNA), las cuales no requieren conocer de manera directa el comportamiento matemático de los componentes del sistema. En su lugar, las redes neuronales aprenden el comportamiento deseado del sistema a través del análisis de datos. En su investigación, los autores implementaron un controlador PID basado en RNA para el control de motores D.C. en robots móviles diferenciales. Los resultados mostraron que este enfoque permitió un mejor desempeño del sistema en comparación con los métodos tradicionales, especialmente en entornos con alta incertidumbre o cuando los parámetros del sistema eran difíciles de modelar con precisión. La RNA logró ajustar el control del motor de manera eficiente, demostrando su capacidad para adaptarse a diferentes condiciones de operación sin necesidad de conocer a fondo la dinámica exacta del sistema.

De igual manera, en (Solaque et al., 2014) se desarrolla un sistema de control, aplicado en un robot móvil de configuración diferencial, para seguir una trayectoria determinada el cual, para dar solución al problema, primero se implementa la cinemática directa del robot para simular el comportamiento de éste; después, dos sistemas de control (holonómico y no-holonómico), que fueron desarrollados e implementados a partir de la cinemática inversa. Posteriormente, se desarrollan

pruebas que permiten comparar el rendimiento de los dos controladores, determinando cuál proporciona la mejor solución.

Estos sistemas de navegación comparten un problema común: la limitación. La flexibilidad de la ruta obliga a los usuarios a seguir rutas fijas (o ir sobre puntos de ruta fijos). En consecuencia, se reserva algo de espacio para uso exclusivo del sistema de transporte. Cualquier cambio de la ruta sólo se puede hacer instalando nuevas marcas y reservando nuevo espacio para el sistema AGV (Martínez-Barberá & Herrero-Pérez, 2014); ya que todas las rutas deben estar señalizadas, la infraestructura debe cubrir toda la zona donde se utiliza el sistema de transporte automático.

Las plataformas móviles autónomas deben estar equipadas con un navegador sistema de navegación que proporciona la capacidad de moverse independientemente en áreas donde la observación de puntos de ruta no es posible. Allí, el controlador de a bordo debe navegar utilizando los datos de sensores locales (navegación a estima).

La autonomía parcial de la plataforma móvil se puede lograr utilizando las tecnologías existentes. Por ejemplo, se utilizó un sensor de efecto Hall para medir las desviaciones de la trayectoria mientras pasa por puntos de ruta magnéticos (Lee & Yang, 2012). Del mismo modo, con el uso de códigos QR y una cámara montada en la plataforma móvil, la desviación de la pista se mide en robots Kiva (E. Guizzo, 2008). El sistema de control local que opera en estos robots identifica las propiedades de la plataforma (como la velocidad angular de rotación) con mediciones de varios sensores a bordo y la observación de puntos de referencia fijos. Como resultado, los robots pueden moverse de forma autónoma a través de la distancia entre puntos de referencia.

Así mismo en (Acosta Calderon et al., 2015), propusieron una solución para eliminar la necesidad de puntos de referencia de hardware en el control de robots móviles. En su propuesta, el sistema permite que la plataforma 'aprenda' la ruta durante su primera ejecución en modo manual, almacenando los datos

proporcionados por los sensores a bordo. Una vez completado este proceso, el robot puede seguir la misma ruta de manera autónoma, basándose en los datos recopilados previamente. Si es necesario cambiar la ruta, el robot debe ser guiado nuevamente por la nueva trayectoria, o también es posible cargar un mapa ya existente que el robot haya seguido previamente. Solución similar, utilizando datos de codificadores y del láser escáner se pueden encontrar en (Martínez-Barberá & Herrero-Pérez, 2014)(Martínez-Barberá & Herrero-Pérez, 2014)(Martínez-Barberá & Herrero-Pérez, 2014). Una ventaja de las soluciones de este tipo es que la plataforma es capaz de realizar algunas operaciones sin referencia a la infraestructura fija de puntos.

En (Baño Azcón, 2003)(Baño Azcón, 2003)(Baño Azcón, 2003), se aborda el control de posición de un robot móvil con tracción diferencial. Se destaca la implementación de controladores digitales para lograr movimientos precisos y seguir trayectorias específicas. Se reconoce la simplicidad del control de posición con características geométricas y se menciona la limitación de ciertos movimientos en sistemas con restricciones cinemáticas, como la tracción diferencial. Se destaca la influencia de la masa del robot como un punto débil y se mencionan mejoras en algoritmos de la máquina de estados que desarrollaron y en el control de velocidad. La investigación futura se enfocará en diseñar un control digital de velocidad más eficiente.

En el estudio (Olivares Ávila et al., 2015)(Olivares Ávila et al., 2015)(Olivares Ávila et al., 2015), se presenta un sistema de localización autónoma para robots móviles que utiliza fusión de sensores propioceptivos. Se emplea un filtro de Kalman extendido para fusionar datos de odómetros, acelerómetros y giroscopio, logrando una mejora del 38% en comparación con sistemas sólo odométricos. Se destaca la importancia de la variable de orientación θ y se sugiere la posibilidad de mejorar el sistema con la inclusión de sensores adicionales, como una brújula. Se plantea la adaptabilidad del sistema para la incorporación de nuevos sensores y se sugieren posibles direcciones para investigaciones futuras, como métodos de ajuste para matrices y aplicaciones en navegación robótica.

En el trabajo de (Leyton et al., 2009)(Leyton et al., 2009)(Leyton et al., 2009), se presenta el desarrollo, implementación y análisis de un Filtro de Kalman "Unscented" (UKF) para la estimación en tiempo real de las variables de estado de un mini-helicóptero robot. Se utiliza el UKF para compararlo con el Filtro de Kalman Extendido (EKF) existente en el modelo. Se destaca la importancia de las matemáticas para comprender el proceso determinístico del UKF, que incorpora la transformación "Unscented" (UT) al EKF. Se resalta que la aplicación de las matemáticas fue crucial para entender el proceso determinístico del UKF, pero se enfrentaron desafíos durante la implementación, especialmente con la técnica de muestreo (UT). La complejidad del proceso de transformación de la matriz de puntos sigma y la multiplicación por parámetros aumentaron el riesgo de inconsistencias en el modelo, afectando el tiempo de operación del UKF en comparación con el EKF.

A pesar de los desafíos, se observó que el UKF mantuvo su trayectoria constante frente a variaciones en las condiciones iniciales, y se concluyó que, a pesar de su mayor complejidad y tiempo de operación, el UKF presenta más ventajas que desventajas en comparación con el EKF. Como trabajo futuro, se sugiere optimizar los algoritmos del UKF para reducir los tiempos de cómputo y explorar características adicionales de los filtros de Kalman, como el procesamiento secuencial de señales a diferentes períodos de muestreo.

En el proyecto de (Sartika et al., 2020)(Sartika et al., 2020)(Sartika et al., 2020), se utiliza una unidad de medición inercial (IMU) de 9 grados de libertad (DOF) compuesta por magnetómetro, acelerómetro y giroscopio para determinar las rotaciones tridimensionales de un objeto y su ángulo de rumbo. Se implementan algoritmos de filtro complementario y filtro de Kalman extendido en una placa Arduino Uno con el protector del sensor ROHM para Arduino.

El experimento demuestra que el uso conjunto de los tres sensores de la IMU proporciona resultados más prometedores que utilizar cada sensor por separado. Se destaca la diferencia de precisión entre el filtro complementario y el filtro de

Kalman, donde este último demuestra reaccionar de manera más rápida y efectiva al cambio del sistema. Para trabajos futuros, se identifica la necesidad de compensar el error de desalineación entre los 9 ejes DOF, ya que el proyecto asume una alineación perfecta. Se sugiere implementar diferentes modelos de filtros, como el Filtro Kalman federado adaptativo difuso (FAFKF) o el Filtro de información extendido (EIF), que podrían ofrecer resultados más precisos.

En (Anota et al., 2020)(Anota et al., 2020)(Anota et al., 2020), se simula la localización de un robot móvil con el Filtro Extendido de Kalman (EKF). Utilizando un láser para medir ángulos a balizas, se observa que el EKF se comporta según lo esperado, ofreciendo mejores estimaciones con menores errores en los sensores y la odometría.

En el proyecto (Rodríguez Muñoz P., 2002)(Rodríguez Muñoz P., 2002)(Rodríguez Muñoz P., 2002), se aplica el Filtro de Kalman al seguimiento de objetos en secuencias de imágenes 2D. Se destaca la importancia de una buena técnica de segmentación para el seguimiento efectivo de los objetos. Además, se enfatiza la necesidad de reducir los espacios de búsqueda en la escena para optimizar el costo computacional del seguimiento. Se sugiere como trabajo futuro la implementación de una interfaz gráfica que sirva como apoyo didáctico al estudio del filtro de Kalman, subrayando así la relevancia de este tipo de filtros en el procesamiento de imágenes y el seguimiento de objetos.

En el proyecto (López Montes, 2016)(López Montes, 2016)(López Montes, 2016), se implementa un método de fusión de sensores utilizando el procedimiento del filtro de Kalman, teóricamente a través de un código en Matlab y prácticamente con seis sensores de temperatura LM35. Las conclusiones destacan que, en la simulación del método de fusión de sensores multi-frecuencia, la señal generada tiene la resolución del sensor con una frecuencia de muestreo más grande y un error comparativamente menor.

En la simulación del método de fusión de sensores redundantes, se concluye que la señal generada tiene un error menor que la generada por el sensor más preciso, gracias al método de intersección de covarianzas implementado. Se señala

que el filtro de Kalman no funciona de manera óptima para sistemas no gaussianos, sugiriendo la implementación de otro procedimiento para tales sistemas. Como trabajo futuro, se propone investigar el comportamiento del filtro de Kalman al variar el número de sensores y considerar nuevos métodos de fusión que brinden confiabilidad en sistemas no gaussianos.

El proyecto de (María Díaz Díaz, 2012)(María Díaz Díaz, 2012)(María Díaz Díaz, 2012) presenta un nuevo Filtro de Kalman para estimar la orientación de vehículos, destacando su formulación en cuaterniones, eliminación de singularidades y evaluaciones trigonométricas, así como su eficiencia computacional. Se propone comparar su eficacia con métodos como EKF o SPKF, explorar la expansión del vector de estado, desarrollar aplicaciones en distintos ámbitos y validar resultados mediante pruebas experimentales. Además, se sugiere realizar comparativas con diversos métodos de fusión para evaluar sus resultados.

1.1.2 Algoritmos de navegación basados en visión artificial

Si bien se han mencionado algunos puntos a considerar en cuanto los AGV'S en general, afrontar la problemática basada en la resolución de la navegación por medio de visión podemos considerar trabajos propuestos como en (Eusebio & Ana Yaveni, 2014)(Eusebio & Ana Yaveni, 2014)(Eusebio & Ana Yaveni, 2014) donde se describe una propuesta de control visual para la formación de robots móviles tipo unicycle bajo el esquema líder-seguidor. El cual considera una sola cámara fija observando el espacio de trabajo de los robots que, en términos de la información procesada, puede ser compartida tanto por el robot líder como por el robot seguidor. De igual manera, en (Yandún & Sotomayor, 2012)(Yandún & Sotomayor, 2012)(Yandún & Sotomayor, 2012) se desarrolla un sistema que controla los desplazamientos de un robot móvil SRV-1 que dispone de un módulo de visión estereoscópica. La aplicación se ejecuta en una computadora que se comunica con el robot a través de Wi-Fi, del cual, para facilitar el diseño, se desarrolla un simulador 3D para determinar las trayectorias del robot.

También, en (Bonilla, 2009)(Bonilla, 2009)(Bonilla, 2009), se presenta el diseño e implementación de un sistema de visión por computadora y control difuso

orientado a la navegación autónoma de un robot móvil con capacidades de localización y mapeo simultáneo en un ambiente controlado. Asimismo, se describe el algoritmo de navegación implementado para el desempeño de diversas tareas, tales como la búsqueda de la ruta más cercana en base al algoritmo de Dijkstra, la detección y evasión de los obstáculos presentes en su entorno y la alineación en el desplazamiento de la ruta basada en visión y lógica difusa. En este caso, el sistema se desarrolla en Matlab utilizando una interfaz gráfica para simular la posición y movimiento del móvil, así como las cajas de herramientas de lógica difusa y de adquisición y tratamiento de imágenes.

Otra propuesta donde se presentan dos esquemas de control visual para robots móviles con ruedas, la podemos considerar en (Torres Muñoz O.S., 2014)(Torres Muñoz O.S., 2014)(Torres Muñoz O.S., 2014) utilizando un sistema de visión montado sobre el robot, donde los esquemas de control presentados se basan en la estimación de una restricción geométrica impuesta entre un conjunto de imágenes de una misma escena donde en particular se trabaja con el tensor trifocal porque es la restricción geométrica más general y además permite desarrollar sistemas de control capaces de funcionar en cualquier tipo de escenas y no únicamente en escenarios planos como en el caso de la homografía.

En el ámbito del control de robots móviles con visión estereoscópica, el trabajo de referencia presenta un sistema integral que aborda tanto el hardware como el software (Fernández Lancha et al., 2010);(Fernández Lancha et al., 2010);(Fernández Lancha et al., 2010);la aplicación, ejecutada en una computadora que se comunica con el robot a través de Wi-Fi, destaca por su enfoque flexible y coordinado mediante la metodología OpenUp. El sistema comprende componentes de hardware como el robot móvil SRV-1, la computadora y la conexión Wi-Fi, mientras que la parte de software incluye una aplicación desarrollada en .NET Framework, implementada en C# con Visual Studio y haciendo uso del motor gráfico TrueVision 3D. Este sistema realiza la planificación de trayectorias, la simulación 3D con obstáculos y la captura de imágenes mediante visión estereoscópica. La implementación de algoritmos descrita en (Hart et al., 1968)para la planificación de trayectorias destaca la capacidad de evitar colisiones, mientras que la comunicación

bidireccional con el robot permite la ejecución de instrucciones en tiempo real. Además, se incorpora la simulación de GPS para establecer un sistema de referencia y se aprovecha el motor gráfico TrueVision 3D para una representación realista en la simulación.

1.1.3 Topologías

Tomando en consideración los diferentes tipos o topologías, en (Armando et al., s/f)(Armando et al., s/f)(Armando et al., s/f)se reportan los principales cuidados que se deben de tener para la operación adecuada de un robot móvil diseñado y construido para el transporte de materiales. De igual manera, se muestra que la maniobrabilidad del vehículo es adecuada para su teleoperación en los ambientes por donde la carga irradiada se debe hacer circular, utilizando para ello los controles que han sido designados de un control remoto de uso general.

1.1.4 Herramientas de control y sector de aplicación

Es importante hacer énfasis en cada una de las herramientas de control que han sido implementadas a lo largo de las diferentes propuestas aquí presentadas, así como el sector de aplicación al que va dirigido; se observa que los dispositivos de control aplicados son el Microcontrolador Atmel ATMEGA 16, con implementación a través de una pequeña computadora o sistema embebido, específicamente una Raspberry PI que cuenta con características más robustas para procesamiento en general.

1.1.5 El ángulo θ para la navegación de vehículos autónomos.

En la navegación de vehículos autónomos, el ángulo θ representa la orientación angular del vehículo respecto a un marco de referencia fijo. La correcta estimación de este ángulo es fundamental para determinar la *pose* del robot, que incluye tanto su posición como su orientación en el espacio. Esta información es crítica para la

navegación precisa, permitiendo al vehículo ajustar su dirección en función de su entorno y trayectorias planificadas(Titterton & Weston, 2004). Sin embargo, la precisión en la estimación del ángulo θ enfrenta desafíos debido a las limitaciones inherentes a los sensores utilizados para la medición, especialmente los giroscopios.

Los giroscopios son sensores clave para medir la tasa de cambio angular, siendo ampliamente utilizados en la obtención del ángulo θ . A pesar de su utilidad, están sujetos a ciertos tipos de ruido que pueden afectar la precisión de las mediciones. Entre ellos se encuentran el *Angle Random Walk* (ARW), que corresponde a un ruido aleatorio que provoca una deriva acumulativa en la orientación angular con el tiempo, y el *Rate Random Walk* (RRW), que es un tipo de ruido a largo plazo que afecta la estabilidad de la tasa de giro medida, comprometiendo la precisión de la estimación (Titterton & Weston, 2004).(Grewal et al., 2001).

Para mejorar la estimación del ángulo θ , se han desarrollado diferentes métodos que combinan la información de varios sensores, como acelerómetros y giroscopios, a través de técnicas de fusión de datos. Uno de los métodos más utilizados es el *filtro complementario*, el cual se basa en la combinación de las mediciones de un acelerómetro, que es confiable a largo plazo pero sensible al ruido de alta frecuencia, y un giroscopio, que ofrece precisión en el corto plazo pero sufre de errores acumulativos debidos a ARW y RRW. Este método proporciona una solución simple pero eficaz para mejorar la precisión en la estimación del ángulo θ mediante la fusión de las señales de ambos sensores (Madgwick et al., 2011).

Otra técnica ampliamente utilizada es el *filtro de Kalman*, que es una herramienta matemática de estimación óptima que integra la información de múltiples sensores y modela la dinámica del sistema para reducir el impacto del ruido. En robótica, el filtro de Kalman es efectivo para mejorar la precisión de la estimación de la pose del robot, especialmente en sistemas no lineales, lo que lo hace adecuado para la navegación de vehículos autónomos(Welch & Bishop, 1995),(Sabatini AM., 2005). Su capacidad para adaptarse a la variabilidad en los modelos dinámicos y la

presencia de ruido en las mediciones lo convierte en una opción preferida en el campo de la robótica móvil.

Diversos estudios han evaluado la efectividad de estas técnicas. Por ejemplo, (Madgwick et al., 2011) investigaron el uso de filtros de orientación inercial para mejorar la precisión de la estimación del ángulo, mientras que (Sabatini AM., 2005) se enfocó en la integración de datos de giroscopios y acelerómetros. Estos trabajos han contribuido a avanzar en el campo de la navegación autónoma, proporcionando métodos robustos para la mejora de la precisión en la estimación del ángulo θ .

1.2 Planteamiento del problema.

Los vehículos de guiado automático (AGV) han sido implementados de manera general para sistemas de fabricación flexible al estar concebidos para la realización repetitiva de transporte de material en áreas de gran demanda, garantizando el transporte de materiales de manera ininterrumpida y sin la intervención directa del hombre. Muchas fábricas han adoptado esta tecnología dentro de sus líneas de ensamblaje y producción tales como el sector automotriz, alimentario y agrícola, entre muchas más; sin embargo, la mayoría de los AGV en servicio realizan sus tareas a través de los métodos convencionales de guiado.

Sin embargo, debido a la inherente naturaleza de estos sensores se acumulan errores en el proceso de integración numérica, generando estimaciones inexactas de la posición y orientación del AGV durante su desplazamiento.

Frente a esta problemática, emerge la siguiente pregunta de investigación:

¿Qué herramientas matemáticas, técnicas de modelado y sensores pueden reducir el impacto del ruido en giroscopios MEMS para mejorar la precisión en la estimación de la orientación y posición de un AGV sin guía magnética a lo largo de una trayectoria prefijada?

1.3 Justificación.

- a) El presente trabajo de investigación representa una contribución científica al campo de la navegación autónoma de robots móviles en general; según se vio en la sección 1.1.5, existen actualmente los siguientes problemas abiertos en cuanto al cálculo del ángulo θ . Los sensores utilizados para medir ángulos, como giróscopos y acelerómetros, suelen tener errores asociados (ruido, deriva, etc.), lo cual introduce incertidumbre en el cálculo de θ .
- b) Para mejorar la precisión de la navegación, suele combinarse la información de varios sensores, como GPS, IMU y cámaras. Sin embargo, la fusión de estos datos a menudo resulta en errores acumulativos en el cálculo de θ .
- c) Es común que los sistemas de navegación autónoma requieran una calibración precisa de los sensores para obtener una referencia exacta del ángulo inicial. La mala calibración puede causar errores constantes en θ .
- d) Factores como el viento, las pendientes o el contacto irregular del terreno pueden afectar la orientación del robot, alterando los valores de θ y, por ende, la precisión en la navegación.

Aquí se trata el problema de la determinación del ángulo de pose θ de un AGV tipo triciclo; este ángulo, conocido como ángulo yaw, es necesario para el cálculo de las coordenadas del AGV basadas en mediciones odométricas y otros sensores “de a bordo”. En conjunto, la pose del vehículo (x, y, θ) es fundamental para cualquier algoritmo de navegación, ya que para navegar éstos requieren de actualizar constantemente su pose, la cual es procesada por variados algoritmos para generar los comandos de control que ocasionan que un AGV siga una trayectoria predefinida sin contar con guías magnéticas u otros artilugios de posición absoluto provistos por el entorno.

Además de lo anterior, esta investigación tiene impacto en la necesidad de avanzar la tecnología de Vehículos de Guiado Autónomo (AGV) en el contexto industrial, en el sentido de que tales vehículos son utilizados en entornos industriales dependiendo de sistemas de guía física, ya sean guías magnéticas,

ópticas o físicas en el suelo; un avance sustancial en la navegación autónoma como el que puede producir esta investigación puede ser transferido a la industria, permitiendo el desarrollo de AGV independientes de guías, dando mayor flexibilidad a las líneas de producción y muy posiblemente una mayor certidumbre en la ejecución de los recorridos de los AGV transportando materiales entre puntos de manufactura en la planta industrial moderna.

Por otro lado, una aportación a la navegación autónoma es de gran relevancia en diversas áreas, incluyendo la industria de vehículos autónomos, drones, automóviles, embarcaciones y sistemas de monitoreo de personas o deportistas en entornos donde otros sistemas de navegación no pueden aplicarse efectivamente.

La resolución exitosa de este problema tiene un impacto potencial en una amplia gama de aplicaciones y es especialmente relevante en el contexto de la creciente demanda de vehículos autónomos que se espera en las próximas décadas debido a las presiones comerciales y tecnológicas.

1.4 Hipótesis.

Hipótesis de investigación (H_i)

La fusión con filtrado de Kalman de la odometría de terreno con dos mediciones de la rotación angular en el plano dadas por giroscopios MEMS alineados al eje z de un AGV tipo triciclo, genera menor error en la estimación de su orientación angular (θ), y por ende, de la estimación de su pose (x,y,θ) en tiempo real, en el contexto de navegación autónoma.

Hipótesis nula (H_0)

La fusión con filtrado de Kalman de la odometría de terreno con dos mediciones de la rotación angular en el plano dadas por giroscopios MEMS alineados al eje z de un AGV tipo triciclo, no genera menor error en la estimación de su orientación

angular (θ), y por ende, de la estimación de su pose (x,y,θ) en tiempo real, en el contexto de navegación autónoma.

1.5 Objetivos.

1.5.1 Objetivo general.

Diseñar un filtro de Kalman que fusione odometría de terreno y dos sensores giroscópicos en el eje z de un AGV tipo triciclo para minimizar los errores en la estimación del ángulo de rotación θ y el mejoramiento de la pose (x,y,θ) del AGV durante su desplazamiento para atenuar los efectos de angle random walk y bias instability en el cálculo del ángulo de giro yaw de un AGV.

1.5.2 Objetivos Particulares

- I. Comparar y seleccionar de una arquitectura (tarjeta de adquisición electrónica) para un adecuado soporte de tipo electrónico y de computación del sistema general.
- II. Construir de un prototipo de AGV tipo triciclo para cubrir de manera integral las pruebas del sistema tomando en cuenta características tales como la configuración de ruedas que se pretende utilizar, actuadores, sensores y estructura en general.
- III. Implementar un protocolo que permita la adecuada comunicación entre todas las partes del sistema asegurando una correcta respuesta de cada una de sus áreas.
- IV. Revisar bibliografía detallada de algoritmos de determinación del ángulo θ , examinar a detalle las ventajas y desventajas que conlleva su implementación, descomponer a manera de módulos cada una de sus partes para su correcto estudio y, por último, considerar los requerimientos de implementación.

- V. Diseñar un filtro de Kalman que fusione odometría de terreno y las señales de razón de rotación angular de dos dispositivos IMU-MEMS, para su aplicación en el prototipo construido.
- VI. Llevar a cabo pruebas de desempeño para obtener datos que validen, o no, la hipótesis de investigación.

1.6 Conclusiones

A lo largo de este capítulo se ha abordado el uso de técnicas avanzadas de fusión de sensores y control distribuido para mejorar la navegación y el comportamiento de robots móviles en entornos dinámicos. Se ha demostrado que el uso de un filtro de Kalman distribuido modificado permite integrar de manera efectiva tanto la información de estado como la información del entorno, lo que resulta en una mayor precisión en la detección de objetos, la construcción de mapas y la planificación de trayectorias.

Uno de los hallazgos clave fue la capacidad del sistema para adaptarse a condiciones cambiantes y fallos de sensores, mejorando su robustez. Durante las pruebas experimentales con robots reales, el sistema fue capaz de reaccionar de manera eficiente a la presencia de obstáculos, reduciendo su velocidad y recalculando rutas alternativas en tiempo real. Esto permitió un control más seguro y preciso en comparación con otros enfoques más simples de navegación.

En términos de los resultados generales, se comprobó que la combinación de algoritmos reactivos con la capacidad de planificación permitió una navegación autónoma eficiente incluso en entornos no estructurados. Además, la arquitectura del sistema demostró ser fácilmente escalable y adaptable a diferentes configuraciones de sensores, lo que sugiere su viabilidad para aplicaciones más complejas en la industria y la robótica móvil avanzada.

Finalmente, se concluye que el enfoque basado en la fusión de múltiples sensores y la implementación de una arquitectura de control distribuido no solo mejora la precisión en la estimación de la pose del robot, sino que también

proporciona una solución robusta y versátil para la navegación autónoma en escenarios de alta incertidumbre.

Capítulo 2. Marco teórico.

En este capítulo se abordará el marco teórico relacionado con la robótica móvil, destacando las principales características que definen a los robots en función de su entorno de operación y sus sistemas de locomoción. Se explorarán los diferentes tipos de entornos en los que los robots móviles pueden operar, tanto interiores como exteriores, así como la clasificación de estos en entornos estructurados y no estructurados. Además, se analizarán los diversos sistemas de tracción y dirección utilizados en robots móviles, desde configuraciones sencillas como triciclos y unicyclos, hasta robots omnidireccionales con ruedas suecas. Finalmente, se revisarán los modelos cinemáticos utilizados para describir el movimiento de los robots y las diferentes configuraciones de ruedas que permiten optimizar la maniobrabilidad y estabilidad en diversos terrenos.

2.1 Robótica Móvil

La mayoría de los robots móviles poseen características particulares que los hacen aptos para una determinada tarea. Es la tarea misma la que determina en una primera etapa las particularidades estructurales del robot que van desde el tipo de rueda, el sistema de tracción y dirección y la forma física del robot. En una segunda etapa la tarea determinará las características sensoriales del robot.

En general los robots móviles distribuyen sus sistemas de tracción y dirección sobre los ejes de sus ruedas de acuerdo con las exigencias de velocidad, maniobrabilidad y características del terreno. La precisión y rapidez con que el robot móvil debe alcanzar su destino implica tener un sistema de tracción confiable y un sistema de dirección que dé maniobrabilidad al robot. Esta confiabilidad y maniobrabilidad que debe tener el robot móvil, determinan las características del sistema de tracción y dirección, no sólo en lo que respecta a la técnica, sino también al número de ruedas necesarias y al tipo y disposición de éstas para lograr una estructura mecánica estable (Campion, 1996), (Jones, 1993).

2.1.1 Entornos en que opera un robot móvil

Es esta característica la que fija mayores restricciones sobre el robot móvil y se agrupan según el área de trabajo y según los objetos presentes en el entono.

Según el área de trabajo el entono del robot puede ser interior o exterior. Es interior cuando el área de trabajo está claramente definida por paredes y cielorrasos. Además, la iluminación es principalmente artificial. Por el contrario, es exterior cuando el área de trabajo no está claramente delimitada y el tipo de iluminación es principalmente natural. Ejemplos clásicos de entornos interiores y exteriores son los grandes edificios públicos y los parques o bosques, respectivamente (Figura 2-1).



Figura 2-1. robots de interior (amigobot) y exterior (seekur) de activemedia.

Según los objetos presentes el entorno del robot puede ser estructurado o no estructurado. Es estructurado cuando los objetos presentes en el entono son estáticos (no cambian de forma ni de posición) y poseen características físicas particulares (forma, color, etc.) que permiten asociarlos con figuras geométricas conocidas como prismas o cilindros o permiten distinguir unos objetos de otros (puertas abiertas, mesas de trabajo, etc.). En el otro sentido el entrono es no estructurado cuando la asociación entre los objetos del entorno y determinadas características físicas no es viable, cuando el entorno es dinámico (que cambia con el transcurrir del tiempo) y estos cambios, además, pueden ser impredecible. Por ejemplo, una biblioteca es un entorno estructurado, debido al poco movimiento de personas y a que las estanterías están siempre en los mismos lugares. Por el contrario, un depósito en una línea de producción es un entorno no estructurado, fundamentalmente porque el entorno cambia a medida que se reciben y envían los pallets.



Figura 2-2. Entorno estructurado (biblioteca) y no estructurado (almacén en línea de producción).

2.1.2 Sistemas de locomoción

El sistema de locomoción es una de las primeras características de un robot que está condicionada por su entorno. De acuerdo con las características de este el robot puede ser terrestre.

- Con patas (Figura 2-3a).
- Con ruedas (Figura 2-3b).
- Con cadenas (Figura 2-3c)



Figura 2-3. Robots terrestres. (a) hermes (robosoft). (b) journey (smu). (c) scour (usf).

O también

- Acuático (UUV) flotante.
- Submarino (Figura 2-4(a))
- Aéreo (UAV) (Figura 2-4(b))

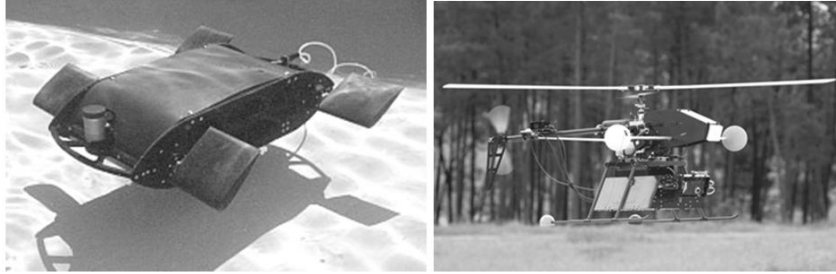


Figura 2-4. Robot acuático (nsf) y robot aéreo (tu Berlín)

2.1.3 Tipos de ruedas

Entre los robots terrestres con ruedas su movilidad está caracterizada por dos factores: el tipo de ruedas que poseen y su disposición sobre una estructura mecánica.

Se asume que, durante el movimiento el plano de la rueda se mantiene vertical y que las ruedas rotan alrededor de su eje (horizontal), que tiene una orientación con respecto a la estructura que puede ser fija o variable.

Se distinguen entre dos clases básicas de ruedas: la rueda convencional y la rueda sueca (swedish wheel). En ambos casos, se supone que el contacto entre la rueda y el terreno se reduce a un único punto del plano.

Para una rueda convencional el contacto entre la rueda y el terreno se supone que satisface la rotación pura sin resbalamiento. Esto significa que la velocidad del punto de contacto es igual a cero (tanto la componente paralela como la componente perpendicular al plano de la rueda).

A su vez entre las ruedas convencionales se distinguen tres tipos:

Rueda fija: El eje de la rueda está fijo a la estructura del robot (Figura 2-5.a). En general está asociada al sistema de tracción del robot.

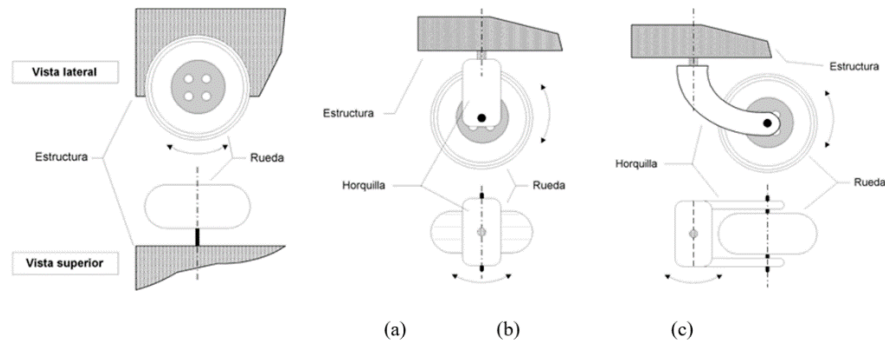


Figura 2-5. Tipos de ruedas. (a) rueda fija. (b) rueda orientable centrada. (c) rueda loca.

Rueda orientable centrada: Es aquella en la que el movimiento del plano de la rueda con respecto a la estructura es una rotación alrededor de un eje vertical que pasa a través del centro de la rueda (Figura 2-5 b). Suele cumplir funciones como rueda de dirección o como rueda de tracción-dirección.

Rueda orientable no-centrada (rueda loca): También conocida como rueda castor (castor wheel) es una rueda orientable con respecto a la estructura, tal que la rotación del plano de la rueda es alrededor de un eje vertical el cual no pasa a través del centro de la rueda (Figura 2-5 c). Su principal función es la de dar estabilidad a la estructura mecánica del robot como rueda de dirección.

Para una rueda sueca (swedish wheel), solo una componente de la velocidad del punto de contacto de la rueda con el terreno se supone igual a cero a lo largo del movimiento (Figura 2-6). Esto le permite, gracias a los rodamientos montados en la superficie de la rueda, desplazarse en dirección perpendicular al plano de la rueda.

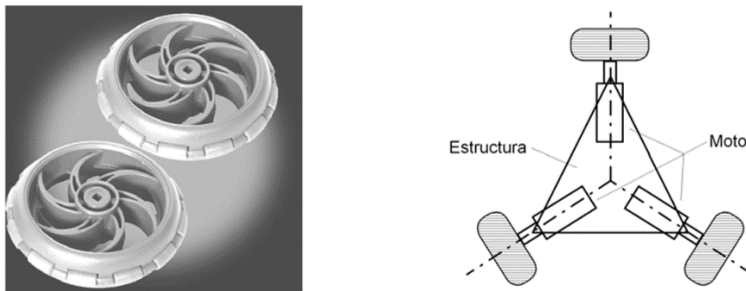


Figura 2-6. Detalle de una rueda sueca y su disposición sobre una estructura mecánica.

2.1.4 Disposición de las ruedas

La combinación de los diversos tipos de ruedas lleva a tener una gran variedad de robots móviles que se diferencian por su grado de maniobrabilidad. A continuación, se presentan brevemente las principales características de diseño de diversos tipos de robot móviles. Esta descripción no pretende ser exhaustiva, sin embargo, se la ha limitado a ejemplos concretos de estructuras no singulares y no redundantes.

2.1.5 Robot omnidireccional

Estos robots tienen máxima maniobrabilidad en el plano; esto significa que ellos pueden moverse en cualquier dirección sin necesidad de reorientarse (Figura 2-7 e Figura 2-8). En contraste, los otros tipos de robots tienen una maniobrabilidad restringida.

La Figura 2-7 muestra el robot omnidireccional Uranus desarrollado en la Universidad de Michigan y el conjunto de movimientos posibles que el mismo puede desarrollar. De acuerdo con la rotación de cada una de las ruedas el robot puede avanzar, girar o desplazarse lateralmente sin necesidad de reorientarse.

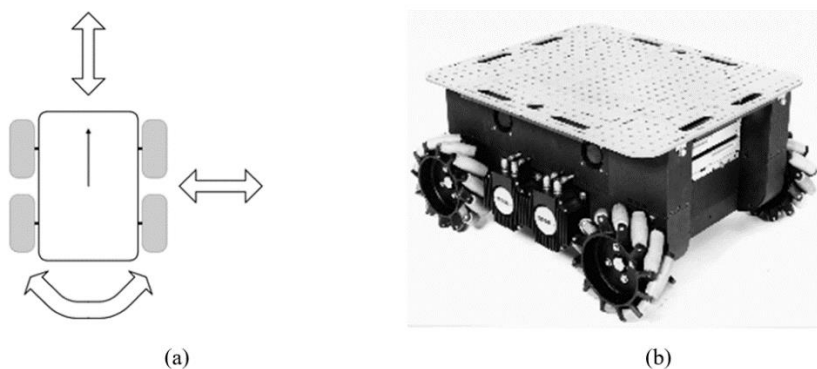


Figura 2-7. Robot omnidireccional con ruedas suecas. (a) maniobrabilidad. (b) robot uranus (universidad de Michigan).

En la Figura 2-8 se presenta otra versión de robot omnidireccional con ruedas orientables centradas. En este caso el robot puede cambiar la dirección de su movimiento simplemente cambiando la orientación de las ruedas. Este movimiento sincronizado se puede lograr por medios mecánicos (Figura 2-8 a) empleando sistemas de tracción y dirección por correas o por medios electrónicos (Figura 2-8 b) mediante señales de accionamientos simultáneas a partir de la electrónica que comanda cada uno de los motores en las ruedas.

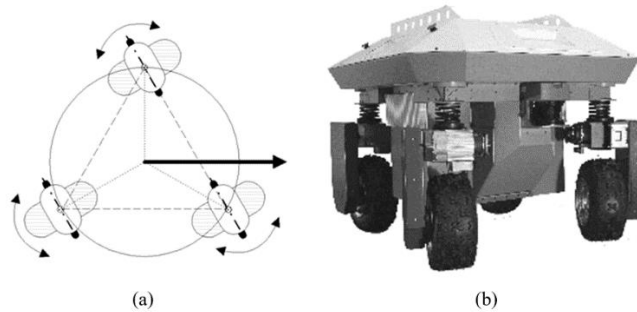


Figura 2-8. Robot omnidireccional con ruedas orientables centradas. (a) disposición sobre una estructura mecánica. (b) robot seekur (activemedia).

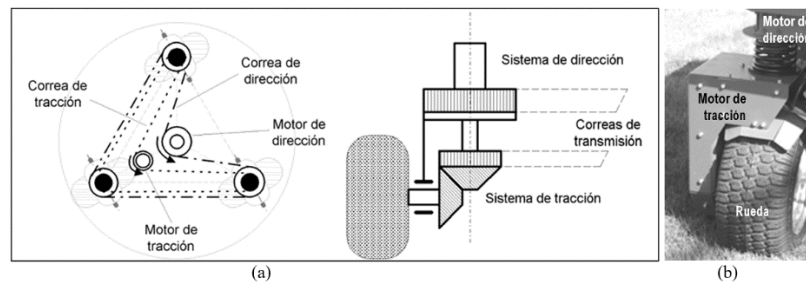


Figura 2-9. Sincronismo entre el sistema de tracción y dirección con ruedas omnidireccionales (synchro drive). (a) mecánico. (b) electrónico.

Las ventajas de un robot omnidireccional se ven desminuidas por la complejidad mecánica y/o electrónicas necesarias para conservar una buena coordinación entre las ruedas y evitar derivas en la pose del robot.

2.1.6 Uniciclo

El robot tipo uniciclo es, en general, el elegido por los investigadores a la hora de probar nuevas estrategias de control por tener una cinemática sencilla. Es una estructura que consta de dos ruedas fijas convencionales sobre el mismo eje, controladas de manera independiente y una rueda loca que le confiere estabilidad (Figura 2-10).

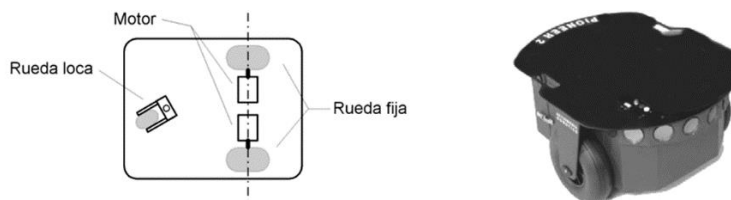


Figura 2-10. (a) Estructura. (b) Robot pioneer (activemedia).

El sistema de tracción-dirección asociado al robot le permite independizar las consignas de velocidad lineal y angular respectivamente. Las ventajas que se derivan de la estructura mecánica y de la electrónica de control hacen de esta configuración la preferida para robots de laboratorio.

2.1.7 Triciclo

El robot tipo triciclo está formado por dos ruedas convencionales fijas sobre un mismo eje y una rueda convencional centrada orientable que concentra las funciones de tracción-dirección (Figura 2-11).

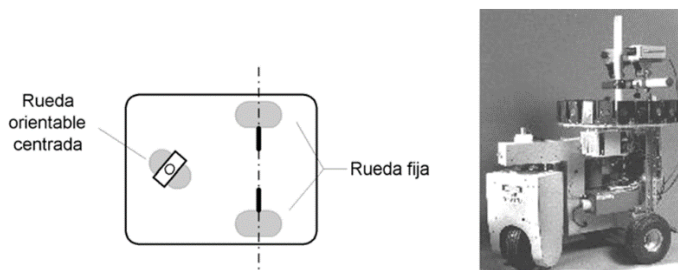


Figura 2-11. Triciclo. (a) Estructura. (b) Robot neptune (universidad carnegie mellon).

Al igual que en el caso anterior, la estructura mecánica y la electrónica de control son sencillas, su tratamiento cinemático resulta de interés en áreas específicas del

control de robots móviles. En cuanto a aplicaciones industriales esta configuración es apta para el transporte de cargas pesadas a baja velocidad.

2.1.7.1 *Parámetros involucrados en el modelo*

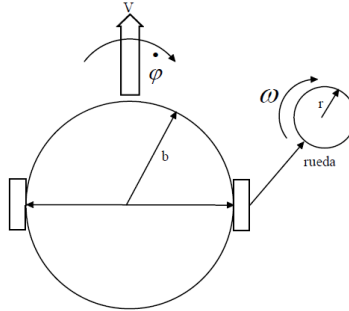


Figura 2-12. Robot Móvil de 2 ruedas.

La Figura muestra tanto los parámetros físicos del robot, como las variables dinámicas. Los parámetros físicos del robot son:

- La longitud del eje (o distancia entre las ruedas) denominada **2b**.
- El radio de cada rueda **r₁** y **r₂**.
- La masa del cuerpo del robot, **M**.
- La masa de cada una de las ruedas, **m**.

Las variables relacionadas con el movimiento del robot son:

- La velocidad lineal que posee el cuerpo del robot, **V**.
- La velocidad angular que posee el cuerpo, **φ**.
- Las velocidades angulares de las ruedas, **ω₁=θ₁** para la rueda derecha y **ω₂=θ₂** para la rueda izquierda.[1]

Además, entre las variables dinámicas se encuentra también la posición absoluta del robot en el espacio. Esta posición queda definida por las coordenadas bidimensionales del centro de masa (x, y) y el ángulo entre la dirección de movimiento del robot y el eje X, denominado φ , tal como se observa en la siguiente Figura.

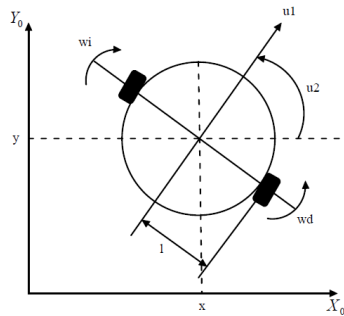


Figura 2-13. Localización en el plano cartesiano del robot.

El modelo también consta de parámetros eléctricos, asociados a los motores DC que se utilizan para entregar la tracción a las ruedas del robot. Para generar el modelo completo, el problema se divide en tres partes: las ecuaciones cinemáticas, las ecuaciones dinámicas y finalmente las ecuaciones eléctricas del sistema del cual este último se detalla en capítulos posteriores.

2.1.7.2 Modelo Cinemático

Los modelos cinemáticos para este tipo de robot son usados dentro del diseño de controladores cuando el robot desempeña tareas o misiones a baja velocidad y con poca carga en relación con su estructura.

Las ecuaciones cinemáticas son aquellas que relacionan la velocidad de giro de cada una de las ruedas con las variables de posición del robot: (x, y, φ).

Considerando al robot como un cuerpo rígido, la velocidad lineal del centro de masa se obtiene a partir del promedio de las velocidades lineales de cada una de sus ruedas. La velocidad lineal de cada rueda se obtiene como el producto de la velocidad angular (velocidad de giro) y el radio de ellas. La velocidad del centro de masa queda definida por:

$$v = \frac{r(\dot{\theta}_1 + \dot{\theta}_2)}{2}$$

El ángulo de giro del robot se determina a partir de las relaciones geométricas entre el movimiento de cada una de las ruedas del robot, tal como se muestra en la siguiente Figura 2-14

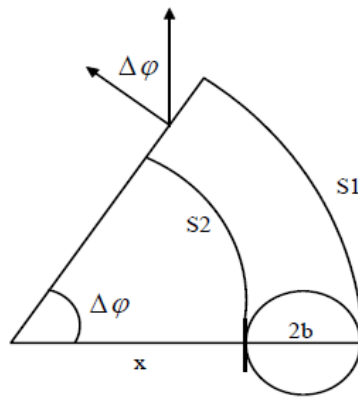


Figura 2-14. Relaciones geométricas para el ángulo de giro.

Según se observa en la Figura 2-11, el ángulo de giro del robot es igual al ángulo del arco sostenido por la trayectoria. Dado que por definición el ángulo de dirección del robot, φ , aumenta en sentido opuesto a las manecillas del reloj, y considerando que la rueda derecha gira a una mayor velocidad que la izquierda, el ángulo de dirección debe aumentar $\Delta\varphi$. Según se aprecia en la Figura 2-11, la rueda izquierda sostiene un arco de radio x , por lo que la distancia recorrida por esa rueda está dada por:

$$S_2 = r\Delta\theta_2 = x\Delta\varphi$$

La rueda derecha se encuentra más lejos del centro de la circunferencia que determina la trayectoria, y recorre una distancia mayor en el mismo tiempo, dada por:

$$S_1 = r\Delta\theta_1 = (x + 2b)\Delta\varphi$$

Calculando el valor de la diferencia S_1-S_2 y dividiendo por el tiempo transcurrido Δt , se obtiene la relación entre la velocidad de giro del robot, y la velocidad de cada una de sus ruedas, como se indica

$$\dot{\varphi} = \lim_{\Delta t \rightarrow 0} \frac{\Delta \varphi}{\Delta t} = \frac{r(\dot{\theta}_1 - \dot{\theta}_2)}{2b}$$

Para obtener las coordenadas de la posición del centro de masa del robot, se requiere primero descomponer la velocidad lineal del robot en las velocidades asociadas a cada eje del plano XY. Esto se presenta en la siguiente ecuación:

$$\dot{x} = V \cos(\varphi)$$

$$\dot{y} = V \sin(\varphi)$$

Para detallar más a fondo se puede representar G que es el centro de masa y se encuentra ubicado a una distancia a por delante del centro del eje que une las ruedas, siendo x y y la posición del punto h respecto al sistema de referencia global, y φ define su orientación respecto a $\{R\}$.

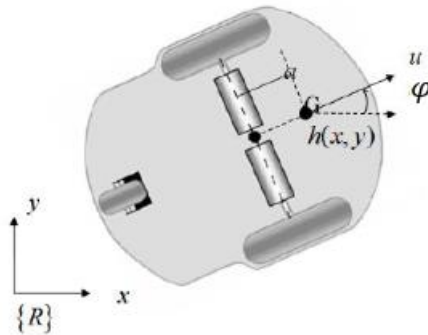


Figura 2-15. Modelo de la estructura cinemática del robot.

El modelo cinemático del robot móvil, considerando la restricción no holonómica, puede ser representado por:

$$\begin{cases} \dot{x} = u \cos(\varphi) - a \omega \sin(\varphi) \\ \dot{y} = u \sin(\varphi) + a \omega \cos(\varphi) \\ \varphi \dot{=} \omega \end{cases}$$

Donde los estados son: x , y , φ y las entradas: u , ω .

Ya se mencionó que los robots móviles se caracterizan por su capacidad de desplazarse de forma autónoma en un entorno desconocido o conocido sólo

parcialmente. Sus aplicaciones cubren una gran variedad de campos y se justifican en la dificultad o imposibilidad de intervención humana, ya sea directa o teleoperada en tiempo real.

El sistema de percepción de un robot móvil permite que este sea capaz de hacer frente a situaciones cambiantes del entorno, como así también de reaccionar ante posibles eventos inesperados mientras navega, lo que exige la utilización de un sistema sensorial que suministre la información del entorno. Esta información del entorno debe ser abundante en calidad y cantidad de tal forma que el robot móvil realice la tarea de manera similar al operador humano. Por ello, un robot móvil rara vez va equipado con un único sensor para realizar todas sus tareas, sino que la práctica más habitual consiste en combinar dentro del sistema sensorial varios sensores que en mayor o menor medida se complementan.

Los sistemas de AGV tienen sus comienzos en 1953, cuando se pensó en hacer realidad un Topología tipo triciclo.

La topología tipo triciclo cuenta con tres ruedas, como se aprecia en la Figura 2-16. Consiste en una rueda delantera utilizada para dar dirección al vehículo y dos ruedas traseras pasivas (o viceversa). La instrumentación puede realizarse en la rueda delantera o en las ruedas traseras, consistiendo en un codificador óptico en el primer caso o de dos codificadores ópticos en el segundo, uno en cada rueda. Cuando se utiliza la rueda delantera como motriz y directora se puede derivar la información de la pose directamente del ángulo de la rueda y de su codificador, o indirectamente mediante odometría diferencial de las ruedas traseras. [6].

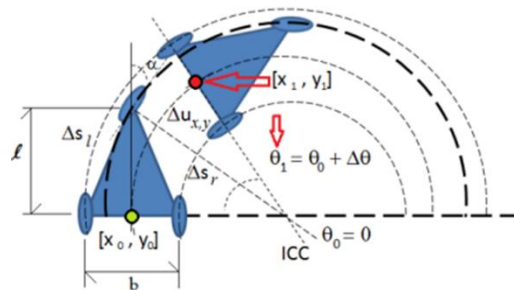


Figura 2-16. Elementos de la cinemática de la configuración tipo triciclo. (Martínez-Romo, 2017).

Los elementos mencionados e identificables en la Figura 2-16 se relacionan entre sí para formar el modelo cinemático:

$$\Delta s_l = \frac{(2\pi r_l) * pl_l}{R} = pl_l * k_1 \quad \text{Ec. (1)}$$

$$\Delta s_r = \frac{(2\pi r_r) * pl_r}{R} = pl_r * k_2 \quad \text{Ec. (2)}$$

$$\Delta \theta = (\Delta s_r - \Delta s_l) * b \quad \text{Ec. (3)}$$

$$\Delta u_{x,y} = (\Delta s_r + \Delta s_l) / 2 \quad \text{Ec. (4)}$$

$$\theta_1 = \left(\theta_0 + \frac{\Delta \theta}{2} \right) \quad \text{Ec. (5)}$$

$$x_k = x_{k-1} - \Delta u_{x,y} * \sin(\theta_1) \quad \text{Ec. (6)}$$

$$y_k = y_{k-1} + \Delta u_{x,y} * \cos(\theta_1) \quad \text{Ec. (7)}$$

Ecuación 2-1. Modelo cinemático (Martínez-Romo, 2017).

En donde:

pl_l es el número de pulsos de codificador óptico de la rueda trasera, lado izquierdo, medido entre intervalos de muestreo.

pl_r es el número de pulsos de codificador óptico de la rueda trasera, lado derecho, medido entre intervalos de muestreo.

k_1 es la constante de compensación odométrica por el error de medición del radio efectivo de la rueda izquierda

k_2 es constante de compensación odométrica por el error de medición del radio efectivo de la rueda izquierda

R , es la resolución del codificador óptico en pulsos por revolución (Martínez-Romo, 2017).

2.1.8 Cuatriciclo

Un problema asociado con la configuración tipo triciclo es que el centro de gravedad del vehículo se posiciona, en algunas ocasiones, en los límites de la superficie de equilibrio, definida por las tres ruedas, cuando el vehículo está en movimiento. Esto produce una pérdida de tracción en el vehículo y es fuente de

error a la hora de estimar la posición del robot. Una solución a este problema lo presenta el sistema de dirección Ackerman. Como se observa en la Figura 2-17 los ejes de las dos ruedas frontales se interceptan en un punto C que pertenece al eje común de las ruedas traseras. El lugar de los puntos en el plano trazados por cada rueda, alrededor de este punto C, es un conjunto de arcos concéntricos donde todos los vectores velocidad instantánea son tangente a estos arcos. Esta estructura, además de brindar mayor estabilidad, evita el deslizamiento en las ruedas y por lo tanto reduce los errores de odometría.

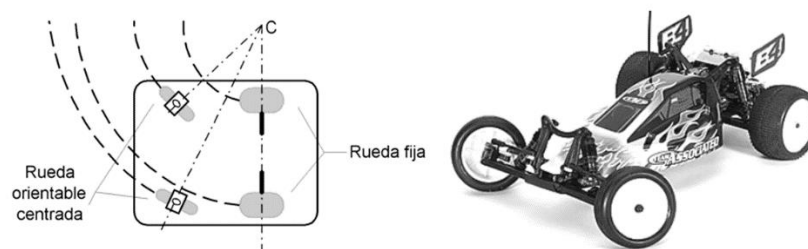


Figura 2-17. Sistema de dirección ackerman.

Si bien su cinemática, su estructura mecánica y su electrónica de control no son tan sencillas esta configuración presenta un gran interés para los amantes de los robots todo terreno en donde los principales desarrollos se producen en la recolección de información sensorial y su posterior tratamiento para lograr reconstrucción de entornos o para el control en tiempo real.

2.1.9 Tracción y dirección

Otros de los aspectos a tener en cuenta en el diseño de un robot móvil es el sistema de tracción y dirección a emplear. El sistema de tracción y dirección no sólo está relacionado con la disposición de ruedas adoptada, sino que también lo está con los algoritmos de control local de los motores y la mecánica asociada a estos. A medida que se requiere, del robot, más confiabilidad (alta maniobrabilidad, máxima tracción en sus ruedas motorizadas, máxima adherencia de todas sus ruedas, etc.) la mecánica, electrónica e informática asociadas es más compleja.

Existen tres sistemas básicos a partir de los cuales se pueden obtener diversas configuraciones:

- Tracción y dirección en ejes independientes (Adamowski, 1990).
- Tracción y dirección en un mismo eje (Cox, 1991).
- Tracción y dirección sobre todos los ejes (Connell, 1990).

Los que se desarrollan a continuación.

2.1.9.1 Tracción y dirección en ejes independientes.

La tracción se efectúa en las ruedas traseras y el control de dirección en las ruedas delanteras o viceversa (Figura 2-18). Si bien el control de dirección es más sencillo, la precisión en la dirección depende de la adherencia de las ruedas correspondientes; esto se debe básicamente a la masa despreciable de estas ruedas respecto al resto de la estructura. Además, posee un radio de giro bastante elevado en relación con otros sistemas, por lo que en este tipo de modelos no se pueden lograr cambios de dirección muy cerrados.

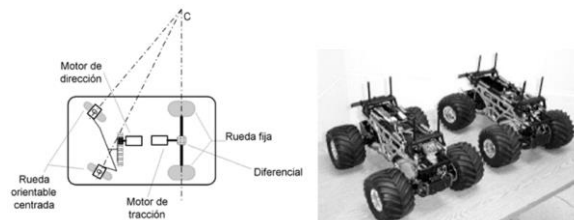


Figura 2-18. Sistema de tracción y dirección en ejes independientes.

2.1.9.2 Tracción y dirección en un mismo eje (Tracción diferencial).

Esto se logra con motores independientes en las ruedas de un mismo eje y ruedas "locas" en el resto de los ejes (Figura 2-19). Este modelo es de construcción sencilla y permite radios de giro del orden del tamaño del vehículo. La única desventaja que posee es que los motores deben ser de características idénticas, para que el control de estos sea simple.

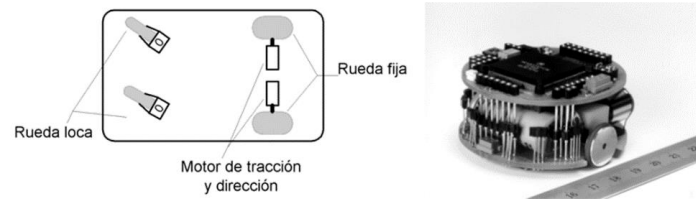


Figura 2-19. Sistema de tracción y dirección sobre un mismo eje.

2.1.9.3 Tracción y dirección sobre todos los ejes.

En la Figura 2-20 se muestra la estructura y un modelo comercial de ActiveMedia. Su aplicación está destinada a terrenos hostiles, donde la velocidad de traslación es menos importante que una buena adherencia al terreno.

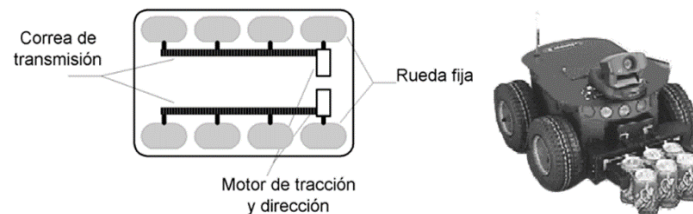


Figura 2-20. Sistema de tracción y dirección sobre todos los ejes.

Esta configuración necesita de un sistema odométrico complejo debido a la incertidumbre en los radios de giro asociada a este sistema de tracción y dirección, aunque existen entre los robots omnidireccionales estructuras que presentan menor complejidad para resolver los errores por odometría.

2.1.10 Configuraciones especiales

En muchos casos el robot tiene alguna característica que lo hace distintivo de los robots convencionales. Algunas configuraciones particulares permiten reducir considerablemente los errores asociados a los deslizamientos en las ruedas como el robot de múltiples grados de libertad con vínculo elástico (Figura 2-21) desarrollado en la Universidad de Michigan, que permite el estudio de los vehículos de transporte de cargas.

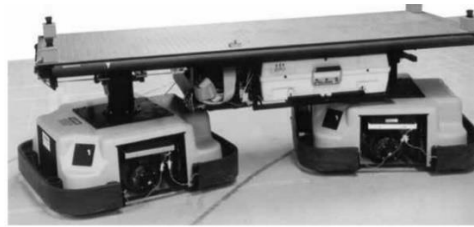


Figura 2-21. Robot de múltiples grados de libertad desarrollado en la universidad de michigan.

Otras configuraciones permiten que pequeños robots individuales se “asocien” para realizar tareas de mayor envergadura que ellos mismos. Este es el caso de los robots colaborativos desarrollados entre Alemania y China.

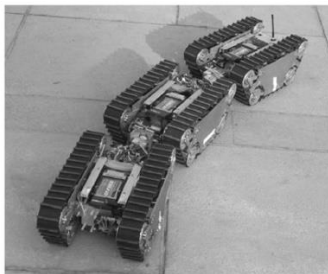


Figura 2-22. Módulos cooperativos (universidad de hamburgo y beihang).

Otros robots son desarrollados con formas especiales para realizar tareas específicas como los robots de exploración espaciales (Figura 2-23).

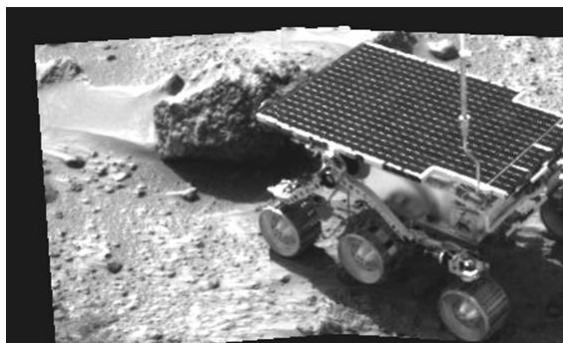


Figura 2-23. robot de exploración espacial pathfinder (jpl).

2.1.11 Tipos de sensores en robótica móvil

Si bien la variedad de sensores que pueden emplearse en un robot móvil está directamente relacionada con el campo de aplicación de éstos, hay una serie de sensores que se pueden considerar más acordes con las funciones del robot y a ellos nos vamos a referir. Para el estudio de los sensores en robótica se los ha clasificado de diversas formas, empleando diversos criterios (Leopoldo Calderón Estévez, 1985) (ver Tabla 2-1).

Los sensores pueden clasificarse utilizando como criterio la interacción sensor-objeto, atendiendo a que la información pueda ser obtenida por contacto directo entre el entorno u objeto y la parte sensorial del robot, o sin mediar contacto físico alguno. De esta manera los sensores podrán clasificarse de contacto y de no contacto.

Tabla 2-1. Clasificación de los sensores en robótica.

Sensores en Robótica	Según el medio relativo al robot	Propioceptivos
		Exteroceptivos
	Según el tipo de interacción robot-objeto	Contacto
		No contacto
	Según el tipo de información	Elementales
		Complejos
	Según el principio de funcionamiento	Carga eléctrica
		Radiación luminosa
		Resistencia
		Inductancia
		Capacitancia
		Radiación térmica
		Otros

Destacamos aquí una última clasificación basada en la relación que tiene el robot con el medio. Cuando este medio es el interno del robot, es decir su estructura mecánica, dichos sensores se denominan propioceptivos o internos; en cambio toda información que provenga del entorno del robot corresponde a los sensores denominados exteroceptivos o externos.

Los sensores propioceptivos han sido cronológicamente los primeros sensores utilizados, encontrándose actualmente en la mayoría de los robots. Suministran la información del estado interno del robot: posición, dirección, sentido, velocidad. En contraposición los sensores exteroceptivos determinan el estado del robot en función de la percepción del entorno.

2.1.11.1 Descriptores estáticos y dinámicos

Existe una amplia gama de técnicas empleadas en la medida de magnitudes físicas, pero el universo de captación del sensor en cuanto al tipo de objetos o fenómenos a detectar y extensión de su campo de sensibilidad, determinan las características o prestaciones del dispositivo de medida. La respuesta del transductor puede requerir únicamente la simple presencia del objeto o fenómeno (detección estática) o puede exigir además la detección del movimiento o variación de este (detección dinámica).

Es así como se puede definir el comportamiento del sensor mediante descriptores. Estos descriptores serán estáticos cuando definan el comportamiento del sensor en régimen permanente o serán dinámicos cuando caractericen la respuesta temporal del sensor antes determinados estímulos.

Los siguientes son descriptores estáticos,

- Rango: valores mínimos y máximos para las variables de entrada y salida.
- Exactitud: la desviación de la lectura de un sistema de medida respecto de una entrada conocida.
- Repetitividad: la capacidad de reproducir una lectura con una precisión dada.

- Reproducibilidad: es igual que la repetitividad, pero las lecturas se realizan bajo condiciones diferentes.
- Resolución: la cantidad más pequeña de incremento que puede ser determinada.
- Error: la diferencia entre el valor medido por el sensor y el valor real.
- Linealidad: cuando la respuesta del sensor es muy semejante a $m \cdot x + h$.
- Sensibilidad: es la razón de cambio de la salida frente a cambios en la entrada.
- Excitación: es la cantidad de corriente o tensión necesaria para el funcionamiento del sensor.
- Estabilidad: una medida de la posibilidad de un determinado sensor de mostrar una misma salida en un rango en el que la entrada permanece constante.
- Ruido: la señal que se acopla a la señal de salida que se podría considerar ideal, que hace que la señal esperada difiera de la real y ocasione problemas en los procesadores que la interpretan. En la Figura 2-24 se presentan los descriptores dinámicos

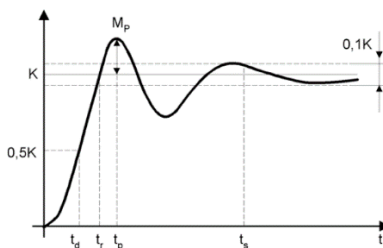


Figura 2-24. Descriptores dinámicos de un sensor.

- Tiempo de retardo, t_d : el tiempo que tarda la señal de salida del sensor en alcanzar el 50% de su valor final.
- Tiempo de crecimiento, t_r : el tiempo que tarda la señal de salida del sensor desde el valor original hasta alcanzar el valor final.
- Tiempo de pico, t_p : el tiempo que tarda la señal de salida del sensor en alcanzar el pico máximo de su sobreoscilación.
- Pico de sobreoscilación, MP : expresa cuánto se eleva la evolución temporal de la señal de salida del sensor respecto al valor final.
- Tiempo de establecimiento, t_s : el tiempo que tarda la señal de salida del sensor en quedar confinada a la banda de 5% alrededor del valor final.

Aparte de los descriptores estáticos y dinámicos, es necesario considerar otros factores que pueden llegar a afectar la respuesta del sensor. Así, por ejemplo, el fabricante suele especificar condiciones ambientales tales como vibraciones, humedad, radiación o rango de temperatura en los cuales se garantiza el correcto funcionamiento.

2.1.12 *Sensores en robots móviles*

Leonard y Durrant-Whyte (Leonard J. and Durrant-Whyte) resumieron el problema de navegación de un robot móvil mediante tres preguntas: “¿Dónde estoy?”, “¿A dónde voy?” y “¿Cómo debo llegar allí?”. La información requerida para responder a estas preguntas debe permitirle al robot móvil realizar tres tareas fundamentales: estimar su posición y orientación, mantener actualizado el mapa del entorno y detectar los posibles obstáculos. Las dificultades para llevar a cabo estas tareas surgen como consecuencia de la muy distinta naturaleza de los procesamiento requeridos en cada una de estas tareas. Así, mientras que para la estimación de la posición y la construcción (o actualización) del mapa del entorno se tienen en cuenta sobre todo características como precisión, resolución espacial,

alcance, etc., en la detección de obstáculos el tiempo entre observaciones normalmente debe ser mucho menor, resultando vital el disponer de la información ya procesada lo más rápidamente posible. En este caso, las características anteriores no son primordiales.

Es por ello por lo que, la condición de que el robot conozca su entorno de trabajo para adaptar su funcionamiento de acuerdo con el estado actual del mismo implica la recolección y procesamiento de información de distinto tipo para utilizarla adecuadamente en el sistema de control. Esta información es generada por los sensores que, según la necesidad particular del trabajo a realizar, podrán variar tanto en número, como en tipo y complejidad (Borenstein J. E., Abril-1996). La calidad y cantidad de información suministrada permitirá controlar al vehículo con trayectorias estables y sin oscilaciones, lo que asegura que el robot móvil alcance el punto destino con el mínimo error sin sufrir choque alguno en el trayecto.

Para que un robot móvil pueda satisfactoriamente afrontar tareas como generar trayectorias, evitar obstáculos, monitorizar la ejecución de la tarea, etc. se requiere que éste sea capaz de determinar su localización o pose (posición y orientación) con respecto a un sistema de referencia absoluto. De forma general, determinar la posición de un robot móvil equivale a encontrar las componentes de translación (x , y , z) y de rotación (ϕ_x , ϕ_y , ϕ_z) del sistema de coordenadas solidario al robot $\{RM\}$ (por tanto, móvil) con respecto a un sistema absoluto $\{R\}$. Aquí sólo se considera el caso bidimensional donde el robot se mueve con tres posibles grados de libertad. Luego el problema se reduce a encontrar la terna (x, y, ϕ_z) asociada al sistema móvil del vehículo, donde (x, y) representan su posición y ϕ_z (ϕ) representa su orientación, tal cual se muestra en la Figura 2-25

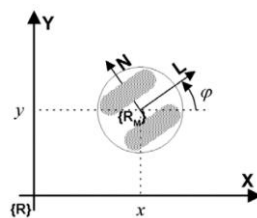


Figura 2-25. Sistema de referencias en un robot móvil.

La mayoría de los robots móviles van provistos de codificadores en los ejes de movimiento que permiten estimar en cada instante la localización de éste empleando un modelo de locomoción. Sin embargo, esta estimación no resulta suficientemente precisa en la mayoría de las aplicaciones. El motivo no es tanto la magnitud de los errores cometidos sino, fundamentalmente, que estos se van acumulando durante la navegación. Esto origina una incertidumbre creciente asociada a la posición y orientación del robot móvil. Debido a esto es necesario acotar los límites de esta incertidumbre mediante un sistema de posicionamiento externo. Estos límites vienen impuestos por el tipo de entorno, la tarea a realizar y la precisión deseada en los movimientos del robot. Es importante destacar que cualquiera sea el sistema de posicionamiento usado la incertidumbre siempre estará presente; la complejidad del sistema de posicionamiento sólo permitirá acotarla en mayor o menor medida.

Una forma resumida, pero inteligente, de presentar los principales sensores y las técnicas asociadas que se emplean en la estimación de la posición y orientación de un robot móvil la dan González Jiménez y Ollero Baturone (González Jiménez, Abril - 1996). En la Tabla 2 se presenta esta clasificación.

2.1.12.1 Estimadores explícitos

Los estimadores explícitos proporcionan la posición y orientación del robot directamente a partir de medidas sin que exista un procesamiento de información para interpretar el entorno.

Dentro de los sistemas de estimación explícita pueden distinguirse dos grupos:

- Estimación basada en medidas internas.
- Estimación basada en estaciones de transmisión

Los primeros trabajan exclusivamente con sensores integrados en el vehículo como codificadores, giroscopios, compases, acelerómetros, etc., y sin ningún tipo de información exterior.

Los segundos, por el contrario, están configurados en base a dos unidades bien diferenciadas. Por un lado, la unidad montada sobre el vehículo y por otro la unidad o unidades externas que deben ser colocadas en posiciones conocidas del entorno.

Tabla 2-2. Posicionamiento de un robot móvil (sensores y técnicas)

Estimadores Explícitos	Medidas Internas	Odometría	- Sensores Doppler - Codificadores ópticos
		Navegación Inercial	- Giroscopios - Acelerómetros
	Estaciones de Transmisión	Fijas	- Ultrasonido - Infrarrojo - Radio frecuencia
		Móviles	
Estimadores basados en la percepción del entorno	Marcas	Artificiales	- Inductivos - Térmicos - Químicos - Infrarrojos - Cámaras de video
		Naturales	- Cámaras de video
	Posicionamiento basado en mapas	Construcción de mapas	
		Técnicas de comparación	- Sistemas ultrasónicos - Sistemas láser - Cámaras de video
		Mapas topológicos y geométricos	

2.1.12.1.1 Estimación explícita basada en medidas internas

La forma más simple de estimar la posición y orientación de un robot móvil consiste en integrar la trayectoria recorrida por éste a partir de una serie de medidas internas: vueltas dadas por las ruedas, velocidades, aceleraciones, cambios de dirección y sentido, etc. En función de la información empleada pueden distinguirse dos grupos:

- Sistemas odométricos.
- Sistemas de navegación inercial

2.1.12.1.2 Sistemas odométricos

La odometría es una técnica antigua que se remonta a la época de Arquímedes y tiene por objeto estimar la posición y orientación de un vehículo a partir del número de vueltas dadas por sus ruedas. La idea fundamental de la odometría es la integración temporal del movimiento, lo cual lleva inevitablemente a la acumulación

de errores. La ventaja de la odometría reside en su simplicidad, bajo costo y en que permite muy altas tasas de muestreo. Sin embargo, además de necesitar una calibración debido al desgaste de las ruedas, esta técnica es vulnerable a las imprecisiones originadas por el deslizamiento de las ruedas, las irregularidades del terreno y las variaciones en la carga transportada.

En la Figura 2-26 se muestran las trayectorias seguidas por las ruedas del robot cuando se produce un desplazamiento del mismo. La rueda izquierda realiza una trayectoria de longitud

$$\Delta x_1 = r_1 \cdot \Delta \phi \quad (1)$$

De la misma forma la rueda derecha describe una trayectoria

$$\Delta x_2 = r_2 \cdot \Delta \phi \quad (2)$$

De tal forma que la trayectoria promedio seguida por el robot (referencial {RM}) tendrá una longitud igual a la semisuma de las dos trayectorias, o sea

$$\Delta x = \frac{\Delta x_1 + \Delta x_2}{2} \quad (3)$$

Y el cambio de orientación se obtiene a partir de la resta de las dos trayectorias y de la separación entre las ruedas, esto es:

$$\Delta \phi \cdot (r_1 - r_2) = (\Delta x_1 - \Delta x_2) \quad (4)$$

$$\Delta \phi = \frac{(\Delta x_1 - \Delta x_2)}{(r_1 - r_2)} = \frac{(\Delta x_1 - \Delta x_2)}{a} \quad (5)$$

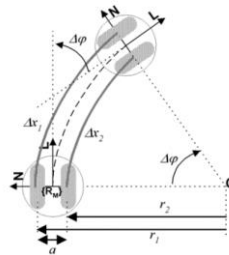


Figura 2-26. Odometría en un robot móvil.

En ciertas aplicaciones es posible reducir los efectos de estos errores empleando sistemas de navegación Doppler, como se muestra en la Figura 2-27 El principio de operación está basado en el desplazamiento de frecuencia observado en una señal cuando la energía radiada se refleja sobre una superficie que está en movimiento con respecto al emisor. El sensor Doppler inclinado a un ángulo β , mide la componente v_D de la velocidad relativa robot- terreno v_A .

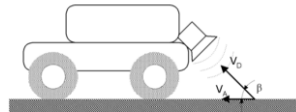


Figura 2-27. Sensor doppler.

Los sistemas marítimos emplean energía acústica mientras que los sistemas aéreo-espaciales emplean energía de radiofrecuencia. Sin embargo, errores en la determinación de la velocidad del vehículo aparecen debido a la interferencia producida por los lóbulos laterales de los sensores, incertidumbre en el ángulo de incidencia y componentes verticales de velocidad introducidas por la reacción del vehículo ante superficies irregulares.

La mayoría de los científicos del área de la robótica móvil coinciden en que la odometría es una parte importante de los sistemas de navegación en robots. En la actualidad es una técnica ampliamente usada en robots móviles y para ello se emplean codificadores ópticos (ver Figura 2-28) de elevada precisión montados sobre los ejes de las ruedas que permiten llevar una cuenta bastante precisa del número de vueltas (y fracción) que estas realizan. Para la estimación se requiere el registro odométrico de al menos dos ruedas del vehículo.

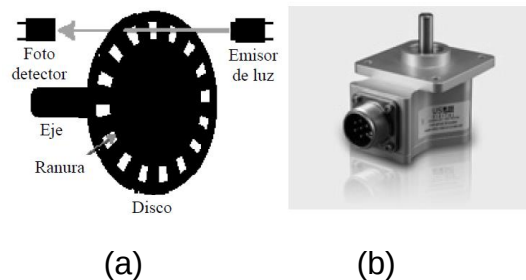


Figura 2-28. Codificador óptico. (a) principio de funcionamiento (b) dispositivo comercial.

2.1.12.1.3 Navegación inercial

Los sistemas de navegación inercial estiman la posición y orientación del vehículo empleando medidas de las aceleraciones y ángulos de orientación.

Los acelerómetros suelen estar basados en sistemas pendulares. La primera integración de las aceleraciones proporciona la velocidad y la segunda la posición. La precisión del acelerómetro resulta crítica debido a la doble integración de las aceleraciones, ya que pequeños errores cometidos por éste repercuten notablemente en la posición estimada. La estimación de la posición se complica más aún cuando la relación señal/ruido es también pequeña debido a que las aceleraciones también son pequeñas.

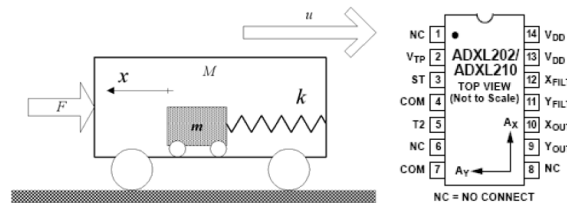


Figura 2-29. Acelerómetro. (a) Principio de funcionamiento. (b) Acelerómetro electrónico.

El acelerómetro es un dispositivo de masa m que se vincula al robot móvil de masa M a través de un resorte de constante elástica k . Una condición para que el acelerómetro funcione correctamente es que $M \ll m$. Cuando el robot móvil se mueve se cumple

$$F = m \cdot a = k \cdot x$$

Donde a es la aceleración del robot móvil y x es la deformación sufrida por el resorte debido a la acción de la fuerza F . Luego se puede obtener el valor de aceleración a , velocidad u y longitud de la trayectoria curvilínea s como:

$$a = k \cdot x / m \rightarrow u = \int a \cdot dt \rightarrow s = \int u \cdot dt$$

Para medir los ángulos de orientación se emplean brújulas o compases magnéticos. Un compás es en esencia un imán con libertad de movimiento para rotar en un plano horizontal. Dada la acción del campo magnético terrestre se

orientará según el meridiano magnético del lugar, indicando la dirección del norte magnético.

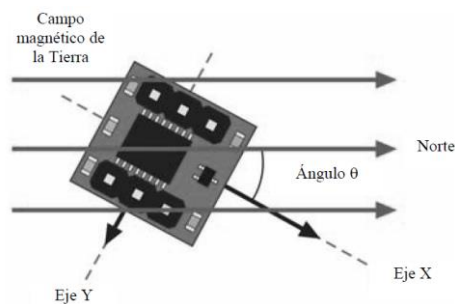


Figura 2-30. Compás electrónico hm55b basado en el efecto hall.

Es posible, también, medir el ángulo de orientación empleando girocompases. Un girocompás es un giróscopo cuyo eje es confinado a permanecer horizontal respecto de la superficie de la Tierra.



Figura 2-31. Girocompás desarrollado por herman anschütz-kaempfe en 1903

Tienen dos ventajas principales sobre las brújulas magnéticas: primero que señalan al norte geográfico, es decir, la dirección del eje de rotación de la Tierra, y no al norte magnético y segundo que no se ven afectados por el metal de la estructura mecánica.

Los acelerómetros y giróscopos se acoplan al vehículo directamente, o bien a través de una base estabilizadora que minimiza los efectos de “cabeceo” de todo el sistema. En el primer caso se requieren medidas adicionales de los ángulos de balanceo (mediante inclinómetros) para realizar la correspondiente corrección de la lectura del acelerómetro.

Los sensores de orientación son de particular importancia en el posicionamiento de robots móviles debido a que ellos pueden ayudar a compensar el problema más

notable del sistema odométrico: en cualquier método de posicionamiento basado en odometría, cualquier pequeño error de orientación transitorio producirá un error de dirección constante y creciente. Por esta razón, será de gran beneficio si el error de orientación puede ser detectado y corregido inmediatamente.

A diferencia de los sistemas odométricos, los sistemas de navegación inercial no se ven afectados por los problemas derivados de la interacción del vehículo con el terreno y pueden corregir los efectos de las ondulaciones e irregularidades del terreno. En la práctica estos sistemas son más confiables y precisos que los sistemas odométricos, aunque son mucho más frágiles y caros que éstos.

Ambas técnicas son sencillas de implementar, pero presentan un grave inconveniente como estimadores de la posición absoluta: los errores son acumulativos y la incertidumbre en la posición crece proporcionalmente al espacio recorrido. Es por ello por lo que ambos sistemas son complementados con algún otro sistema de posicionamiento absoluto que reduzca periódicamente dicha incertidumbre.

2.1.12.1.4 Estimación explícita basada en estaciones de transmisión

Este tipo de sistemas de posicionamiento absoluto, también conocidos como balizas de radio- frecuencia (RF), se emplea desde hace varias décadas en aplicaciones marítimas y aeronáuticas, si bien su uso se remonta a varios siglos: Las estrellas fueron utilizadas desde muy antiguo como un sistema de referencia para estimar la posición y más tarde los faros construidos por el hombre. En los últimos años, con la aparición de robots móviles para exteriores, estos sistemas han cobrado un especial interés dentro de la comunidad científica vinculada a esta área. La ventaja de esta técnica consiste en proporcionar la localización absoluta del vehículo en un área suficientemente grande sin requerir estructuración alguna del entorno. Estas características hacen, a estos sistemas, especialmente adecuados para ser empleados en aplicaciones donde el vehículo ha de moverse en entornos muy diversos y donde debe recorrer grandes distancias.

La configuración de estos sistemas está basada en un receptor (o un trancceptor) a bordo del vehículo y un conjunto de estaciones transmisoras de RF ubicadas en lugares conocidos distantes del vehículo.

Los avances tecnológicos han mejorado enormemente la capacidad de los sistemas basados en estaciones de transmisión con la incorporación del láser, ultrasonido y los transmisores de RF. Es importante notar que ninguno de los sistemas de RF puede ser empleado con certeza en interiores.

Se pueden distinguir dos grupos fundamentales:

Sistemas de posicionamiento mediante estaciones fijas. Sistemas de posicionamiento mediante estaciones móviles.

b.1) Estaciones fijas

Los sistemas de posicionamiento mediante estaciones fijas son básicamente de dos tipos:

Triangulación: Conocidos como métodos de navegación hiperbólica o pasivos. Este método fue desarrollado durante la Segunda Guerra Mundial y la implementación original se denominó LORAN (del inglés LOng RAnge Navigation) navegación de largo alcance. Este sistema compara los tiempos de llegada de dos señales idénticas transmitidas simultáneamente desde transmisores de alta potencia localizados en sitios de coordenadas conocidas como se muestra en la Figura 2-32

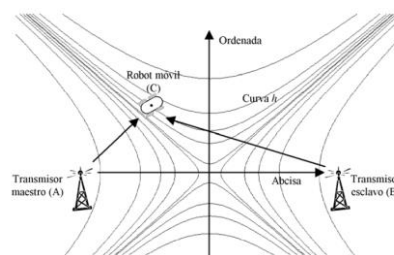


Figura 2-32. Para cada punto sobre una línea hiperbólica, la distancia ($abc - ac$) = k , siendo k constante.

El transmisor maestro envía una señal de identificación al robot móvil y a otras estaciones transmisoras que ofician de esclavos. Cuando éstas reciben la señal del maestro la retransmiten añadiendo su propia señal de identificación que también es

recibida por el robot móvil, pero un instante de tiempo después. La diferencia de tiempo entre la recepción de las dos señales en el robot móvil se corresponde directamente con las distancias entre las estaciones y el robot móvil y se asocian a una constante h que se relaciona a una curva hiperbólica en la cual se encuentra posicionado el robot móvil cuyos focos son las estaciones de transmisión, es decir:

$$(\overline{AB} + \overline{BC}) - (\overline{AC}) = h$$

Para evitar ambigüedades en el posicionamiento del robot tres o más estaciones de transmisión son necesarias para encontrar el punto de intersección de las líneas hiperbólicas.

La versión más moderna es LORAN-C que funciona en frecuencias del espectro electromagnético entre 90 KHz y 100 KHz. Aunque su uso ha decaído notablemente debido a la aparición del GPS no ha perdido vigencia como sistema alternativo de posicionamiento debido a la cobertura que el sistema posee actualmente (ver la Figura 2-33). Rusia utiliza un sistema casi idéntico llamado CHAYKA, que usa la misma banda de frecuencias.



Figura 2-33 Cobertura del sistema loran.

Trilateración: conocidos como sistemas activos. Éstos miden el retardo de propagación para un número de transeptores localizados en lugares de coordenadas conocidas como se muestra en la Figura 2-34, Los sistemas activos cubren desde los 100 m. hasta casi los 100 Km con una exactitud de 2 m.

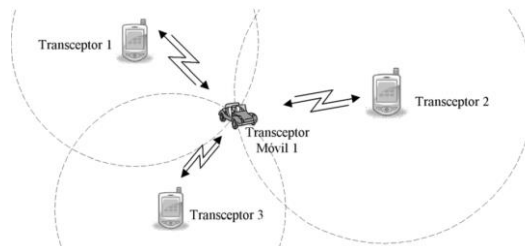


Figura 2-34. Esquema del funcionamiento de un sistema de trilaterización.

El sistema opera de la siguiente manera: El transceptor móvil emite una señal de identificación que es captada por varios transceptores fijos, que la devuelven adicionando su propio código de identificación. El retardo que existe entre el instante en que el robot móvil emite la señal y el instante en que recibe la respuesta de las estaciones fijas determina la distancia que separa al robot de las estaciones fijas. Son necesarias al menos tres estaciones fijas para evitar ambigüedades en la localización del robot móvil.

Los esquemas de trilaterización ultrasónicos ofrecen una exactitud aceptable y son una solución de bajo costo al problema del posicionamiento de robots móviles. Debido al alcance relativamente corto del ultrasonido, estos sistemas son adecuados para operar en pequeñas áreas de trabajo y sólo si no existen obstáculos significativos que interfieran con la propagación de la señal.

Los sistemas de posicionamiento ópticos generalmente involucran algún tipo de mecanismo de búsqueda operando en conjunción con dispositivos de referencia estratégicamente ubicados en lugares de coordenadas conocidas dentro del ambiente de operación del robot móvil. Uno de los principales problemas asociados con estos sistemas es la necesidad de preservar la visibilidad entre el robot y la baliza.

2.1.12.1.5 Estaciones móviles

Los sistemas de posicionamiento mediante estaciones móviles operando desde satélites son hoy en día los de mayor interés para robots móviles. El primer sistema de posicionamiento mediante satélites está basado en el cambio de frecuencia experimentado por las señales de radio transmitidas por éstos, conocido como Efecto Doppler. En la actualidad este sistema ha sido reemplazado por uno mucho más potente denominado GPS (Global Positioning System) cuya estructura se muestra en la Figura 2-35

El GPS (su nombre correcto es NAVSTAR-GPS) emplea una constelación de 24 satélites (incluyendo tres de reserva) orbitando la tierra cada 12 horas a una altura de 20.200 Km. aproximadamente. Cuatro satélites son colocados en cada una de seis órbitas cuyos planos están separados 55° a partir del plano del Ecuador. Cada uno de ellos transmite dos señales de radio en alta frecuencia, moduladas por un pseudo-ruido binario en las que se codifican en forma compleja información sobre el instante en que la señal fue transmitida e información orbital entre otras cosas. Empleando al menos tres satélites, el receptor calcula por trilaterización la altitud, latitud y longitud del vehículo de forma “instantánea y continua” (tiempos entre 30 y 60 nseg.). También puede determinar la velocidad a partir del desplazamiento en frecuencias mediante el efecto Doppler.

Aunque conceptualmente muy simple, esta filosofía de operación del GPS introduce, al menos, cuatro desafíos tecnológicos:

- La sincronización de los relojes entre los satélites individuales y los receptores GPS.
- Localización precisa en tiempo-real de los satélites.
- Medición exacta del tiempo de propagación de la señal.
- Suficiente relación señal-ruido para una operación eficaz en presencia de posibles interferencias.

La precisión del GPS depende de numerosos factores, algunos comunes a otros sistemas de posicionamiento mediante radiofrecuencias (prestaciones del receptor,

inestabilidad en el recorrido de la transmisión, posición relativa del receptor respecto a las estaciones de transmisión, etc.), y otros específicos del GPS (alteración de la velocidad de propagación de la señal de radio, errores en el posicionamiento orbital, etc.). Esta precisión llega a ser inferior a los 20m para vehículos en movimiento e inferiores a 10m para medidas estacionarias. Sin embargo, el empleo del GPS diferencial, basado en el empleo de una estación terrena fija de coordenadas conocidas, está permitiendo incrementar la precisión hasta un par de metros.

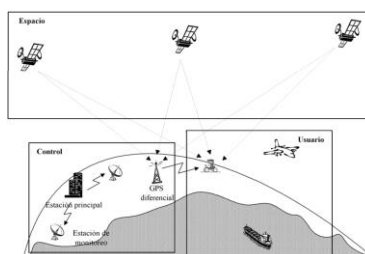


Figura 2-35. El sistema de posicionamiento global (gps) consiste de tres segmentos fundamentales: espacio, control y usuario.

La antigua Unión Soviética tenía un sistema similar llamado GLONASS, ahora gestionado por la Federación Rusa. Actualmente la Unión Europea está desarrollando su propio sistema de posicionamiento por satélite, denominado GALILEO.

2.1.12.2 Estimadores basados en la percepción del entorno

Los estimadores basados en la percepción del entorno emplean sensores que suministran información sobre éste a partir de la cual se infiere la localización del robot móvil mediante comparación de esta información con otros datos o modelo conocido del entorno.

Los sensores empleados en este tipo de estimadores pueden clasificarse en dos grupos:

Sensores activos: Son aquellos que emiten algún tipo de energía al medio (por ejemplo: luz o ultrasonidos). Éstos proporcionan directamente medidas de distancias al entorno. Entre los sistemas más empleados se encuentran los sonares (sensores de ultrasonido) y los scanners (sensores láser).

Sensores pasivos: Se limitan a captar la energía existente en el medio. Entre los sistemas más empleados se encuentran las cámaras de vídeo y los sensores infrarrojos.

Además de las características comunes a otras aplicaciones como la resolución, precisión, alcance, etc., para aplicaciones en robots móviles es necesario tener muy en cuenta otras características del sensor como la inmunidad a las variaciones de las condiciones ambientales, la robustez ante vibraciones, el tamaño, consumo, desgaste y seguridad de funcionamiento.

2.1.12.2.1 Estimación mediante marcas o balizas

En general las marcas (también conocidas como mojones o balizas especiales) son características del entorno de operación que un robot puede reconocer desde sus entradas sensoriales. Aunque puede entenderse que este proceso conlleva la percepción del entorno, la posición no se estima a partir del análisis o interpretación del entorno percibido, sino que es determinado de una forma más o menos directa en base al principio de triangulación, bien a partir de medidas de distancias, de ángulos ó combinaciones de los dos.

Las marcas naturales son aquellos objetos o características propios del ambiente y que tienen una función distinta a la de facilitar la navegación del robot. El principal problema en el posicionamiento mediante marcas naturales es poder detectar y extraer características distintivas del entorno de trabajo a partir de la forma en que se haya estructurado éste. El sistema sensorial por excelencia es la visión computarizada. La mayoría de los sistemas de visión empleados en la navegación mediante marcas naturales tratan de identificar segmentos verticales de longitud apreciable como son los marcos de una puerta, la intersección de paredes u objetos característicos como las fuentes de luz en el cielo raso. Sin embargo, la visión computarizada es un área demasiado amplia y diversa para poder resumirla en unas pocas líneas y sólo se hace referencia a ella en forma superficial.

En el caso de las marcas artificiales, éstas son formas geométricas (rectángulos, líneas, círculos, etc.) que, además, pueden incluir información adicional (por ejemplo, en forma de código de barras), tienen una posición fija conocida en relación

con la cual el robot móvil puede estimar su posición y su único propósito es facilitar la navegación de éste.

Es necesario que el robot móvil conozca las marcas y su posición dentro del área de trabajo. Con el objeto de simplificar el problema de la interpretación de las marcas se considera que la posición y orientación actual del vehículo son conocidas aproximadamente, tal que el robot sólo necesita observar y comparar la información de las marcas en un área limitada. El procedimiento general para la estima de la posición en función de marcas se muestra en la Figura 2-36

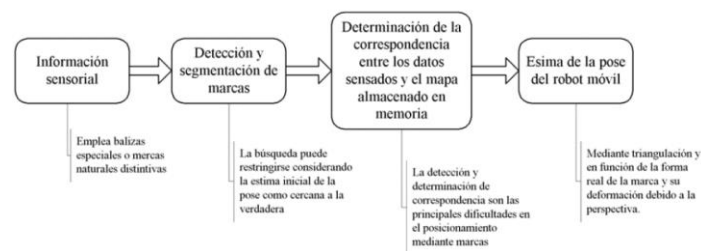


Figura 2-36 Procedimiento general para el posicionamiento mediante marcas.

Si bien puede entenderse que este proceso conlleva la percepción del entorno, la posición no se estima del análisis o interpretación del entorno percibido, sino que es determinado de una forma más o menos directa en base al principio de triangulación, bien a partir de medidas de distancias, de ángulos o combinaciones de las dos. El número mínimo de marcas requeridas dependerá del tipo de sistema empleado, como se muestra en la Figura 2-37

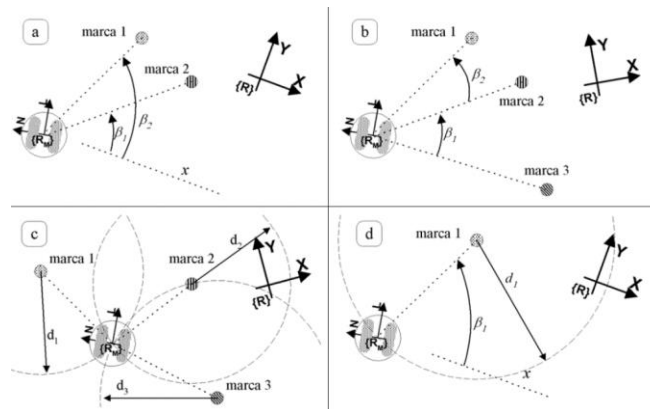


Figura 2-37. Diferentes posibilidades de triangulación. A) ángulos absolutos de referencia. B) ángulos observados entre marcas. C) distancias observadas a las marcas. D) ángulo y distancia a una marca.

Otro tipo de sistema de navegación que ha sido ampliamente usado en la industria es la denominada Línea de Navegación. La Línea de Navegación puede entenderse como una marca continua, sin embargo, en la mayoría de los casos el sensor empleado en estos sistemas necesita estar muy cerca de la línea, tanto que el movimiento del robot está limitado a la inmediata vecindad de la ésta. Existen diversas formas de implementar este sistema que se basan en principios electromagnéticos, ópticos (reflexión), térmicos o químicos (Russell A. , 1995).

La exactitud obtenida mediante estos métodos depende de la confiabilidad de la información extraída de los parámetros geométricos de la marca. Ésta depende fundamentalmente de la posición y el ángulo relativos entre el robot y la marca.

2.1.12.2.2 Posicionamiento basado en mapas del entorno

El posicionamiento basado en mapas del entorno (también conocida como “map matching”) es una técnica en la cual el robot emplea sus sensores para crear un mapa de su entorno local. Este mapa local es luego comparado con un mapa global previamente almacenado en memoria. Si alguna correspondencia entre los mapas es encontrada (comparación de mapas) el robot puede computar su posición y orientación real en el ambiente. El mapa pre- almacenado puede ser un modelo CAD del ambiente, o éste puede ser construido a priori a partir de la información sensorial.

El procedimiento básico para lograr el posicionamiento basado en el mapa del entorno se muestra en la Figura 2-38

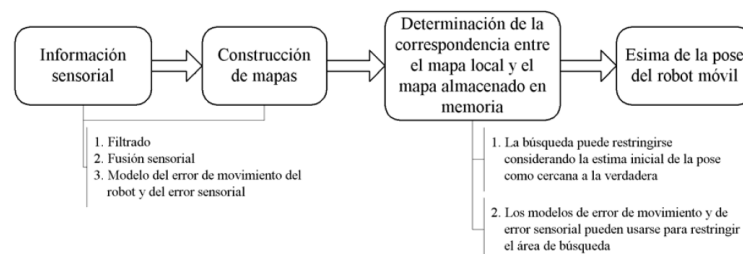


Figura 2-38. Posicionamiento basado en mapas.

La mayoría de los sensores empleados con este propósito involucran algún método de medición de distancia. Básicamente existen tres métodos diferentes:

- Tiempo de vuelo.
- Desplazamiento de fase.
- Desplazamiento de frecuencia.

Las principales ventajas del posicionamiento basado en mapas del entorno son las siguientes:

- Este método emplea la configuración natural de los ambientes interiores en edificios o laboratorios para obtener información acerca de la posición del vehículo sin modificar el área de operación.
- El posicionamiento basado en mapas puede emplearse para generar una actualización del mapa del ambiente.
- El posicionamiento basado en mapas permite al robot conocer un nuevo ambiente y mejorar la exactitud de la estima de su posición a través de la exploración.

Las desventajas del posicionamiento basado en mapas del entorno están relacionadas con los requerimientos especificados para una navegación satisfactoria:

- El entorno debe ser altamente estático y las características de éste deben ser fáciles de reconocer, de tal modo que puedan emplearse para la comparación.
- El sensor empleado para la elaboración del mapa debe ser bastante preciso (en relación con la tarea a desarrollar).
- Es necesaria una importante capacidad de sensado y procesamiento de información.

Es preciso notar que la mayoría de los trabajos actuales en este sentido están limitados a ambientes relativamente simples y a experiencias de laboratorio.

Existen fundamentalmente dos puntos de partida para el proceso de posicionamiento basado en mapas. Uno es la existencia de un mapa almacenado en memoria, y otro es que el robot debe construir su propio mapa del ambiente.

2.1.12.2.3 Construcción de mapas

Rencken (Rencken, 1993) definió el problema de construir el mapa como: “Dados la posición del robot y un conjunto de mediciones, ¿qué están viendo los sensores?”. Obviamente, la habilidad del robot para construir el mapa está totalmente limitada a su capacidad de sensado.

Hoppen (Hoppen) et al. listaron los tres pasos principales a seguir en el procesamiento de la información sensorial para construir el mapa del entorno:

- Extracción de características desde los datos sin procesar.
- Fusión de los datos de varios tipos de sensores.
- Generación automática de un modelo del entorno con diferentes grados de abstracción.

Un problema relacionado con la construcción del mapa es la exploración autónoma. En orden a elaborar un mapa, el robot debe explorar su entorno para mapear áreas no registradas. Habitualmente se supone que el robot comienza su exploración sin tener conocimiento del ambiente. Luego se sigue una determinada estrategia de movimiento dirigida a maximizar la cantidad de área mapeada en la menor cantidad de tiempo. Dicha estrategia de movimiento se llama estrategia de exploración y depende fuertemente del tipo de sensor empleado.

2.1.12.2.4 Técnicas de comparación de datos

Uno de los aspectos más importantes de la navegación basada en mapas es la comparación de datos, es decir, el procedimiento por el cual se establece la correspondencia entre un mapa local actual y el mapa global almacenado en memoria. Los algoritmos empleados en la comparación de mapas pueden ser clasificados como:

- Algoritmos basados en la extracción de características.

- Algoritmos basados en técnicas icónicas.

Los algoritmos basados en la extracción de características resuelven el problema de la estimación de la posición del robot a partir de la extracción de un conjunto de características de cada uno de los conjuntos a comparar, para luego buscar los pares de correspondencia entre elementos de ambos conjuntos de características. Por el contrario, los algoritmos basados en técnicas icónicas trabajan directamente con los dos conjuntos de datos, buscando en este caso la correspondencia directa de sus elementos.

Ambos enfoques presentan ventajas y desventajas:

- Los algoritmos basados en la extracción de características requieren una estructuración de los datos que posibilite la identificación de objetos o formas determinadas (segmentos, círculos, polígonos) mientras que los algoritmos basados en técnicas icónicas comparan directamente los datos sensados.
- Los algoritmos basados en la extracción de características son más rápidos que los algoritmos basados en técnicas icónicas, dado que los primeros sólo comparan sub- conjuntos de características mientras que en el segundo se comparan todos los puntos de los dos conjuntos de datos.
- Los algoritmos icónicos son más precisos que los basados en extracción de características desde que la extracción de características implica una reducción de la cantidad de información.
- Ambos estimadores son robustos ante errores debido a los sensores.

2.1.12.2.5 Mapas topológicos y mapas geométricos

En el posicionamiento basado en mapas hay dos representaciones comunes: Los mapas topológicos y los mapas geométricos. Un mapa geométrico representa los objetos de acuerdo con sus relaciones geométricas absolutas (Moravec, 1985), (Borenstein J. a., 1991) (Figura 2-39 a). Éste puede ser una representación en forma de grilla, o más abstracto aún, un mapa de polígonos y líneas (Crowley, 1989). En contraste, los mapas topológicos se basan en la relación geométrica entre las características observadas más que en su posición absoluta (Courtney, 1994) (Figura 2-39 b). El resultado se presenta bajo la forma de un grafo donde los nodos representan las características observadas y los arcos representan la relación entre las características.

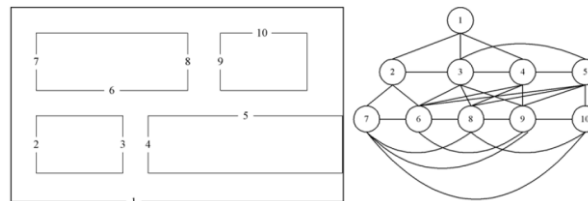


Figura 2-39. Distintos tipos de mapas. (a) geométrico. (b) topológico.

A diferencia de los mapas geométricos, los mapas topológicos pueden ser contruidos y actualizados sin considerar la estima de la posición del robot. Esto significa que cualquier error en esta representación será independiente de cualquier error en las estimas de la posición del robot. Esto permite generar mapas de grandes dimensiones sin preocuparse por los errores odométricos del robot, dado que, todas las conexiones entre nodos son relativas. Luego que el mapa ha sido establecido, el proceso de posicionamiento es esencialmente el proceso de comparación y correspondencia del mapa local con la ubicación apropiada sobre el mapa almacenado en memoria.

Los mapas geométricos a menudo se emplean en esquemas de navegación occidental es decir una navegación basada en una planificación en base a coordenadas geométricas y a trayectorias pre-fijadas.

Los mapas topológicos se emplean en esquemas de navegación polinésica en donde la navegación está basada en objetivos sensoriales que son asociados a patrones de reacción.

Por ejemplo, para que el robot móvil se mueva desde la puerta 2 a la puerta 5 en la Figura 2-39, en el caso de un mapa geométrico la posición inicial del robot móvil debe coincidir con la posición de la puerta 2 y la posición destino será la coordenada de la puerta 5 según el mapa. Para el caso de un mapa topológico la posición inicial es dada por la percepción de la puerta 2 y la posición destino estará dada por el cumplimiento de objetivos intermedios, esto es ubicar y alcanzar la puerta 3, cumplido este objetivo ubicar y alcanzar la puerta 4, cumplido este objetivo ubicar y alcanzar la puerta 5.

2.2 Arquitectura de un AGV.

Los sistemas mecánicos tanto manuales como automatizados que se encuentran en el mercado incluyen una diversidad de elementos y características que los hace costosos por mencionar algunas de estas características: precisión con que fueron maquinados y ensamblados, los motores y controladores, así como también a los mecanismos de transmisión de movimiento.

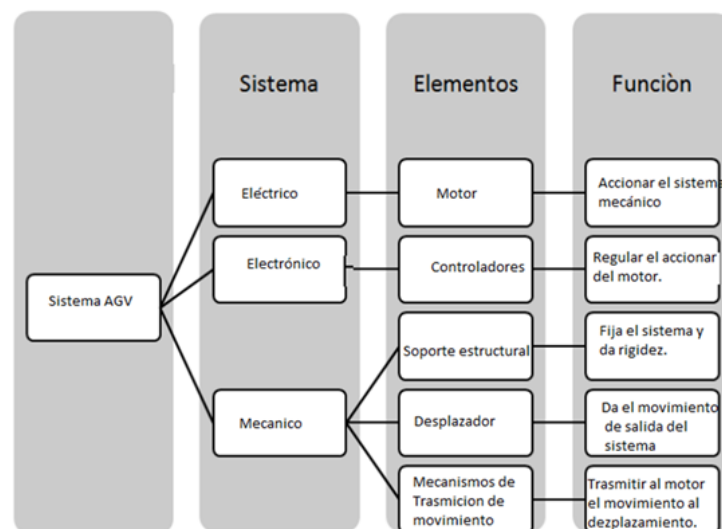


Figura 2-40. Arquitectura del AGV.

2.3 Topología tipo triciclo.

La topología tipo triciclo cuenta con tres ruedas, como se aprecia en la Figura 2-41. Consiste de una rueda delantera utilizada para dar dirección al vehículo y dos ruedas traseras pasivas (o viceversa). La instrumentación puede realizarse en la rueda delantera o en las ruedas traseras, consistiendo de un codificador óptico en el primer caso o de dos codificadores ópticos en el segundo, uno en cada rueda. Cuando se utiliza la rueda delantera como motriz y directora se puede derivar la información de la pose directamente del ángulo de la rueda y de su codificador, o indirectamente mediante odometría diferencial de las ruedas traseras. [6].

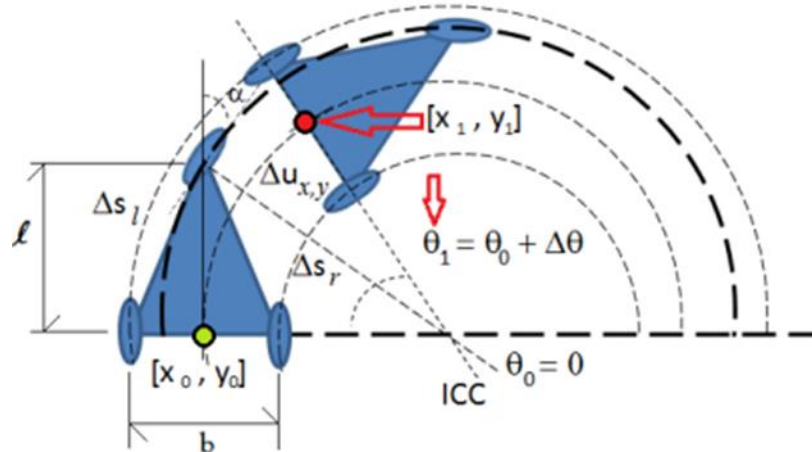


Figura 2-41. Elementos de la cinemática de la configuración tipo triciclo. (Martínez-Romo, 2017).

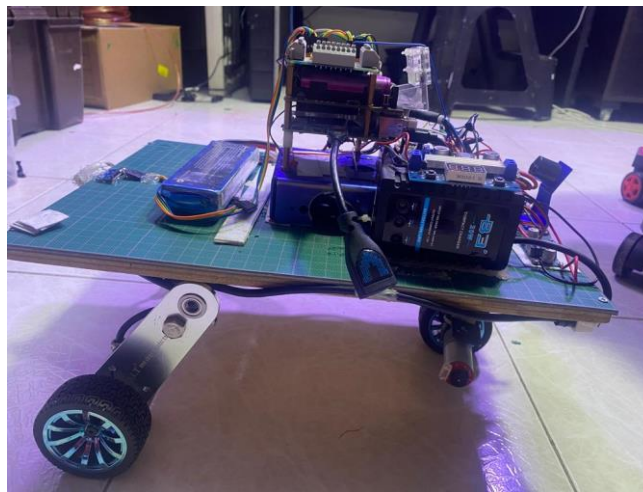


Figura 2-42. Vehículo de guiado automático AGV (Automatic Guided Vehicle).

Los elementos mencionados e identificables en la Figura 2-42 se relacionan entre sí para formar el modelo cinemático:

$$\Delta S_l = \frac{(2\pi r_l) * pl_l}{R} = pl_l * k_1 \quad \text{Ec. (1)}$$

$$\Delta S_r = \frac{(2\pi r_r) * pl_r}{R} = pl_r * k_2 \quad \text{Ec. (2)}$$

$$\Delta \theta = (\Delta S_r - \Delta S_l) * b \quad \text{Ec. (3)}$$

$$\Delta u_{x,y} = (\Delta S_r + \Delta S_l) / 2 \quad \text{Ec. (4)}$$

$$\theta_1 = \left(\theta_0 + \frac{\Delta \theta}{2} \right) \quad \text{Ec. (5)}$$

$$x_k = x_{k-1} - \Delta u_{x,y} * \sin(\theta_1) \quad \text{Ec. (6)}$$

$$y_k = y_{k-1} + \Delta u_{x,y} * \cos(\theta_1) \quad \text{Ec. (7)}$$

Ecuación 2-2. Modelo cinemático (Martínez-Romo, 2017).

En donde:

- pl_l es el número de pulsos de codificador óptico de la rueda trasera, lado izquierdo, medido entre intervalos de muestreo.
- pl_r es el número de pulsos de codificador óptico de la rueda trasera, lado derecho, medido entre intervalos de muestreo.
- k_1 es la constante de compensación odométrica por el error de medición del radio efectivo de la rueda izquierda
- k_2 es constante de compensación odométrica por el error de medición del radio efectivo de la rueda izquierda
- R , es la resolución del codificador óptico en pulsos por revolución (Martínez-Romo, 2017).

2.3.1 Diseño de un sistema AGV

Los vehículos guiados automáticamente conocidos hoy en día como AGV son un desarrollo que se viene presentando desde 1953 inventado por Berrett Electronics.

El diseño de un sistema AGV es una tarea compleja que cuenta con muchas variables que impactan en el funcionamiento de este y que son difíciles de predecir. Las decisiones que se tomen deben estar ligadas a otras variables y todas estas decisiones tienen implicaciones importantes en el sistema. Los principales aspectos a tener en cuenta en un diseño de un sistema AGV son las siguientes:

- Las trayectorias.
- Manejo del tráfico de los AGV.
- El número de estaciones de carga y descarga de materiales.
- Los requerimientos de los vehículos
- Enrutamiento de los vehículos.
- Planificación de uso de los vehículos.
- Posiciones ideales para los vehículos.
- Manejo de baterías.
- Manejo de fallas

En donde las trayectorias definen por donde se debe mover el AGV, el manejo del tráfico incluye todas las estrategias de seguridad que se deben tener para que los vehículos no colisionen entre sí o con algún otro objeto u o persona, los requerimientos de los vehículos determinan que características deben tener los AGV para poder encajar a la perfección con el sistema (tamaño, forma, sensores, sistema de guiado, programación), la planificación se refiere a que debe hacer cada robot en qué momento, las posiciones ideales se refieren a los sitios más seguros de tránsito, el manejo de las baterías está ligado a que baterías se deben tener, cuál va a ser su forma de carga, en que parte de la planta se van a cargar y por último el manejo de fallas se refiere a como se va a manejar las fallas en el sistema.

2.3.2 Objetivos de un sistema AGV

Los objetivos principales que debe seguir la implementación de un sistema AGV son los siguientes:

- Minimizar el tiempo requerido para terminar todos los trabajos.
- Minimizar los tiempos de trayecto de los vehículos cargados o descargados.
- Minimizar los costos de movimiento.
- Minimizar los tiempos de espera entre cargas

2.3.3 Interacción de un AGV con subsistemas

Un AGV se implementa fácilmente en los sistemas de manufactura flexible (FMS) los cuales son sistemas altamente automatizados capaces de producir diferentes partes o productos sin tiempos muertos significativos debidos a los cambios. Este sistema es una combinación de la automatización flexible, tecnología de grupos, maquinas herramienta CNC y sistemas de manejo automatizado de materiales.

2.3.4 Ventajas de los sistemas AGV

Flexibilidad: Estos sistemas son los más flexibles comparados con otros sistemas utilizados en el manejo de materiales. Entre sus ventajas se puede nombrar la alta flexibilidad en los cambios que se pueden hacer en sus recorridos lo cual permite un mejor uso del espacio.

Mayor confiabilidad: Son los más confiables a comparación de otros tipos de sistemas utilizados para el manejo de materiales. Fácilmente el AGV puede ser sustituido por otro en caso de inconvenientes sin causar un paro en el sistema de manufactura.

Ahorro de Inversión: El costo de operación de los AGV es menor que el de otros sistemas de manejo de materiales ya que solo se necesita una persona para su programación. Es fácil de realizarle mantenimiento, consume menos potencia y es muy raro que falle logrando que el tiempo muerto del sistema sea bajo gracias a su funcionamiento continuo.

Movimientos libres: La trayectoria a seguir asegura suavidad, flexibilidad y confiabilidad en las operaciones. Al ser un robot sus movimientos van a estar

completamente estandarizados, y si la programación es la adecuada el manejo de materiales se hará de forma segura.

Aplicaciones de los AGV: Estos vehículos cuentan con varias aplicaciones, entre las más importantes existe el transporte, carga y descarga de materiales y limpieza.

2.3.5 Componentes de un sistema AGV

El vehículo (AGV): Encargado de cumplir las órdenes realizando movimientos de materiales dentro de un sistema.

La trayectoria guía: Dirige el vehículo para que se mueva a lo largo de la trayectoria establecida.

La unidad de control: Supervisa y dirige los procedimientos y actividades del sistema.

La interfaz de computadora: Se comunica con el sistema y las computadoras.

2.3.6 Sistemas de guía

Existen diversos sistemas de guiado para el AGV que se pueden utilizar, el uso de estos depende del ambiente que se va a manejar, su aplicación, necesidades y costos de inversión. La trayectoria guía debe ser flexible sin obstaculizar otros tipos de tráfico.

Sistema de guiado por cable: El AGV sigue la trayectoria por medio de un cable energizado perforado en el piso detectando sus señales por medio de una antena del AGV.

Sistema de guía óptica: Partículas fluorescentes incoloras sobre el piso de concreto siguiendo el trayecto establecido. El vehículo contiene foto sensores los cuales leen la trayectoria.

Sistema de guía inercial: Por medio de un microprocesador conduce la trayectoria programada. Para este sistema de guía se utilizan sensores de

movimiento que pueden ser acelerómetros y sensores de rotación giroscópicos los cuales ayudan a la rotación del robot.

Sistema de guía infrarroja: El AGV reconoce mediante visión artificial una tira de espejo catadióptrico, realizando sus movimientos para seguir la ruta.

Sistema de guiado por cinta reflexiva: En este tipo de guiado se pueden utilizar diferentes tipos de cintas reflexivas las cuales pueden ser de diferentes colores. Esta cinta se posiciona en el suelo siguiendo la ruta que el robot debe realizar. Mediante un sensor óptico el robot detecta las diferentes intensidades de luz que se reflejan en el suelo y sigue el camino deseado.

Sistema de guiado magnético: El sistema de guiado magnético funciona de la misma manera que el guiado por cinta reflexiva solo que para este caso se utiliza una cinta magnética que es la que va a determinar el posicionamiento y la ruta del robot en el entorno.

Sistema de guiado laser: Existen diferentes tipos de sistema de guiado por láser que se pueden utilizar para las trayectorias de un AGV. El más sencillo de los guiados por láser consta de un láser fijo en el robot. Este sistema es poco flexible ya que el rayo que sale del láser solo va en una dirección definiendo una sola trayectoria previamente establecida. Por otra parte, se encuentra el sistema de guiado laser que permite ser más flexible con las trayectorias del AGV reflectores en el entorno en donde se pretende tener el AGV. Estos reflectores hacen de espejo y se colocan principalmente en las esquinas para ayudar al robot a realizar los giros predeterminados.

Reconocimiento de guía con códigos de barra: Este sistema utiliza los códigos de barra para que el robot siga la trayectoria. Para este sistema el robot tiene que estar equipado con una cámara en la parte inferior la cual está adaptada para reconocer los códigos de barra que se posicionan a lo largo de la ruta.

Sistema de cubos reflectores: En este sistema dos o tres cubos reflectores son posicionados en las esquinas del área por donde se pretende que se mueva el AGV. Un láser rotacional es posicionado en la parte superior del robot el cual determina las posiciones de los cubos reflectores y manda la información a un

computador para determinar el movimiento. Este sistema de cubos reflectores es similar al sistema de guiado por láser con espejos.

No existe una forma predeterminada para identificar el sistema de guiado ideal para un sistema, aunque es importante reconocer e identificar cual es el mejor que se adapte a las características del entorno de trabajo.

Se identifican tres sistemas que componen el sistema AGV; eléctrico, electrónico y mecánico. El sistema eléctrico es el motor, el sistema electrónico representa el control del motor y el sistema mecánico todos los componentes de soporte y de transmisión movimiento.

2.4 Odometría de terreno

La odometría es un método que utiliza codificadores para medir la rotación de la rueda y / o la orientación de la dirección. La odometría tiene la ventaja de que es totalmente autónoma y siempre es capaz de proporcionar al vehículo una estimación de su posición. La desventaja de la odometría es que el error de posición crece sin límite a menos que una referencia independiente se utiliza periódicamente para reducir el error. [2].

2.5 Navegación Inercial.

Este método utiliza giroscopios y acelerómetros para medir la velocidad de rotación y aceleración. Las mediciones se integran una vez (o dos veces) para obtener la posición. Los sistemas de navegación inercial también tienen la ventaja de que son autónomos. En el lado negativo, los datos de los sensores inerciales fluctúan con el tiempo debido a la necesidad de integrar los datos de la velocidad para obtener la posición; cualquier pequeño error constante aumenta sin límite después de la integración. Por lo tanto, los sensores inerciales no son adecuados para un posicionamiento preciso durante un periodo prolongado de tiempo. Otro problema con la navegación inercial es el alto costo del equipo. (Martínez-Romo, 2017).

2.6 Fusión de Sensores

La fusión de sensores es una parte integral del diseño de sistemas autónomos.

- Combina dos o más fuentes de datos de una manera que genere una mejor comprensión del sistema.
- Más consistencia.
- Más preciso
- Más confiable (podría ser con un solo sensor)

La medición proporciona la comprensión del sistema y la fuente de datos que podría ser un modelo matemático, porque como diseñador se tiene cierto conocimiento del mundo físico y se puede codificar ese conocimiento en el algoritmo de fusión para mejorar la medición del sensor.

2.6.1 Sistema autónomo.

Necesita interactuar con el mundo que lo rodea como lo es:

- **Sensado:** Mide directamente el entorno con un sensor que recopila información del sistema en un mundo externo.
- **Percepción:** interactuar con los datos y convertirlos en algo que se pueda entender.
- **Planificar:** encontrar el camino.
- **Actuar:** Seguir la ruta, calcular las mejores acciones para que el sistema siga esa ruta.

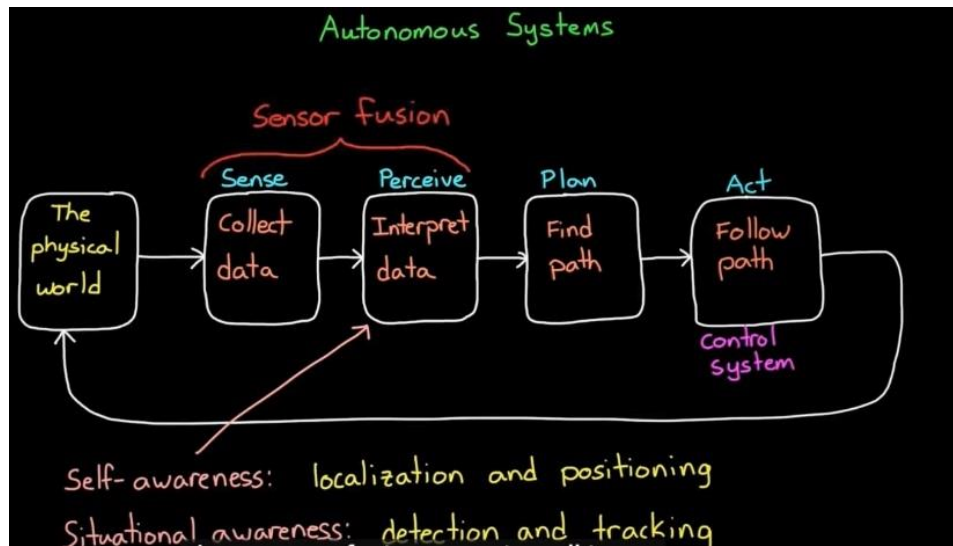


Figura 2-43. Fusión de sensores en sistemas autónomos

La fusión de sensores es muy buena, se puede percibir y percibir, ya que interviene en estas dos capacidades, es el proceso de tomar múltiples mediciones de sensores combinándolas para mezclar información adicional del modelo matemático.

2.6.2 4 formas en que la fusión de sensores puede ayudar

- Puede aumentar la calidad de los datos.
 - Puede aumentar la fiabilidad
- Puede estimar estados no medidos
 - Simplemente significa que el sistema no tiene un sensor que pueda medir directamente el estado que le interesaba.
- Puede aumentar el área de cobertura
 - Las mediciones adicionales se fusionan para producir un campo total de sensor más grande, ahora es más que probable que estas mediciones no sean promedio ni se combinen matemáticamente de ninguna manera, ya que generalmente es útil saber qué sensor se está registrando y objeto para tener una idea de dónde que el objeto es relativo. Para definir una orientación:

- Marco de referencia
- Especificar rotación
- Roll, Pitch, Yaw.
- Direcciones, coseno, matriz, (DCM).
- Cuaterniones.

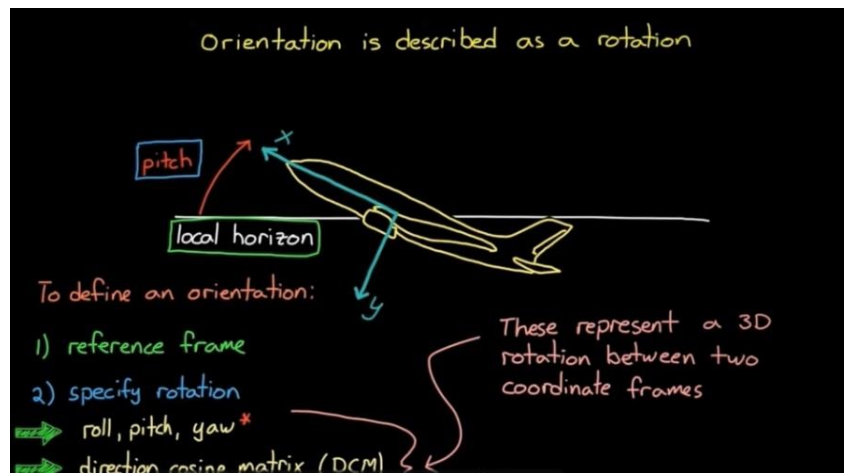


Figura 2-44. Orientación descrita en una rotación

2.6.3 Acelerómetro

Lo importante es que todos estos grupos de números representan una rotación tridimensional entre 2 marcos de coordenadas diferentes. El propio marco de coordenadas del objeto se fija al cuerpo y gira con él.

En otro ejemplo, digamos que queremos saber la orientación del teléfono que está sentado en una mesa, para que el marco de coordenadas del cuerpo del teléfono en relación con el marco de coordenadas local noreste y hacia abajo podamos encontrar la orientación absoluta.

La orientación del cuerpo es simplemente la rotación entre el marco del cuerpo y el marco hacia el noreste puede construir la matriz de coseno de dirección directamente.

2.6.4 Problemas del acelerómetro

No solo miden la gravedad, sino que miden todas las aceleraciones lineales, por lo que, si el sistema se mueve mucho, va a descartar la estimación de dónde está usted, por ejemplo, si no gira realmente el sensor pero el espectador está saltando por todos lados el lugar y esto podría no ser un gran problema si su sistema no acelera en gran medida como un teléfono mientras está navegando a gran altitud.

La aceleración lineal no es el único problema, incluso las rotaciones pueden desviar la estimación porque no se encuentra en el centro de las rotaciones detectará una aceleración cuando el sistema gira, por lo que tenemos que encontrar una manera de lidiar con estas corrupciones.

2.6.5 Magnetómetro

El magnetómetro se ve afectado por perturbaciones en el campo magnético.

Si las perturbaciones magnéticas son parte del sistema y giran con el magnetómetro, entonces se pueden calibrar, estas son las llamadas fuentes de hierro duro y suave.

La calibración es importante para recopilar un montón de mediciones en diferentes orientaciones y luego encontrar los coeficientes de calibración que los ajustarán a una esfera ideal.

2.6.6 Resolviendo el otro problema de las aceleraciones lineales corruptas

Podemos multiplicar la medición de velocidad angular por el tiempo de muestra para obtener el cambio de ángulo durante ese tiempo y luego, si supiéramos la orientación del teléfono en el tiempo de muestra anterior, podríamos agregarle este ángulo delta y tener una estimación actualizada de la orientación actual y si el objeto no se escribe, entonces el ángulo delta sería cero y la orientación no cambiaría, por lo que todo funciona y repitiendo este proceso para el siguiente tiempo de muestra y el siguiente que conocerá la orientación del teléfono a lo largo del tiempo, este

proceso se denomina cálculo muerto y, esencialmente, solo integra la medición giroscópica. (1)

En este punto, tenemos dos formas diferentes de estimar la orientación, una usando el acelerómetro y el magnetómetro y la otra usando solo el giroscopio y cada una tiene sus respectivos beneficios y problemas, y aquí es donde entra en juego la fusión de sensores. combine estas dos estimaciones de una manera que enfatice cada una de sus fortalezas y minimice sus debilidades ahora hay una serie de algoritmos de fusión de sensores que podemos usar como un filtro complementario o un filtro kalman o los filtros madgwick o mahony más especializados pero comunes.

Cada uno de ellos hace esencialmente lo mismo que inicializan la actitud, usan el campo magnético y la gravedad para corregir la deriva del giroscopio.

Estimar el sesgo del sensor es extremadamente importante porque el sesgo se desplaza con el tiempo, esto significa que incluso si calcula el sesgo del sensor antes de operar su sistema y codifica ese valor de calibración en su software, no será preciso por mucho tiempo y cualquier sesgo que no tengamos t remove se integrará y hará que la estimación se aleje de la verdad cuando confiamos en ese sensor.

Es un proceso matemático interactivo que utiliza un conjunto de ecuaciones y entradas de datos consecutivas para estimar rápidamente el verdadero valor, posición, velocidad, etc. del objeto que se está midiendo, cuando los valores medidos contienen errores aleatorios, incertidumbres o variaciones impredecibles. El filtro de Kalman toma información de una observación y toma un estado de una situación, por ejemplo, la temperatura. Cuando quiera saberlo, siga obteniendo mediciones en un intervalo periódico y Xo quiere seguir actualizando donde cree, va a predecir y va a mezclar esa medición real de predicción y va a usar el filtro de Kalman para calcule cuánto de la predicción y cuánto mide la medición que desea impartir en la nueva predicción para el próximo intervalo y, a partir de ahí, debería poder rastrear con mucha precisión el filtro de Kalman minimiza el error que puede tener en una actualización normal mecanismo donde simplemente toma la

observación puesta en su ecuación y aquí tiene la nueva posición porque a veces la medición se actualiza muy erráticamente y no es una muy buena manera de rastrear algo.

Está prediciendo cómo cambiarán los estados con el tiempo en función de un modelo que tenga y, junto con los estados, también realiza un seguimiento de la confiabilidad de la predicción en función del ruido del proceso que le ha dado y cuanto más tiempo tenga que predecir el filtro el estado es la menor confianza que tiene en el resultado que cuando se produce una nueva medición que tiene su propio ruido de medición asociado, el filtro compara la predicción con la medición y corrige su estimación en función de la confianza relativa en ambos.

2.7 Filtro de Kalman

Sirve para poder identificar el estado oculto (no medible) de un sistema dinámico lineal.

El filtro de Kalman en una medición en un sistema dinámico lineal, descrito por las ecuaciones, se ha realizado en el momento k . El requisito es utilizar la información contenida en la nueva medición y_k para actualizar la estimación del estado desconocido $\mathbf{x}_k$. Sea $\hat{\mathbf{x}}_k^-$ la estimación a priori del estado, que ya está disponible en el momento k . Con un estimador lineal como objetivo, podemos expresar la estimación a posteriori $\hat{\mathbf{x}}_k$ como una combinación lineal de la estimación a priori y la nueva medición, como se muestra descrita por (Haykin, 2001):

$$\hat{\mathbf{x}}_k = \mathbf{G}_k^{(1)} \hat{\mathbf{x}}_k^- + \mathbf{G}_k \mathbf{y}_k, \quad (2.5)$$

Donde se deben determinar los factores de la matriz multiplicadora $\mathbf{G}_k^{(1)}$ y $\mathbf{G}_k$ deben ser 89ínimas. Para encontrar estas dos matrices, invocamos el principio de ortogonalidad establecido en el Teorema 1.2- (Haykin, 2001)

El vector de error de estado está definido por:

$$\tilde{\mathbf{x}}_k = \mathbf{x}_k - \hat{\mathbf{x}}_k. \quad (2.6)$$

Aplicando el principio de ortogonalidad a la situación actual, podemos escribir

$$E [\tilde{x}_k y_i^T] = \mathbf{0} \text{ for } i = 1, 2, \dots, k-1. \quad (2.7)$$

Usando las ecuaciones. (2.3), (2.5) y (2.6) en (2.7), obtenemos

$$E [(\mathbf{x}_k - \mathbf{G}_k^{(1)} \hat{\mathbf{x}}_k - \mathbf{G}_k \mathbf{H}_k \mathbf{x}_k - \mathbf{G}_k \mathbf{w}_k) y_i^T] = \mathbf{0} \text{ for } i = 1, 2, \dots, k-1. \quad (2.8)$$

Dado que el ruido de proceso $\mathbf{w}_k$ y el ruido de medición $\mathbf{v}_k$ no están correlacionados, se deduce que usando esta relación y reordenando los términos, podemos reescribir la Ec. (2.8) como :

$$E[(\mathbf{I} - \mathbf{G}_k \mathbf{H}_k - \mathbf{G}_k^{(1)}) \mathbf{x}_k y_i^T + \mathbf{G}_k^{(1)} (\mathbf{x}_k - \hat{\mathbf{x}}_k^-) y_i^T] = \mathbf{0}, \quad (2.9)$$

Donde $\mathbf{I}$ es la matriz de identidad. Desde el principio de ortogonalidad, ahora notamos que

$$E[(\mathbf{x}_k - \hat{\mathbf{x}}_k^-) y_i^T] = \mathbf{0}.$$

Por consiguiente, Eq. (2.9) se simplifica a

$$(\mathbf{I} - \mathbf{G}_k \mathbf{H}_k - \mathbf{G}_k^{(1)}) E[\mathbf{x}_k y_i^T] = \mathbf{0} \quad \text{for } i = 1, 2, \dots, k-1. \quad (2.10)$$

Para valores arbitrarios del estado $\mathbf{x}_k$ y observable y_i , Eq. (2.10) solo se puede satisfacer si la escala si los factores de escala $\mathbf{G}_k^{(1)}$ y $\mathbf{G}_k$ están relacionados de la siguiente manera:

$$\mathbf{I} - \mathbf{G}_k \mathbf{H}_k - \mathbf{G}_k^{(1)} = \mathbf{0},$$

O, de manera equivalente, $\mathbf{G}_k^{(1)}$ se define en términos de $\mathbf{G}_k$ como

$$\mathbf{G}_k^{(1)} = \mathbf{I} - \mathbf{G}_k \mathbf{H}_k. \quad (2.11)$$

Sustituyendo la ecuación. (2.11) en (2.5), podemos expresar la estimación a posteriori del estado en el tiempo k como:

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-) \quad (2.12)$$

A la luz de lo cual, la matriz $\mathbf{G}_k$ se denomina la ganancia de Kalman.

Ahora queda el problema de derivar una fórmula explícita para $\mathbf{G}_k$.

Dado que, desde el principio de ortogonalidad, tenemos:

$$E[(\mathbf{x}_k - \hat{\mathbf{x}}_k) \mathbf{y}_k^T] = \mathbf{0}, \quad (2.13)$$

Resulta que:

$$E[(\mathbf{x}_k - \hat{\mathbf{x}}_k)\hat{\mathbf{y}}_k^T] = \mathbf{0}, \quad (2.14)$$

Donde $\hat{\mathbf{y}}_k^T$ es una estimación de $\mathbf{y}_k$ dada la medición anterior $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_{k-1}$.

Definir el proceso de innovaciones.

$$\tilde{\mathbf{y}}_k = \mathbf{y}_k - \hat{\mathbf{y}}_k. \quad (2.15)$$

El proceso de innovación representa una medida de la "nueva" información contenida en $\mathbf{y}_k$; también puede expresarse como:

$$\begin{aligned} \tilde{\mathbf{y}}_k &= \mathbf{y}_k - \mathbf{H}_K \hat{\mathbf{x}}_K^- \\ &= \mathbf{H}_K \mathbf{x}_k + \mathbf{v}_k - \mathbf{H}_K \hat{\mathbf{x}}_K^- \\ &= \mathbf{H}_K \tilde{\mathbf{x}}_k^- + \mathbf{v}_k. \end{aligned} \quad (2.16)$$

Por lo tanto, restando la ecuación. (2.14) de (2.13) y luego usando la definición de la ecuación, (2.15), podemos escribir

$$E[(\mathbf{x}_k - \hat{\mathbf{x}}_k) \tilde{\mathbf{y}}_k^T] = \mathbf{0}. \quad (2.17)$$

Usando las ecuaciones. (2.3) y (2.12), podemos expresar el vector de error de estado $\mathbf{x}_k - \hat{\mathbf{x}}_k$ como

$$\begin{aligned} \mathbf{x}_k - \hat{\mathbf{x}}_k &= \tilde{\mathbf{x}}_k^- - \mathbf{G}_k (\mathbf{H}_K \tilde{\mathbf{x}}_k^- + \mathbf{v}_k) \\ &= (\mathbf{I} - \mathbf{G}_k \mathbf{H}_K) \tilde{\mathbf{x}}_k^- - \mathbf{G}_k \mathbf{v}_k. \end{aligned} \quad (2.18)$$

Por lo tanto, sustituyendo las ecuaciones. (2.16) y (2.18) en (2.17), obtenemos:

$$E\{[(\mathbf{I} - \mathbf{G}_k \mathbf{H}_K) \tilde{\mathbf{x}}_k^- - \mathbf{G}_k \mathbf{v}_k](\mathbf{H}_K \tilde{\mathbf{x}}_k^- + \mathbf{v}_k)^T\} = \mathbf{0}. \quad (2.19)$$

Dado que el ruido de medición $\mathbf{v}_k$ es independiente del estado $\mathbf{x}_k$, por lo tanto, el error $\tilde{\mathbf{x}}_k^-$, la expectativa de la Ec. (2.19) se reduce a:

$$(\mathbf{I} - \mathbf{G}_k \mathbf{H}_K) E[\tilde{\mathbf{x}}_k^- \tilde{\mathbf{x}}_k^{T-}] \mathbf{H}_k^T - \mathbf{G}_k E[\mathbf{v}_k \mathbf{v}_k^T] = \mathbf{0}. \quad (2.20)$$

Definir la matriz de covarianza a priori:

$$\mathbf{P}_k^- = E[(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)(\mathbf{x}_k - \hat{\mathbf{x}}_k^-)^T]$$

$$= E[\tilde{\mathbf{x}}_k^- \cdot \tilde{\mathbf{x}}_k^{T-}]. \quad (2.21)$$

Luego, invocando las definiciones de covarianza de las Ecs. (2.14) y (2.21), podemos reescribir la ecuación. (2.20) como:

$$(\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) \mathbf{P}_k^- \mathbf{H}_K^T - \mathbf{G}_K \mathbf{R}_k = \mathbf{0}.$$

Resolviendo esta ecuación para $\mathbf{G}_k$, obtenemos la fórmula deseada:

$$\mathbf{G}_k = \mathbf{P}_k^- \mathbf{H}_k^T [\mathbf{H}_K \mathbf{P}_k^- \mathbf{H}_K^T + \mathbf{R}_k]^{-1}, \quad (2.22)$$

Donde el símbolo $[\cdot]^{-1}$ denota la inversa de la matriz dentro de los corchetes. La ecuación (2.22) es la fórmula deseada para calcular la ganancia de Kalman $\mathbf{G}_k$, que se define en términos de la matriz de covarianza a priori $\mathbf{P}_k^-$. (Haykin, 2001).

Para completar el procedimiento de estimación recursiva, se considera la propagación de la covarianza del error, que describe los efectos del tiempo sobre las matrices de covarianza de los errores de estimación.

Esta propagación implica dos etapas de cálculo:

La matriz de covarianza a priori $\mathbf{P}_k^-$ en el tiempo k está definida por la Ec. (2.21), Dado $\mathbf{P}_k^-$, calcule la matriz de covarianza a posteriori $\mathbf{P}_k$, que, en el tiempo k , está definida por:

$$\begin{aligned} \mathbf{P}_k &= E[\tilde{\mathbf{x}}_k \tilde{\mathbf{x}}_k^T] \\ &= E[(\mathbf{x}_k - \hat{\mathbf{x}}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_k)^T]. \end{aligned} \quad (2.23)$$

Dada la matriz de covarianza a posteriori “antigua” $\mathbf{P}_{k-1}$, calcule la matriz de covarianza a priori “actualizada” $\mathbf{P}_k^-$.

Para continuar con la etapa 1, sustituimos la ecuación (2.18) en (2.23) y observe que el proceso de ruido $\mathbf{v}_k$ es independiente del error de estimación a priori $\tilde{\mathbf{x}}_k^-$.

Así obtenemos ¹

$$\begin{aligned} \mathbf{P}_k &= (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) E[\tilde{\mathbf{x}}_k^- \tilde{\mathbf{x}}_k^{T-}] (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K)^T + \mathbf{G}_K E[\mathbf{v}_k \mathbf{v}_k^T] \mathbf{G}_K^T \\ &= (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) \mathbf{P}_k^- (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K)^T + \mathbf{G}_K \mathbf{R}_k \mathbf{G}_K^T \end{aligned} \quad (2.24)$$

Ampliación de términos en la ecuación. (2.24) y luego usando la Ec. (2.22), podemos reformular la dependencia de la matriz de covarianza a posteriori $\mathbf{P}_k$ de la matriz de covarianza a priori $\mathbf{P}_k$ en la forma simplificada.

$$\begin{aligned}\mathbf{P}_k &= (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) \mathbf{P}_k^- - ((\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) \mathbf{P}_k^- \mathbf{H}_K^T \mathbf{G}_K^T + \mathbf{G}_K \mathbf{R}_k \mathbf{G}_K^T \\ &= (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) \mathbf{P}_k^- - \mathbf{G}_K \mathbf{R}_k \mathbf{G}_K^T + \mathbf{G}_K \mathbf{R}_k \mathbf{G}_K^T \\ &= (\mathbf{I} - \mathbf{G}_K \mathbf{H}_K) \mathbf{P}_k^-\end{aligned}\tag{2.25}$$

Para la segunda etapa de la propagación de la covarianza del error, primero reconocemos que la estimación a priori del estado se define en términos de la estimación "antigua" o posteriori de la siguiente manera:

$$\hat{\mathbf{x}}_k^- = \mathbf{F}_{k,k-1} \hat{\mathbf{x}}_{k-1}\tag{2.26}$$

Por tanto, podemos utilizar las ecuaciones. (2.1) y (2.26) para expresar el error de estimación a priori en otra forma más:

$$\begin{aligned}\tilde{\mathbf{x}}_k^- &= \mathbf{x}_k - \hat{\mathbf{x}}_k^- \\ &= (\mathbf{F}_{k,k-1} \mathbf{x}_{k-1} + \mathbf{w}_{k-1}) - (\mathbf{F}_{k,k-1} \hat{\mathbf{x}}_{k-1}) \\ &= \mathbf{F}_{k,k-1} (\mathbf{x}_{k-1} - \hat{\mathbf{x}}_{k-1}) + \mathbf{w}_{k-1} \\ &= \mathbf{F}_{k,k-1} \tilde{\mathbf{x}}_{k-1} + \mathbf{w}_{k-1}.\end{aligned}\tag{2.27}$$

Por consiguiente, usando la ecuación. (2.27) en (2.21) y observando que el ruido del proceso $\mathbf{w}_k$ es independiente de $\tilde{\mathbf{x}}_{k-1}$, obtenemos

$$\begin{aligned}\mathbf{P}_k^- &= \mathbf{F}_{k,k-1} E[\tilde{\mathbf{x}}_{k-1} \tilde{\mathbf{x}}_{k-1}^T] \mathbf{F}_{k,k-1}^T + E[\mathbf{w}_{k-1} \mathbf{w}_{k-1}^T] \\ &= \mathbf{F}_{k,k-1} \mathbf{P}_{K-1} \mathbf{F}_{k,k-1}^T + \mathbf{Q}_{k-1},\end{aligned}\tag{2.28}$$

Que define la dependencia de la matriz de covarianza a priori $\mathbf{P}_k^-$ de la matriz de covarianza a posteriori "antigua" $\mathbf{P}_{k-1}$.

Con las Ecs. (2.26), (2.28), (2.22), (2.12) y (2.25) a la mano, ahora podemos resumir la estimación recursiva del estado. En ausencia de datos observados en el tiempo $k = 0$, podemos elegir la estimación inicial del estado como

$$\hat{\mathbf{x}}_0 = E[\mathbf{x}_0],$$

Modelo de espacio de estados:

$$\mathbf{x}_{k+1} = \mathbf{F}_{k+1,k} \mathbf{x}_k + \mathbf{w}_k,$$

$$\mathbf{y}_k = \mathbf{H}_k \mathbf{x}_k + \mathbf{v}_k,$$

Donde $\mathbf{w}_k$ y $\mathbf{v}_k$ son procesos independientes de ruido gaussiano de media cero de las matrices de covarianza $\mathbf{Q}_k$ y $\mathbf{R}_k$, respectivamente.

Inicialización: Para $k = 0$, establezca.

$$\hat{\mathbf{x}}_0 = E[\mathbf{x}_0],$$

$$\mathbf{P}_0 = E[(\mathbf{x}_0 - E[\mathbf{x}_0])(\mathbf{x}_0 - E[\mathbf{x}_0])^T].$$

Cálculo: Para $k = 1, 2, \dots$, calcule:

Propagación de estimación de estado.

$$\hat{\mathbf{x}}_k^- = \mathbf{F}_{k,k-1} \hat{\mathbf{x}}_{k-1}^-;$$

Propagación de covarianza de error.

$$\mathbf{P}_k^- = \mathbf{F}_{k,k-1} \mathbf{P}_{k-1} \mathbf{F}_{k,k-1}^T + \mathbf{Q}_{k-1};$$

Matriz de ganancia de Kalman.

$$\mathbf{G}_k = \mathbf{P}_k^- \mathbf{H}_k^T [\mathbf{H}_k \mathbf{P}_k^- \mathbf{H}_k^T + \mathbf{R}_k]^{-1};$$

Actualización de la estimación del estado.

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k^- + \mathbf{G}_k (\mathbf{y}_k - \mathbf{H}_k \hat{\mathbf{x}}_k^-);$$

Actualización de covarianza de error.

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{G}_k \mathbf{H}_k) \mathbf{P}_k^-.$$

Y el valor inicial de la matriz de covarianza a posteriori como

$$\mathbf{P}_0 = E[(\mathbf{x}_0 - E[\mathbf{x}_0])(\mathbf{x}_0 - E[\mathbf{x}_0])^T]. \quad (2.30)$$

Esta elección para las condiciones iniciales no solo es intuitivamente satisfactoria, sino que también tiene la ventaja de producir una estimación insesgada del estado $\mathbf{x}_k$. (Haykin, 2001).

2.8 Varianza de Allan

El uso de la varianza estándar en el análisis de la dispersión de variables dependientes del tiempo puede conducir a problemas de divergencia cuando el número de mediciones tiende a infinito. Dicha divergencia puede ser originada por una fuerte correlación entre mediciones que introduce ruidos no blancos en las series de mediciones. En el caso de la metrología de tiempo y frecuencia, por ejemplo, la presencia de ruidos no blancos como el llamado ruido en frecuencia de paso aleatorio (*Random Walk Frequency Noise*) introduce una rápida divergencia en el análisis de estabilidad de frecuencia cuando se usa la varianza estándar. El uso de la llamada Varianza de Allan se ha generalizado a nivel internacional para expresar la estabilidad de osciladores ya que es convergente para los principales ruidos no blancos presentes en señales de frecuencia y en series de tiempo. (R., 2020).

La Varianza de Allan es la herramienta usada para el análisis de mediciones de Tiempo y Frecuencia siendo un estimador de la dispersión de las mediciones, determinando así, la estabilidad del oscilador bajo calibración.

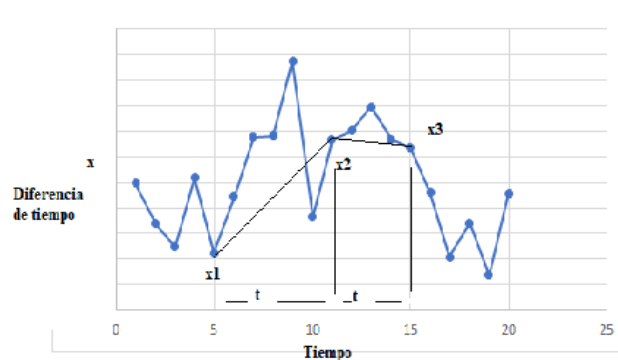


Figura 2-45. Diferencia de tiempo en desplazamiento.

$$y_1 = \frac{x_{i+1} - x_i}{T}$$

$$\Delta y_i = y_{i+1} - y_i$$

$$\sigma_y^2 = \frac{1}{2} \{(\Delta y_i)^2\}$$

Frecuencia:

$$\Delta^2 x_i = x_{i+2} - 2x_{i+1} + x_i$$

$$\sigma_y^2 = \frac{1}{2T^2} \{(\Delta x_i)^2\}$$

Fase:

Varianza de Allan para Mediciones de Frecuencia.

$$\sigma_y^2(T) = \frac{1}{2T_0(N-1)} \sum_{i=1}^{N-1} \{(y_{i+1} - y_i)^2\}$$

Donde :

σ_y^2 Es la varianza de Allan.

N- Es el número de datos espaciados.

T- Es el tiempo de observación = mT_0 .

y_1 Es la i-ésima medición de fase.

$m = 2^n$ cálculos posibles .

Varianza de Allan para Mediciones de Diferencia de Fase.

$$\sigma_y^2(T) = \frac{1}{2(N-2m)T^2} \sum_{i=1}^{N-2m} (x_{i+2m} - 2x_{i+m} + x_i)^2$$

σ_y^2 Es la varianza de Allan.

N- Es el número de datos espaciados en T_0 .

x_i Es la i-ésima medición de fase.

T- Es el tiempo de observación = mT_0 .

$m = 2^n$ cálculos posibles.

$$x^2 = (df) \frac{s_y^2}{\sigma_y^2}$$

Para $df < 100$

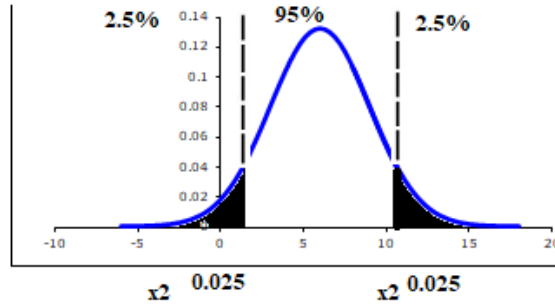


Figura 2-46. Campana de gauss en varianza de Allan.

Barra de incertidumbre:

$$\frac{s_y^2(df)}{x^2(.975)} < \sigma_y^2 < (df) \frac{s_y^2(df)}{x^2(.25)}$$

Barra inferior Barra superior

Dependencia temporal de $\sigma_y(\tau)$

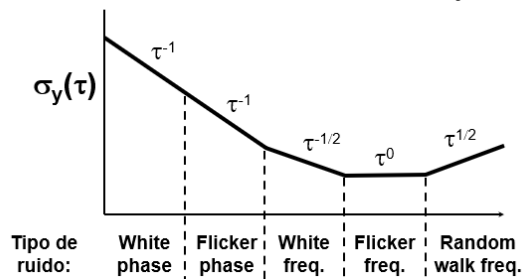


Figura 2-47. Dependencia temporal de σ_y en el tiempo.

2.9 Sistemas Embebidos

Los sistemas de computación se encuentran en cualquier parte. Es probable que nadie se sorprenda al saber que millones de sistemas de computación son producidos y vendidos cada año, como por ejemplo computadores de escritorios o computadores personales. Sin embargo, sorprende saber que billones de sistemas de computación de propósito específico son construidos y vendidos cada año; los sistemas embebidos se encuentran alrededor de nuestras vidas, en forma de teléfonos móviles, equipos médicos, sistemas de navegación aérea, reproductores MP3, impresoras, automóviles, etc. Cada vez que se mira alrededor es posible identificar un dispositivo que contiene un microprocesador, y probablemente se ha

encontrado un sistema embebido. Por estas razones resulta conveniente estudiarlos.

Existen numerosas definiciones de sistemas embebidos, algunas son:

- “Un sistema embebido es cualquier dispositivo que incluye un computador programable, pero en sí mismo no es un computador de propósito general” (Wolf, 2008).
- “Un sistema embebido es un sistema electrónico que contiene un microprocesador o microcontrolador; sin embargo, no pensamos en ellos como un computador” (Morton, 2000).
- “Las personas usan el término sistema embebido para referirse a cualquier sistema de cómputo escondido en algún producto o dispositivo” (Simon, 1999).
- “Un sistema embebido es un sistema cuya función principal no es computacional, pero es controlado por un computador integrado. Este computador puede ser un microcontrolador o un microprocesador. La palabra embebido implica que se encuentra dentro del sistema general, oculto a la vista, y forma parte de un todo de mayores dimensiones” (Wilmshurst, 2003).

Un sistema embebido posee hardware de computador junto con software embebido como uno de sus componentes más importantes. Es un sistema computacional dedicado para aplicaciones o productos. Puede ser un sistema independiente o parte de un sistema mayor, y dado que usualmente su software está embebido en ROM (Read Only Memory) no necesita memoria secundaria como un computador. Un sistema embebido tiene tres componentes principales:

- Hardware.
- Un software primario o aplicación principal. Este software o aplicación lleva a cabo una tarea en particular, o en algunas ocasiones una serie de tareas.

- Un sistema operativo que permite supervisar la(s) aplicación(es), además de proveer los mecanismos para la ejecución de procesos. En muchos sistemas embebidos es requerido que el sistema operativo posea características de tiempo real.

Es importante resaltar que el software que se ejecuta en un sistema embebido es diseñado bajo algunas restricciones importantes:

- Cantidades pequeñas de memoria, generalmente en el orden de los KB.
- Capacidades limitadas de procesamiento, generalmente los procesadores poseen velocidades que no superan los Mhz.
- La necesidad de limitar el consumo de energía en cualquier instante bien sea en estado de ejecución o no.

2.9.1 3.7.1 Características de un sistema embebido.

Los sistemas embebidos poseen ciertas características que los distinguen de otros sistemas de cómputo, a continuación, estudiaremos las más importantes (Vahid, 2002):

- Funcionamiento específico. Un sistema embebido usualmente ejecuta un programa específico de forma repetitiva. Por ejemplo, un pager, siempre en un pager. En contraste, un sistema de escritorio ejecuta una amplia variedad de programas, como hojas de cálculo, juegos, etc.; además nuevos programas son añadidos frecuentemente. Por supuesto puede haber excepciones, podría ocurrir que el programa del sistema embebido fuese actualizado a una nueva versión. Por ejemplo, un teléfono celular podría actualizarse de alguna manera.
- Fuertes limitaciones. Todos los sistemas de computación poseen limitaciones en sus métricas de diseño, pero en los sistemas embebidos son muy fuertes. Una métrica de diseño es una medida de algunas características de implementación, como: costo, tamaño, desempeño, y consumo de energía. Los sistemas embebidos

generalmente deben ser poco costosos, poseer un tamaño reducido, tener un buen desempeño para procesar datos en tiempo real, y además consumir un mínimo de energía para extender el tiempo de vida de las baterías o prevenir la necesidad de elementos adicionales de enfriamiento.

- **Reactivos y tiempo real.** Muchos sistemas embebidos deben ser reactivos o reaccionar ante cambios en el ambiente, además de realizar algunos cálculos en tiempo real sin ningún retraso, es decir, se deben tener resultados en tiempos fijos ante cualquier eventualidad. Por ejemplo, el módulo de control de viaje de un automóvil continuamente monitorea la velocidad y los sensores de frenos, reaccionando ante cualquier eventualidad. Ante un estímulo anormal, el módulo de control debe realizar los cálculos de forma precisa y acelerada para garantizar la entrega de los resultados dentro de un tiempo límite, una violación en este tiempo podría ocasionar la pérdida del control del automóvil. En contraste, un sistema de escritorio se enfoca en realizar cálculos con una frecuencia no determinada y la demora de estos no producen fallas en el sistema.

2.9.2 3.7.2 Hardware de sistemas embebidos

El término hardware en cualquier sistema se refiere a los componentes físicos que lo forman o constituyen; estos componentes permiten realizar un conjunto de tareas al ejecutar programas o software. Los componentes físicos de un sistema embebido por lo general difieren en algunos aspectos de los que conforman un sistema de propósito general, como un computador de escritorio en: tamaño, capacidad de cómputo, requerimientos de energía, etc. Por esta razón es de gran importancia conocer el funcionamiento del hardware para poder desarrollar sistemas embebidos y las aplicaciones que se ejecutarán en él. A continuación, se estudiarán los aspectos más relevantes del hardware comúnmente utilizado en sistemas embebidos.

2.9.2.1 Microprocesadores

La tecnología de procesadores trata de la arquitectura del núcleo computacional usado para implementar las funcionalidades deseadas de un sistema. En particular, un microprocesador “es un componente LSI que realiza una gran cantidad de funciones o tareas en una sola pieza de circuito integrado” (Zazks, 2008).

El término LSI (Large Scale Integration) se refiere a la tecnología que permite integrar varios miles de transistores en un solo circuito integrado. Otra definición de microprocesador podría ser “componente LSI que asocia en una sola pieza de circuito integrado las funciones de una unidad aritmética y la unidad de control asociada” (Zazks, 2008) .

2.9.2.2 Procesadores de propósito general

Un procesador de propósito general o microprocesador es un dispositivo programable adaptable a una gran variedad de aplicaciones. Una característica de ellos es que poseen memoria para la ejecución de programas, este componente es necesario ya que no se conoce a priori cuál programa será ejecutado. Otra característica es que poseen un camino de datos genérico, conformado por varios registros y una o varias ULAs (Unidad Lógica Aritmética). Todo esto hace posible la ejecución de aplicaciones de diversos propósitos (Hallinan, 2006).

2.9.2.3 Procesadores de propósito específico

Un procesador de propósito específico es un circuito digital diseñado para ejecutar exactamente un programa. Por ejemplo, un codificador/decodificador JPEG, el cual ejecuta un programa que comprime y descomprime marcos de video. Un sistema embebido frecuentemente utiliza este tipo de procesadores, ya que se ajusta completamente a las funcionalidades requeridas. A menudo este tipo de procesadores es denominado coprocesador, acelerador y/o periférico (Díaz, 2006).

2.9.2.4 Procesadores específicos de aplicación

Un procesador específico de aplicación (ASIP – Application Specific Instruction set Processor) es un compromiso entre las opciones anteriores. Un ASIP es un procesador programable optimizado para una clase particular de aplicaciones que

comparten características comunes, tales como procesamiento de señales digitales o telecomunicaciones. De este modo es posible optimizar el camino de datos para una clase de aplicaciones en particular, al agregar unidades funcionales especiales para aplicaciones comunes y eliminar otras de poco uso.

2.9.2.5 Microcontrolador

“Un microcontrolador es un circuito integrado que consiste en muchas de las mismas cualidades que una computadora de escritorio, tales como CPU (Central Process Unit), la memoria, etc., pero no incluye ningún dispositivo de comunicación con humanos (monitor, teclado, etc.)” (Gilliland, 1999).

Un microcontrolador es un tipo de procesador ASIP, usado desde hace varias décadas. Un microcontrolador es un microprocesador que ha sido optimizado para aplicaciones de control embebidas. Estas aplicaciones típicamente monitorean y fijan numerosas señales de control de un único bit, pero no realizan cálculos exhaustivos. Así, los microcontroladores tienden a tener caminos de datos sencillos que se destacan en operaciones a nivel de bit, para su lectura y escritura a nivel externo. Además, tienden a incorporar en el chip del microprocesador varios componentes periféricos comunes en el control de aplicaciones, tales como: periféricos de comunicación serial, temporizadores, contadores, modulador de amplitud de pulso, etc. Esta incorporación de periféricos permite la implementación de un único chip, lo que genera productos de menor costo. Algunos ejemplos de microcontroladores son los 18F452 Y 18F4550 de Microchips.

2.9.2.6 Sensores

Un sensor es un dispositivo eléctrico y/o mecánico que convierte magnitudes físicas en valores medibles de dicha magnitud. Los sensores van a aportar información tanto del entorno como del estado interno del componente que mide.

La señal medida usualmente debe transformarse para poder ser interpretada. Este proceso se realiza en tres fases:

- Un fenómeno físico es captado por un sensor, como consecuencia, muestra en su salida una señal eléctrica equivalente al fenómeno captado.
- La señal eléctrica es modificada por un sistema de acondicionamiento de señal, cuya salida es un voltaje que será convertido usando un convertidor analógico/digital (A/D) para ser tratado.
- El convertidor A/D es sensible sólo a rangos limitados de tensiones, frecuentemente 0 a 5V.

El convertidor hace que la salida continua se convierta en una salida discreta. En sistemas embebidos suelen utilizarse diversos tipos de sensores como, por ejemplo, sensores de luz, sensores de contacto, sensores de temperatura, etc.

2.9.2.7 Memoria

Cualquiera de las funcionalidades de los sistemas embebidos está compuestas de tres aspectos: procesamiento, almacenamiento y comunicación. El procesamiento es la transformación de los datos, el almacenamiento es la retención de los datos para su posterior uso, y la comunicación es la transferencia de los datos. Cada uno de estos aspectos debe ser implementado. Se usan procesadores para el procesamiento, memoria para el almacenamiento, y buses para la comunicación.

2.9.3 3.7.3 Principios básicos de un sistema embebido

El diseñador de un sistema embebido debe construir una implementación que satisfaga la necesidad deseada. Pero una dificultad adicional es construir una implementación que simultáneamente optimice diversas métricas de diseño. Una implementación consiste en un microprocesador con un programa que lo acompañe, una conexión de compuertas digitales, o una combinación de ambos. Una métrica de diseño es una característica medible de la implementación del sistema. Métricas comunes incluyen:

- Costo NRE (Nonrecurring Engineering). El costo monetario de diseñar el sistema por primera vez. Una vez que el sistema está diseñado, cualquier número de unidades puede ser manufacturado sin incurrir en costo adicionales de diseño.
- Costo unitario. El costo monetario de manufacturar cada copia del sistema, excluyendo el costo NRE.
- Tamaño. El espacio físico requerido para el sistema, usualmente medido en bytes para software, y compuertas o transistores para hardware.
- Desempeño. El tiempo de ejecución del sistema.
- Energía. La cantidad de energía consumida por el sistema, que puede determinar el tiempo de vida de la batería.
- Tiempo para crear un prototipo. El tiempo necesario para construir una versión funcional del sistema.
- Tiempo de mercadeo. El tiempo requerido para desarrollar un sistema al punto en que pueda ser lanzado al mercado.
- Mantenimiento. La habilidad de modificar el sistema luego de su lanzamiento inicial, especialmente por diseñadores que no participaron en el diseño original.
- Correctitud. Confianza en que la funcionalidad del sistema es la correcta.

- Seguridad. La probabilidad de que el sistema no cause daño. Al diseñar un sistema embebido deben tenerse en cuenta las métricas mencionadas; sin embargo, las métricas típicamente compiten entre sí. La mejora de una métrica generalmente conlleva al degrado de otra. Por ejemplo, si se reduce el tamaño de una implementación su desempeño puede decaer, tal como se muestra en la Figura 2-48.

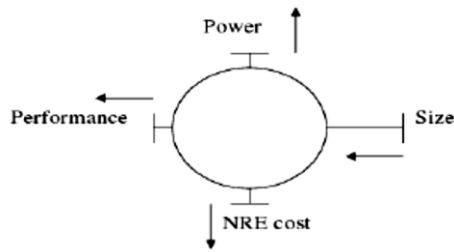


Figura 2-48. Competencia de métrica de diseño

2.9.4 3.7.4 Aplicaciones de un sistema embebido

Los sistemas embebidos se encuentran en una variedad de dispositivos electrónicos comunes, tales como consumibles electrónicos (teléfonos celulares, pagers, cámaras digitales, video juegos portátiles, calculadoras, PDAs, etc.), electrodomésticos (hornos microondas, máquinas contestadoras, termostatos, lavadoras, etc.), equipos de oficina (fax, copiadoras, impresoras, scanners), equipos de negocios (caja registradora, sistemas de alarma, lectores de tarjeta y cajeros automáticos), y automóviles (control de transmisión, control de viaje, inyección de combustible, ABS, etc.). Podría decirse que prácticamente cualquier dispositivo que se ejecute con electricidad o ya tiene un sistema computacional embebido o próximamente lo tendrá.

2.10 IMU: Inertial Measurement Unit.

Una unidad de medición inercial o IMU (del inglés inertial measurement unit), es un dispositivo electrónico que mide e informa acerca de la velocidad, orientación y fuerzas gravitacionales de un aparato, usando una combinación de acelerómetros y giróscopos. Las unidades de medición inercial son normalmente usadas para

maniobrar aviones, incluyendo vehículos aéreos no tripulados, entre muchos otros usos, y además naves espaciales, incluyendo transbordadores, satélites y aterrizadores. (naylor mechatronics, s.f.)

2.10.1 MEMS

Una primera definición general para los MEMS es que se trata de sistemas embebidos y especializados en miniatura que involucran uno o más componentes o estructuras micro maquinadas que actúan como sensores o actuadores para habilitar funciones en un nivel superior dentro de la estructura de un sistema más complejo.

Sistemas Microelectromecánicos (Microelectromechanical Systems, MEMS) se refieren a la tecnología electromecánica, micrométrica y sus productos, y a escalas relativamente más pequeñas (escala nanométrica) se fusionan en sistemas nanoelectromecánicos (Nanoelectromechanical Systems, NEMS) y Nanotecnología. MEMS también se denominan 'Micro Máquinas' (en Japón) o 'Tecnología de Micro Sistemas' - MST (en Europa). Los MEMS son independientes y distintos de la hipotética visión de la nanotecnología molecular o Electrónica Molecular. MEMS en general varían en tamaño desde un micrómetro (una millonésima parte de un metro) a un milímetro (milésima parte de un metro). En este nivel de escala de tamaño, las construcciones de la física clásica no son siempre ciertas. Debido a la gran superficie en relación al volumen de los MEMS, los efectos de superficie como electrostática y viscosidad dominan los efectos de volumen tales como la inercia o masa térmica. El análisis de elementos finitos es una parte importante del diseño de MEMS. La tecnología de sensores ha hecho progresos significativos debido a los MEMS. La complejidad y el rendimiento avanzado de los sensores MEMS ha ido evolucionando con las diferentes generaciones de sensores MEMS.

El potencial de las máquinas muy pequeñas fue apreciado mucho antes de que existiera la tecnología que pudiera construirlas - véase, por ejemplo, la famosa lectura de 1959 de Feynman "Hay mucho espacio en lo pequeño". Los MEMS se convirtieron en prácticos una vez que pudieran ser fabricados utilizando

modificación de tecnologías de fabricación de semiconductores, normalmente utilizadas en electrónica. Estos incluyen moldeo y galvanoplastia, grabado húmedo (KOH, TMAH) y grabado en seco (RIE y DRIE), el mecanizado por electro descarga (EDM), y otras tecnologías capaces de fabricar dispositivos muy pequeños.

Existen diferentes tamaños de empresas con importantes programas MEMS. Las empresas más grandes se especializan en la fabricación de componentes de bajo costo alto volumen o paquetes de soluciones para los mercados finales como el automotriz, biomedicina, y electrónica. El éxito de las pequeñas empresas es ofrecer valor en soluciones innovadoras y absorber el costo de fabricación con altos márgenes de ventas. Tanto las grandes como las pequeñas empresas realizan trabajos de I + D para explorar la tecnología MEMS.

Uno de los mayores problemas de los MEMS autónomos es la ausencia de micro fuentes de energía con alta densidad de corriente, potencia y capacidad eléctrica. (investigacion y ciencia, 2019)

Los avances en el campo de los semiconductores están dando lugar a circuitos integrados con características tridimensionales e incluso con piezas móviles. Estos dispositivos, llamados Sistemas Microelectromecánicos (MEMS), pueden resolver muchos problemas que un microprocesador más el software o configuración no ASIC (Chip integrados de aplicación específica) no pueden. La tecnología MEMS puede aplicarse utilizando un sin número de diferentes materiales y técnicas de fabricación; la elección dependerá del tipo de dispositivo que se está creando y el sector comercial en el que tiene que operar.

2.10.1.1 Part 1. Acelerómetro

Para entender esta unidad se comienza con el acelerómetro. Cuando se piensa en acelerómetros, a menudo es útil imaginar una caja en forma de cubo con una bola dentro. Puede imaginar algo más como una pelota:

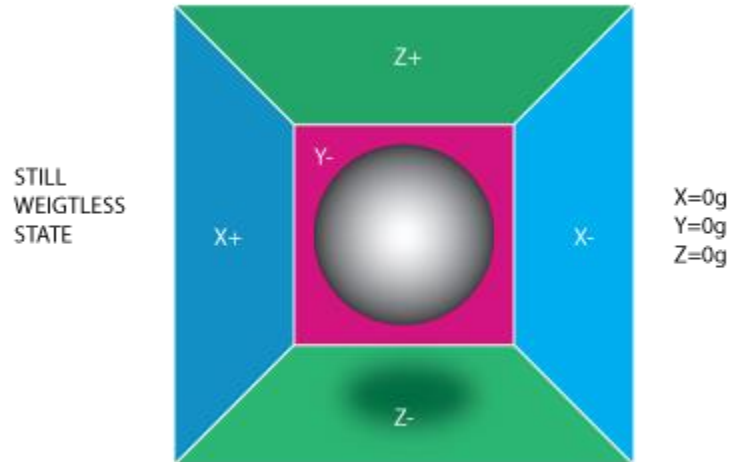


Figura 2-49. Representación de un acelerómetro

Si se toma esta caja en un lugar sin campos de gravitación o sin otros campos que puedan afectar la posición de la pelota, la pelota simplemente flotará en el centro de la caja. Imaginar al menos una nave espacial que orbita alrededor del planeta donde todo está en un estado sin peso. En la imagen de arriba puede ver que se asigna a cada eje un par de paredes (eliminamos la pared Y + para poder mirar dentro de la caja). Imaginar que cada pared es sensible a la presión. Si se mueve de repente la caja hacia la izquierda (la aceleramos con una aceleración de $1g = 9.8m/s^2$), la pelota golpeará la pared X-. Luego medimos la fuerza de presión que la bola aplica a la pared y emitimos un valor de -1g en el eje X.

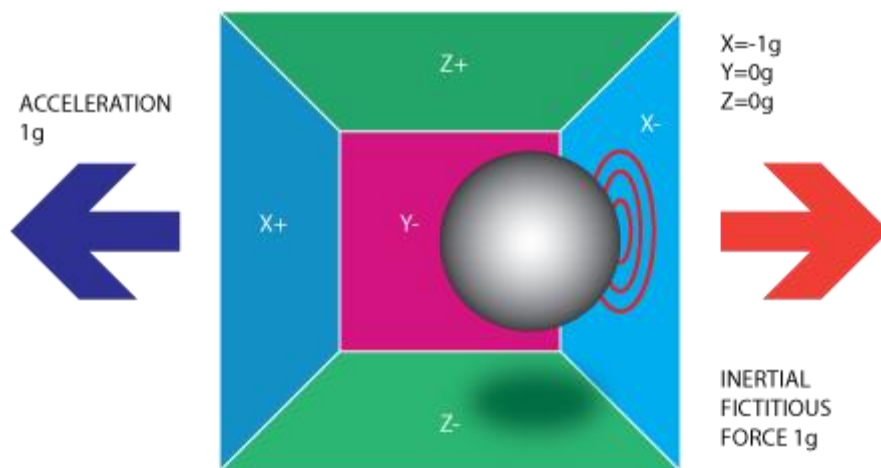


Figura 2-50. Movimiento de la representación de un acelerómetro

Tener en cuenta que el acelerómetro detectará realmente una fuerza que se dirige en la dirección opuesta al vector de aceleración. Esta fuerza a menudo se llama fuerza inercial o fuerza ficticia. Una cosa que debe aprender de esto es que un acelerómetro mide la aceleración indirectamente a través de una fuerza que se aplica a una de sus paredes (según nuestro modelo, podría ser un resorte u otra cosa en los acelerómetros de la vida real). Esta fuerza puede ser causada por la aceleración, pero como veremos en el siguiente ejemplo, no siempre es causada por la aceleración.

Si tomamos se toma el modelo anterior y se coloca en la tierra, la bola caerá sobre la pared Z y aplicará una fuerza de 1 g en la pared inferior, como se muestra en la imagen a continuación:

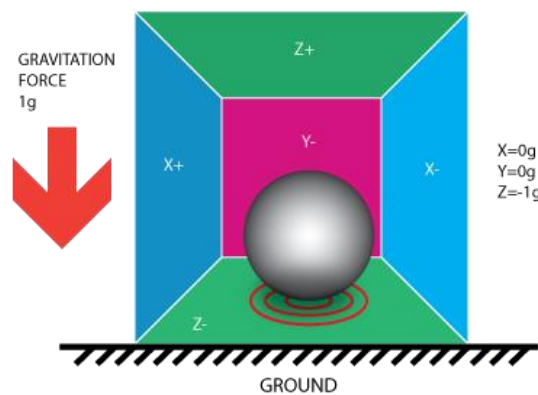


Figura 2-51. Representación de la fuerza en el acelerómetro

En este caso, el cuadro no se mueve, pero aún obtenemos una lectura de -1g en el eje Z. La presión que la pelota ha aplicado sobre la pared fue causada por una fuerza de gravitación. En teoría, podría ser un tipo diferente de fuerza; por ejemplo, si imagina que nuestra bola es metálica, colocar un imán al lado de la caja podría mover la bola para que golpee otra pared. Esto se dijo solo para demostrar que, en esencia, el acelerómetro mide la fuerza, no la aceleración. Simplemente sucede que la aceleración causa una fuerza de inercia que es capturada por el mecanismo de detección de fuerza del acelerómetro.

Si bien este modelo no es exactamente cómo se construye un sensor MEMS, a menudo es útil para resolver problemas relacionados con el acelerómetro. En

realidad, hay sensores similares que tienen bolas metálicas en el interior, se llaman interruptores de inclinación, sin embargo, son más primitivos y generalmente solo pueden decir si el dispositivo está inclinado dentro de un rango o no, no el grado de inclinación.

Hasta ahora se ha analizado la salida del acelerómetro en un solo eje y esto es todo lo que obtendrá con un acelerómetro de un solo eje. El valor real de los acelerómetros triaxiales proviene del hecho de que pueden detectar fuerzas de inercia en los tres ejes. Volvamos a nuestro modelo de caja y giremos la caja 45 grados hacia la derecha. La pelota tocará 2 paredes ahora: Z- y X- como se muestra en la imagen a continuación:

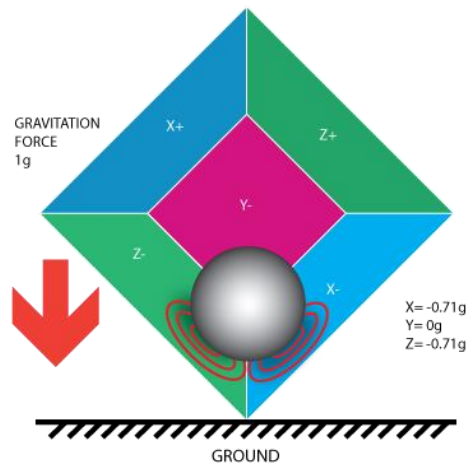


Figura 2-52. Representación de el comportamiento de un acelerómetro triaxial

En el modelo anterior, se fija la fuerza de gravitación y se gira la caja imaginaria. En los últimos 2 ejemplos se ha analizado la salida en 2 posiciones de caja diferentes, mientras que el vector de fuerza se mantuvo constante. Si bien esto fue útil para comprender cómo interactúa el acelerómetro con las fuerzas externas, es más práctico realizar cálculos si fijamos el sistema de coordenadas a los ejes del acelerómetro e imaginamos que el vector de fuerza gira a nuestro alrededor.

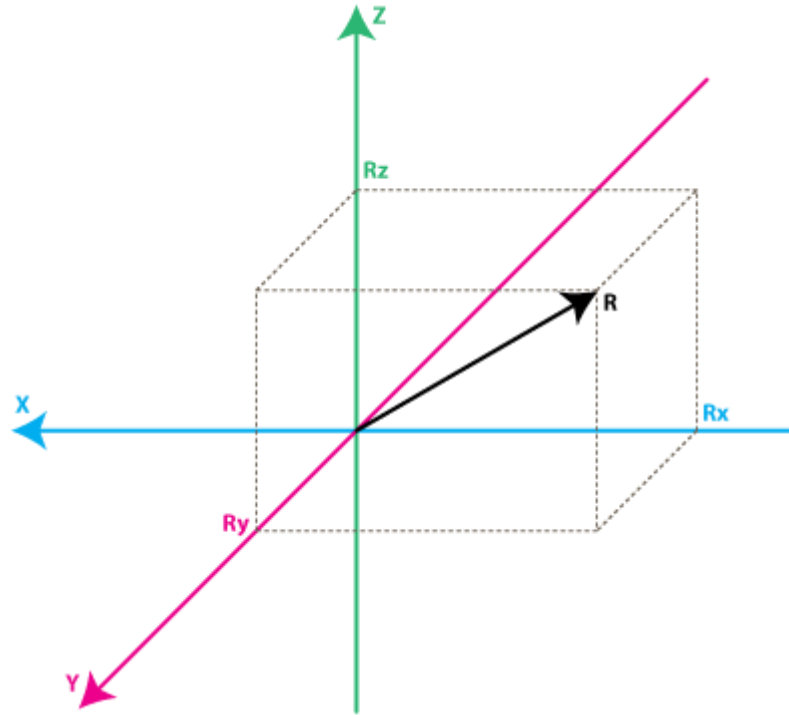


Figura 2-53. Sistema de coordenadas a los ejes del acelerómetro

Revisando el modelo anterior, se conservó los colores de los ejes para que se pueda hacer una transición mental del modelo anterior al nuevo. Solo se visualiza que cada eje en el nuevo modelo es perpendicular a las caras respectivas de la caja en el modelo anterior. El vector R es el vector de fuerza que está midiendo el acelerómetro (podría ser la fuerza de gravitación o la fuerza de inercia de los ejemplos anteriores o una combinación de ambos). Rx, Ry, Rz son proyecciones del vector R en los ejes X, Y, Z. Tener en cuenta la siguiente relación:

$$R^2 = R_x^2 + R_y^2 + R_z^2 \quad (\text{Eq. 1})$$

Que es básicamente el equivalente del teorema de Pitágoras en 3D.

Los valores Rx, Ry, Rz están realmente relacionados linealmente con los valores que generará un acelerómetro de la vida real y que se puede usar para realizar varios cálculos.

La mayoría de los acelerómetros se clasificarán en dos categorías: digital y analógica.

Los acelerómetros digitales le proporcionarán información utilizando un protocolo en serie como I2C, SPI o USART, mientras que los acelerómetros analógicos emitirán un nivel de voltaje dentro de un rango predefinido que debe convertir a un valor digital utilizando un módulo ADC.

Un ejemplo simple, se supone que para este ejemplo el módulo ADC de 10 bits y da los siguientes valores para los tres canales (ejes) del acelerómetro:

$$AdcRx = 586, AdcRy = 630, AdcRz = 561$$

Cada módulo ADC tendrá un voltaje de referencia, supongamos en este caso es 3.3V. Para convertir un valor ADC de 10 bits a voltaje, se utiliza la siguiente fórmula:

$$VoltsRx = \frac{AdcRx \cdot Vref}{1023}$$

Aplicando esta fórmula a los 3 canales se obtiene:

$$VoltsRx = \frac{586 \cdot 3.3}{1023} = 1.89$$

$$VoltsRy = \frac{630 \cdot 3.3}{1023} = 2.03$$

$$VoltsRz = \frac{561 \cdot 3.3}{1023} = 1.81$$

Cada acelerómetro tiene un nivel de voltaje de cero-g, se puede encontrar en las especificaciones, este es el voltaje que corresponde a 0 g. Para obtener un valor de voltaje con signo necesitamos calcular el cambio desde este nivel. Digamos que el nivel de voltaje de 0g es $V_{zeroG} = 1.65V$. se los cambios de voltaje del voltaje de cero- g de la siguiente manera:

$$\Delta VRx = 1.89V - 1.65V = 0.24V$$

$$\Delta VRy = 2.03V - 1.65V = 0.38V$$

$$\Delta VRz = 1.81V - 1.65V = 0.16V$$

Ahora se tiene las lecturas de acelerómetro en voltios, todavía no está en g (9.8 m/s^2), para hacer la conversión final se aplica la sensibilidad del acelerómetro,

generalmente expresada en mV/g. para este caso la sensibilidad = 478.5mV/g = 0.4785V/g. Los valores de sensibilidad se pueden encontrar en las especificaciones del acelerómetro. Para obtener los valores de fuerza finales expresados en g se utiliza la siguiente fórmula:

$$Rx = \frac{\Delta v Rx}{Sens}$$

$$Rx = 0.24V / 0.4785V/g = \sim 0.5g$$

$$Ry = 0.38V / 0.4785V/g = \sim 0.79g$$

$$Rz = 0.16V / 0.4785V/g = \sim 0.33g$$

Entonces la formula general a un componente del vector de fuerza expresado en g. **(Eq.2)**

$$Rx = \frac{\frac{AdcRx * Vref}{1023 - VCeroG}}{Sens}$$

$$Ry = \frac{\frac{AdcRx * Vref}{1023 - VCeroG}}{Sens}$$

$$Rz = \frac{\frac{AdcRx * Vref}{1023 - VCeroG}}{Sens}$$

Ahora que se tienen las 3 componentes que se definen en el vector de fuerza de inercia, si el dispositivo no está sujeto a otras fuerzas que no sean la gravitación, se puede suponer que esta es la dirección de nuestro vector de fuerza de gravitación. Si se desea calcular la inclinación del dispositivo con respecto al suelo, se puede calcular el ángulo entre este vector y el eje Z. Si también está interesado en la dirección de inclinación por eje, se puede dividir este resultado en 2 componentes: inclinación en los ejes X e Y que se pueden calcular como el ángulo entre el vector de gravitación y los ejes X/Y.

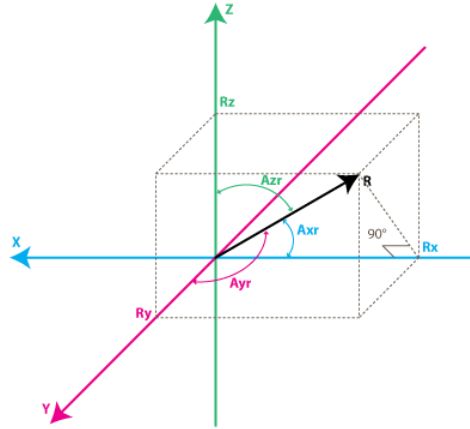


Figura 2-54. Vector de fuerzas del sistema de coordenadas a los ejes del acelerómetro

Los ángulos de interés son los ángulos entre los ejes X, Y, Z y el vector de fuerza R. Para ejemplo se definen estos ángulos como Axr, Ayr, Azr. Se Puede observar en el triángulo rectángulo formado por R y Rx que:

$$\cos (A_{xr}) = R_x/R, \text{ y de manera similar:}$$

$$\cos (A_{yr}) = R_y/R$$

$$\cos (A_{zr}) = R_z /R$$

Se puede deducir de la ecuación 1 que $R = \sqrt{R_x^2 + R_y^2 + R_z^2}$.

Se puede los ángulos usando la función arccos () (la función inversa cos ()):

$$A_{xr} = \arccos (R_x/R)$$

$$A_{yr} = \arccos (R_y/R)$$

$$A_{zr} = \arccos (R_z/R)$$

Algunas anotaciones más útiles:

$$\text{CosX} = \cos(A_{xr}) = R_x/R$$

$$\text{CosY} = \cos(A_{yr}) = R_y/R$$

$$\text{CosZ} = \cos(A_{zr}) = R_z/R$$

Esto se llama Coseno de dirección, y básicamente representa el vector unitario (vector con longitud 1) que tiene la misma dirección que el vector R. Puede verificar fácilmente que:

$$\sqrt{\cos^2 X + \cos^2 Y + \cos^2 Z} = 1$$

2.10.1.2 Part 2- Giroscopio

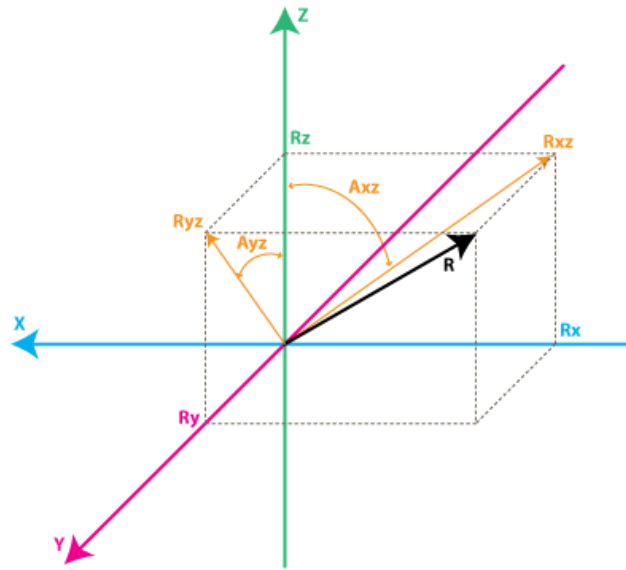


Figura 2-55. sistema de coordenadas a los ejes del giroscopio

Cada canal de giroscopio mide la rotación alrededor de uno de los ejes. Por ejemplo, un giroscopio de 2 ejes medirá la rotación alrededor (o algunos pueden decir "acerca de") los ejes X e Y. Para expresar esta rotación en números, Primero se define:

Rxz: Es la proyección del vector de fuerza de inercia R en el plano XZ

Ryz: Es la proyección del vector de fuerza de inercia R en el plano YZ

Del triángulo rectángulo formado por Rxz y Rz, se usa el teorema de Pitágoras y se obtiene:

$$R_{xz}^2 = R_x^2 + R_z^2$$

y de manera similar:

$$R_{yz}^2 = R_y^2 + R_z^2$$

También se debe de tener en cuenta que:

$R^2 = R_{xz}^2 + R_y^2$, esto puede derivarse de la ecuación 1 y las ecuaciones anteriores, o puede derivarse del triángulo rectángulo formado por R y Ryz

$$R^2 = R_{yz}^2 + R_x^2$$

No se utilizarán estas fórmulas en este documento, pero es útil tener en cuenta la relación entre todos los valores del modelo.

En su lugar, se va a definir el ángulo entre el eje Z y Rxz, los vectores Ryz de la siguiente manera:

Axz: es el ángulo entre Rxz (proyección de R en el plano XZ) y el eje Z

Ayz: es el ángulo entre el Ryz (proyección de R en el plano YZ) y el eje Z

Ahora se está cerca de lo que mide el giroscopio. El giroscopio mide la tasa de cambios de los ángulos definidos anteriormente. En otras palabras, generará un valor que está relacionado linealmente con la velocidad de cambio de estos ángulos. Para explicar esto, sé supondrá que se ha medido el ángulo de rotación alrededor del eje Y (que sería el ángulo de Axz) en el tiempo t0, y lo definimos como Axz0, luego medimos este ángulo en un momento posterior t1 y fue Axz1. La tasa de cambio se calculará de la siguiente manera:

$$Vel_{Axz} = \frac{Axz1 - Axz0}{t1 - t0}$$

Si expresamos Axz en grados y el tiempo en segundos, este valor se expresará en deg/s. Esto es lo que mide un giroscopio.

En la práctica, un giroscopio (a menos que sea un giroscopio digital especial) rara vez da un valor expresado en deg/s. Al igual que para el acelerómetro, se obtendrá un valor de ADC que se deberá convertir a deg/s utilizando una fórmula similar a la **Eq.2** que se ha definido para acelerómetro.

Eq.3

$$Vel_{Axz} = \frac{\frac{ADCGyroXZ * vREF}{1023 - vCeroVel}}{Sensibilidad}$$

$$Vel_{Ayz} = \frac{\frac{ADCGyroXZ * vREF}{1023 - vCeroVel}}{Sensibilidad}$$

AdcGyroXZ, AdcGyroYZ: Se obtienen del módulo ADC y representan los canales que miden la rotación de proyección del vector R en XZ respectivamente en planos YZ, lo que equivale a decir que la rotación se realizó alrededor de los ejes Y y X respectivamente.

Vref: Es el voltaje de referencia de ADC que se usa para este ejemplo es 3.3V.

VzeroVel: Es el voltaje de tasa cero, en otras palabras, el voltaje que genera el giroscopio cuando no está sujeto a ninguna rotación, para la placa Genérica es por ejemplo, 1.23V (puede encontrar estos valores en las especificaciones, pero se confía en las especificaciones, la mayoría de los giroscopios sufrirán un ligero desplazamiento después de la soldadura, por lo que mida la tasa VzeroVel para cada salida del eje con un voltímetro, por lo general, este valor no cambiará con el tiempo una vez que se haya soldado el giroscopio, si varía, Se debe indicar al usuario que mantenga el dispositivo en una posición fija durante el arranque para que los giroscopios se calibren.

Sensibilidad: Es la sensibilidad del giroscopio que se expresa en mV/(deg/s) a menudo escrito como mV/deg/s, básicamente dice cuántos mV aumentará la salida del giroscopio, si aumenta la velocidad de rotación en un deg/s. La sensibilidad de la placa genérica es, 2mV/deg/

Un ejemplo, se supondrá que la medición del ADC del módulo devuelve los siguientes valores:

$$\text{AdcGyroXZ} = 571$$

$$\text{AdcGyroYZ} = 323$$

Usando la fórmula anterior y los parámetros de especificaciones de la placa se obtiene que:

$$\text{RateAxz} = (571 * 3.3\text{V} / 1023 - 1.23\text{V}) / (0.002\text{V} / \text{deg} / \text{s}) = \sim 306 \text{ deg} / \text{s}$$

$$\text{RateAyz} = (323 * 3.3\text{V} / 1023 - 1.23\text{V}) / (0.002\text{V} / \text{deg} / \text{s}) = \sim -94 \text{ deg} / \text{s}$$

En otras palabras, el dispositivo gira alrededor del eje Y (o se puede decir que gira en el plano XZ) con una velocidad de 306 deg/s y alrededor del eje X (o se puede decir que gira en el plano YZ) con una velocidad de - 94 deg/s. Se debe tener en cuenta que el signo negativo significa que el dispositivo gira en la dirección opuesta a la dirección positiva convencional. Por convención, un sentido de rotación es positivo. Una la hoja de especificaciones del giroscopio mostrará qué dirección es positiva; de lo contrario, se tendrá que encontrarla experimentando con el dispositivo y observando qué dirección de rotación aumenta el voltaje en el pin de salida. Esto se hace mejor usando un osciloscopio ya que tan pronto como detenga la rotación, el voltaje volverá a caer al nivel de tasa cero. Si se está utilizando un multímetro, se deberá mantener una velocidad de rotación constante durante al menos unos segundos y observar el voltaje durante esta rotación, luego compararlo con el voltaje de velocidad cero. Si es mayor que el voltaje de tasa cero, significa que la dirección de rotación es positiva.

2.10.1.3 Part 3. Poniendo Todo Junto. Combinando datos de acelerómetro y giroscopio.

El primer paso para usar un dispositivo IMU combinado que combine un acelerómetro y un giroscopio es alinear sus sistemas de coordenadas. La forma más fácil de hacerlo es elegir el sistema de coordenadas del acelerómetro como su sistema de coordenadas de referencia. La mayoría de las hojas de datos del

acelerómetro mostrarán la dirección de los ejes X, Y, Z en relación con la imagen del chip o dispositivo físico. Por ejemplo, aquí están las direcciones de los ejes X, Y, Z como se muestra en las especificaciones de una placa genérica:

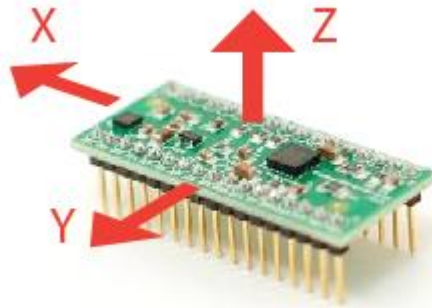


Figura 2-56. Placa genérica de un IMU

Los siguientes pasos son:

- Identificar las salidas del giroscopio que corresponden a los valores Vel_{Axz} , Vel_{Ayz} discutidos anteriormente.
- Determinar si estas salidas deben invertirse debido a la posición física del giroscopio en relación con el acelerómetro.
- No suponer que, si un giroscopio tiene una salida marcada como X o Y, corresponderá a cualquier eje en el sistema de coordenadas del acelerómetro, incluso si esta salida es parte de una unidad IMU. La mejor manera es probarlo.
- Comenzar por colocar el dispositivo en posición horizontal. Las salidas X e Y del acelerómetro emitirían un voltaje de cero g (por ejemplo, para la placa genérica esto es 1.65V).
- Luego se puede comenzar a girar el dispositivo alrededor del eje Y, otra forma de decirlo es que gira el dispositivo en el plano XZ, de modo que las salidas del acelerómetro X y Z cambien y la salida Y permanezca constante.

- Mientras gira el dispositivo a una velocidad constante, observe qué salida del giroscopio cambia, las otras salidas del giroscopio deben permanecer constantes.
- La salida del giroscopio que cambió durante la rotación alrededor del eje Y (rotación en el plano XZ) proporcionará el valor de entrada para AdcGyroXZ, a partir del cual calculamos VelAxz.
- El paso final es asegurar que la dirección de rotación corresponda a nuestro modelo, en algunos casos puede que tenga que invertir el valor de VelAxz debido a la posición física del giroscopio en relación con el acelerómetro.
- Realizar nuevamente la prueba anterior, girando el dispositivo alrededor del eje Y, esta vez controlando la salida X del acelerómetro (AdcRx en nuestro modelo). Si AdcRx crece (los primeros 90 grados de rotación desde la posición horizontal), entonces AdcGyroXZ debería disminuir. Esto se debe al hecho de que se está monitoreando el vector de gravitación y cuando el dispositivo gira en una dirección, el vector girará en dirección opuesta (en relación con el sistema de coordenadas del dispositivo, que se está usando). Entonces, de lo contrario, se debe invertir VelAxz, puede lograr esto introduciendo un factor de signo en la ecuación 3, de la siguiente manera:

$$\text{RateAxz} = \text{InvertAxz} * (\text{AdcGyroXZ} * \text{Vref} / 1023 - \text{VzeroRate}) / \text{Sensitivity} ,$$

donde se invierteAxz es 1 o -1

Se puede hacer la misma prueba para VelAyz, girando el dispositivo alrededor del eje X, y puede identificar qué salida del giroscopio corresponde a VelAyz, y si es necesario invertirla. Una vez que tenga el valor para InvertAyz, debe usar la siguiente fórmula para calcular VelAyz:

$$\text{RateAyz} = \text{InvertAyz} * (\text{AdcGyroYZ} * \text{Vref} / 1023 - \text{VzeroRate}) / \text{Sensibilidad}$$

A partir de este momento, se considera que se ha configurado el IMU de tal manera que se pueda calcular los valores correctos para A_{xr} , A_{yr} , A_{zr} (como se define en la Parte 1. Acelerómetro) y Vel_{Axz} , Vel_{Ayz} (como se define en la Parte 2- Giroscopio)

A continuación, se analiza las relaciones entre estos valores que resultan útiles para obtener una estimación más precisa de la inclinación del dispositivo con respecto al plano del suelo.

No siempre se puede confiar en los datos del acelerómetro al 100%. Hay varias razones, ya que el acelerómetro mide la fuerza de inercia, tal fuerza puede ser causada por la gravitación (e idealmente solo por la gravitación), pero también puede ser causada por la aceleración (movimiento) del dispositivo. Como resultado, incluso si el acelerómetro está en un estado relativamente estable, sigue siendo muy sensible a la vibración y al ruido mecánico en general. Esta es la razón principal por la cual la mayoría de los sistemas IMU usan un giroscopio para suavizar los errores del acelerómetro.

Sin embargo, el giroscopio no está libre de ruido, ya que mide la rotación, es menos sensible a los movimientos mecánicos lineales, el tipo de ruido que sufre el acelerómetro, sin embargo, los giroscopios tienen otros tipos de problemas como, por ejemplo, la deriva (no volver al valor de velocidad cero) cuando se detiene la rotación). Sin embargo, al promediar los datos que provienen del acelerómetro y el giroscopio, podemos obtener una estimación relativamente mejor de la inclinación actual del dispositivo de la que obtendríamos utilizando solo los datos del acelerómetro.

En los próximos pasos se presentará un algoritmo inspirado en algunas ideas utilizadas en el filtro de Kalman, Antes de eso se tiene que ver primero que se quiere calcular en el algoritmo. La dirección del vector de fuerza de gravitación $R = [R_x, R_y, R_z]$ de la que se pueden derivar otros valores como A_{xr} , A_{yr} , A_{zr} o $\cos X$, $\cos y$, $\cos Z$ que dará una idea sobre la inclinación del dispositivo. En relación con el plano de tierra,

Del cual uno podría concluir que ya se tienen los valores de Rx, Ry, Rz de la ecuación 2 en la Parte 1 pero estos valores se derivan únicamente de los datos del acelerómetro, por lo que si se usaran directamente en su aplicación, se podría obtener más ruido del que la aplicación pueda tolerar. Para ello se debe redefinir las medidas del acelerómetro de la siguiente manera:

Racc: Es el vector de fuerza de inercia medido por el acelerómetro, que consta de los siguientes componentes (proyecciones en los ejes X, Y, Z):

$$RxAcc = (AdcRx * Vref / 1023 - VzeroG) / Sensitivity$$

$$RyAcc = (AdcRy * Vref / 1023 - VzeroG) / Sensitivity$$

$$RzAcc = (AdcRz * Vref / 1023 - VzeroG) / Sensitivity$$

Hasta ahora se tiene un conjunto de valores medidos que se pueden obtener únicamente de los valores de ADC del acelerómetro. Se llamará a este conjunto de datos un "vector" y se usará en la siguiente notación.

$$Racc = [RxAcc, RyAcc, RzAcc]$$

Debido a que estos componentes de Racc se pueden obtener de los datos del acelerómetro, se puede considerar como una entrada al algoritmo.

Se debe de tener en cuenta que debido a que Racc mide la fuerza de gravitación, estará en lo correcto si supone que la longitud de este vector definida de la siguiente manera es igual o cercana a 1 g.

$$| Racc | = \text{SQRT} (RxAcc^2 + RyAcc^2 + RzAcc^2),$$

Se tiene que actualizar este vector de la siguiente manera:

$$Racc \text{ (normalizado)} = [RxAcc / | Racc |, RyAcc / | Racc |, RzAcc / | Racc |].$$

Esto asegurará que la longitud de su vector Racc normalizado sea siempre 1

A continuación, se presenta un nuevo vector y se llamará

$$Rest = [RxEst, RyEst, RzEst]$$

Este será el resultado del algoritmo, estos son valores corregidos basados en datos de giroscopio y en datos estimados anteriores.

Esto es lo que hará al algoritmo:

- el acelerómetro dice: "Ahora estás en la posición Racc"
- luego se corrige esta información con los datos del giroscopio, así como con los datos anteriores de Rest y se genera un nuevo vector estimado Rest.

- Se considera que Rest es "mejor apuesta" en cuanto a la posición actual del dispositivo.

Se comenzará con la secuencia confiando en que el acelerómetro y asignando:

$$\text{Rest}(0) = \text{Racc}(0)$$

Por cierto, se debe de recordar que Rest y Racc son vectores, por lo que la ecuación anterior es solo una forma simple de escribir 3 conjuntos de ecuaciones y evitar la repetición:

$$\text{RxEst}(0) = \text{RxAcc}(0)$$

$$\text{RyEst}(0) = \text{RyAcc}(0)$$

$$\text{RzEst}(0) = \text{RzAcc}(0)$$

A continuación, se hacen mediciones regulares a intervalos de tiempo iguales de T segundos, y se obtienen nuevas mediciones que se definen como Racc (1), Racc (2), Racc (3) y así sucesivamente. También se emiten nuevas estimaciones en cada intervalo de tiempo REST(1), REST(2), REST(3), etc.

Supongamos que estamos en el paso n. Tenemos dos conjuntos de valores conocidos que nos gustaría usar:

Rest(n-1) – estimado previo, with Rest(0) = Racc(0)

Racc(n) – actual medición del acelerometro

Antes de que se pueda calcular Rest(n), se debe de introducir un nuevo valor medido, que se puede obtener del giroscopio y una estimación previa.

Se llamará Rgyro, y también es un vector que consta de 3 componentes:

$$R_{gyro} = [R_{xGyro}, R_{yGyro}, R_{zGyro}]$$

Se calculará este vector un componente a la vez. Comenzando con RxGyro.

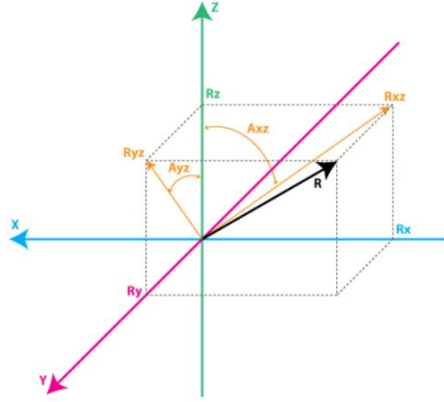


Figura 2-57. vector de un componente a la vez. RxGyro.

Se comienza observando la siguiente relación en el modelo del giroscopio, a partir del triángulo rectángulo formado por Rz y Rxz se puede deducir que:

$$\tan (Axz) = Rx / Rz \Rightarrow Axz = \text{atan2} (Rx, Rz)$$

Atan2 devuelve valores en el rango de $(-\pi, \pi)$ en lugar de $(-\pi / 2, \pi / 2)$ como los devuelve atan, y toma 2 argumentos en lugar de uno. Permite convertir los dos valores de Rx, Rz a ángulos en el rango completo de 360 grados $(-\pi$ a $\pi)$.

Entonces, conociendo RxEst (n-1) y RzEst (n-1) se puede encontrar:

$$Axz (n-1) = \text{atan2} (RxEst (n-1), RzEst (n-1)).$$

Se tiene que recordar que el giroscopio mide la tasa de cambio del ángulo de Axz. Entonces se puede estimar el nuevo ángulo Axz (n) de la siguiente manera:

$$Axz (n) = Axz (n-1) + \text{RateAxz} (n) * T$$

Recordar que se puede obtener RateAxz de las lecturas del ADC de giroscopio. Una fórmula más precisa puede usar una tasa de rotación promedio calculada de la siguiente manera:

$$\text{RateAxzAvg} = (\text{RateAxz} (n) + \text{RateAxz} (n-1)) / 2$$

$$Axz(n) = Axz(n-1) + RateAxzAvg * T$$

De la misma manera se puede encontrar:

$$Ayz(n) = Ayz(n-1) + RateAyz(n) * T$$

Se puede escribir la longitud del vector Rgyro de la siguiente manera:

$$|Rgyro| = \text{SQRT}(RxGyro^2 + RyGyro^2 + RzGyro^2)$$

Además, debido a que se normalizo el vector Racc, se puede suponer que su longitud es 1 y que no ha cambiado después de la rotación, por lo que es relativamente seguro escribir:

$$|Rgyro| = 1$$

Se puede adoptar una notación temporal más corta para los cálculos a continuación:

$$x = RxGyro, y = RyGyro, z = RzGyro$$

Usando las relaciones anteriores se puede escribir:

$$x = x / 1 = x / \text{SQRT}(x^2 + y^2 + z^2)$$

Ahora se divide el numerador y el denominador de la fracción por $\text{SQRT}(x^2 + z^2)$

$$x = (x / \text{SQRT}(x^2 + z^2)) / \text{SQRT}((x^2 + y^2 + z^2) / (x^2 + z^2))$$

Se debe de tener en cuenta que $x / \text{SQRT}(x^2 + z^2) = \sin(Axz)$, entonces:

$$x = \sin(Axz) / \text{SQRT}(1 + y^2 / (x^2 + z^2))$$

Ahora multiplicar el numerador y el denominador de la fracción dentro de SQRT por z^2

$$x = \sin(Axz) / \text{SQRT}(1 + y^2 * z^2 / (z^2 * (x^2 + z^2)))$$

SE debe de tener en cuenta que $z / \text{SQRT}(x^2 + z^2) = \cos(Axz)$ ey $y / z = \tan(Ayz)$, así que finalmente:

$$x = \sin(Axz) / \text{SQRT}(1 + \cos(Axz)^2 * \tan(Ayz)^2)$$

Volviendo a la notación, se obtiene:

$$RxGyro = \sin(Axz(n)) / \sqrt{1 + \cos(Axz(n))^2 * \tan(Ayz(n))^2}$$

de la misma manera se encuentra que:

$$RyGyro = \sin(Ayz(n)) / \sqrt{1 + \cos(Ayz(n))^2 * \tan(Axz(n))^2}$$

Nota: es posible simplificar aún más esta fórmula. Al dividir ambas partes de la fracción entre $\sin(Axz(n))$ obtienes:

$$RxGyro = 1 / \sqrt{1 / \sin(Axz(n))^2 + \cos(Axz(n))^2 / \sin(Axz(n))^2 * \tan(Ayz(n))^2}$$

$$RxGyro = 1 / \sqrt{1 / \sin(Axz(n))^2 + \cot(Axz(n))^2 * \sin(Ayz(n))^2 / \cos(Ayz(n))^2}$$

$$\text{ahora sumar y restar } \cos(Axz(n))^2 / \sin(Axz(n))^2 = \cot(Axz(n))^2$$

$$RxGyro = 1 / \sqrt{1 / \sin(Axz(n))^2 - \cos(Axz(n))^2 / \sin(Axz(n))^2 + \cot(Axz(n))^2 * \sin(Ayz(n))^2 / \cos(Ayz(n))^2 + \cot(Axz(n))^2}$$

y al agrupar los términos 1 y 2 y luego 3 y 4 se obtiene:

$$RxGyro = 1 / \sqrt{1 + \cot(Axz(n))^2 * \sec(Ayz(n))^2}, \text{ donde } \cot(x) = 1 / \tan(x) \text{ y } \sec(x) = 1 / \cos(x)$$

Esta fórmula usa solo 2 funciones trigonométricas y puede ser computacionalmente menos costosa. Si tiene el programa Mathematica puede verificarlo

$$\text{evaluando FullSimplify} [\sin[A]^2 / (1 + \cos[A]^2 * \tan[B]^2)]$$

Ahora, finalmente podemos encontrar:

$$RzGyro = \text{Signo}(RzGyro) * \sqrt{1 - RxGyro^2 - RyGyro^2}.$$

Donde: $\text{Sign}(RzGyro) = 1$ cuando $RzGyro \geq 0$, y $\text{Sign}(RzGyro) = -1$ cuando $RzGyro < 0$.

Una forma simple de estimar esto es tomar:

$$\text{Signo (RzGyro)} = \text{Signo (RzEst (n-1))}$$

En la práctica, tenga cuidado cuando RzEst (n-1) esté cerca de 0. En este caso, puede omitir la fase giroscópica y asignar: Rgyro = Descanso (n-1). Rz se usa como referencia para calcular los ángulos de Axz y Ayz y cuando está cerca de 0, los valores pueden desbordarse y provocar malos resultados. Estará en el dominio de grandes números de coma flotante donde las implementaciones de funciones $\tan()$ / $\arctan()$ pueden carecer de precisión.

Así que recapitulemos lo que tenemos hasta ahora, estamos en el paso n de nuestro algoritmo y hemos calculado los siguientes valores:

Racc - lecturas actuales de nuestro acelerómetro

Rgyro: obtenido de Rest (n-1) y lecturas actuales del giroscopio

¿Qué valores utilizamos para calcular la estimación actualizada Rest (n)? Probablemente hayas adivinado que usaremos ambos. Utilizaremos un promedio ponderado para que:

$$\text{Resto (n)} = (\text{Racc} * w1 + \text{Rgyro} * w2) / (w1 + w2)$$

Podemos simplificar esta fórmula dividiendo tanto el numerador como el denominador de la fracción por w1.

$$\text{Resto (n)} = (\text{Racc} * w1 / w1 + \text{Rgyro} * w2 / w1) / (w1 / w1 + w2 / w1)$$

y después de sustituir $w2 / w1 = w\text{Gyro}$ obtenemos:

$$\text{Descanso (n)} = (\text{Racc} + \text{Rgyro} * w\text{Gyro}) / (1 + w\text{Gyro})$$

En la fórmula anterior, wGyro nos dice cuánto confiamos en nuestro giroscopio en comparación con nuestro acelerómetro. Este valor se puede elegir experimentalmente, por lo general, los valores entre 5 y 20 generarán buenos resultados.

La principal diferencia de este algoritmo con respecto al filtro de Kalman es que este peso es relativamente fijo, mientras que en el filtro de Kalman los pesos se actualizan permanentemente en función del ruido medido de las lecturas del acelerómetro. El filtro Kalman se enfoca en brindarle "los mejores" resultados teóricos, mientras que este algoritmo puede brindarle resultados "suficientemente buenos" para su aplicación práctica. Puede implementar un algoritmo que ajuste w_{Gyro} dependiendo de algunos factores de ruido que mida, pero los valores fijos funcionarán bien para la mayoría de las aplicaciones.

Estamos a un paso de obtener nuestros valores estimados actualizados:

$$RxEst(n) = (RxAcc + RxGyro * w_{Gyro}) / (1 + w_{Gyro})$$

$$RyEst(n) = (RyAcc + RyGyro * w_{Gyro}) / (1 + w_{Gyro})$$

$$RzEst(n) = (RzAcc + RzGyro * w_{Gyro}) / (1 + w_{Gyro})$$

Ahora normalicemos este vector nuevamente:

$$R = \text{SQRT}(RxEst(n)^2 + RyEst(n)^2 + RzEst(n)^2)$$

$$RxEst(n) = RxEst(n) / R$$

$$RyEst(n) = RyEst(n) / R$$

$$RzEst(n) = RzEst(n) / R$$

2.10.2 *El MPU9250*

Descripción

El módulo MPU-9250, realiza detección de movimiento en 9 ejes, ya que combina giroscopio de 3 ejes(3DOF), un acelerómetro de 3 ejes(3DOF) y un magnetómetro AK8963 de 3 ejes (3DOF) en el mismo chip DMP (Procesador digital de movimiento) capaz de realizar complejos algoritmos para la captura del

movimiento de estos 9 ejes(9DOF). La comunicación puede realizarse tanto por bus SPI como por bus I2C, por lo que es sencillo obtener los datos medidos.

El módulo MPU-9250, es usado para ubicación, control, navegación de aplicaciones de interfaz usuario “sin contacto”, para atajos gestuales, juegos habilitados con movimientos, auriculares y juegos portátiles, controles remotos 3D, para pantallas y decodificadores conectados a Internet, sensores portátiles para salud, fitness y deportes.

Frecuentemente se encuentran integrados en módulos que incorporan la electrónica necesaria para conectarla de forma sencilla a un Arduino. En la mayoría de los módulos, esto incluye un regulador de voltaje que permite alimentar directamente a 5V.

ESPECIFICACIÓN Y CARACTERÍSTICAS

Módulo Modelo: GY-9250 /GY -6500

Sensor: MPU-9250/6500

Voltaje de alimentación: 3V a 5V (con regulador interno)

Grados de libertad (DoF): 9

Dimensiones: 26 mm x 15.5 mm x 2-6 mm (sin header)

Interfaz:

I2C (400kHz) y SPI (1MHz)

Registros de interrupción SPI (20MHz.)

Características del Giroscopio

Rango de Giroscopio: 250/500/1000/2000 grados/segundo

Corriente de Operación: 3.2mA

Corriente en modo sleep: 8uA

Características del Acelerómetro

Rango de Aceleración: $\pm 2 \pm 4 \pm 8 \pm 16$ g

Convertidor ADC: 16 bit

Corriente de Operación: 450 uA

Corriente en modo sleep: 8uA

Características del Magnetómetro

Funcionamiento por Efecto Hall

Corriente de Operación: 280µA a 8Hz

Rango del campo magnético: $\pm 4800\text{uT}$ (completo)

Sensibilidad: 0,6 $\mu\text{T/LSB}$

Resolución de salida:

14 bit (0.6 $\mu\text{T/LSB}$)

16 bit (15 $\mu\text{T/LSB}$)

2.11 Motores de DC (Corriente continua)

Los motores de corriente continua son los más comunes y económicos, y se pueden encontrar en la mayoría de los juguetes a pilas, constituidos, por lo general, por dos imanes permanentes fijados en la carcasa y una serie de bobinados de cobre ubicados en el eje del motor, que habitualmente suelen ser tres y a su vez son ampliamente usados a nivel industrial. Los motores de corriente continua permiten un amplio rango de velocidad y pueden proporcionar un alto par-motor con control más sencillo y económico que cualquier motor de corriente alterna. En la actualidad los métodos de control de velocidad se han ido desarrollando considerablemente y los más comunes son el control de velocidad por corriente de campo y el control de velocidad por corriente de armadura, que son técnicas de control no lineal.

Para poder analizar estos métodos se requiere del conocimiento físico del sistema, unidades de las constantes que aparecen en el modelo, selección adecuada de las variables de estado y conocimientos de desarrollo de ecuaciones

diferenciales utilizando la transformada de Laplace punto que se tocarán en capítulos posteriores.

2.12 Tipos de sensores

De forma similar a los seres vivos, los sensores facilitan la información necesaria para que los robots interpreten el mundo real. Todo robot debe tener al menos un sensor con el que interactuar. La mayoría de los sistemas robóticos incluyen al menos sensores de obstáculos (bumpers) y algún sensor de guiado por infrarrojos o ultrasonidos. Los sensores avanzados, además de detectar algo, son capaces de reportar una medida de lo detectado, como puede ser un sensor de temperatura, o un medidor de distancias ultrasónico, por ejemplo.

Para entender un poco mejor describiremos lo que es 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.

El número de sensores disponibles para las distintas magnitudes físicas es tan elevado que no se puede proceder racionalmente a su estudio sin clasificación previamente de acuerdo con algún criterio.

Desde el punto de vista de ingeniería electrónica, es más atractiva la clasificación de los sensores de acuerdo con el parámetro variable: resistencia, capacitivo, inductancia, añadiendo luego los sensores generadores de tensión, carga o corriente, y otros tipos. Este tipo de clasificación permite reducir en pocos grupos y se puede hacer un buen estudio de los acondicionadores ordinarios para las magnitudes más frecuentes.

2.12.1.1 Sensor laser

Un sensor de este tipo es un dispositivo que permite detectar con una gran precisión la posición en la que se encuentra un objeto en relación con un punto; tiene una importante aplicación al detectar la presencia significativa de pequeños objetos que pierden.

2.12.1.2 Ultrasónicos

El sensor ultrasónico emite cíclicamente un impulso acústico de alta frecuencia y corta duración. Este impulso se propaga a la velocidad del sonido por el aire. Al encontrar un objeto, es reflejado y vuelve como eco al sensor ultrasónico. Este último calcula internamente la distancia hacia el objeto, basado en el tiempo transcurrido entre la emisión de la señal acústica y la recepción de la señal de eco.

2.12.1.3 Encoders

La medida de posiciones lineales y angulares ha sido hasta el momento el único campo con abundantes desarrollos comerciales de sensores con salida digital, incluso antes de la era del microprocesador. Aun así, un grupo de estos sensores, solos denominados encoders incrementales, es de tipo “casi digital”.

2.12.1.4 Sensor de contacto

Los sensores de contacto son los dispositivos más simples de todos los sensores que podemos encontrarnos, ya que son interruptores que se activan o desactivan si se encuentran en contacto con un objeto, por lo que de esta manera reconoce la presencia de un objeto en un determinado lugar.

Su simplicidad de construcción añadido a su robustez, los hacen muy empleados en la construcción de robótica móvil. La descripción de este tipo de sensores es muy simple, el bumper es un conmutador de 2 posiciones con muelle de retorno a la posición de reposo y con una palanca de accionamiento más o menos larga según el modelo elegido.

2.13 Sistemas de alimentación de baterías

Los AGV's pueden contar con diferentes tipos de energía como lo pueden ser la energía eléctrica o combustibles fósiles. En el caso de este proyecto de tesis la fuente de energía que mejor se adaptaría a el contexto es la energía eléctrica ya que los combustibles se utilizan principalmente para AGV's de gran tamaño y es más fácil contar con un sistema de carga de este tipo ya que se puede realizar la carga desde cualquier tomacorriente.

2.13.1 *Técnicas de carga de las baterías*

Dentro de las técnicas de carga que se pueden realizar con energía eléctrica en los AGV's se tiene dos principalmente que son:

- Carga por oportunidad
- Carga de ciclo completo

La carga por oportunidad es el tipo de carga que se realiza cuando el vehículo está en espera y para este caso se cuentan con diferentes puntos dentro del sistema donde el AGV puede realizar las cargas de su batería, mientras que la carga de ciclo completo es el tipo de carga que se realiza cuando el AGV tiene que desplazarse a un sitio especial dentro del sistema para realizar su carga completa de las baterías. En el caso de este proyecto el sistema de carga ideal para el funcionamiento del robot sería una carga de ciclo completo ya que esta carga permitiría que el robot cargara sus baterías durante la noche cuando nadie lo está utilizando y durante el día se le pueda dar el uso necesario. Además, este tipo de carga permite que solo haya un punto de recarga dentro del sistema mientras que la carga por oportunidad necesita más puntos de recarga lo que aumentaría los costos de implementación del sistema de AGV.

Igualmente, esta técnica de carga de baterías se realizará de acuerdo al método "Carga en Cliente" donde el AGV se desplaza hasta el punto de carga de baterías y este es conectado directamente a la fuente de energía sin tener que extraer las baterías del robot.

2.13.2 *Batería*

Para las baterías del robot se usarán las baterías de litio ya que este tipo de baterías son las que más se utilizan en el mercado por su buen funcionamiento, su buena relación potencia/peso, su gran autonomía de trabajo, el poco mantenimiento que debe realizárseles y su poco impacto ambiental.

2.14 Sistemas de monitoreo Continuo

Son aquellos sistemas de visión que necesitan recibir una respuesta de la inspección visual en tiempo real. Esto implica que la retroalimentación de las imágenes que se analizan es continua.

Los sistemas de monitoreo continuo se utilizan en aplicaciones en las que es crítico que exista una secuencia totalmente continua de imágenes, que únicamente se logra analizando una secuencia de video.

El proceso de captura de imágenes se logra por medio de secuencias de video utilizando cámaras que transmiten la señal a una tarjeta capturadora de video, un software para visión por computadora tiene la tarea de realizar los procesos de análisis y decisión por medio de imágenes. El hecho de que la respuesta sea en tiempo real implica que estos sistemas requieren equipos con alta capacidad de procesamiento.

2.15 Sistema de Monitoreo Periódico

Los sistemas que realizan una inspección utilizando secuencia de imágenes que no son totalmente continuas, reciben el nombre de sistemas de inspección periódica.

Esta arquitectura es apropiada para aplicaciones en las que no se requiere una respuesta en tiempo real, ya que el tiempo de procesamiento y análisis no es crítico.

Al utilizar una web cam esta se encargará de realizar el análisis en las imágenes. Un ejemplo es un sistema de detección de defectos en una línea de inspección de un producto que se mueve a través de una banda eléctrica a razón de un producto cada 4 segundos. El sistema utilizaría entonces una web cam para capturar una imagen cada 4 segundos, la almacene en disco, se realice el proceso de análisis de la imagen y se clasifique el producto en aceptado o defectuoso.

Se debe tener presente las limitantes que posee, por no contar con una secuencia completa de imágenes y no tener la capacidad de procesamiento necesaria para una respuesta más rápida.

Una ventaja que tiene esta arquitectura frente a la de monitoreo continuo es que el costo de esta es menor.

2.16 Etapa de Control

Se trata de un sistema que mantiene comunicación entre el hardware de control y las variables con las cuales se mantiene la posición de la plataforma. Están ligadas a la imagen que tiene el usuario de la condición actual de la plataforma, sin embargo, no son las mismas. Es decir, el sistema de control es independiente de la representación que se usa para notificar al usuario la posición de los motores. Es por eso que se habla del sistema de control como un sistema aparte.

La prioridad del sistema de control es mantener una constante comunicación con la plataforma y no con el usuario. Ya que un movimiento fuera de control de la plataforma sería difícilmente controlado por el usuario, a diferencia de lo que podría hacer el programa por sí mismo. Además de las requisiciones del usuario, están sujetas a su propia velocidad de respuesta y a la prueba y error del usuario. Es por eso por lo que el sistema tiene como prioridad a la plataforma en sí mismo y no al usuario.

2.17 Conclusiones

A lo largo de este capítulo se han explorado los fundamentos y avances en la navegación autónoma de robots móviles, particularmente en relación con los Vehículos de Guiado Automático (AGV). Se han discutido las problemáticas clave asociadas a la precisión en la estimación de la pose de los robots, así como las soluciones propuestas en la literatura, destacando la relevancia de técnicas de fusión sensorial y métodos de control avanzados.

El análisis del estado del arte resalta la necesidad de abordar las limitaciones de los métodos convencionales de navegación y el potencial de enfoques modernos como los filtros de Kalman, la visión artificial y las arquitecturas de control distribuidas. Estos enfoques no solo mejoran la precisión en la navegación, sino que también incrementan la adaptabilidad y la robustez de los robots en entornos dinámicos y no estructurados.

Con este marco teórico, se justifica la investigación propuesta en este trabajo, centrada en el diseño e implementación de soluciones que optimicen la precisión y la eficiencia de los AGV en contextos industriales. Este esfuerzo no solo contribuye al avance de la robótica móvil, sino que también presenta un impacto significativo en la mejora de procesos industriales mediante la incorporación de vehículos autónomos más flexibles y eficientes.

Capítulo 3. Desarrollo

En este capítulo se abordan los aspectos clave en el diseño y desarrollo del prototipo de un AGV (Automated Guided Vehicle). Se analiza en detalle la estructura del AGV, la cual debe soportar componentes esenciales como las tarjetas electrónicas, sensores, motores, y la batería. Se exponen los requerimientos y especificaciones técnicas necesarias para construir un AGV robusto y funcional, incluyendo materiales de construcción ligeros que permitan una mayor movilidad y un sistema de comunicación inalámbrica confiable. Además, se describe el diseño de hardware, los componentes de control, y el proceso de integración de los sistemas de sensado y procesamiento de datos, principalmente mediante el uso de una Raspberry Pi 4 como unidad central de procesamiento.

3.1 Desarrollo físico del prototipo de AGV.

Las bases de diseño y construcción para el desarrollo de un AGV (Vehículo Guiado Automáticamente) se deben de establecer para una estructura funcional y optimizada. En donde se definen los requisitos fundamentales y se analizan los criterios de diseño necesarios para asegurar que la estructura soporte de manera adecuada los componentes esenciales —como sensores, motores y unidades de control—, logrando un equilibrio entre resistencia y ligereza.

Además, se deben de considerar las condiciones técnicas y los materiales que contribuyen a la eficacia operativa del sistema. Con esta base, el diseño del AGV se orienta hacia una configuración flexible, permitiendo futuras adaptaciones y mejoras que conduzcan a una implementación más robusta y eficiente.

3.1.1 *Requerimientos del AGV*

Para el desarrollo de su estructura es necesario considerar una serie de factores que van a influenciar el desempeño del AGV. Los requerimientos básicos que se necesitan para su construcción están detallados a continuación:

- El AGV debe tener una buena resistencia para poder soportar todo el peso de los componentes que estarán encima del mismo.

- La estructura del AGV tiene que ser de un material ligero para que pueda tener mayor movilidad y de esta manera facilitar el trabajo que realiza el robot.
- El tamaño de la estructura del AGV dependerá de los elementos que tendrá instalados, tanto en el interior como en su exterior.
- Los motores deben tener un torque lo suficientemente fuerte para poder vencer el estado de inercia del AGV.
- La comunicación inalámbrica tiene que ser generada a través de una tarjeta wifi la cual está integrada en la Raspberry que es el cerebro del proyecto que permita una comunicación clara y muy eficaz para evitar errores durante el intercambio de datos entre el computador y el robot móvil.
- Las baterías deben tener un amperaje lo suficiente alto para soportar el consumo de energía del robot móvil por un lapso prolongado.

3.1.2 Diseño propuesto

Para el diseño de la estructura del robot móvil, se diseñó una plataforma que permite la integración y montaje de los componentes electrónicos y mecánicos esenciales, como las tarjetas driver para los motores, sensores y el sistema de comunicación inalámbrica, que incluye una cámara. La estructura está construida con material ligero que facilita la movilidad del AGV sin comprometer su resistencia.

Los componentes principales (drivers, sensores, batería, y unidad de procesamiento) se han seleccionado cuidadosamente para garantizar un rendimiento eficiente. En las siguientes secciones se detallan específicamente las características y funciones de cada uno de estos elementos, proporcionando una visión integral de cómo contribuyen al desempeño global del robot.

3.1.3 Diseño de la estructura del AGV

La representación visual en la Figura busca transmitir una concepción que se asemeja bastante al prototipo definitivo. Además, muestra los diversos componentes que lo constituyen, incluyendo una disposición de tipo triciclo, con la zona de tracción y dirección ubicada en la parte frontal, junto con un compartimiento de conexiones diseñado para albergar los dispositivos de control.

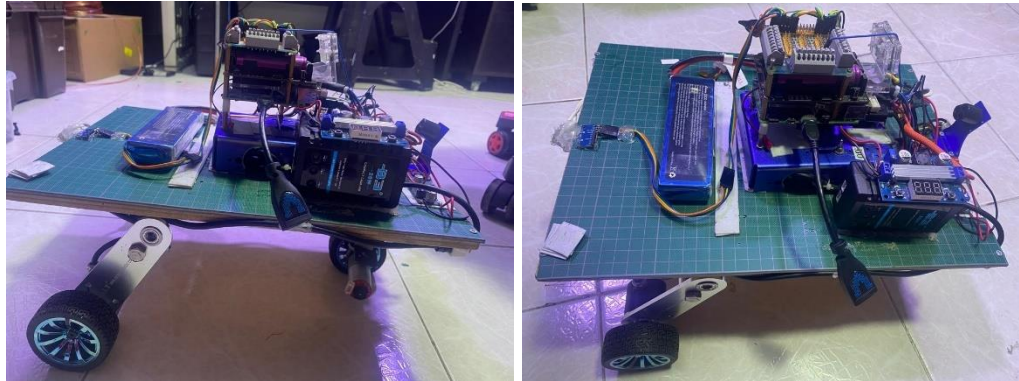


Figura 3-1 Imagen de prototipo 3

Los elementos mencionados e identificables los modelos se relacionan entre sí para formar el modelo cinemático.

Dimensiones

Las dimensiones específicas del prototipo se establecen en 40 cm de largo, 35 cm de ancho y 25 cm de altura, según lo ilustrado en la Figura 3-2. Es importante señalar que, si bien existe un margen de tolerancia considerable en el proceso de construcción, ciertos elementos cruciales de este diseño están sujetos a posibles ajustes en la versión definitiva.

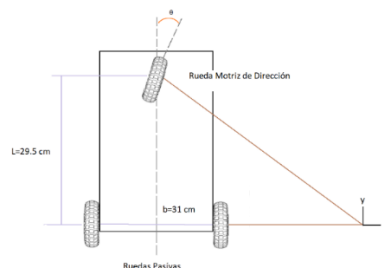


Figura 3-2 Dimensiones del AGV

Entre estos elementos, destacan el gabinete de conexionado, que juega un papel central en la integración de los dispositivos de control, así como las dimensiones finales de los motores que impulsarán el prototipo. Además, la elección del tipo de ruedas a utilizar, aunque inicialmente planificada, podría experimentar modificaciones para optimizar el rendimiento y la eficiencia del sistema en la iteración final del diseño. Estos ajustes permiten una mayor flexibilidad y adaptabilidad del prototipo a posibles mejoras y requisitos específicos que puedan surgir durante el proceso de desarrollo.

3.1.4 Elementos físicos del sistema

En la Tabla 3.1 se detallan los componentes esenciales requeridos para la construcción del prototipo. Es importante señalar que las cantidades indicadas son precisas para cada elemento, sin incluir piezas adicionales. En casos donde no se especifica la cantidad, se debe a la posibilidad de que el tipo y la cantidad de dichos elementos puedan variar durante el proceso, permitiendo así una flexibilidad en la adaptación del diseño a posibles ajustes y mejoras.

Tabla 3-1 Material del AGV

Material	Cantidad
Elevador De Voltaje Mt3608 Boost Step Up 2a	1
Motor de engranaje con Encoder 12vdc 600 RPM Llanta Base Cople Rasp	1
Soporte Encoder Medidor Distancia Encoder 20mm Plc Arduino	2
Codificador Rotatorio Incremental Industrial De 5v 24v	2
Modulo Regulador De Voltaje XL4016 Step Down 250w 8a Display	1
Adaptador Para Llanta Coples P Motorreductor Hexagonal 12mm	3
Modulo De Expansion Para Raspberry Gpio Raspberry Pi3 3b Pi4	1
Pila recargable Efest IMR IMR18650V1 Cilíndrica	1
Base de aluminio	1

Soporte de camara	1
soporte de estructura para raspberry	1
Servomotor Mg995 15kg 360° Giro Continuo Engrane Metal	1
Llantas De Goma Para Adaptador Hexagonal 65mm, Robótica	3
Batería Lipo Gens Ace 3300mah 45c 3s1p 11.1v Conector Ec3	1
Mpu-6050 Imu De 6dof	2

3.1.5 Integración

La finalización del montaje da lugar a la siguiente ilustración la cual denota que el AGV ya ensamblado con todos los dispositivos y tarjetas conectados y en funcionamiento para trabajar:

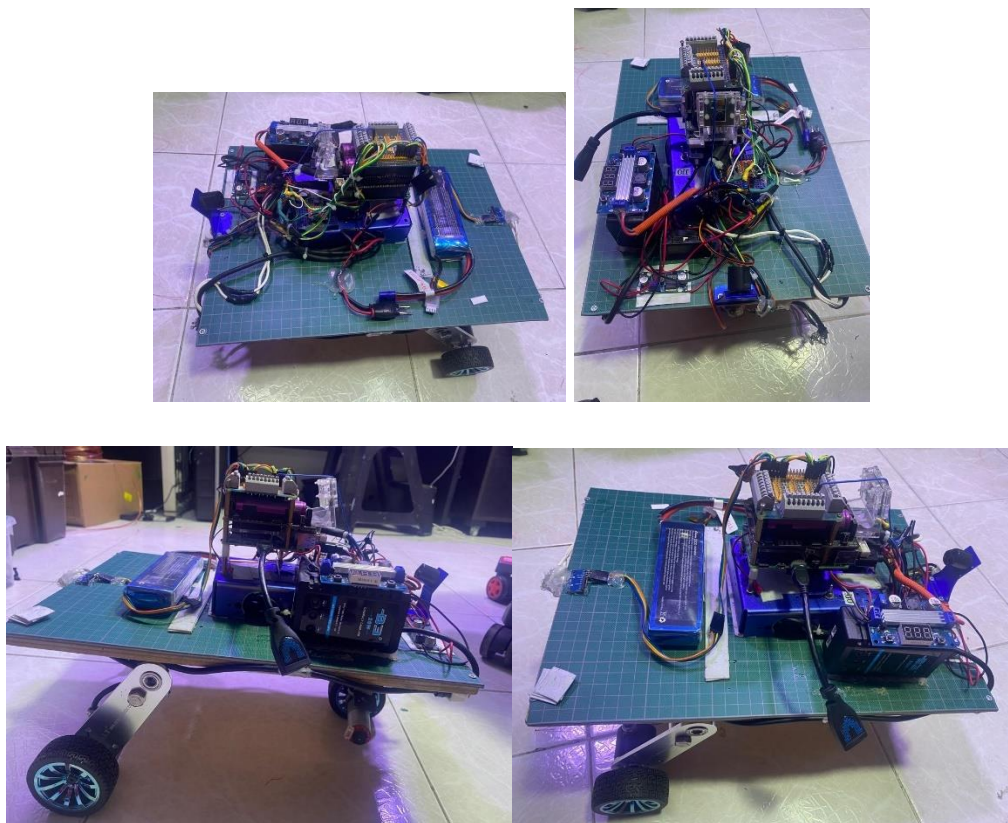


Figura 3-3 Vista lateral izquierda, B) Vista lateral derecha

3.2 Desarrollo electrónico y plataforma de software del AGV.

Para garantizar el funcionamiento integral del AGV, es fundamental establecer una arquitectura de software robusta y bien configurada, con un enfoque en el uso de una Raspberry Pi 4 como unidad central para ejecutar los algoritmos de fusión de sensores. Esta arquitectura permite que el AGV interprete y responda a su entorno en tiempo real, asegurando un desempeño preciso y confiable. Esta configuración inicial establece las bases para el procesamiento de datos eficiente y la sincronización de todos los componentes en un sistema integrado y adaptable, orientado a optimizar las capacidades de navegación y autonomía del AGV.

3.2.1 Arquitectura de software del AGV

Primeramente, se debe de observar el sistema completo para iniciar con las configuraciones. En la figura 3-4 se puede observar un breve diseño de la interacción del sistema. Se muestra el sistema de la tarjeta Raspberry, donde los sensores como los mpu 9250 y los encoder óptico del sistema están procesados en lenguaje Python en la misma raspberry.



Figura 3-4 Configuración del sistema

Para configurar los requisitos previos, se adecua los entornos para ejecutar este proyecto de investigación:

- Sensores
- Placa de desarrollo Raspberry pi 4 b+
- Python.

3.2.2 Elección placa de desarrollo

Después de revisado las diferentes placas de desarrollo comparando todas las características se toma la decisión de crear dicho proyecto con la tarjeta Raspberry pi 4 b+.

Raspberry pi 4

La Raspberry Pi 4 es una pequeña computadora de placa única (SBC, por sus siglas en inglés) desarrollada por la Raspberry Pi Foundation. A continuación, se presentan algunas de las características clave de la Raspberry Pi 4:

- **Procesador:**
 - CPU: Broadcom BCM2711, Quad-core Cortex-A72 (ARM v8) a 1.5 GHz.
- **Memoria:**
 - Opciones de memoria RAM: 2 GB, 4 GB o 8 GB LPDDR4-3200 SDRAM.
- **Conectividad:**
 - Gigabit Ethernet.
 - Wi-Fi 802.11ac.
 - Bluetooth 5.0 BLE (Baja Energía).
- **Puertos USB:**
 - 2 puertos USB 3.0.
 - 2 puertos USB 2.0.
- **Puertos de video:**
 - 2 puertos micro HDMI (admite hasta dos monitores 4K).
- **Almacenamiento:**
 - Ranura para tarjeta microSD para almacenamiento.
 - Puerto USB-C para la alimentación y datos.

- **GPIO (General Purpose Input/Output):**
 - 40 pines GPIO para conectar periféricos y expansiones.
- **Audio:**
 - Conector para auriculares de 3.5 mm.
 - Salida de audio por HDMI.
- **Dimensiones:**
 - Tamaño: 85.6 mm x 56.5 mm.
- **Sistema Operativo:**
 - Admite una variedad de sistemas operativos, incluidos Raspbian (ahora llamado Raspberry Pi OS), Ubuntu, y otros sistemas basados en Linux.
- **Rendimiento gráfico:**
 - VideoCore VI GPU que admite OpenGL ES 3.x.
- **Alimentación:**
 - Se puede alimentar mediante un adaptador de corriente USB-C (5V/3A).
- **Enfriamiento:**
 - La Raspberry Pi 4 tiene una capacidad mejorada de manejar el calor en comparación con modelos anteriores, pero aún puede beneficiarse de soluciones de enfriamiento pasivas o activas.
- **Puertos de cámara y pantalla:**
 - Conectores CSI para cámaras y DSI para pantallas.

La Raspberry Pi 4 es una computadora de placa única que, a pesar de su tamaño compacto y su costo accesible, ofrece una amplia gama de funcionalidades, convirtiéndola en una herramienta versátil para una variedad de aplicaciones. Este pequeño dispositivo cuenta con un potente procesador Quad-core Cortex-A72 a 1.5 GHz y opciones de memoria RAM de 2 GB, 4 GB o 8 GB, lo que le permite realizar tareas de computación de propósito general.



Figura 3-5 Raspberry 4

Su conectividad es notable, con puertos USB 3.0 y 2.0, Ethernet Gigabit, Wi-Fi 802.11ac y Bluetooth 5.0 BLE. Además, la Raspberry Pi 4 tiene capacidad para manejar hasta dos monitores 4K gracias a sus dos puertos micro HDMI, lo que la hace adecuada para proyectos multimedia y estaciones de trabajo ligeras.

La presencia de pines GPIO facilita la conexión de sensores, actuadores y otros dispositivos, lo que convierte a la Raspberry Pi 4 en una opción popular para proyectos de bricolaje y electrónica. Su capacidad para funcionar como servidor doméstico permite compartir archivos, alojar sitios web y realizar funciones de servidor en el hogar.



Alternate Function					Alternate Function
	3.3V PWR	1		2	5V PWR
I2C1 SDA	GPIO 2	3		4	5V PWR
I2C1 SCL	GPIO 3	5		6	GND
	GPIO 4	7		8	UART0 TX
	GND	9		10	UART0 RX
	GPIO 17	11		12	GPIO 18
	GPIO 27	13		14	GND
	GPIO 22	15		16	GPIO 23
	3.3V PWR	17		18	GPIO 24
SPI0 MOSI	GPIO 10	19		20	GND
SPI0 MISO	GPIO 9	21		22	GPIO 25
SPI0 SCLK	GPIO 11	23		24	GPIO 8
	GND	25		26	GPIO 7
	Reserved	27		28	Reserved
	GPIO 5	29		30	GND
	GPIO 6	31		32	GPIO 12
	GPIO 13	33		34	GND
SPI1 MISO	GPIO 19	35		36	GPIO 16
	GPIO 26	37		38	GPIO 20
	GND	39		40	GPIO 21
					SPI0 CS0
					SPI0 CS1
					SPI1 CS0
					SPI1 MOSI
					SPI1 SCLK

Figura 3-6 Pinout raspberry 4

Además, la Raspberry Pi 4 se ha ganado un lugar en el ámbito educativo al servir como una herramienta valiosa para aprender programación y conceptos informáticos. Su bajo consumo de energía y su tamaño la hacen ideal para proyectos de bajo costo y de bajo impacto ambiental.

3.2.3 Sistema electrónico

El sistema electrónico del vehículo guiado automático (AGV) integra varios componentes críticos que le permiten navegar de manera autónoma, manteniendo precisión y estabilidad en su trayectoria.

3.2.3.1 Descripción general

Los componentes principales del sistema incluyen:

1. Encoders en las ruedas:

Los encoders son sensores de posición que monitorean el desplazamiento de cada rueda del AGV. Estos dispositivos cuentan los pulsos generados por el movimiento rotacional de las ruedas, permitiendo calcular la distancia recorrida y la velocidad del vehículo. La información de los encoders se utiliza para actualizar en tiempo real la posición y orientación del AGV en el plano cartesiano, facilitando el seguimiento de su trayectoria y las correcciones de rumbo necesarias.

2. Sensores IMU (MPU6050):

El sistema incluye dos unidades de medida inercial (IMUs) MPU6050, configuradas para trabajar en paralelo. Cada IMU proporciona datos de giroscopio en el eje z , lo cual es esencial para ajustar la orientación y estabilizar el vehículo durante el movimiento. Ambos sensores están calibrados individualmente para reducir errores de medición y ruido. La calibración asegura una base de referencia precisa y minimiza el impacto de factores ambientales, como cambios de temperatura, que podrían afectar las lecturas. Los datos obtenidos de las IMUs ayudan a mantener la orientación deseada del AGV, corrigiendo desviaciones en trayectorias dinámicas.

3. Servomotor de dirección:

El AGV usa un servomotor para ajustar el ángulo de dirección. Este servomotor recibe señales PWM (modulación por ancho de pulso) que le permiten cambiar su posición con precisión, logrando orientar el vehículo hacia sus objetivos. La señal de control es ajustable para orientar el servomotor entre -90 y 90 grados, lo cual se convierte en pulsos que definen la dirección del movimiento.

4. Motor de tracción y control de velocidad:

El motor de tracción del AGV está controlado por PWM, lo que permite ajustar su velocidad en función de los requisitos de cada maniobra. La velocidad es controlada en tiempo real según los datos de posición y dirección, proporcionando un manejo estable al aproximarse a los puntos objetivo.

5. Unidad de procesamiento (Raspberry Pi):

La Raspberry Pi actúa como el cerebro del AGV, ejecutando cálculos cinemáticos y algoritmos de control. Esta unidad procesa datos de sensores, coordina el movimiento del vehículo, y maneja la comunicación en tiempo real para que el AGV responda de inmediato a cualquier variación en su trayectoria. La capacidad de procesamiento de la Raspberry Pi permite la integración fluida de los diferentes módulos, asegurando una navegación autónoma precisa y estable.

Este sistema electrónico, con sus componentes y sus métodos de integración y calibración, permite que el AGV mantenga una trayectoria precisa y se adapte a cambios en el entorno.

Diagrama de bloques

La Figura 3-7 muestra la interconexión en bloques del sistema.

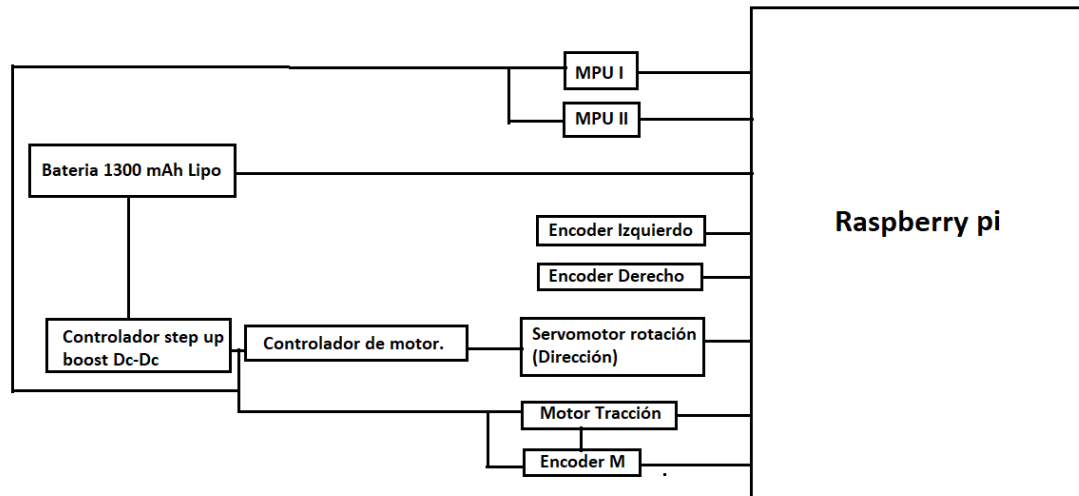


Figura 3-7 Diagrama de bloques del sistema en general.

3.2.3.2 Interfaz de Odometría

La interfaz de odometría del sistema es una herramienta esencial para el seguimiento de la posición y orientación del vehículo guiado automático (AGV) en el plano, en función de los datos recolectados por los sensores de movimiento. Esta interfaz cumple varios roles fundamentales que permiten calcular la posición, la orientación y la distancia recorrida por el AGV en tiempo real, a continuación, se describe los puntos que desarrolla esta interfaz:

Procesa los datos de los encoders de las ruedas, que proporcionan el número de pulsos de rotación acumulados. Los encoders, ubicados en cada rueda, registran el movimiento en intervalos definidos, permitiendo calcular la distancia que ha recorrido el vehículo y su velocidad de avance en cada actualización. Este proceso de lectura es constante y se ajusta en tiempo real.

- Usa los datos de los encoders para calcular el desplazamiento promedio del AGV y la variación en el ángulo de orientación. Aplicando un modelo

cinemático, se estiman las nuevas coordenadas (x,y) y el ángulo (θ) del vehículo. Este modelo se ajusta automáticamente, manteniendo el cálculo de orientación dentro de los rangos válidos, y permite que el AGV se posicione y oriente correctamente en su entorno.

- Manejo del flujo constante de datos de los sensores y los actualiza en tiempo real. Cada vez que se recibe una lectura de los encoders, el sistema procesa la información, actualiza las coordenadas y almacena las posiciones y ángulos resultantes en listas o vectores. Esto permite revisar el historial de movimientos y realizar un análisis de trayectoria o de variabilidad en la posición al final de cada ciclo de navegación.

3.2.3.3 Interfaz de Giroscopio

La interfaz del giroscopio (IMU) es fundamental para que el vehículo guiado automático (AGV) pueda monitorear y ajustar su orientación durante la navegación. En este sistema, la interfaz de los sensores de medida inercial (IMUs) proporciona datos precisos de rotación y orientación, permitiendo que el AGV mantenga un rumbo estable y reaccione a cualquier cambio de trayectoria, a continuación se describe los puntos que desarrolla esta interfaz:

- Integración de dos sensores MPU6050 que operan en paralelo. Cada IMU proporciona lecturas de giroscopio en el eje (z) , capturando cualquier rotación o cambio de orientación en ese eje. La configuración de dos IMUs permite obtener una lectura redundante y simultánea, mejorando la precisión en entornos donde pueden presentarse fluctuaciones de datos o ruido.
- Cada sensor IMU está calibrado para eliminar errores de offset que surgen debido a las variaciones de fábrica o factores ambientales como la temperatura. Estos offsets se aplican directamente en la interfaz para ajustar las lecturas en tiempo real, proporcionando una referencia precisa desde la cual medir cualquier cambio en la orientación del AGV.

- Los IMUs lee los datos de rotación en el eje (z) y aplica la calibración correspondiente para obtener valores precisos. Las lecturas se ajustan restando los valores de offset predefinidos, lo que garantiza que la medida de rotación comience en una referencia de cero cuando el AGV está en reposo. Este ajuste continuo permite que la interfaz filtre el ruido y reduzca las desviaciones en las mediciones de orientación.
- Utiliza un sistema de procesamiento paralelo para obtener lecturas simultáneas de ambos sensores IMU. Esto es particularmente útil en aplicaciones que requieren un monitoreo en tiempo real de la orientación, como el control de dirección en robots móviles. La capacidad de ejecutar ambos sensores en paralelo asegura que el AGV pueda reaccionar de inmediato a cualquier cambio de orientación, ajustando su dirección para mantener la estabilidad en su trayecto.

3.2.3.4 Diagrama electrónico

El diagrama electrónico para el sistema del AGV descrito en el documento incluye los componentes esenciales de navegación, control de movimiento, y procesamiento, mostrando cómo están interconectados para lograr la navegación autónoma. A continuación, se describe una propuesta de estructura para el diagrama:

- **Unidad de Procesamiento (Raspberry Pi):**
 - Interfaz I2C para la comunicación con las IMUs (MPU6050).
 - Conexión GPIO para recibir las señales de los encoders de las ruedas.
 - Interfaz PWM para controlar el servomotor de dirección y el motor de tracción.
 - La Raspberry Pi centraliza el procesamiento de datos y el control de los periféricos, ejecutando los algoritmos de navegación.
- **Sensores IMU (MPU6050):**

- Cada IMU se conecta a un bus I2C diferente (por ejemplo, IMU A en el bus 1 y IMU B en el bus 4) para evitar conflictos de dirección.

- Ambos sensores requieren alimentación de 3.3V o 5V, según el modelo específico, con un pin GND común a la Raspberry Pi.

- **Encoders de las ruedas:**

- Cada encoder envía señales de pulsos al sistema de GPIO de la Raspberry Pi, donde el conteo de pulsos se convierte en medidas de distancia y velocidad.

- Conectados al voltaje necesario (generalmente 5V) y un pin GND común a la Raspberry Pi.

- **Motor de tracción (con controlador PWM):**

- La Raspberry Pi envía señales PWM al controlador del motor para ajustar la velocidad de avance del AGV.

- Un módulo de control de potencia (puente H o controlador de motor) maneja la corriente y voltaje para el motor de tracción.

- Conectado a una fuente de 12V, con tierra común a todo el sistema.

- **Servomotor de dirección:**

- Recibe señales PWM desde la Raspberry Pi para ajustar su ángulo de dirección entre -90° y 90°, permitiendo el control de la orientación del vehículo.

- Conectado a una fuente de 5V, con un pin GND común.

Este diagrama muestra la estructura de control del AGV, donde los sensores y actuadores están coordinados mediante la Raspberry Pi para lograr un control preciso y adaptable de la navegación autónoma. Este diseño modular permite la adición de otros sensores o componentes en el futuro sin comprometer la funcionalidad básica del sistema.

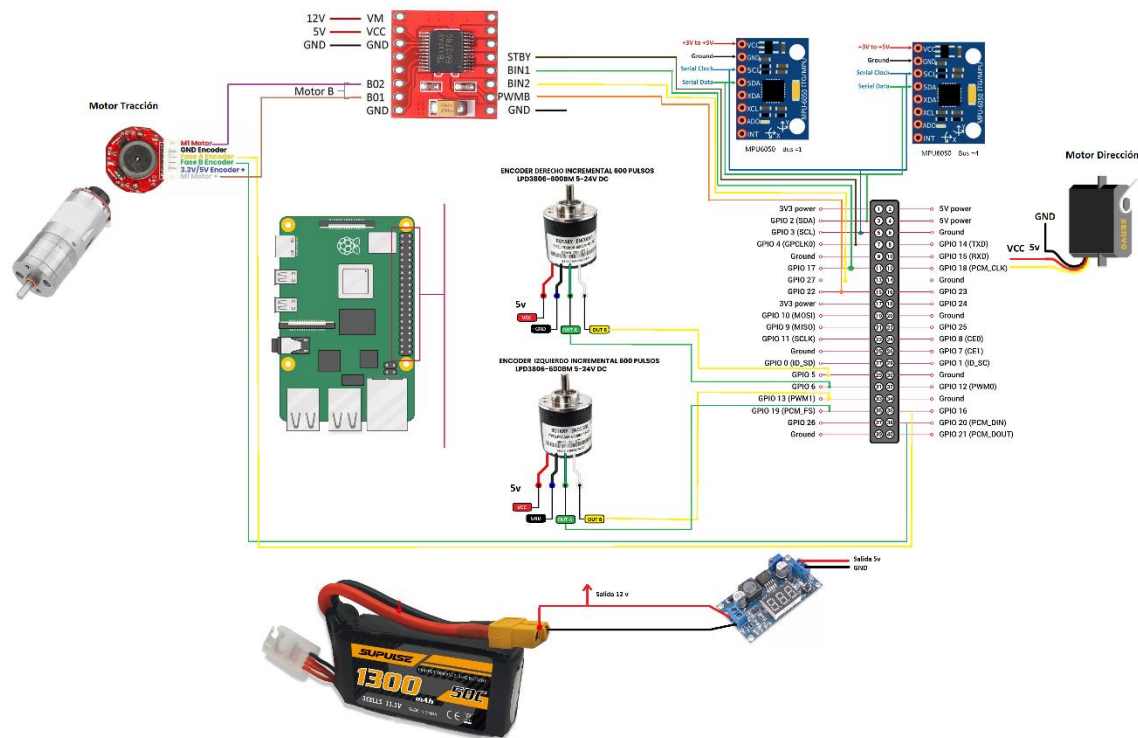


Figura 3-8 Diagrama Electrónico del sistema

3.3 Desarrollo e implementación del algoritmo.

La implementación del programa de control del vehículo guiado automático (AGV) destaca la precisión y sofisticación en la programación y algoritmos utilizados para la navegación autónoma y la integración de múltiples sensores. Este apartado detalla la importancia de los algoritmos cinemáticos y los módulos de control del motor, encoders y servomotores en el marco del sistema de navegación y posicionamiento del AGV.

Estos algoritmos no solo aseguran una navegación confiable al establecer y mantener el rumbo hacia cada punto objetivo, sino que también aprovechan la capacidad de la Raspberry Pi para ejecutar cálculos y comunicaciones en tiempo real. Esto permite el monitoreo continuo de la posición y la orientación mediante datos de los encoders y IMUs, asegurando que el vehículo pueda responder de inmediato a variaciones en su trayectoria.

Además, el uso de estrategias como el mapeo de grados a pulsos para el control de servo y la corrección continua de desviaciones en la posición, subraya la precisión con la que estos algoritmos controlan el movimiento del vehículo. La implementación de estos métodos en un sistema Raspberry Pi muestra cómo la integración de sensores y motores puede permitir una navegación autónoma eficiente y robusta en entornos reales. Este análisis destaca la importancia de los cálculos cinemáticos, los ajustes de control, y el procesamiento paralelo en la navegación autónoma, estableciendo una base sólida para el desarrollo de sistemas avanzados de control y guiado en robots móviles.

La arquitectura de control del AGV incorpora funciones para manejar lecturas en tiempo real de sensores IMU MPU6050, que proveen datos críticos de giroscopios y para ajustar la orientación y estabilidad del vehículo. La calibración de estos sensores, fundamental para compensar errores y reducir el ruido de las lecturas, se realiza mediante algoritmos diseñados para establecer una base de referencia precisa, permitiendo que el sistema pueda leer cambios en orientación con una desviación mínima. Esta precisión es vital para asegurar que el AGV pueda mantener su curso y corregir cualquier desviación en trayectorias dinámicas.

El código implementa un modelo cinemático para la estimación de la posición en tiempo real del AGV, considerando su orientación y desplazamiento en el plano cartesiano. Esto se realiza mediante cálculos trigonométricos avanzados y el uso de vectores de desplazamiento. La estimación del centro de curvatura, el ajuste de orientación y el cálculo del ángulo de giro son componentes clave que permiten a este sistema ajustar la trayectoria del AGV, alineándose con objetivos predefinidos y realizando maniobras complejas con precisión. Esta capacidad de adaptación permite que el AGV navegue eficientemente en entornos estructurados y no estructurados.

Además, la integración de encoders en las ruedas permite monitorear el desplazamiento preciso de cada rueda, facilitando el cálculo de la distancia recorrida y la velocidad del AGV. Esto, combinado con el control de servomotores para la

dirección, asegura un manejo estable de las maniobras de giro y ajuste en tiempo real de la dirección del vehículo.

La programación implementada en este apartado va más allá de la simple movilidad, proporcionando un marco robusto para la navegación autónoma en tiempo real, utilizando una mezcla de algoritmos de control, cinemática avanzada y ajustes de orientación basados en datos de sensores. Cada sección del código se coordina para permitir que el AGV mantenga una trayectoria precisa, se alinee con los objetivos de movimiento y pueda realizar ajustes en respuesta a cambios en el entorno, demostrando la importancia de la integración y sincronización de sensores y actuadores en un sistema de navegación autónoma complejo.

3.3.1 Función de procesamiento y lectura de 2 sensores IMU (Inertial Measurement Unit) MPU6050

La función está diseñada para manejar y procesar lecturas de dos sensores IMU (Inertial Measurement Unit) modelo MPU6050 en paralelo, calibrándolos y obteniendo datos de giroscopio en z. Este tipo de configuración es útil cuando se desea realizar lecturas simultáneas en sistemas que requieren conocer con precisión la orientación o los cambios de ángulo en un entorno tridimensional, como robots o vehículos autónomos.

1. Calibración de los IMUs

El proceso de calibración reduce las desviaciones de medición en aceleración y rotación causadas por factores ambientales o variaciones del propio sensor.

- **Inicialización de Variables de Calibración:**

Se definen los offsets *accel_offsets* y *gyro_offsets* para los valores de aceleración y giroscopio en los ejes x, y, y z. Estos serán los valores finales que se usarán para ajustar las lecturas de cada sensor.

Se definen sumas acumulativas *accel_sums* y *gyro_sums* que almacenarán los valores de lectura de cada eje en el proceso de calibración.

- **Toma de Muestras:**

La función *calibrate_sensor()* realiza *num_samples* lecturas (en este caso, 100) de los datos de aceleración y giroscopio en los tres ejes (*x, y, z*) de cada IMU.

Cada muestra se acumula en *accel_sums* y *gyro_sums*, sumando así un promedio de lecturas de los sensores para reducir el impacto de cualquier fluctuación o ruido instantáneo en las lecturas.

- **Cálculo de Offsets:**

- Después de recoger las muestras, se calculan los valores promedio de aceleración y giroscopio en cada eje dividiendo cada suma acumulada entre el número de muestras *num_samples*.

- En el caso del eje *z* de la aceleración, se resta un valor de 1 en el promedio para compensar el efecto de la gravedad (aproximadamente 1g), ya que el sensor suele percibir este valor como una fuerza constante en el eje vertical mientras está quieto.

- **Almacenamiento de Offsets:**

- Una vez calculados los offsets, estos se guardan en un archivo de texto (*calibration_data.txt*). Esto permite que los valores se reutilicen en futuras sesiones sin necesidad de recalibrar los sensores cada vez que se inician.

- Cada offset (de aceleración y giroscopio para ambos sensores) se escribe en el archivo con un nombre específico, lo que facilita su identificación y posterior carga.

Importancia de la Calibración

La calibración de los sensores IMU es fundamental para obtener datos precisos, ya que:

- **Corrige Errores de Offset:** Cada sensor puede tener un desplazamiento en sus valores en reposo debido a sus características físicas y de fabricación, generando un error de offset que distorsiona las lecturas.

- **Compensa el Ruido:** Al promediar múltiples lecturas, la calibración reduce el impacto del ruido aleatorio, proporcionando una base de referencia más estable.

- **Ajuste por Factores Ambientales:** Cambios en la temperatura o en el entorno también pueden afectar las lecturas; una calibración periódica ayuda a minimizar este impacto.

En el código, la función lee los offsets de calibración de un archivo txt. Estos offsets permiten ajustar los valores de salida del giroscopio de cada sensor a partir de una referencia de cero conocida. Para cada IMU, se carga un "offset de giroscopio" en el eje z, el cual se almacena en variables separadas: *gyro_offsets_mpuA* y *gyro_offsets_mpuB*. Este valor se resta de las lecturas en cada IMU para eliminar posibles desviaciones iniciales, estabilizando así la medición.

2. Lectura de Datos del Giroscopio

- Los sensores se inicializan, asignando uno a la dirección I2C estándar (0x68) en el bus 1 y otro en el bus 4, lo que permite diferenciarlos.
- Las funciones leen datos de cada sensor. En cada lectura, se accede al valor del giroscopio en el eje z, al cual se le aplica el offset correspondiente para obtener la medida calibrada.
- Luego, cada función imprime el valor ajustado.

3. Ejecución de los Sensores en Paralelo

- Se crea dos procesos paralelos (uno para cada IMU) que permite que ambas lecturas ocurran simultáneamente. Esto es necesario en aplicaciones que demandan datos en tiempo real, como el control de movimiento en robots, donde se necesitan datos inmediatos de ambos sensores para detectar cambios en la orientación de manera precisa.
- Cada proceso ejecuta, iniciando ambos al mismo tiempo para recolectar datos sin interferencias ni retrasos.

3.3.2 Función de estimación del robot en el Plano

1. Inicialización de Variables y Parámetros

El código inicializa varias variables de configuración:

- ***d***: que es la separación entre el eje delantero y trasero del vehículo. Este parámetro es crítico, ya que afecta al radio de giro y, por tanto, a la trayectoria del vehículo (representada en metros).
- ***phi***: Ángulo de orientación inicial del robot, indica la dirección en la que el vehículo está orientado respecto a un eje arbitrario de referencia, que define hacia dónde está apuntando el eje trasero en un espacio 2D.
- ***(Rx, Ry)***: Coordenadas iniciales del eje trasero, que representan el punto de partida en el plano.
- ***(Tx, Ty)***: Coordenadas del objetivo, el punto al que el robot intenta llegar, que representa el punto de destino hacia el cual se dirige el vehículo.

Además, se crean vectores para almacenar los resultados de cada parámetro calculado en cada iteración del movimiento del robot, permitiendo analizar los datos resultantes tras la ejecución.

2. Cálculo de Coordenadas del Eje Delantero (*Px, Py*)

Usando el ángulo *phi*, el código calcula la posición (*Px, Py*) en el eje delantero del vehículo. Este cálculo considera que la orientación *phi* afecta la posición del eje delantero debido a la separación *d* entre ambos ejes:

$$Px = Rx - d \cdot \sin(\phi)$$

$$Py = Ry + d \cdot \cos(\phi)$$

Este punto (Px, Py) es importante porque marca un punto de referencia en la dirección del vehículo, ayudando a definir cómo se alineará para moverse hacia el objetivo.

3. Determinación del Punto Medio y la Pendiente Hacia el Objetivo

Se calcula el punto medio (Bx, By) entre el punto (Px, Py) y el destino (Tx, Ty) para utilizarlo como una referencia en la trayectoria del robot. Este punto también ayuda a establecer la pendiente Tm entre la posición del robot y el objetivo:

$$Tm = \frac{Ty - Py}{Tx - Px}$$

Esta pendiente representa la dirección directa hacia el objetivo. Además, se calcula una pendiente perpendicular Bm , que se obtiene como el opuesto de la inversa de Tm , y es útil para encontrar el centro de curvatura.

4. Cálculo del Centro de Curvatura (Cx, Cy)

Se utilizan ecuaciones de líneas para encontrar el punto de intersección (Cx, Cy) entre las líneas formadas por el eje trasero y la pendiente Bm . Este punto de intersección se define como el centro de curvatura, que es el punto alrededor del cual el vehículo debe girar para alinear su dirección con el destino:

$$Cx = \frac{(bB - bR)}{(Rm - Bm)}$$

$$Cy = Cx \cdot Bm + bB$$

La posición (Cx, Cy) es crucial para determinar el ángulo de giro y la trayectoria de curvatura del robot.

5. Ángulo de Giro y Dirección de Movimiento

Luego, se calculan ángulos que permiten al vehículo alinearse con el objetivo. Se usa Pm , una pendiente temporal que depende del centro de curvatura, para obtener un ángulo de giro relativo, *result3*, que ayuda a calcular el cambio de dirección necesario para que el robot gire hacia el objetivo:

$$result = \frac{Pm - Rm}{1 + Rm \cdot Pm}$$

$$result3 = \text{atan}(result) \cdot 57.29577951(\text{engrados})$$

6. Distancia Recorrida por la Rueda Delantera (Barrido)

El cálculo de la distancia de desplazamiento barrido utiliza la geometría entre el eje trasero y el centro de curvatura (Cx, Cy). Esta distancia representa la cantidad de movimiento o “barrido” que realiza la rueda delantera a lo largo de la trayectoria de giro:

$$Di2 = \frac{|Py - Cy|}{\sin(\phi + \text{atan}(result))}$$

$$barrido = 2 \cdot |\text{atan}((Pm - Bm)/(1 + Pm \cdot Bm))| \cdot Di2$$

7. Cálculo de la Nueva Orientación del Robot

Se calcula un ángulo c_theta , que representa la nueva orientación del eje mayor del robot después de realizar el giro:

$$c_theta = \phi + 2 \cdot \text{atan}((Pm - Bm)/(1 + Pm \cdot Bm)) - \pi/2$$

Con este ángulo, se actualizan las coordenadas del centro del eje trasero (Rx, Rys) usando trigonometría para reflejar la nueva posición del robot después del giro. Estas coordenadas son las posiciones finales estimadas al término del movimiento.

8. Ángulos y Correcciones Finales

Finalmente, se convierte el ángulo de orientación $pend_out$ de radianes a grados ($pend_out_grados$) para facilitar la interpretación, y se calcula $angle_run$, un ángulo que representa el cambio general en la dirección de movimiento:

$$pend_out_grados = pend_out \cdot 57.29577951$$

$$angle_run = Pm \cdot 57.29577951$$

9. Almacenamiento de Resultados y Preparación para la Próxima Iteración

Cada uno de los valores calculados (coordenadas, distancias, ángulos) se almacena en listas para poder revisar y analizar los resultados al final de la ejecución del código. Se actualizan ϕ y Ry con los nuevos valores de orientación y posición, lo que permite repetir el proceso en caso de que se realicen movimientos adicionales hacia nuevos objetivos.

Este código representa el cálculo de una trayectoria circular para el robot en movimiento, determinando cómo debe ajustarse su posición y orientación a medida que se acerca a un objetivo. Los cálculos geométricos y trigonométricos permiten al vehículo establecer un centro de curvatura, calcular la distancia y el ángulo de giro necesarios para alinearse con el objetivo, y actualizar sus coordenadas y ángulos de forma iterativa para adaptarse a los cambios en su entorno.

3.3.2.1 *Estimación Teórica del Cálculo de Puntos Objetivo para la Navegación Autónoma del Robot*

El presente código proporciona un marco teórico para la estimación y cálculo de puntos objetivo en la navegación de un robot móvil en el plano cartesiano. Este análisis teórico se centra en la estimación de los parámetros necesarios para que el vehículo autónomo avance desde una posición inicial hasta un objetivo predeterminado, haciendo uso de una geometría de control y guiado que permite ajustar su orientación y calcular la distancia recorrida. Los cálculos aquí expuestos sirven como base para un análisis posterior de implementación y optimización del modelo de navegación.

1. Parámetros Iniciales y Condiciones de Partida

El modelo comienza con la definición de parámetros esenciales, como:

- Las **coordenadas iniciales del centro del eje trasero** del robot (R_x , R_y), que establecen la posición de partida en el plano.

- Un **ángulo de orientación inicial (ϕ)**, que determina la dirección hacia la cual está orientado el robot en relación a un sistema de referencia arbitrario.
- La **distancia entre ejes (d)**, que es la separación entre los ejes delantero y trasero del vehículo. Este parámetro es fundamental, ya que establece el radio de giro, lo cual influye directamente en la trayectoria estimada del robot.

Asimismo, el modelo incorpora las coordenadas del punto objetivo (T_x, T_y), que es el destino hacia el cual se dirige el robot. Estos valores se utilizan en los cálculos subsecuentes para determinar la posición y orientación necesarias para que el vehículo alcance dicho objetivo de manera controlada.

2. Cálculo Teórico del Centro de Giro y el Punto de Medición

Para estimar la trayectoria hacia el objetivo, el algoritmo calcula teóricamente el **punto proyectado del eje delantero** según la orientación inicial y la distancia entre ejes. Este punto proyectado (P_x, P_y) se emplea en la determinación del **centro de giro (C_x, C_y)**, un punto crucial que actúa como el centro de curvatura alrededor del cual el robot simularía su giro.

El centro de giro se obtiene a partir de la intersección de las líneas rectas que pasan por (R_x, R_y) y (P_x, P_y), así como entre el punto medio de la línea entre el origen y el objetivo (B_x, B_y) y el punto objetivo (T_x, T_y). Este cálculo teórico genera un punto fijo en el plano en torno al cual se establece la curva de desplazamiento, lo que permite definir la trayectoria del robot.

3. Estimación del Ángulo de Giro (Theta) y Evaluación de la Orientación Final

El algoritmo estima el **ángulo de giro (Theta)**, representando el cambio en la orientación del robot necesario para alinear su trayectoria hacia el objetivo. Este ángulo se obtiene mediante relaciones trigonométricas y se utiliza para calcular la **pendiente y orientación final teórica** del vehículo al alcanzar el objetivo. Este cálculo permite evaluar el cambio de dirección que el robot deberá seguir, basándose en la diferencia entre su orientación actual y el ángulo hacia el objetivo.

4. Cálculo de la Distancia Teórica Recorrida (Barrido)

El cálculo teórico de la **distancia recorrida por la rueda delantera** (barrido) es fundamental en este modelo, ya que estima el desplazamiento necesario en función del ángulo de giro y de la separación entre ejes. Esta distancia recorrida acumulada permite evaluar el avance teórico del vehículo en cada iteración, proporcionando una medida teórica de la distancia necesaria para alcanzar el objetivo.

5. Actualización Teórica de la Posición y Orientación Final

El algoritmo recalcula las coordenadas teóricas del centro del eje trasero, simulando una nueva posición del robot en función del desplazamiento. En este modelo, se ajustan los valores de las coordenadas RyR_yRy y RxR_xRx usando el ángulo final de orientación (c_theta) y el desplazamiento relativo estimado desde el centro de giro. Este ajuste permite simular teóricamente la posición y orientación final del robot una vez alcanzado el punto objetivo, sirviendo como referencia para futuras implementaciones y ajustes en los modelos de control.

6. Registro de Datos y Análisis Teórico

El modelo guarda los valores generados en vectores que pueden analizarse en función de la trayectoria y los cambios de orientación en cada iteración. Este registro sistemático permite analizar la variabilidad de los desplazamientos teóricos, sirviendo como base para la evaluación de futuros modelos de navegación que se desarrollen a partir de esta estimación inicial.

3.3.3 Función de actualización del modelo cinemático de topología triciclo

El robot sigue un modelo de cinemático de topología triciclo. A continuación, se describen las ecuaciones utilizadas para calcular la posición y orientación del robot conforme a la lectura de los encoders.

Variables y Parámetros Clave

- $tick_left, tick_right$: número de pulsos obtenidos de los encoders de las ruedas izquierda y derecha.
- $last_tick_left, last_tick_right$: valor de pulsos previos para cada rueda, usado para calcular el desplazamiento.
- r_izq, r_der : radios de las ruedas izquierda y derecha.
- l : distancia entre las dos ruedas del robot (ancho del robot).
- $theta$: ángulo de orientación actual del robot.
- x, y : coordenadas actuales del robot en el plano.

1. Cálculo de la distancia recorrida por cada rueda

Cada rueda del robot recorre una distancia proporcional a los pulsos de su encoder. La distancia recorrida por cada rueda donde la distancia recorrida es proporcional al número de pulsos detectados por el encoder. Cada distancia se calcula usando el radio de cada rueda, el número de pulsos ($ticks$) desde la última actualización y la resolución del encoder:

$$dist_{right} = 2\pi \cdot r_{der} \cdot \frac{tick_{right} - last_tick_{right}}{ticks_per_revolution}$$

$$dist_{left} = 2\pi \cdot r_{izq} \cdot \frac{tick_{left} - last_tick_{left}}{ticks_per_revolution}$$

Donde:

- $ticks_per_revolution$ es la cantidad de pulsos del encoder por vuelta de rueda.

2. Cálculo del desplazamiento promedio del vehículo

El desplazamiento promedio del vehículo, $Duxy$, se toma como el promedio de las distancias recorridas por ambas ruedas. Esto representa el avance efectivo del vehículo en su dirección frontal:

$$Duxy = \frac{\text{dist}_{\text{left}} + \text{dist}_{\text{right}}}{2}$$

3. Cálculo del cambio de orientación (giro)

El cambio en la orientación del vehículo, Dth , se calcula como la diferencia entre las distancias recorridas por las ruedas derecha e izquierda, dividido por la distancia entre ejes l . Esto da una medida de cuánto ha girado el vehículo:

$$Dth = \frac{\text{dist}_{\text{right}} - \text{dist}_{\text{left}}}{l}$$

4. Actualización de la posición (x, y) y orientación θ

Para calcular la nueva posición y orientación del vehículo, se considera el desplazamiento promedio $Duxy$ y el cambio en la orientación Dth .

La posición en x se actualiza restando el desplazamiento en la dirección lateral:

$$x_{\text{nuevo}} = x_{\text{anterior}} - Duxy \cdot \sin\left(\theta + \frac{Dth}{2}\right)$$

La posición en y se actualiza sumando el desplazamiento en la dirección frontal:

$$y_{\text{nuevo}} = y_{\text{anterior}} + Duxy \cdot \cos\left(\theta + \frac{Dth}{2}\right)$$

La orientación θ se actualiza con el cambio de orientación calculado:

$$\theta_{\text{nuevo}} = \theta_{\text{anterior}} + Dth$$

5. Ajuste de la orientación (θ) para mantenerla en el rango $([-\pi, \pi])$

Finalmente, el código ajusta el valor de (θ) para que permanezca dentro del rango $([-\pi, \pi])$:

Estas líneas aseguran que (θ) no exceda los límites de un círculo completo, manteniendo la orientación dentro del rango estándar de un ángulo en radianes en los encoders.

El conjunto de ecuaciones que describe el movimiento del robot en función de los pulsos de los encoders de las ruedas es:

$$\text{dist_right} = 2 \pi \cdot r_{\text{der}} \cdot \frac{\text{tick_right} - \text{last_tick_right}}{\text{ticks_per_revolution}}$$

$$\text{dist_left} = 2 \pi \cdot r_{\text{izq}} \cdot \frac{\text{tick_left} - \text{last_tick_left}}{\text{ticks_per_revolution}}$$

$$duxy = \frac{\text{dist_left} + \text{dist_right}}{2}$$

$$dth = \frac{\text{dist_right} - \text{dist_left}}{l}$$

$$x_{\text{nuevo}} = x_{\text{anterior}} - duxy \cdot \sin\left(\theta + \frac{Dth}{2}\right)$$

$$y_{\text{nuevo}} = y_{\text{anterior}} + duxy \cdot \cos\left(\theta + \frac{Dth}{2}\right)$$

$$\theta_{\text{nuevo}} = \theta_{\text{anterior}} + dth$$

3.3.4 Función de Actualización de encoders

La función de actualización permite gestionar los encoders del sistema robótico. Los encoders son sensores que miden la rotación de las ruedas, lo cual es fundamental para calcular la posición y dirección del robot. Esta clase incluye:

- Configuración de los pines de los encoders, donde se detecta los eventos de cambio de señal en los pines, y establece el estado inicial de los pines (*leftPin* y *rightPin*).
- Registro del cambio de estado en los encoders para calcular la dirección de rotación (*adelante o atrás*) y el valor acumulado de rotaciones, que se ajusta según el sentido del giro.

- La devolución del valor actual de rotación de los encoders.

La función gestiona la detección y el registro de pulsos incremental, lo cual es esencial para medir la posición o el desplazamiento del control de movimiento. Al inicializarse, esta función se configura para leer las señales de los canales del encoders y activar los eventos en ambos flancos (ascendente y descendente). Cada vez que uno de estos pines detecta un cambio, la función se activa para determinar la nueva posición relativa de los encoders, estableciendo el estado actual de los pines y eligiendo la dirección del movimiento, dependiendo del estado previo y la transición detectada, este método también incrementa o decrementa un contador de pulsos acumulado que refleja el desplazamiento neto. Este contador se ajusta dinámicamente con cada transición y, en caso de que se haya definido, se ejecuta un llamado de vuelta opcional que recibe como parámetros el valor del contador y la dirección. Para acceder al valor acumulado de los pulsos en cualquier momento, la función permite obtener esta lectura.

La implementación de esta clase resulta útil para aplicaciones donde es crucial el monitoreo preciso y en tiempo real de la traslación.

3.3.5 Función de Configuración para el Motor y PWM

Esta función es necesaria para configurar los parámetros del motor y el control de PWM (modulación por ancho de pulso) en donde se describe la funcionalidad en el siguiente apartado:

- Configuración de los pines del motor y definición de la frecuencia PWM, que controla la velocidad del motor de traslación.
- Inicialización el PWM en un pin específico, controlando así la potencia del motor.

En esta función se calcula la distancia que ha recorrido el robot en función del número de ticks (pulsos) de los encoders y actualiza la posición total del motor de traslación.

- La distancia se calcula usando la fórmula:

$$\text{dist_delantero} = 2 \times \pi \times r_delant_delantero \times \frac{(\text{ticks actuales} - \text{ticks previos})}{\text{ticks por revolución}}$$

Donde:

- $(2 \times \pi \times r_delant_delantero)$ Es la circunferencia de la rueda.
- $\left(\frac{\text{ticks actuales} - \text{ticks previos}}{\text{ticks por revolución}}\right)$ da la fracción de vuelta que la rueda ha girado.

Esta distancia es agregada a la distancia total recorrida.

Además de que la función actúa cada vez que el encoder detecta un cambio en el valor en donde si ha cambiado y es mayor que el valor anterior en 1, el contador consecutivo aumenta y cuando el contador llega a 10 cambios consecutivos, se llama a para mostrar los valores actuales y el contador se reinicia. Estas funciones definen el movimiento del motor delantero o de traslación hacia adelante o en reversa a una velocidad específica.

3.3.6 Función de Servomotor de dirección

Esta función controlar el ángulo que ajusta la dirección del robot. En donde maneja los pulsos PWM para el servomotor, manteniendo un control estable sobre su posición, en donde se convierten entre grados y pulsos para controlar la posición. Este servomotor requiere un pulso específico para moverse a un ángulo determinado, y estas conversiones son necesarias para manejar con precisión.

La función convierte un ángulo en grados a pulsos, que es el formato que el servo necesita para posicionarse correctamente, que puede variar entre -90 y 90 grados y así devolver un valor en pulsos que se envía al servomotor para que se mueva al ángulo deseado. La fórmula utilizada es:

$$\text{pulsos} = 489 + (\text{grados} + 90) \times \frac{2000}{180}$$

Aquí, 489 representa el valor en pulsos correspondiente al ángulo de -90 grados, y se ajusta el rango de -90 a 90 grados a un rango de 0 a 180 grados y $\frac{2000}{180}$ es el

factor de escala que mapea los grados a un rango de 2000 pulsos, que es adecuado para la mayoría de los servomotores, los cuales suelen trabajar en un rango de 500 a 2500 pulsos. Así mismo la función convierte pulsos de un servo a grados, donde el valor en microsegundos recibido del servomotor corresponde a un ángulo específico y devuelve el ángulo en grados, mapeando los pulsos de vuelta a grados entre -90 y 90.

$$\text{grados} = (\text{pulsos} - 1500) \times \frac{180}{2000}$$

Donde *1500* representa el pulso correspondiente al centro del rango (0 grados), y *pulsos - 1500* ajusta el rango de pulsos para que el centro 1500 sea cero.

La finalidad de esta función es convertir de grados a pulsos cuando se quiere mover el servo a un ángulo específico y de pulsos a grados cuando se desea saber el ángulo correspondiente a un valor de pulso determinado.

3.3.7 Bloque de ejecución del programa

Este bloque principal es el núcleo de control del movimiento del robot, diseñado para desplazarse hacia una serie de posiciones especificadas y ajustar su orientación en función de valores predeterminados. Todo el código permite manejar errores inesperados durante la ejecución, asegurando qué si ocurre algún problema, el programa puede limpiar y detener adecuadamente todos los procesos sin dejar el sistema en un estado inestable.

La lógica comienza con una lista que representa las posiciones objetivo en el eje Y hacia donde el vehículo debe moverse. Para cada posición, el bucle espera hasta que la variable que representa la posición actual del vehículo en el eje Y, alcance o supere el valor del objetivo. Esto significa que el vehículo se moverá paso a paso hasta llegar al destino específico.

Dentro del bloque se obtiene el índice que es crucial para obtener el ángulo correspondiente. Estos ángulos generan el valor que representa el ángulo en el que el vehículo debería orientarse al moverse hacia el objetivo. Este pulso representa la señal que se envía al servo para ajustarlo a la orientación deseada. A continuación,

se envía la señal PWM al servomotor, indicando el ajuste del ángulo del vehículo. Se introduce una pausa muy breve de 1 ms para evitar que el servomotor se ajuste demasiado rápido y cause inestabilidad en el movimiento.

El motor delantero se activa a una velocidad preestablecida en porcentaje para avanzar el vehículo. Esto permite un avance controlado hacia la posición objetivo deseado. Mientras no se alcance el valor del objetivo, el ciclo de ajuste de ángulo y avance continúa, asegurando que el vehículo mantenga su orientación y avance controlado.

Cuando Y alcanza o excede el valor del objetivo, el motor se detiene usando. A continuación, se los valores actuales de las posiciones, que representan la posición actual en el plano, y θ , que es la orientación angular calculada del vehículo en grados a partir de su valor en radianes. Así mismo se obtienen los datos de los sensores de inercia (IMU) del sistema, lo que proporciona información sobre la orientación y dinámica del vehículo en tiempo real en conjunto con los datos muestreados de los encoders.

Cada vez que el robot llega a su objetivo, después de un retraso de 5 segundos (para permitir cualquier ajuste o monitoreo adicional), el bucle continúa con el siguiente valor de objetivos programados. Una vez que todas las posiciones objetivo han sido alcanzadas, el programa limpia los recursos para liberar cualquier control usado y garantizar que el sistema esté en un estado seguro.

Como paso final de seguridad y configuración, el ángulo del servomotor se ajusta a 0 grados para llevar al vehículo a una posición neutral o de reposo. Este ajuste finaliza cualquier orientación residual y establece un estado base de orientación con esto se aseguran de que los procesos secundarios que pudieran estar ejecutándose sean detenidos, evitando que sigan consumiendo recursos del sistema una vez finalizado el programa.

3.4 Navegación punto a punto

En el ámbito de los vehículos autónomos, la navegación precisa punto a punto es una de las tareas más desafiantes y fundamentales. Este proceso requiere un entendimiento detallado del modelo cinemático del vehículo, ya que su desempeño depende directamente de la correcta estimación de su pose, es decir, la combinación de su posición y orientación en el espacio. Los vehículos tipo triciclo, en particular, presentan características únicas que exigen una modelación matemática precisa para calcular los desplazamientos y giros necesarios que los lleven de un punto inicial a un destino específico.

El modelo cinemático de un vehículo de tipo triciclo considera aspectos como el radio de las ruedas, la separación entre los ejes y la resolución de los encoders, elementos que, en conjunto, determinan su capacidad para interpretar y reaccionar ante el entorno. Las ecuaciones derivadas de este modelo permiten calcular no solo el desplazamiento lineal y los cambios de orientación, sino también identificar parámetros clave, como el centro instantáneo de curvatura (ICC), necesario para describir movimientos en trayectorias curvas. Este enfoque no solo garantiza precisión en la navegación, sino que también optimiza el uso de recursos y mejora la eficiencia general del sistema.

El objetivo de este punto es proporcionar una comprensión integral del modelo cinemático de un AGV tipo triciclo y su capacidad para alcanzar puntos objetivo con alta precisión. Este conocimiento sirve como base para sistemas de navegación autónoma más complejos, donde se integran herramientas avanzadas, como la fusión de datos de sensores y algoritmos de optimización de trayectorias.

3.4.1 Navegación basada en el modelo cinemático

En la navegación de vehículos autónomos, es fundamental el uso de un modelo cinemático preciso que permita estimar la posición y orientación del vehículo en el espacio a medida que se desplaza. Es esencial definir sus características físicas, ya que estas afectan directamente las ecuaciones cinemáticas y la precisión de la

estimación de su posición y orientación. A continuación, se detallan los parámetros principales utilizados en el modelo:

- **Radio de las Llantas (r):**

El radio de cada llanta afecta la distancia que el vehículo recorre por cada rotación de la rueda. En este modelo, se usa un valor específico para cada llanta:

- ($r_{\text{der}} = 0.033605 \text{ m}$): Radio de la llanta derecha.
- ($r_{\text{izq}} = 0.033605 \text{ m}$): Radio de la llanta izquierda.

Estos radios permiten calcular los incrementos de desplazamiento de cada rueda por separado.

- **Radio de la Rueda Delantera ($r_{\text{delantera}}$):** El radio de la rueda delantera es esencial para calcular el desplazamiento lineal y el giro del vehículo. El radio efectivo de esta rueda afecta directamente el avance del AGV en cada rotación de la rueda.

- ($r_{\text{delantera}} = 0.033605 \text{ m}$): Radio de la llanta delantera.

- **Separación entre Llantas Traseras (b):**

La distancia entre las ruedas traseras, representada por (b), es la distancia entre los centros de las dos ruedas traseras, estas ruedas no participan en la tracción ni en la dirección. Sin embargo, esta separación es crucial para calcular el centro de rotación y los cambios en la orientación durante el desplazamiento

- ($b = 0.2654 \text{ m}$): separación entre llantas traseras.

- **Distancia entre el Eje Delantero y el Eje Trasero(l):**

Representa la distancia entre el eje trasero (donde están las ruedas pasivas) y el eje de la rueda delantera, que proporciona la tracción y el giro. Esta distancia afecta el radio de giro y la estabilidad direccional del AGV.

- ($l = 0.2635 \text{ m}$): separación entre el eje delantero y el eje trasero.

- **Resolución del Encoder en la Rueda Delantera (R):**

La resolución del encoder montado en la rueda delantera es fundamental para medir con precisión el desplazamiento del vehículo. El valor de (R) determina la precisión de las mediciones del ángulo de la rueda delantera, que se utiliza en el cálculo del avance y del giro del AGV.

Las ecuaciones cinemáticas están desarrolladas en función de la distancia recorrida por las ruedas traseras y el ángulo de orientación del vehículo. A continuación, se muestran las ecuaciones conforme a los siguientes parámetros:

- **Radio de las ruedas traseras:** $r_{\text{der}} \text{ y } r_{\text{izq}} = 0.033605 \text{ metros}$
- **Radio de la rueda delantera:** $r_{\text{delantera}} = 0.033605 \text{ metros}$
- **Separación entre ruedas traseras:** $b = 0.2654 \text{ metros}$
- **Separación entre eje trasero y rueda delantera:** $l = 0.2635 \text{ metros}$

1.- Centro de Origen y Definiciones Iniciales:

- El origen de las coordenadas $[x_0, y_0]$ está en el centro del eje trasero del vehículo, que tiene una separación (b) entre las dos llantas traseras.
- La distancia (l) representa la separación entre el eje trasero y la rueda de tracción delantera, que también tiene un ángulo de orientación (α).

2.- Movimiento y Centro Instantáneo de Curvatura (ICC):

- Cuando el vehículo avanza, el centro instantáneo de curvatura (ICC) se encuentra en la intersección de la proyección del eje de rotación de la rueda delantera y la proyección del eje trasero del vehículo.
- La orientación inicial del eje trasero es (θ_0) (generalmente 0°), y conforme el vehículo avanza, su orientación cambia a un nuevo ángulo (θ_1) , siguiendo un arco de círculo.

3.- Distancia Recorrida por las Ruedas Traseras Δs_l y Δs_r :

- Cada rueda trasera tiene un encoder óptico que mide la cantidad de pulsos((pl_l) y (pl_r)) durante el intervalo de muestreo.

- El número de pulsos de los encoders, junto con la resolución (R) y los factores de corrección (k_1) y (k_2), permite calcular la distancia que recorre cada rueda.
- Estas distancias están dadas por:

$$\Delta s_l = \frac{2\pi r_{izq} \cdot pl_l}{R} = pl_l \cdot k_1$$

$$\Delta s_r = \frac{2\pi r_{der} \cdot pl_r}{R} = pl_r \cdot k_2$$

4.- Actualización de la Posición y Orientación del Vehículo:

- El desplazamiento promedio ($\Delta u_{x,y}$) y el cambio en orientación ($\Delta\theta$) se usan para actualizar la posición ((x, y)) y el ángulo θ del vehículo.
- La orientación se ajusta en cada paso como:

$$\Delta\theta = \frac{\Delta s_r - \Delta s_l}{b}$$

- Las posiciones (x) y (y) se actualizan de la siguiente manera:

$$x_k = x_{k-1} - \Delta u_{x,y} \cdot \sin(\theta)$$

$$y_k = y_{k-1} + \Delta u_{x,y} \cdot \cos(\theta)$$

5.- Calibración y Ajustes:

- La constante (b) y los factores (k_1) y (k_2) se calibran para minimizar el error en la odometría, es decir, asegurar que la posición estimada coincida con la medida por sensores externos (como cámaras o ultrasonidos).
- Este modelo cinemático describe la forma en que el vehículo tipo triciclo calcula su posición y orientación en función del desplazamiento de sus ruedas y la orientación de la rueda delantera. Las ecuaciones se ejecutan

recurrentemente para actualizar la pose $([x, y, \theta])$ cuando se reciben nuevos datos de los encoders.

El resumen de las ecuaciones es el siguiente:

$$\Delta s_l = \left(\frac{2 \pi r_{izq} \cdot pl_l}{R} \right) = pl_l \cdot k_1$$

$$\Delta s_r = \left(\frac{2 \pi r_{izq} \cdot pl_l}{R} \right) = pl_r \cdot k_2$$

$$\Delta u_{x,y} = \frac{\Delta s_r + \Delta s_l}{2}$$

$$\Delta \theta = \Delta s_r - \Delta s_l \cdot b$$

$$\theta = \theta + \frac{\Delta}{2}$$

$$x_k = x_{k-1} - \Delta u_{x,y} \cdot \sin \theta_1$$

$$y_k = y_{k-1} - \Delta u_{x,y} \cdot \cos \theta_1$$

3.4.1.1 Ejemplo de Navegación entre dos puntos

En particular, los AGVs de tipo triciclo emplean un sistema de odometría para estimar la distancia recorrida por cada rueda trasera. Para ejemplificar el proceso de cálculo de la pose de un AGV tipo triciclo, se establecen las siguientes condiciones iniciales y de destino:

Datos iniciales:

- Ángulo inicial ϕ : (0°)
- Coordenada inicial en x : $Rx = 0.00 \text{ m}$
- Coordenada inicial en y : $Ry = 0.00 \text{ m}$

Datos de destino:

- Coordenada en x del objetivo: $Tx = 0.25 \text{ m}$

- Coordenada en y del objetivo: $Ty = 0.70$ m

Con estos parámetros de entrada, el cálculo de la trayectoria y la pose se realiza considerando la orientación inicial del AGV y la distancia entre los ejes delantero y trasero, que en este caso es de $d = 0.265$ metros.

Resultados del cálculo de la pose

- Coordenada de referencia del AGV (Cy): El cálculo determina que la coordenada (Cy) del punto de referencia del AGV es 0.0. Esto indica que, al inicio del desplazamiento, el punto de referencia en (y) permanece alineado con la coordenada inicial.

Coordenadas finales del punto de medición del AGV:

- **Rxs:** 0.121 metros, que representa la posición en el eje (x) al finalizar el desplazamiento.
- **Rys:** 0.4683 metros, que representa la posición en el eje (y) al finalizar el desplazamiento.

Estas coordenadas indican que el AGV se ha desplazado a una nueva ubicación cercana al destino, pero con un ligero desfase respecto a las coordenadas Tx y Ty .

Barrido:

La distancia de *barrido* o *trayectoria curva* que recorre la llanta delantera a lo largo del desplazamiento hacia el punto objetivo es de 0.5071 metros. Este valor proporciona una medida de la distancia total que ha cubierto el AGV, siguiendo una curva en lugar de un trayecto lineal, acorde con el modelo de triciclo que emplea.

Ángulo de salida:

El ángulo de orientación final del vehículo en relación con el eje x al término del movimiento es de -29.04 indicando una ligera inclinación negativa en la orientación de la trayectoria, además hay un ángulo calculado de -15.74 que describe el ángulo necesario para que el AGV realice la alineación óptima con la trayectoria hacia el objetivo final.

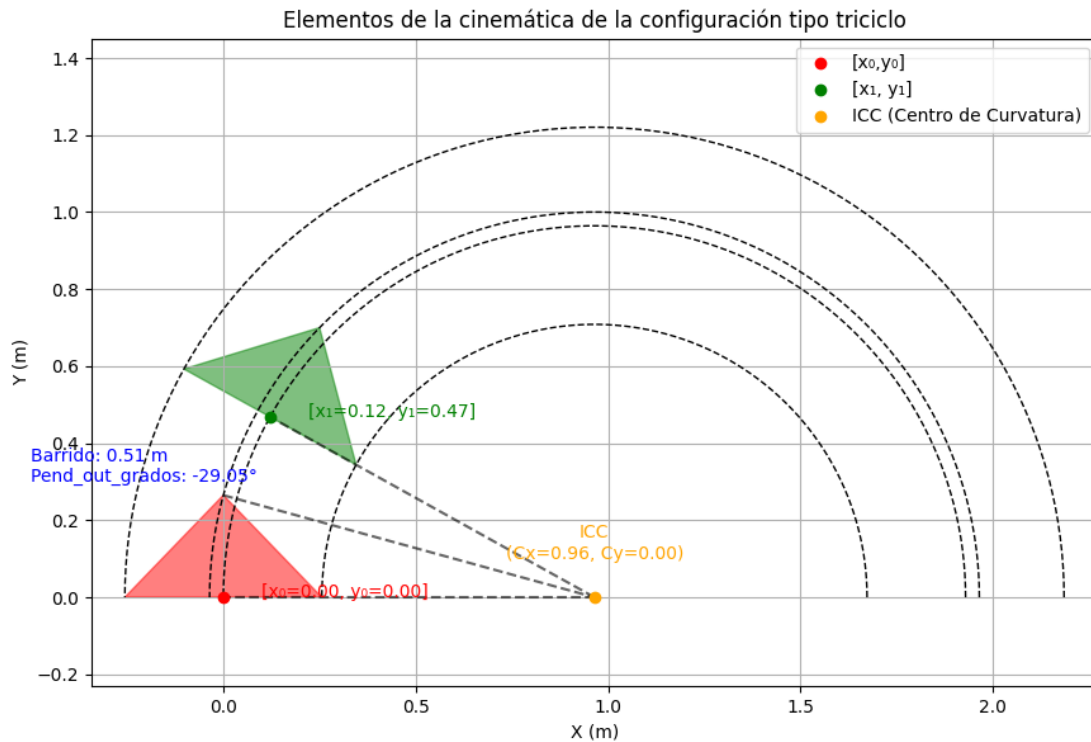


Figura 3-9 La pose del AGV. Distancia en metros.

3.4.2 Varianza de Allan

Adquisición de datos

La varianza de Allan es una técnica estadística utilizada para analizar y caracterizar las propiedades del ruido en señales, particularmente en dispositivos como sensores inerciales, relojes y osciladores. Fue desarrollada para estudiar cómo el ruido afecta las mediciones en diferentes escalas de tiempo y proporcionar una herramienta para identificar y cuantificar tipos específicos de errores, como el ruido aleatorio (*bias*), el deslizamiento del sesgo (*bias instability*) y los paseos aleatorios (*Random Walk*).

En esencia, la varianza de Allan mide la estabilidad de un sistema o sensor al calcular las diferencias promediadas entre clústeres consecutivos de datos para varias duraciones temporales (τ). Este enfoque permite descomponer el ruido en componentes que dominan en diferentes rangos de tiempo:

- En intervalos cortos, el ruido aleatorio tiende a prevalecer.
- En intervalos medios, se observa el efecto de errores sistemáticos como la estabilidad del sesgo.
- En intervalos largos, errores acumulativos como caminatas aleatorias se vuelven más evidentes.

Se realizó la experimentación donde se analiza los datos obtenidos de los sensores IMU 6050 para caracterizar las propiedades del ruido y la estabilidad de sus mediciones. Estos datos son esenciales en aplicaciones que requieren precisión, como sistemas de navegación, estabilización y robótica.

Al iniciar el proceso, el programa toma como entrada un archivo CSV que contiene las lecturas de velocidad angular del IMU 6050 en reposo. Estas lecturas representan la forma en que el sensor percibe los cambios de orientación en el tiempo. Sin embargo, como cualquier sensor, el IMU 6050 está sujeto a varios tipos de ruido inherente, que pueden distorsionar las mediciones y generar errores acumulativos si no se comprenden y manejan adecuadamente.

El primer paso del programa es convertir estas velocidades angulares en ángulos acumulados mediante una integración. Esto permite simular cómo el sensor mediría cambios de orientación a lo largo del tiempo. A partir de estas mediciones acumuladas, el programa inicia el cálculo de la desviación de Allan.

La desviación de Allan es una herramienta que agrupa los datos en clústeres de diferentes duraciones temporales (τ). Para cada intervalo de tiempo, el programa calcula la varianza promedio de las mediciones. Esta varianza describe la consistencia del sensor en un rango de tiempos específicos. Por ejemplo, en intervalos cortos, el ruido aleatorio suele dominar, mientras que en intervalos largos, emergen errores sistemáticos como el deslizamiento del sesgo.

Una vez calculada la desviación de Allan para diferentes valores de τ , el programa procede a analizar los resultados y buscar patrones característicos de ruido. Entre estos, se encuentran:

- **Caminata aleatoria angular (Angle Random Walk o ARW):** Este tipo de ruido se manifiesta como fluctuaciones rápidas y aleatorias en las mediciones del sensor. En la desviación de Allan, el ARW se identifica por una pendiente característica en el gráfico log-log.
- **Estabilidad del sesgo (Bias Instability):** Representa el punto en el que el sesgo del sensor es más estable, es decir, donde el error debido a fluctuaciones de largo plazo es mínimo. Este valor es crucial porque define el rendimiento más confiable que se puede esperar del sensor en condiciones ideales.
- **Caminata aleatoria (Random Walk):** Este tipo de error describe cómo pequeñas acumulaciones de ruido pueden sumarse y crear desviaciones significativas en periodos prolongados.

El programa no solo calcula estas métricas, sino que también visualiza los resultados en gráficos logarítmicos. Estos gráficos muestran cómo la desviación de Allan varía con los intervalos de tiempo y destacan puntos críticos como el *ARW*, el *bias instability* y el *random walk*. Esto permite a los usuarios interpretar fácilmente la calidad y las limitaciones del sensor.

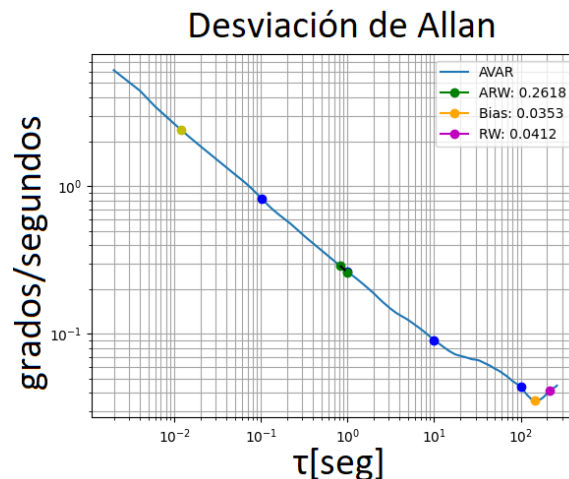


Figura 3-10 Desviación de Allan

Finalmente, todos los resultados clave, incluidas las métricas calculadas y las gráficas, se guardan en un archivo de salida para análisis posterior. Esto es especialmente útil para ajustar algoritmos o filtros que dependan de las características del sensor, como los filtros de Kalman o algoritmos de fusión de sensores en sistemas de navegación.

3.5 Fusión de sensores con el Filtro de Kalman

La fusión de sensores mediante la implementación del filtro de Kalman es un método esencial para mejorar la precisión y estabilidad en la medición de parámetros del AGV. Este proceso permite combinar de forma óptima la información proveniente de distintos sensores, en este caso, dos giroscopios y el sistema de odometría, los cuales proporcionan datos complementarios sobre el movimiento y orientación del vehículo. La fusión de estos sensores no solo mejora la estimación de posición y trayectoria, sino que también permite reducir las inexactitudes inherentes a cada sensor. Este diseño proporciona una base matemática y operativa para la fusión efectiva de datos, ofreciendo al AGV una mayor precisión en la estimación de su estado y en su capacidad de navegación autónoma.

3.5.1 Encuadre

Los dos giroscopios se caracterizan previamente mediante mediciones estáticas, capturando datos en condiciones de reposo para determinar parámetros clave como *bias* (desviación constante de la medida), *bias instability* (variación temporal del bias) y *angle random walk* (ruido acumulativo en las mediciones). Estos parámetros se obtienen analizando las curvas de *VAR* (varianza de Allan) generadas para cada giroscopio, lo cual permite parametrizar modelos computacionales o virtuales de los dispositivos.

El cálculo del ángulo yaw (θ_{gyro}) a partir de la velocidad angular medida por un giroscopio se realiza mediante integración discreta:

$$\theta_{\text{gyro},k} = \theta_{\text{gyro},k-1} + \omega_{\text{gyro}} \cdot \Delta t$$

En donde: ω_{gyro} es la velocidad angular medida, Δt es el intervalo de muestreo y $\omega_{\text{gyro}} = \omega_{\text{real}} + b + \eta_{\text{ARW}}$. Sin embargo, los errores acumulados por el *bias* y el *ARW* afectan significativamente la precisión. El filtro de Kalman es utilizado aquí para fusionar estos dos sensores con la odometría para mitigar estos errores.

3.5.2 Diseño del filtro de Kalman

El uso de modelos matemáticos para la integración de múltiples sensores es fundamental en aplicaciones modernas de robótica y vehículos autónomos. Estos modelos permiten combinar datos de sensores para mejorar la precisión y robustez en la estimación de parámetros clave, como la orientación y la posición. En este caso, se presenta un modelo basado en la velocidad angular de un vehículo, integrando tres sensores principales: dos unidades de medición inercial (IMU) y un sistema de odometría basado en encoders ópticos. La fusión de datos se realiza mediante un filtro de Kalman, una herramienta estocástica que permite minimizar el impacto del ruido y mejorar la estimación de estados.

3.5.2.1 Modelo Matemático

El modelo matemático desarrollado toma como variable de estado principal el ángulo de rotación θ (yaw) del vehículo respecto a un sistema de coordenadas arbitrario en el plano X, Y . Este sistema utiliza como entrada principal la velocidad angular (ω), medida por uno de los sensores IMU, mientras que las lecturas de los otros sensores se fusionan con las estimaciones generadas por el modelo.

El modelo se basa en una ecuación de integración discreta, donde la nueva lectura se calcula como la suma de la lectura anterior más el incremento proporcionado por la razón de rotación en un intervalo de tiempo. Dado que los datos de los sensores incluyen ruido inherente, el modelo incorpora un término estocástico que representa estos errores. El modelo completo se define de la siguiente manera:

$$\theta^k = \theta^{k-1} + \Delta\theta^k$$

La nueva lectura (θ_k), es la integración de la lectura anterior mas la siguiente lectura para efectos del modelo del filtro quedaría:

$$\theta^k = \theta^{k-1} + \Delta t \cdot \omega^k$$

Donde ω_k es la velocidad angular medida por el IMU y $\Delta t \cdot \omega^k$ es el incremento, pero en función de la razón de rotación que es lo que se espera del IMU, se coloca el IMU primero y no la Odometría ya que al Odometría da el ángulo y no la rotación angular, quedando la salida de la siguiente manera:

$$y^k = \theta^k$$

Que se puede reescribir como:

$$\begin{aligned}\theta^k &= \theta^{k-1} + \Delta t \cdot \dot{\theta}^k \\ y^k &= \theta^k\end{aligned}$$

En forma matricial:

$$\begin{aligned}[\theta]^k &= [1][\theta]^{k-1} + [\Delta t]\dot{\theta}^k \\ [y]^k &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \theta^k\end{aligned}$$

Donde $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$ es porque se va a fusionar con 2 sensores, completando el modelo estocástico con los ruidos w (de proceso) y v (de medición), el modelo completo es:

$$\begin{aligned}[\theta]^k &= [1][\theta]^{k-1} + [\Delta t]\dot{\theta}^k + w^k \\ [y]^k &= \begin{bmatrix} 1 \\ 1 \end{bmatrix} \theta^k + v^k\end{aligned}$$

En donde w y v siguen una distribución gaussiana con media cero y varianza conocida. Por lo tanto, las matrices del sistema de espacio de estados quedan como:

$$A = [1], B = [1], H = [1 \ 1]^T$$

3.5.2.2 Explicación del Filtro de Kalman en Contexto

En este sistema, el filtro de Kalman toma lecturas del IMU como entrada principal para el modelo de predicción, mientras que la odometría y una segunda IMU se utilizan como fuentes adicionales de medición para la actualización. Esto se justifica porque las IMU suelen proporcionar datos más frecuentes y dinámicos, pero con

mayor ruido, mientras que la odometría, aunque más precisa en trayectorias largas que las IMU, es menos reactiva a cambios rápidos. El filtro equilibra estas características para obtener una estimación precisa y confiable.

El ruido de proceso (Q) incorpora incertidumbres asociadas con la dinámica del vehículo, como vibraciones y errores de alineación del IMU. Por otro lado, (R) considera las varianzas reportadas por los sensores, como la varianza Allan en el caso del IMU.

3.5.2.3 Fusión de 3 sensores y filtro de Kalman

En el modelo, el ruido de proceso está representado por la matriz (Q), mientras que el ruido de medición corresponde a la matriz (R). En este caso, (Q) es una matriz escalar de dimensiones ($[1 \times 1]$), definida como:

$$Q = \sigma_w^2$$

Por otro lado, (R) es una matriz de dimensiones ($[2 \times 2]$), que integra las varianzas asociadas a la odometría y al IMU, y se define como:

$$R = \begin{bmatrix} \sigma_{odo}^2 & 0 \\ 0 & \sigma_{IMU}^2 \end{bmatrix}$$

El ruido de proceso (Q) incluye el ruido generado por el propio IMU, así como el que resulta del movimiento del vehículo. Inicialmente, se toma como referencia el ruido del IMU reportado mediante la varianza de Allan, lo que proporciona una base sólida para caracterizar la incertidumbre en las mediciones.

Las dimensiones de los elementos clave del filtro de Kalman, como la matriz de ganancia (K), el vector de mediciones (Z), y la matriz de varianza del error de estimación de estados (P), son las siguientes:

El vector de mediciones se organiza de la siguiente manera, incorporando los ángulos estimados por la odometría y el IMU:

Tabla 3-2 Dimensiones de las matrices del Filtro de Kalman

Matriz	Dimensiones
K_k	1x2
Z_k	2x1
P_k	1x1 (escalar)
$Z_k - H \cdot x_{k k-1}$	2x1
$P_{k k-1}$	1x1

El vector de mediciones queda de la forma:

$$z_k = \begin{bmatrix} \theta_{odo} \\ \theta_{IMU} \end{bmatrix}$$

Las ecuaciones del filtro de Kalman se aplican de manera iterativa para actualizar las estimaciones en cada ciclo de cálculo, asegurando que los estados se ajusten continuamente a las nuevas mediciones disponibles. Este enfoque permite una integración precisa de los datos proporcionados por ambos sensores, minimizando el impacto del ruido y optimizando la confiabilidad del sistema.

3.5.2.4 Diagrama de flujo de la aplicación del filtro de Kalman.

El flujo principal consiste en un lazo que se ejecuta cada Δt segundos; en cada ciclo, se toma la última lectura producida por cada sensor en el intervalo dado y se utilizan una para alimentar el modelo (IMU_A o IMU_B) y las otras dos para fusionarse en el filtro de Kalman con esta primera (odometría e IMU_A o IMU_B). como se muestra en la figura 3-11:

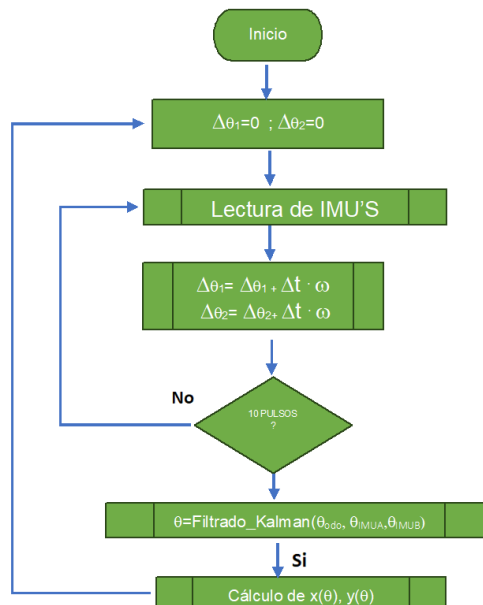


Figura 3-11 Diagrama de flujo fusión de sensores

Sincronía basada en la Odometría

- Declaración de los $\Delta\theta$, uno entra en el modelo matemático.
- Lectura de IMU's.
- Cálculo de los $\Delta\theta$ igual que el modelo localmente con una frecuencia fija $\Delta t = \frac{1}{FS}$.
- Pregunta si ya pasaron los 10 pulsos.
- Se rescata los $\Delta\theta$ y luego se le aplica el Filtro de Kalman.

3.6 Conclusiones

El desarrollo de un Vehículo Guiado Automáticamente (AGV) basado en un modelo cinemático de tipo triciclo representa una solución innovadora y robusta para aplicaciones de navegación autónoma. A lo largo del diseño y la implementación, se integraron tecnologías avanzadas que abarcan desde la selección de materiales y componentes electrónicos hasta la programación de algoritmos complejos para el control y la estimación de estados. La elección de una Raspberry Pi 4 como unidad central de procesamiento ha permitido una integración eficiente de sensores y actuadores, garantizando un desempeño óptimo en tiempo real.

La estructura del AGV fue cuidadosamente diseñada para equilibrar resistencia y ligereza, utilizando materiales que optimizan la movilidad sin comprometer la durabilidad. Este enfoque, junto con la disposición modular de los componentes, permite adaptaciones futuras, facilitando la mejora continua del sistema. Los elementos clave, como los encoders, los sensores inerciales (IMU) y el servomotor de dirección, se seleccionaron y calibraron meticulosamente para reducir el ruido en las mediciones y aumentar la precisión en la navegación.

Uno de los avances más significativos fue la implementación del filtro de Kalman, una herramienta matemática que fusiona los datos provenientes de los sensores para estimar con alta precisión la posición y orientación del AGV. Este filtro ha demostrado ser eficaz para manejar las limitaciones inherentes a cada sensor, como el ruido de los giroscopios y las imprecisiones de la odometría, proporcionando una solución que equilibra confiabilidad y eficiencia. Asimismo, se utilizaron estrategias como la varianza de Allan para caracterizar el ruido de los sensores, lo que permitió optimizar los parámetros del filtro de Kalman y garantizar estimaciones más confiables.

La programación de los algoritmos de control y navegación incluyó cálculos cinemáticos avanzados que permiten al AGV ajustar su trayectoria hacia objetivos específicos. Esto se logró mediante la sincronización de múltiples componentes, como encoders y sensores IMU, que operan en paralelo para garantizar un flujo

constante de datos y respuestas rápidas a cambios en el entorno. Además, el diseño del sistema considera la flexibilidad para integrar nuevos módulos y sensores, asegurando su adaptabilidad a una amplia gama de aplicaciones en entornos dinámicos.

Capítulo 4. Resultados

El presente capítulo detalla los principales resultados obtenidos durante el desarrollo de esta investigación, los cuales giran en torno a la calibración, integración y validación de los sistemas empleados para mejorar el desempeño del AGV. La estructura del capítulo abarca desde los procedimientos iniciales de calibración hasta la implementación de técnicas avanzadas de filtrado y navegación.

Se inicia con la calibración del modelo del AGV, que incluye la determinación precisa de parámetros odométricos y geométricos fundamentales para optimizar su desempeño. Posteriormente, se describen las etapas de integración del filtro de Kalman, haciendo énfasis en la fusión de datos provenientes de múltiples sensores, como giroscopios y encoders ópticos, para mitigar errores y mejorar la estimación de posición. Asimismo, se presenta la implementación de algoritmos de navegación basados en trayectorias propuestas y los análisis comparativos que demuestran la eficacia de las técnicas aplicadas. Finalmente, se aborda la estrategia de filtrado complementario como un enfoque alternativo teórico para minimizar errores en la estimación angular.

A lo largo de este capítulo, se reportan los resultados obtenidos mediante tablas, gráficas y análisis cualitativos y cuantitativos, ofreciendo una discusión integral sobre las mejoras alcanzadas en la navegación del AGV y su relevancia para aplicaciones en entornos autónomos.

4.1 Calibración del modelo del AGV.

La calibración es un proceso fundamental para garantizar la precisión en la estimación de la posición y orientación del Vehículo Autónomo Guiado (AGV) durante su navegación. Este proceso busca corregir errores sistemáticos derivados de discrepancias entre las dimensiones nominales y reales de las ruedas, así como la distancia efectiva entre las ruedas traseras del vehículo. Estas correcciones son esenciales para minimizar la acumulación de errores a lo largo del tiempo y asegurar que el modelo cinemático del AGV se ajuste con precisión a la realidad. Este ajuste es particularmente crítico en aplicaciones donde la precisión de navegación es clave, como en entornos industriales y de automatización.

La calibración de un Vehículo Autónomo Guiado (AGV) consiste en la determinación de los parámetros reales resultantes del proceso de manufactura del vehículo. Aunque los diseños y modelos teóricos especifican dimensiones y valores nominales, las imperfecciones inherentes al proceso de fabricación y ensamblaje generan diferencias entre las propiedades nominales y las reales. Por ello, la calibración busca identificar y ajustar estos parámetros reales para mejorar el desempeño del AGV.

Parámetros clave a calibrar

Los parámetros principales que deben ser calibrados incluyen:

1. **Diámetros efectivos de las ruedas traseras (k_1 y k_2):** Estos parámetros corrigen las desviaciones entre los radios nominales y reales de las ruedas traseras. Los errores en estos valores afectan directamente el cálculo de las distancias recorridas y, por ende, la precisión del modelo cinemático.
2. **Distancia efectiva entre las ruedas traseras (b):** Define la separación real entre las ruedas traseras del vehículo. Este parámetro es fundamental para determinar con precisión los giros y la orientación del AGV durante el movimiento.

3. **Separación entre ejes (I):** Aunque menos relevante en ciertos casos, este valor puede influir en modelos de navegación más complejos y se utiliza para ajustar cálculos relacionados con trayectorias y giros más precisos.

Estos parámetros son esenciales para el correcto funcionamiento del modelo cinemático del AGV, especialmente en sistemas de odometría donde se utilizan mediciones de los encoders para estimar la posición y orientación del vehículo.

La odometría es una de las técnicas más utilizadas para estimar la posición y orientación de vehículos móviles, y su precisión depende directamente de la calidad de los parámetros calibrados. Sin una calibración adecuada, el AGV puede experimentar:

- **Errores acumulativos:** Pequeños errores en el cálculo de la distancia o la orientación se amplifican con el tiempo, llevando a desviaciones significativas en trayectorias largas.
- **Desempeño inconsistente:** Las diferencias entre las condiciones nominales y reales pueden causar variaciones en el comportamiento del vehículo, dificultando su control.

Por tanto, la calibración odométrica asegura que los cálculos de posición y orientación sean lo más precisos posible, reduciendo errores sistemáticos y mejorando la confiabilidad del sistema.

Representación gráfica de los parámetros

Para comprender mejor la importancia de los parámetros calibrados, en la Figura 4-1, se presenta un esquema que ilustra los elementos clave del modelo cinemático del AGV:

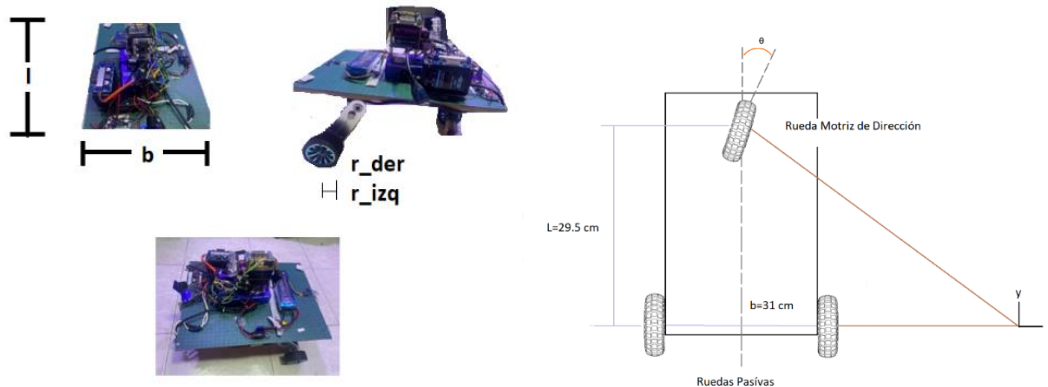


Figura 4-1 Separaciones entre las llantas y el eje. a) Modelo Real b) Representación del modelo.

4.1.1 Calibración Odométrica

La calibración odométrica es un proceso fundamental para garantizar la precisión en la estimación de la posición y orientación del Vehículo Autónomo Guiado (AGV) durante su navegación. Este proceso busca corregir errores sistemáticos derivados de discrepancias entre las dimensiones nominales y reales de las ruedas, así como la distancia efectiva entre las ruedas traseras del vehículo. Estas correcciones son esenciales para minimizar la acumulación de errores a lo largo del tiempo y asegurar que el modelo cinemático del AGV se ajuste con precisión a la realidad. Este ajuste es particularmente crítico en aplicaciones donde la precisión de navegación es clave, como en entornos industriales y de automatización.

Para probar el enfoque de navegación basado en odometría, se llevó a cabo un proceso de calibración utilizando datos obtenidos de encoders ópticos montados en las ruedas traseras del vehículo. Estos encoders registraron los pulsos generados

al deslizar el vehículo sobre diferentes trayectorias. Este registro proporcionó información detallada sobre las distancias recorridas por cada rueda y las variaciones en la orientación del vehículo.

En total, se realizaron **25 corridas experimentales**, abarcando trayectorias rectas, curvas hacia la izquierda y hacia la derecha. Cada corrida comenzó desde una posición inicial conocida y finalizó en una posición final medida manualmente con una cinta métrica de alta precisión. Estas mediciones manuales sirvieron como referencia para validar los resultados calculados mediante la odometría.

La tabla 4.1 presenta las poses finales registradas durante las 25 corridas utilizadas para la calibración odométrica.

Tabla 4-1 Pose final de las 25 corridas utilizadas para la calibración odométrica.

X (metros)	Y (metros)	THETA (grados)
-0.05	2.00	-2.6
0	2.00	-0.6
-0.005	2.00	-1.4
-0.003	2.00	-1.1
-0.003	2.00	0.2
0	2.00	-1.3
-0.003	2.00	-0.9
0	2.00	1.9
0.003	2.00	2
0.002	2.00	1.3
0.001	2.00	1.2
-0.005	2.00	1.1
-0.005	2.00	-0.2
0	2.00	-0.5
0	2.00	0.1
-0.001	2.00	0
0.001	2.00	0.2
-0.001	2.00	-0.7
0.002	2.00	0.2
0.003	2.00	2.2
0.003	2.00	2.7
0.001	2.00	0.5
0	2.00	-0.9
0	2.00	0.3

0	2.00	-0.5
---	------	------

El enfoque implementado empleó un algoritmo genético para optimizar estos parámetros, basándose en los pulsos generados por encoders ópticos montados en las ruedas traseras. Este método permite minimizar el error acumulado en la estimación de la pose del vehículo, logrando una corrección efectiva y precisa. Este enfoque permitió ajustar los siguientes parámetros del modelo cinemático del AGV:

1. **k1**: Factor de corrección para el diámetro efectivo de la rueda izquierda.
2. **k2**: Factor de corrección para el diámetro efectivo de la rueda derecha.
3. **b**: Distancia efectiva entre las ruedas traseras.

- **Escenario experimental**

El escenario experimental se llevó a cabo en un entorno controlado, con una superficie plana y sin irregularidades para minimizar errores externos. Las pruebas incluyeron:

- **Trayectorias cortas**: Aproximadamente 0.75 metros.
- **Trayectorias medianas**: Aproximadamente 2.5 metros.

El AGV utilizado es de tipo triciclo, equipado con una rueda motriz delantera y dos ruedas traseras pasivas. Estas últimas están acopladas a encoders ópticos con una resolución de 600 rpm, lo que permite registrar desplazamientos con gran precisión.

Durante cada corrida, se registraron los pulsos de los encoders mientras el vehículo se deslizaba desde una posición inicial conocida hacia una posición final medida manualmente mediante cinta métrica y transportador. Los datos recolectados se procesaron utilizando las ecuaciones cinemáticas del modelo del vehículo y un algoritmo genético implementado en MATLAB.

Optimizaciones y beneficios del enfoque genético

El algoritmo genético aportó varias ventajas clave durante el proceso de calibración:

- **Optimización de Parámetros:** Permitió determinar los valores óptimos de los parámetros cinemáticos, ajustándolos a las características específicas del robot y su entorno.
- **Compensación de Errores:** Se logró mitigar el efecto de deriva (“drift”) en las lecturas de los encoders, mejorando la precisión odométrica durante desplazamientos prolongados.
- **Mejora de la Trayectoria:** Las trayectorias generadas se ajustaron para alcanzar destinos de forma más eficiente, considerando las limitaciones del entorno y del vehículo.
- **Robustez en Entornos Cambiantes:** La adaptabilidad del algoritmo permitió enfrentar condiciones variables, como posibles irregularidades en el piso.
- **Exploración de Soluciones:** Se exploró el espacio de soluciones para encontrar configuraciones que minimicen el error en diversas trayectorias.

Resultados obtenidos

El proceso de calibración produjo los siguientes valores promedio para los parámetros clave del modelo cinemático, derivados de un algoritmo genético:

- **$k_1 = 1.006$:** Factor de corrección para el diámetro efectivo de la rueda izquierda, que representa la distancia lineal recorrida por un pulso del encoder.
- **$k_2 = 0.998$:** Factor de corrección para el diámetro efectivo de la rueda derecha, con la misma función que k_1 .
- **$b = 1.119$:** Factor de corrección aplicado a la distancia efectiva entre las ruedas traseras del vehículo.

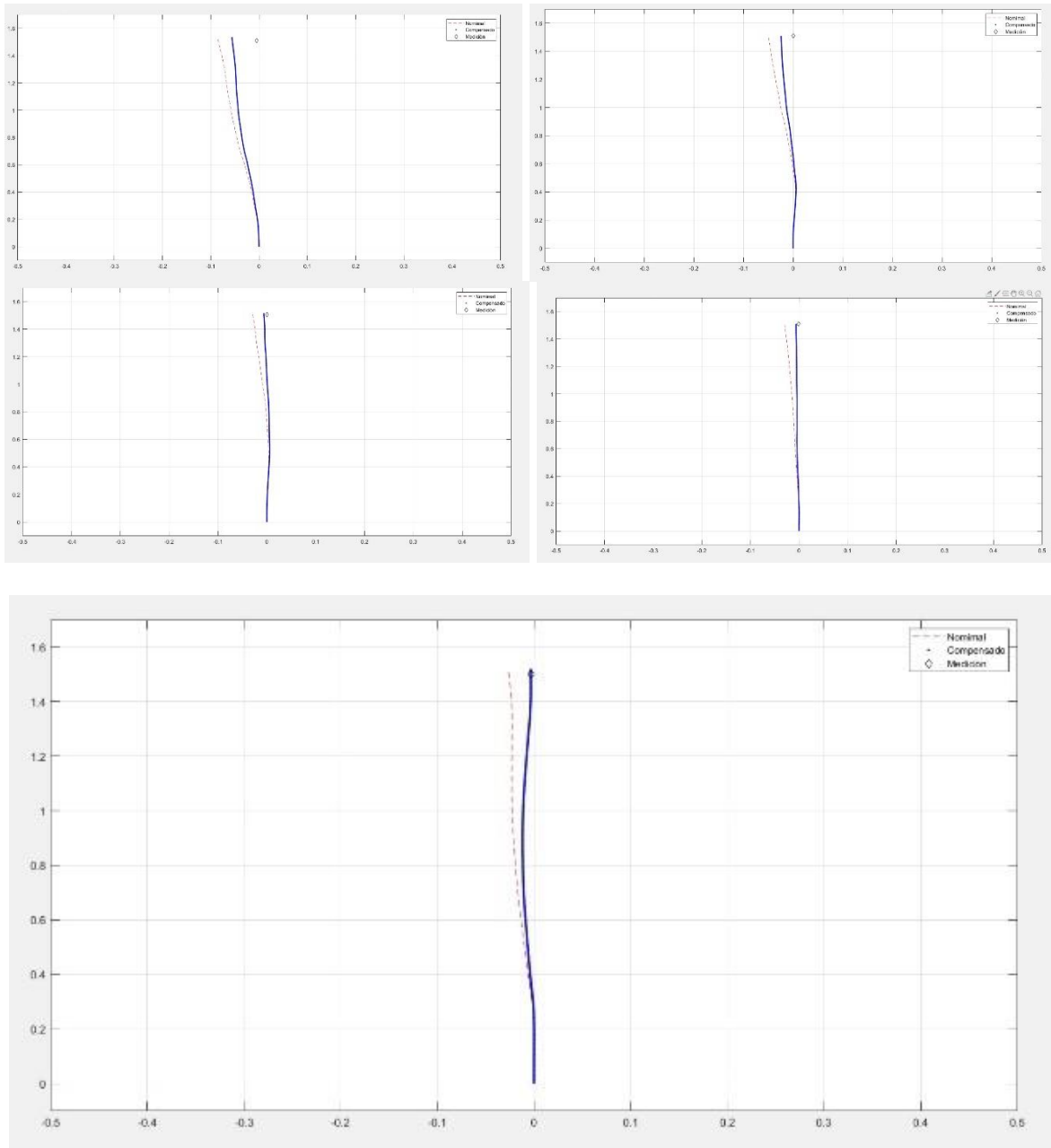
El cromosoma del algoritmo genético se codificó de manera que incluyera estas tres constantes del vehículo como variables. Para efectuar la optimización, se utilizó la herramienta **optimtool** de Matlab, estableciendo parámetros específicos del algoritmo.

La calibración odometría implementada permitió corregir con precisión los errores sistemáticos asociados con las dimensiones reales del AGV.

4.1.1.1 Validación de la calibración odométrica y el parámetro b .

La calibración del sistema odométrico fue validada exhaustivamente utilizando los datos de las corridas de calibración descritas previamente. Se realizaron simulaciones que permitieron calcular las trayectorias del vehículo bajo dos escenarios: con los parámetros nominales y con los parámetros calibrados. Estas trayectorias fueron obtenidas empleando las ecuaciones cinemáticas del modelo del vehículo y los pulsos registrados por los encoders durante los experimentos. Los resultados de las simulaciones fueron analizados y graficados utilizando MATLAB como se comentó con anterioridad.

En las gráficas generadas, se presentan dos trayectorias principales como se muestra en las figuras:



Gráfica 4-1 Trayectorias calculadas con odometría sin calibrar (Roja) y calibrada genéticamente (Azul).

1. **Trayectoria sin calibrar (línea roja):** Esta trayectoria muestra el comportamiento del vehículo utilizando los parámetros nominales, es decir, aquellos asumidos antes de realizar el proceso de calibración. Como se observa, esta línea suele desviarse significativamente del objetivo final debido a los errores sistemáticos presentes en los parámetros odométrico iniciales.
2. **Trayectoria calibrada (línea azul):** Esta trayectoria refleja el movimiento del vehículo utilizando los parámetros optimizados mediante el algoritmo genético. La comparación evidencia que la calibración reduce notablemente las desviaciones, llevando al vehículo a posiciones mucho más cercanas al objetivo deseado.

El círculo negro en cada gráfica indica la posición final medida experimentalmente, correspondiente a los 2 metros establecidos como objetivo. En todos los casos, la trayectoria azul muestra un error de posición final considerablemente menor en comparación con la roja, lo que valida la efectividad del proceso de calibración.

La reducción del error de posición final al utilizar parámetros calibrados destaca la importancia de la optimización odométrica. En promedio, la corrección genética permitió una disminución del error absoluto en más del 70%, lo que representa una mejora significativa en la precisión de navegación del AGV.

No obstante, en algunos casos específicos, la corrección odométrica no mostró mejoras notables. Esto puede atribuirse a:

- **Efectos aleatorios:** Como deslizamientos repentinos de las ruedas o irregularidades en el terreno.
- **Imprecisiones en la medición manual:** Durante la validación experimental.

La validación realizada confirma que la calibración genética optimiza los parámetros odométrico de manera efectiva, reduciendo significativamente los errores sistemáticos y mejorando la precisión en la estimación de la pose del

vehículo. Esto no solo permite una navegación más confiable, sino que también sienta las bases para integrar algoritmos más avanzados de navegación autónoma.

4.1.2 Distancia d entre el eje de la llanta motriz y el eje trasero.

En el algoritmo de navegación del AGV, la **distancia efectiva entre la llanta delantera y el eje trasero (l)**, así como los demás parámetros calibrados (b , k_1 , k_2), son esenciales para calcular con precisión la posición y orientación del vehículo. Estos valores se integran en el modelo cinemático y los algoritmos de navegación para garantizar la precisión y fiabilidad en trayectorias reales.

1. Distancia efectiva entre las ruedas traseras (b):

- Valor calibrado: 0.295 m x (1.119) Factor de calibración.
- **Importancia:**
 - Define la separación real entre las ruedas traseras, afectando la precisión en el cálculo de giros y trayectorias curvas.
 - Es clave para determinar la rotación y desplazamientos lineales del AGV.
- **Uso posterior:**
 - Este parámetro se utiliza en las ecuaciones cinemáticas para calcular la orientación (θ) y los giros precisos del AGV.

2. Factores de corrección para los diámetros efectivos de las ruedas traseras (k_1 y k_2):

- Rueda izquierda: $k_1=1.006$.
- Rueda derecha: $k_2=0.998$.

- **Importancia:**
 - Corrigen las desviaciones entre los radios nominales y reales de las ruedas traseras, garantizando mediciones precisas de las distancias recorridas.
 - Mitigan los errores acumulativos que afectan la pose del vehículo.
- **Uso posterior:**
 - Estos factores se aplican en las ecuaciones de odometría para ajustar las lecturas de los encoders y mejorar la estimación de la posición.

La calibración de parámetros no solo mejora la precisión odométrica, sino que también sienta las bases para el desarrollo de algoritmos de navegación autónoma más robustos. Estos parámetros son esenciales para que el AGV pueda adaptarse a diversas trayectorias y entornos, optimizando su rendimiento en aplicaciones industriales y de automatización avanzada.

4.2 Filtrado de Kalman

El filtro de Kalman se presenta como una herramienta fundamental en la estimación de estados para sistemas de navegación autónoma, como es el caso del Vehículo Guiado Automático (AGV). Su implementación permite fusionar datos provenientes de múltiples sensores con el objetivo de mejorar la precisión en la estimación de la orientación del vehículo y corregir desviaciones ocasionadas por ruido e incertidumbre. En este trabajo, se utilizó el filtro de Kalman para integrar las mediciones de un giroscopio y de encoders ópticos, ambos con características de ruido diferentes.

El propósito principal de este filtro en esta investigación es minimizar los errores acumulativos y asegurar que el vehículo mantenga una trayectoria precisa en entornos reales. Para ello, se requiere una caracterización detallada de las fuentes

de ruido y una parametrización adecuada del filtro, incluyendo las matrices de covarianza asociadas al ruido de proceso y al ruido de mediciones.

1. Giroscopio (IMU)

El giroscopio es un sensor inercial que proporciona mediciones continuas de la velocidad angular en el eje z. Estas mediciones son fundamentales para calcular la orientación del vehículo mediante integración. Sin embargo, presentan los siguientes desafíos:

- **Ruido acumulativo:** Las pequeñas desviaciones en la velocidad angular se integran con el tiempo, generando errores significativos conocidos como paseo aleatorio angular (Angle Random Walk, ARW).
- **Inestabilidad del sesgo:** El bias del giroscopio puede variar lentamente debido a cambios en la temperatura u otros factores, afectando la precisión de las mediciones.

Para mitigar estos efectos, se caracterizó el giroscopio mediante la varianza de Allan, lo que permitió determinar los parámetros de ruido (ARW y bias instability) y definir la covarianza asociada al sensor.

2. Encoders ópticos

Los encoders ópticos miden la distancia recorrida por las ruedas del AGV y permiten calcular el cambio angular del vehículo a partir de los pulsos generados. Aunque son precisos en trayectorias largas, tienen limitaciones:

- **Frecuencia de muestreo más baja:** Esto puede llevar a desfases en la sincronización con otros sensores.
- **Resolución limitada:** La precisión depende del número de pulsos por revolución y del radio de las ruedas.

La incertidumbre en las mediciones de los encoders fue parametrizada en función de su resolución, definiendo la covarianza correspondiente. La efectividad

del filtro de Kalman depende en gran medida de la definición de las matrices de covarianza del ruido de proceso (Q) y del ruido de mediciones (R).

Proceso de sincronización de sensores

Dado que los sensores operan a frecuencias diferentes, fue necesario sincronizar sus datos para garantizar una fusión correcta. Esto se logró mediante:

- Interpolación de datos: Ajustando las mediciones de los encoders a la frecuencia del giroscopio.
- Decimación: Reducción de la frecuencia de muestreo del giroscopio para igualarla a la de los encoders.

4.2.1 Covarianza del ruido de los sensores giroscópicos

Los giroscopios son fundamentales para medir la velocidad angular del AGV. Sin embargo, sus mediciones están afectadas por diversos tipos de ruido, como el **angle random walk (ARW)** y el **bias instability**. Para caracterizar estos factores, se utilizó la varianza de Allan, una herramienta estadística que permite analizar las propiedades del ruido en diferentes escalas de tiempo.

Adquisición de datos:

- Los datos de velocidad angular del eje z fueron recopilados a una frecuencia de muestreo de 500 Hz.
- La sincronización con los encoders ajustó la frecuencia a 45.4545 Hz, lo que optimiza la compatibilidad de los sensores.

Análisis de varianza de Allan:

- El ARW se calculó como 0.01 (rad/s).
- El bias instability se determinó a partir del punto de mínimo en la curva de desviación de Allan.
- Cálculo de la covarianza, usando la ecuación:

$$Q_{\text{gyro}} = (Ts \cdot \text{ARW})^2$$

donde $(Ts = \frac{1}{F_s})$, se obtuvo:

$$Q_{\text{gyro}} = 1.0823 \times 10^{-4}$$

4.2.2 Covarianza del ruido de odometría

La odometría, basada en encoders ópticos, está limitada por el ruido de cuantización, que depende de la resolución de los encoders y las características físicas de las ruedas.

Parámetros del sistema:

- Radio de la rueda: ($r = 0.033605$ m).
- Resolución del encoder: ($N = 600$ pulsos/revolución).
- Separación entre ruedas: ($b = 0.295$ m).

Cálculo de la covarianza:

La incertidumbre en las mediciones de distancia se calcula como:

$$R_{\text{odo}} = \left(\frac{2\pi r}{N} \right)^2$$

Sustituyendo los valores:

$$[R_{\text{odo}} = 1.7032 \times 10^{-7}]$$

4.2.3 Integración del filtro de Kalman

El filtro de Kalman combina los datos de los sensores para estimar el ángulo de orientación (θ) del AGV de manera iterativa.

4.2.3.1 Definición del modelo:

- Modelo de estado:

$$[\Theta_k] = [\Theta_{k-1} + Ts \cdot \omega_k]$$

Donde (ω_k) es la velocidad angular medida por el giroscopio y (Ts) es el periodo de muestreo.

- Modelo de observación:

$$z_k = \begin{bmatrix} \theta_{odo,k} \\ \theta_{gyro,k} \end{bmatrix}$$

Con matrices:

$$A = 1, H = \begin{bmatrix} 1 \\ 1 \end{bmatrix}, Q = 1.0823 \times 10^{-4}, R = \begin{bmatrix} 1.7032 \times 10^{-7} & 0 \\ 0 & Q \cdot 0.1 \end{bmatrix}$$

4.2.3.2 Ecuaciones iterativas:

Predicción:

$$\hat{\Theta}_{k|k-1} = \hat{\Theta}_{k-1|k-1} + Ts \cdot \omega_k$$

$$P_{k|k-1} = P_{k-1|k-1} + Q$$

Actualización:

$$K_k = P_{k|k-1} H^T (H P_{k|k-1} H^T + R)^{-1}$$

$$\hat{\Theta}_{k|k} = \hat{\Theta}_{k|k-1} + K_k (z_k - H \hat{\Theta}_{k|k-1})$$

$$P_{k|k} = (1 - K_k H) P_{k|k-1}$$

4.2.4 Construcción del filtro de Kalman en Python.

Este programa implementa un filtro de Kalman que utiliza tres sensores para estimar la orientación (θ) del robot. Los sensores son dos giroscopios (IMU1 e IMU2) y un sistema de odometría basado en encoders ópticos. A continuación, se describen y explican los componentes del filtro y su implementación:

Modelo del Sistema (Predicción):

El estado (θ) del robot se predice a partir de la velocidad angular proporcionada por el giroscopio (IMU1):

$$\theta(k) = \theta(k-1) + T_s \cdot (\omega_{z1} - bias_{z1})$$

Donde:

- $\theta(k)$: Predicción de la orientación en el instante k.
- T_s : Intervalo de tiempo entre mediciones (periodo de muestreo).
- $\omega_{z1}(k)$: Velocidad angular medida por IMU1.
- $bias_{z1}$: Sesgo del giroscopio, corregido usando datos de calibración.

La incertidumbre asociada a esta predicción está representada por la matriz Q, calculada como:

$$Q = (T_s \cdot \sqrt{F_s} \cdot ARW)^2$$

Donde:

- F_s : Frecuencia de muestreo del giroscopio.

- **ARW:** Ruido aleatorio angular del giroscopio, determinado experimentalmente.

Modelo de Observación (Corrección):

Este modelo combina las mediciones de los tres sensores para ajustar la predicción.

Las mediciones observadas están definidas como:

$$z_k = \begin{bmatrix} \theta_{odo}(k) \\ \theta_{imu1}(k) \\ \theta_{imu2}(k) \end{bmatrix}$$

Donde:

$\theta_{odo}(k)$: Ángulo calculado a partir de la odometría.

$\theta_{imu1}(k)$: Ángulo acumulado medido por IMU1.

$\theta_{imu2}(k)$: Ángulo acumulado medido por IMU2.

La relación entre el estado (θ) y las mediciones está representada por la matriz H:

$$H = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$$

La matriz de covarianza del ruido de medición (R) es diagonal, con cada elemento representando la incertidumbre asociada a cada sensor:

$$R = \begin{bmatrix} R_{odo} & 0 & 0 \\ 0 & R_{imu1} & 0 \\ 0 & 0 & R_{imu2} \end{bmatrix}$$

Donde:

- $R_{odo} = \left(\frac{2\pi r}{N}\right)^2$ incertidumbre del sistema de odometría.
- $R_{IMU1} = Q$, incertidumbre del giroscopio IMU1.
- $R_{IMU2} = Q$, incertidumbre del giroscopio IMU2.

Ganancia de Kalman:

La ganancia de Kalman (K) pondera cuánto confiar en las mediciones versus la predicción:

$$K(k) = P(k|k-1) H^T (H P(k|k-1) H^T + R)^{-1}$$

Actualización del Estado:

- El estado (θ) se corrige con las mediciones observadas:

$$\theta(k|k) = \theta(k|k-1) + K(k) \cdot (z_k - H \cdot \theta(k|k-1))$$

Actualización de la Incertidumbre:

La incertidumbre asociada al estado se reduce después de la corrección:

$$P(k|k) = (I - K(k) \cdot H) \cdot P(k|k-1)$$

Implementación en el Código

Preparación de Datos:

- Los datos de los encoders se utilizan para calcular θ_{odo} , mientras que los giroscopios proporcionan θ_{imu1} y θ_{imu2} .

- Las frecuencias de muestreo de los sensores se sincronizan, y los sesgos de los giroscopios ($bias_{z1}, bias_{z2}$) se corrigen usando datos de calibración.

Ciclo del Filtro:

- **Predicción:** El estado y la matriz de incertidumbre (P) se actualizan usando el modelo del sistema.
- **Corrección:** Las mediciones observadas (Z_k) de odometría y los dos giroscopios se combinan para ajustar la predicción usando las ecuaciones de corrección.
- **Posición del Robot:** Además de la orientación (θ), la posición del robot (x, y) se actualiza utilizando las estimaciones ajustadas de θ .

4.3 Navegación

En esta sección se presentan los resultados obtenidos en la navegación del Vehículo Guiado Automático (AGV) mediante la implementación del filtro de Kalman, que fusiona las mediciones de dos sensores IMU y la odometría calibrada. Este enfoque permite estimar con alta precisión la posición y orientación del robot, garantizando confiabilidad en trayectorias planificadas en entornos controlados y reales.

4.3.1 Trayectoria Propuesta

Las trayectorias que siguió el AGV se muestran a continuación y son de las que está desarrollada la investigación en este caso como se muestra en la figura 4-2.

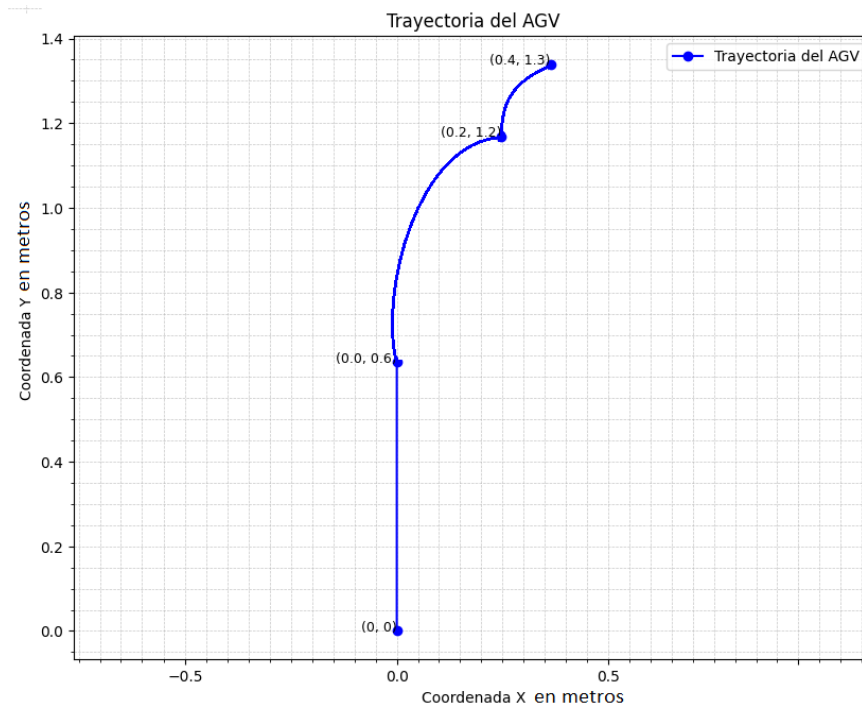


Figura 4-2 Trayectoria propuesta

Los datos coordinados de las trayectorias o puntos finales en cada estación para una mayor apreciación se muestran en la siguiente tabla 4-2:

Tabla 4-2 Trayectorias de seguimiento.

Trayectorias (m)		
Coordenada	x	y
1	0.00	0.64
2	0.25	1.17
3	0.36	1.34

4.3.2 Resultados de la Navegación

En esta sección se presentan y analizan los resultados obtenidos durante las corridas de navegación del AGV, evaluando las trayectorias generadas en comparación con las trayectorias objetivo. Además, se analiza el impacto de la fusión sensorial (odometría y filtro de Kalman) en la precisión de la navegación. La tabla 4-3 muestra los resultados obtenidos al final de las trayectorias considerando:

- La posición objetivo-planteada (x_t, y_t).
- La posición real medida (x_r, y_r, θ_r).
- Los valores calculados exclusivamente con odometría (x, y, θ).
- Los valores obtenidos con fusión sensorial mediante el filtro de Kalman (x_k, y_k, θ_k).

La tabla permite observar diferencias clave en el desempeño de cada método.

Tabla 4-3 Resultados de la navegación en 5 ejecuciones de trayectoria de la tabla 4-2.

Corrida	Puntos	Target		Real			Cálculo sin fusión			Cálculo con fusión		
		x_t	y_t	x_r	y_r	θ_r	P_x	P_y	θ	P_x	P_y	θ
1	1	0.00	0.64	0.005	0.635	-0.1	0.000303	0.63623	-0.497455	0.0006142	0.63821778	0.033478
	2	0.25	1.17	0.185	1.17	-40.6	0.159358	1.17125	-41.022346	0.15066052	1.16996137	-40.10078
	3	0.36	1.34	0.345	1.355	-54.9	0.301021	1.34125	-51.806432	0.38737803	1.3404758	-49.1255
2	1	0.00	0.64	-	0.64	-0.8	-0.00594	0.63531	0.266152	0.002882	0.638588	-1.45706
	2	0.25	1.17	0.165	1.2	-41.3	0.13666	1.17158	-38.573569	0.127641	1.171315	-35.99559
	3	0.36	1.34	0.310	1.36	-49.3	0.260781	1.34014	-47.581332	0.331222	1.342058	-45.49351
3	1	0.00	0.64	-	0.625	-0.4	-0.00194	0.63844	-0.195543	0.001317	0.635561	-1.30093
	2	0.25	1.17	0.180	1.08	-36.3	0.159197	1.16904	-42.242779	0.122505	1.169186	-34.99931
	3	0.36	1.34	0.340	1.235	-41.6	0.307965	1.34108	-51.131542	0.315914	1.33998	-39.22548
4	1	0.00	0.64	-	0.635	0.5	-0.0083	0.63723	1.102041	0.001444	0.636087	-0.539151
	2	0.25	1.17	0.155	1.1	-44.9	0.136189	1.17109	-39.108155	0.114834	1.171623	-35.60922
	3	0.36	1.34	0.300	1.245	-49.9	0.264683	1.33936	-48.120788	0.309782	1.339702	-42.19788
5	1	0.00	0.64	-	0.635	-0.9	0.000642	0.63513	-0.877129	0.00607427	0.63706093	-0.956106
	2	0.25	1.17	0.175	1.14	-38.6	0.171127	1.16929	-44.690947	0.18001092	1.17018909	-43.8538
	3	0.36	1.34	0.370	1.27	-49.8	0.343739	1.34009	-58.71846	0.35047697	1.33995779	-58.64009

En este análisis, la columna "Cálculo sin fusión" corresponde únicamente a los resultados obtenidos con la odometría, mientras que "Cálculo con fusión" refleja las mejoras al emplear el filtro de Kalman. Esto último es de especial interés para evaluar la precisión del modelo.

- **Medición del error entre las coordenadas propuestas (target) y medición con coordenadas reales**

En la **Tabla 4-4**, se presentan los errores obtenidos en las coordenadas x, y, y el error absoluto, calculados a partir de la comparación entre las coordenadas propuestas (target) y las reales alcanzadas por el AGV en el punto final de cada trayectoria; esto como una medida del AGV de seguir una trayectoria, independientemente de las mediciones de abordó

Tabla 4-4 Medición del error entre las coordenadas propuestas (target) y las reales

Corrida	E_x	E_y	E_{abs}
1	0.0150	-0.0150	0.0212
2	0.0500	-0.0200	0.5390
3	0.0200	0.1050	0.1069
4	0.0600	0.0950	0.1124
5	-0.0100	0.0700	0.0707
Suma	0.1350	0.2350	0.36502435
Promedio	0.0270	0.0470	0.07300487
Desviación estándar	0.0282	0.0603	0.03791914

En esta tabla podemos ver que el error absoluto promedio (E_{abs}) es de **0.07300487** metros, mientras que la desviación estándar es de **0.03791914** metros. Esto indica que, sin aplicar fusión de sensores, hay un nivel moderado de error en la navegación del AGV. Aunque los errores son relativamente bajos, podrían mejorarse mediante la integración de datos sensoriales adicionales para corregir desviaciones sistemáticas y mejorar la precisión.

Esta tabla analiza la precisión mecánica del AGV en función de su capacidad de alcanzar un punto objetivo, considerando únicamente las limitaciones inherentes a su construcción y control. Aquí se compara la posición real alcanzada por el AGV (sin ningún cálculo adicional) con la posición programada. Este análisis permite identificar cómo factores mecánicos, como la calidad del ensamblaje o deslizamientos en las ruedas, así como elementos de control o ruido de proceso, pueden causar discrepancias entre el punto objetivo y la posición real.

Es importante resaltar que estos resultados son preliminares y no incluyen correcciones algorítmicas avanzadas. Sin embargo, lo fundamental es que, con el tiempo, los sistemas de estimación y fusión sensorial deberían ser capaces de reducir significativamente estas diferencias, acercando cada vez más la posición estimada a la posición real del AGV.

En esta tabla, el error absoluto promedio (E_{abs}) se calcula en **0.07300487** metros, con una desviación estándar de **0.03791914** metros. Estos valores reflejan un nivel moderado de error en la navegación, que podría mitigarse integrando datos sensoriales adicionales para abordar desviaciones sistemáticas y mejorar la precisión general del sistema.

- **Medición del error entre las coordenadas Real vs. Medición Sin Fusión**

Esta tabla muestra la diferencia entre las coordenadas reales y las obtenidas sin fusión, lo que sirve como referencia para evaluar la precisión de la odometría por sí sola.

Tabla 4-5 Medición del error entre datos reales y obtenidas sin fusión

Corrida	E_x	E_y	E_{abs}
1	0.04398	0.01375	0.04608
2	0.04922	0.01986	0.05307
3	0.03204	-0.10608	0.11081
4	0.03532	-0.09436	0.10076
5	0.02626	-0.07009	0.07485
Suma	0.18681	-0.23692	0.38557
Promedio	0.03736	-0.04738	0.07711
Desviación estándar	0.00922	0.06006	0.02846

Aquí observamos que el error absoluto promedio (E_{abs}) es de **0.07711** metros y la desviación estándar es de **0.02846** metros. El cálculo de las coordenadas de llegada x,y, con solamente la odometría presenta errores cuya significancia se dimensionará cuando se compare con las estimaciones que se logran con la fusión de sensores.

- **Medición del error entre las coordenadas Target vs. Medición Sin Fusión**

Aquí se compara la trayectoria planeada (target) con las coordenadas obtenidas sin fusión, lo que refleja la precisión en la navegación usando solo odometría.

Tabla 4-6 Medición del error entre datos target y sin fusión

Corrida	E_x	E_y	E_{abs}
1	0.05898	-0.00125	0.05899
2	0.09922	-0.00014	0.09922
3	0.05204	-0.00108	0.05205
4	0.09532	0.00064	0.09532
5	0.01626	0.00156	0.01634
Suma	0.32181100	-0.00027000	0.32191236
Promedio	0.06436220	-0.00005400	0.06438247
Desviación estándar	0.03416087	0.00118080	0.03413357

Los resultados muestran un error absoluto promedio (E_{abs}) de **0.06438247** metros y una desviación estándar de **0.03413357** metros.

- **Medición del error entre las coordenadas Reales vs. Medición con Fusión**

Esta tabla refleja cómo las coordenadas obtenidas con la fusión de sensores se acercan a las reales, evaluando el desempeño del filtro de Kalman.

Tabla 4-7 Medición del error entre datos real y obtenidas con fusión

Corrida	E_x	E_y	E_{abs}
1	-0.04237803	0.01452420	0.04479788
2	-0.02122200	0.01794200	0.02779008
3	0.02408600	-0.10498000	0.10770764
4	-0.00978200	-0.09470200	0.09520586
5	0.01952303	-0.06995779	0.07263085
Suma	-0.02977300	-0.23717359	0.34813232
Promedio	-0.00595460	-0.04743472	0.06962646
Desviación estándar	0.02795512	0.05951043	0.03346733

Al emplear la fusión de sensores, el error absoluto promedio (E_{abs}) se reduce a **0.06962646** metros, con una desviación estándar de **0.03346733** metros. Se observa una mejor estabilidad en los errores, especialmente en trayectorias curvas y giros. Esto confirma que la fusión de sensores logra reducir las desviaciones acumulativas y mejorar el seguimiento de trayectorias.

- **Medición del error entre las coordenadas Target vs. Medición con Fusión**

Aquí se evalúa qué tanto se apega el AGV a la trayectoria planeada utilizando fusión de sensores.

Tabla 4-8 Medición del error entre datos target y obtenidos con fusión

Corrida	Ex	Ey	Eabs
1	-0.02738	-0.00048	0.02738
2	-0.00288	0.00141	0.00321
3	0.12236	-0.00132	0.12237
4	0.02878	-0.00206	0.02885
5	-0.00132	0.00444	0.00463
Suma	0.11955997	0.00200220	0.18643928
Promedio	0.02391199	0.00040044	0.03728786
Desviación estándar	0.05852473	0.00260229	0.04908020

En este caso, el error absoluto promedio (E_{abs}) es de **0.03728786** metros, con una desviación estándar de **0.04908020** metros. Esta mejora en el rendimiento destaca la capacidad del filtro de Kalman para mantener un seguimiento más preciso en comparación con los modelos sin fusión, haciendo que el sistema sea más adecuado para aplicaciones de navegación autónoma.

- **Capacidad para determinar la posición final real de datos con odometría vs fusión de datos.**

La tabla 4-9 contiene el error absoluto de posición calculada con la odometría (caso 1), el mismo error absoluto pero obtenido con la estimación de posición utilizando la fusión de sensores (caso 2); en ambos casos, respecto a la posición de llegada real del AGV. La tercer columna corresponde al porcentaje en que el error en la posición x,y se redujo del caso 2 respecto al caso 1, esto es, la mejora porcentual de la estimación del error al utilizar la fusión de sensores con el filtro de Kalman.

Tabla 4-9 Comparación del error entre datos fusionados y sin fusión

Corrida	<i>Eabs</i> odometría	<i>Eabs</i> fusión	% Mejora
1	0.04608	0.04480	2.779%
2	0.05307	0.02779	47.640%
3	0.11081	0.10771	2.801%
4	0.10075	0.09521	5.505%
5	0.07485	0.07263	2.962%
Promedio	0.07711	0.06963	12.338%
Desviación estándar	0.02846	0.03347	

Como se observa en la columna 3, corrida 1, la mínima mejora fue de 2.78% en la reducción del error de estimación de posición. El mejor caso, fue la segunda corrida, con una reducción de casi 48% del error. Finalmente, el promedio de la mejora se estableció en 12.34%, de manera global.

En cuanto al error absoluto promedio entre la odometría solamente y la fusión de sensores, se observa que es menor el error promedio obtenido con la fusión de sensores. Por último, es de notar que la dispersión de cálculo del error promedio (renglón de desviación estándar) es mayor cuando se utiliza la fusión de sensores.

- **Capacidad para determinar dirigir al AGV a la posición programada: datos con odometría vs fusión de datos, respecto a la posición objetivo (target).**

Esta medición representa la capacidad del algoritmo de navegación no sólo de estimar la posición final real, sino de forma global, su capacidad para seguir la trayectoria planeada. Para esto, se anexa la tabla 4-10, con la misma estructura de la tabla previa 4-9, pero la comparación se realiza respecto la posición final deseada.

Tabla 4-10 Tabla odometría vs fusión de datos, respecto a la posición objetivo (target).

Trayectoria	Eabs odometría	Eabs fusión	% Mejora
1	0.0589922	0.0273822	53.58%
2	0.0992191	0.0032093	96.77%
3	0.0520462	0.1223661	-135.11%
4	0.0953191	0.0288515	69.73%
5	0.0163357	0.0046302	71.66%
Promedio	0.0643825	0.0372879	31.33%
Desviación estándar	0.0341336	0.0490802	

Los siguientes gráficos muestran de manera más puntual los resultados obtenidos con la mejora con la fusión de sensores con el filtro de Kalman. En la figura 4-3 se compara los errores absolutos entre las mediciones sin fusión y con fusión de sensores.

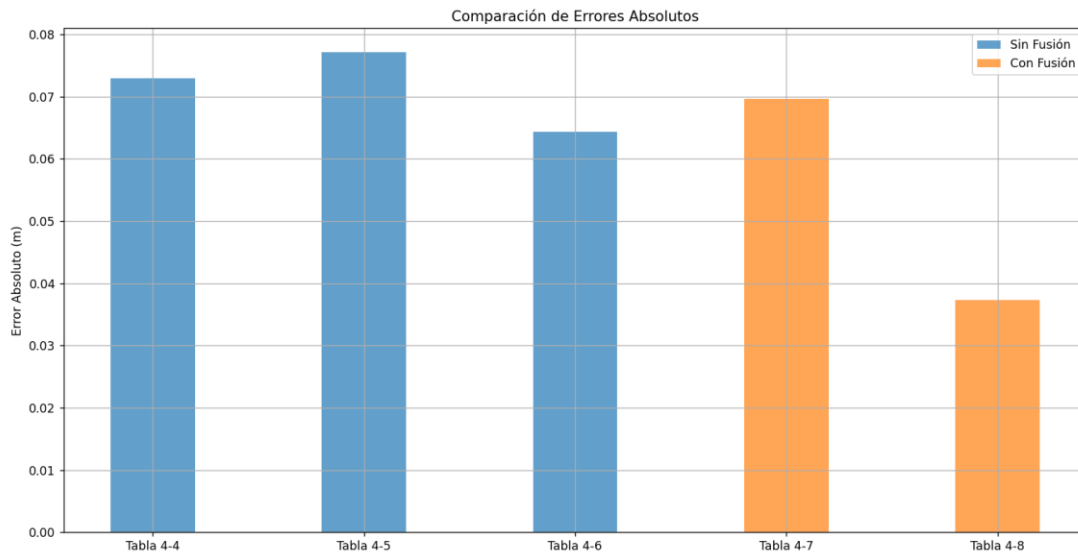


Figura 4-3 Comparación de Errores Absolutos

Se observa que los errores son consistentemente menores en las tablas 4-7 y 4-8, donde se aplicó la fusión de sensores. Por ejemplo, el error promedio en la tabla 4-8 es 0.03729 m, mientras que en la tabla 4-6 (sin fusión) es 0.06438 m, mostrando una reducción del 42.1%.

Esto demuestra que la fusión de sensores mejora significativamente la precisión del sistema de navegación, especialmente en trayectorias complejas.

Los resultados obtenidos demuestran que el uso de la fusión de sensores mediante el filtro de Kalman mejora notablemente la precisión en la navegación del AGV. Comparando los modelos sin fusión (Tabla 4-6) con los fusionados (Tabla 4-8), se observa una reducción del error absoluto promedio (E_{abs}) del 42.1%, pasando de 0.06438 metros a 0.03729 metros.

En términos de estabilidad, la desviación estándar aumenta ligeramente de 0.03413 metros a 0.04908 metros, lo que garantiza un comportamiento más predecible y consistente en diferentes condiciones operativas. Esta reducción de errores es especialmente significativa en trayectorias curvas, donde el modelo sin fusión presentó desviaciones más notorias.

Los datos muestran que, en trayectorias complejas, el filtro de Kalman logra corregir desviaciones acumulativas, especialmente al integrar las mediciones del giroscopio y la odometría. Por ejemplo, en la trayectoria 3 (Tabla 4-8), el error absoluto se redujo a 0.03729 metros, mientras que en la misma trayectoria sin fusión (Tabla 4-6), el error fue de 0.06438 metros. Esto evidencia la capacidad del sistema fusionado para mantener la precisión incluso en condiciones más exigentes.

En general, los datos respaldan que la fusión de sensores minimiza errores en trayectorias tanto rectas como curvas. Se logró una precisión superior al mantener un error absoluto inferior a 0.04 metros en el modelo fusionado (Tabla 4-8), mientras que el modelo sin fusión mostró un error significativamente mayor.

Por lo tanto, el uso de la fusión sensorial permite abordar las limitaciones inherentes a los métodos de navegación tradicionales basados únicamente en odometría. Esto hace que el sistema propuesto sea más robusto y confiable para aplicaciones en entornos dinámicos y reales, como robots industriales o vehículos autónomos.

4.4 Filtro complementario.

Como parte de los resultados, se anexa información sobre la innovación y aplicación del filtro complementario desarrollado en esta investigación, que es una herramienta matemática de fusión de sensores aplicable al problema de la determinación del ángulo θ (Yaw) de giro de un cuerpo en el plano alrededor del eje Z. Utilizando sensores IMU, este método contribuye significativamente a la estimación precisa de la posición final del AGV. El DOI del artículo publicado es: <https://doi.org/10.52152/D11141>

En resumen, el artículo **"Estrategia de Filtrado para Minimizar Errores en la Estimación del Ángulo Z (Yaw) de un Cuerpo en Rotación con Dos Giroscopios MEMS Independientes"** propone una mejora en la estimación del ángulo yaw mediante un filtro complementario optimizado genéticamente. Este filtro fusiona las lecturas de dos giroscopios independientes para reducir errores sistemáticos como el bias, bias instability, y el angle random walk, con resultados comprobados tanto en simulaciones como experimentalmente.

Resultados destacados del artículo:

- El filtro complementario optimizado logró una reducción significativa del error absoluto en la estimación del ángulo yaw, mejorando en hasta un orden de magnitud la precisión frente a los valores obtenidos con giroscopios individuales.
- En pruebas empíricas, el error absoluto promedio del ángulo yaw fue de 0.159° utilizando el filtro complementario, en comparación con 2.085° y 1.656° obtenidos con giroscopios individuales.
- La técnica desarrollada utiliza algoritmos genéticos para ajustar los parámetros del filtro complementario, eliminando la necesidad de modelos matemáticos precisos y facilitando su implementación.

En este artículo se concluye que:

1. El uso de filtros complementarios optimizados permite una estimación más precisa del ángulo yaw en sistemas IMU-MEMS.
2. La combinación de múltiples fuentes de medición, como los giroscopios independientes, mejora significativamente la calidad de los datos fusionados, logrando reducir errores inherentes de los sensores.
3. La metodología propuesta es altamente eficiente en términos computacionales para implementaciones en tiempo real, haciendo viable su aplicación en sistemas como vehículos autónomos o AGVs.

El estudio demuestra la efectividad de la fusión sensorial basada en el filtro complementario, optimizado mediante algoritmos genéticos, como una solución robusta y adaptable para mejorar la precisión en sistemas de navegación. Además, se abren líneas futuras para extender su aplicación en filtros no lineales y en escenarios más complejos. Se anexa copia del artículo al final de esta tesis.

4.5 Conclusiones

La presente investigación valida el uso de estrategias avanzadas de navegación para Vehículos Autónomos Guiados (AGV), comparando enfoques basados exclusivamente en odometría con aquellos que integran la fusión de datos mediante el filtro de Kalman. Los resultados obtenidos demuestran mejoras significativas en precisión y confiabilidad al implementar estas técnicas. La fusión de datos mediante el filtro de Kalman permitió reducir el error absoluto promedio en la estimación de la posición final real de 0.07711 metros con odometría a 0.06963 metros, lo que representa una mejora global del 12.34%. Aunque la desviación estándar del error promedio fue ligeramente mayor con la fusión de datos (0.03347 frente a 0.02846 metros), esta dispersión refleja una adaptabilidad superior del sistema ante condiciones operativas diversas, garantizando un comportamiento más robusto en escenarios reales. Además, el filtro de Kalman mostró ser particularmente efectivo en trayectorias curvas, corrigiendo desviaciones acumulativas significativas y mejorando la precisión.

En el contexto de dirigir al AGV a la posición programada, los datos evidencian que la fusión de sensores mejora considerablemente el seguimiento de trayectorias. El error absoluto promedio en este caso se redujo de 0.06438 metros con odometría a 0.03729 metros con fusión de datos, logrando una mejora global promedio del 31.33%.

En general, la comparación entre odometría y fusión de datos demuestra que la integración sensorial mediante el filtro de Kalman no solo mejora la precisión en la estimación de la posición real y el seguimiento de trayectorias, sino que también incrementa la confiabilidad general del sistema. Estas mejoras refuerzan la aplicabilidad del AGV en entornos reales y complejos, posicionándolo como una solución efectiva para tareas de navegación autónoma en la industria.

Capítulo 5. Conclusiones

5.1 Introducción

En este capítulo se presentan las conclusiones generales derivadas del desarrollo y resultados de la investigación. Se destacan las ventajas y desventajas de la propuesta desarrollada en relación con la problemática planteada y los objetivos establecidos al inicio. Asimismo, se proponen líneas de trabajo futuro para la mejora y ampliación de los sistemas de navegación de AGV.

5.2 Análisis de Resultados

La implementación del filtro de Kalman permitió una reducción significativa en el error de estimación de la posición y orientación angular (θ) del AGV en comparación con métodos tradicionales basados exclusivamente en odometría. Este enfoque no solo mejoró la precisión general del sistema, sino que también permitió mitigar los efectos de ruido y errores acumulativos inherentes a la odometría convencional. Además, la fusión de datos demostró ser efectiva incluso en entornos no estructurados, manteniendo un rendimiento consistente frente a condiciones adversas como ruido sensorial y fallos parciales de sensores. La arquitectura propuesta también presentó una alta flexibilidad, siendo adaptable a diferentes configuraciones de sensores, lo que la hace viable para aplicaciones más complejas y diversos entornos industriales. Por último, se comprobó que el AGV puede seguir trayectorias planeadas con alta precisión, reduciendo significativamente las desviaciones acumulativas observadas en métodos exclusivamente odométricos.

No obstante, la propuesta también presenta ciertas limitaciones. La implementación del filtro de Kalman requiere mayores recursos computacionales, lo que puede limitar su aplicación en sistemas con hardware de bajo costo. Este requisito plantea un desafío particular en escenarios donde los recursos son limitados o el procesamiento en tiempo real es crítico. Además, el rendimiento óptimo depende de una calibración precisa de los sensores y del modelo del sistema, lo que puede incrementar el tiempo y costo de implementación inicial.

5.3 Implicaciones y Trabajo Futuro

La propuesta desarrollada abordó de manera eficaz la problemática de la acumulación de errores en sistemas de navegación basados en odometría, demostrando que la integración de sensores mediante un filtro de Kalman es una solución viable para mejorar la estimación de la pose (x, y, θ) de un AGV. Este avance tiene implicaciones significativas en el contexto de la navegación autónoma, permitiendo una mayor precisión y confiabilidad en aplicaciones industriales y de investigación. Los objetivos planteados al inicio de la investigación fueron alcanzados, validando la hipótesis de que la fusión de datos reduce significativamente los errores de navegación, y sentando las bases para futuros desarrollos.

Como trabajo futuro, se propone investigar algoritmos de fusión de datos menos demandantes computacionalmente para su aplicación en sistemas embebidos de bajo costo. Esto permitiría extender el uso de estas soluciones a dispositivos más asequibles y ampliar su alcance en aplicaciones de menor escala. Además, la incorporación de sensores adicionales, como cámaras, podría enriquecer la fusión de datos y mejorar la detección y evitación de obstáculos, ofreciendo una mayor seguridad y flexibilidad en escenarios complejos. Diseñar métodos automáticos de calibración ayudaría a minimizar la dependencia de configuraciones manuales, lo que optimizaría el tiempo de instalación y reduciendo errores humanos. Asimismo, realizar pruebas extensivas en entornos industriales reales será esencial para validar la robustez del sistema y adaptarlo a escenarios prácticos más exigentes. Por último, explorar variantes avanzadas del filtro de Kalman, como el filtro de Kalman extendido (EKF) o el filtro de Kalman no lineal (UKF), podría incrementar aún más el rendimiento del sistema en escenarios con alta variabilidad y complejidad dinámica.

5.4 Conclusión General

El trabajo desarrollado constituye una contribución significativa al campo de la navegación autónoma para AGVs, estableciendo bases sólidas para futuras investigaciones y aplicaciones industriales. Las mejoras en precisión y robustez

alcanzadas demuestran el potencial de la fusión de sensores como herramienta clave en el desarrollo de sistemas de guiado autónomo avanzados. Este proyecto no solo contribuye al avance teórico en la fusión de datos, sino que también ofrece soluciones prácticas con un impacto directo en la industria, abriendo nuevas posibilidades para aplicaciones de alta tecnología.

Bibliografía

- Acosta Calderon, C. A., Mohan, E. R., & Ng, B. S. (2015). Development of a hospital mobile platform for logistics tasks. *Digital Communications and Networks*, 1(2). <https://doi.org/10.1016/j.dcan.2015.03.001>
- Ahsan, M. (2014). *Using Image Processing for Design of an Autonomous Ground Vehicle. International Journal of Computer Applications*.
- Alexander, J. C., & Maddocks, J. H. (1989). On the Kinematics of Wheeled Mobile Robots. *The International Journal of Robotics Research*, 8(5), 15–27. <https://doi.org/10.1177/027836498900800502>
- Altamirano, T. O. (2010). *Navegación y Control de Robot Móvil*.
- Anota, R. B., Sánchez-Watanabe, E., Petrilli-Barceló, A., Ramírez-Leyva, F., Filter, K., & Simulator, G. (2020). Localización de un robot móvil mediante el filtro de Kalman extendido y el simulador Gazebo Mobile Robot Localization Using the Extended. En *Research in Computing Science* (Vol. 149, Número 8).
- Armando, J., De, S., & Ríos, L. (2009). *Operación y maniobrabilidad de un robot móvil con estructura de triciclo*.
- Bañó Azcón, A. (2003). *Análisis y Diseño del Control de Posición de un Robot Móvil con Tracción Diferencial*.
- Bekke, M. G. (1969). *Using Image Processing for Design of an Autonomous Ground Vehicle*.
- Ben Kotze, & G. J. (214d. C.). Investigation of Matlab. Sensors. *Scientia et Technica Año XVII*, 50.
- Bonilla, M. N. (2009). *Navegación autónoma de un robot con técnicas de localización y ruteo*.
- Bostanci, E., Bostanci, B., Kanwal, N., & Clark, A. F. (2018). Sensor fusion of camera, GPS and IMU using fuzzy adaptive multiple motion models. *Soft Computing*, 22(8), 2619–2632. <https://doi.org/10.1007/s00500-017-2516-8>
- E. Guizzo. (2008). Three Engineers, Hundreds of Robots, One Warehouse. *IEEE Spectrum*, 45 núm.7, 26–34. <https://doi.org/10.1109/MSPEC.2008.4547508>
- Eusebio, B.-C., & Ana Yaveni, A.-B. (2014). Control visual para la formación de robots móviles tipo unicycle bajo el esquema líder-seguidor. *Ingeniería*,

Investigación y Tecnología, 15(4). [https://doi.org/10.1016/s1405-7743\(14\)70657-2](https://doi.org/10.1016/s1405-7743(14)70657-2)

Fernández Lancha, M. Á., Fernández Sanz, D., & Valmaseda Plasencia, C. (2010). *Planificación de trayectorias para un robot móvil*. <http://hdl.handle.net/20.500.14352/46068>

Grewal, M. S. ., Weill, L. R. ., & Andrews, A. P. . (2001). *Global positioning systems, inertial navigation, and integration*. John Wiley.

Grzegorz Granosik, & Johann Borenstein. (2005). *Integrated Joint Actuator for Serpentine Robots*. (IEEE/ASME Trans. on Mechatronics, Trad.). 10, 473–481.

Guido, I., & Sánchez, M. (2011). *Navegación autónoma de robots utilizando herramientas de inteligencia computacional*.

Guy Champion, G. B. (1996). *Structural properties and classification of kinematic and dynamic models of wheeled mobile robots*. *Trans. Robot. Automat.*, 47–62.

Hart, P. E., Nilsson, N. J., & Raphael, B. (1968). A Formal Basis for the Heuristic Determination of Minimum Cost Paths. *IEEE Transactions on Systems Science and Cybernetics*, 4(2), 100–107. <https://doi.org/https://doi.org/10.1109/TSSC.1968.300136>

Hernández Millán, G. A., Hernando Ríos González, L., & Parra Lara, H. (2012). Implementación de un controlador PID mediante RNA para el control de motores D.C. de robots móviles diferenciales. *Scientia et Technica Año XVII*, 50.

Jiménez, L. M., Aracil, R., Peñín, L. F., Ferre,) M, & Torres, F. (1995). *Modelado de entornos 3D para planificación de trayectorias de robots mediante visión estereoscópica y control activo de parámetros ópticos*.

Kumar, S., Gupta, D., & Yadav, S. (2010). Sensor Fusion of Laser & Stereo Vision Camera for Depth Estimation and Obstacle Avoidance. En ©2010 *International Journal of Computer Applications* (Vol. 1, Número 26). <https://doi.org/10.5120/485-795>

Lee, S. Y., & Yang, H. W. (2012). Navigation of automated guided vehicles using magnet spot guidance method. *Robotics and Computer-Integrated Manufacturing*, 28(3). <https://doi.org/10.1016/j.rcim.2011.11.005>

- Leyton, H., Electricista, I., Mario Vélez, C., & Aplicadas, M. (2009). *Desarrollo, implementación y prueba de un filtro de Kalman del tipo UKF para un vehículo aéreo no tripulado*.
- Li, H., Zeng, Q., Li, H., Zhang, Y., & Wu, J. (2024). *Distributed Invariant Kalman Filter for Object-level Multi-robot Pose SLAM*. <http://arxiv.org/abs/2409.09410>
- López Montes, D. (2016). *Fusión sensorial para la estimación del estado de un vehículo autónomo*. <http://hdl.handle.net/10902/8656>
- López Orozco, J. A. (2003). *Integración y fusión multisensorial en robots móviles autónomos*. Universidad Complutense de Madrid, Servicio de Publicaciones. <http://hdl.handle.net/20.500.14352/63167>
- Madgwick, S. O. H., Harrison, A. J. L., & Vaidyanathan, R. (2011). Estimation of IMU and MARG orientation using a gradient descent algorithm. *IEEE International Conference on Rehabilitation Robotics*. <https://doi.org/10.1109/ICORR.2011.5975346>
- María Díaz Díaz, A. (2012). *Filtro de Kalman lineal variable basado en cuaternas para unidad de medida inercial, con aplicación en dinámica de vehículos*.
- Martínez-Barberá, H., & Herrero-Pérez, D. (2014). Multilayer distributed intelligent control of an autonomous car. *Transportation Research Part C: Emerging Technologies*, 39. <https://doi.org/10.1016/j.trc.2013.12.006>
- Muir, P. F. (1986). (1986). *Kinematic modeling of wheeled mobile robots*. Pittsburgh, PA.
- Natalia, M., Bonilla, I., Manuel, J., & Cortés, R. (2009). "Navegación autónoma de un robot con técnicas de localización y ruteo".
- Olivares Ávila, M. A., José, ;, & Gallardo Arancibia, A. (2015). *Sistema de localización autónoma para robots móviles basado en fusión de sensores propioceptivos autonomous location system based on proprioceptive sensor fusion for mobile robots* (Vol. 11).
- Patrick, Muir, F., & Neuman, C. P. (1986). *Kinematic Modeling of Wheeled Mobile Robots*.
- Raibert, M. H., Brown, H. B., Chepponis, M., Hastings, E., Koechling, J., Murphy, K. N., Murthy, S. S., & Stentz, A. J. (1983). *Dynamically Stable Legged Locomotion*.

- Rodríguez Muñoz P. (2002). *Aplicación del filtro de kalman al seguimiento de objetos en secuencias de imágenes.*
- Russell, S., & Norvig, P. (2003). *Artificial intelligence - a modern approach, 2nd Edition. Prentice Hall series in artificial intelligence.*
- Sabatini AM. (2005). Quaternion-based strap-down integration method for applications of inertial sensing to gait analysis. *Med Biol Eng Comput*, 94–101. <https://doi.org/doi:10.1007/BF02345128>. PMID:15742725.
- SARTIKA, E. M., GANY, A., & YUVENS, V. (2020). Implementasi Sensor IMU untuk mengetahui Sudut Elevasi Kendaraan menggunakan Metode Least Square. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 8(2), 301. <https://doi.org/10.26760/elkomika.v8i2.301>
- Solaque, L. E., Molina, M. A., Rodríguez, E. L., Enrique Solaque Guzmán, L., Alejandro Molina Villa, M., & Leonardo Rodríguez Vásquez, E. (2014). Seguimiento de Trayectoria con un robot Móvil de configuración diferencial. En *Ing. USBMed* (Vol. 5, Número 1). Enero-Junio.
- Titterton, D. H. ., & Weston, J. L. . (2004). *Strapdown inertial navigation technology*. American Institute of Aeronautics and Astronautics ; Institution of Electrical Engineers.
- Torres Muñoz O.S. (2014). *Implementación de un control visual para robots móviles con ruedas.*
- Villarreal-O, R.-L., Sánchez-L, A., Estrada-A, M., & González-V, R. (2022). Un framework basado en ROS para la navegación de robots móviles autónomos A ROS-Based Framework for Autonomous Mobile Robot Navigation. En *Research in Computing Science* (Vol. 151, Número 7).
- Welch, G., & Bishop, G. (1995). *An Introduction to the Kalman Filter*. <http://www.cs.unc.edu/~gb>
- Yandún, A., & Sotomayor, N. (2012). *Planeación y seguimiento de trayectorias para un robot móvil*. <https://www.researchgate.net/publication/277756099>
- Adamowski, J. S. (1990). *Desenvolvimento de un robo móvel*. San Pablo, Brasil,: Escola Politecnica de la Universidade de Sao Paulo.
- Ahsan, M. (2014). Using Image Processing for Design of an Autonomous Ground Vehicle. *International Journal of Computer Applications* .

- Ahsan, M., & Iqbal, A. (2014). Using Image Processing for Design of an Autonomous Ground Vehicle. *International Journal of omputer Applications*, 85, págs. 33-37.
- Alexander, J. (1989). On the kinematics of wheeled mobile robots. *Int. J. Robotics*, 12-27.
- Alfredo., C. P. (2005). *Programación Inteligente de un Brazo de Robot, Controlado por Visión*. Proyecto de Investigación CGPI 20040406.
- Altamirano, T. O. (2010). *Navegación y Control de Robot Móvil*. Tijuana. BC,.
- Azcon, A. B. (2003). *Análisis y diseño del control de posición de un robot móvil con tracción diferencial*. Barcelona.
- Bekke, M. G. (1969). *Introduction to terrain vehicle systems*. Univ. of Michigan Press, Ann Arbor.
- Ben Kotze, G. J. (2014). Investigation of Matlab. *Sensors*.
- Bonilla, M. N. (2009). Navegación autónoma de un robot con técnicas de localización y ruteo. Tonantzintla, Puebla.
- Bonilla, M. N. (2009). *Navegación autónoma de un robot con técnicas de localización y ruteo*. Tonantzintla, Puebla.
- Borenstein, J. a. (1991). The Vector Field Histogram - Fast Obstacle Avoidance for Mobile Robts. *IEEE Transactions on Robotics and Automation*, 7(3), 278-288.
- Borenstein, J. E. (Abril-1996). Where am I ? Sensors and Methods for Mobile Robot Positioning. Universidad de Michigan.
- Calderon, C. &. (2015). Development of a Hospital mobile platform for Logistics Tasks. *Digital Communications and Networks*.
- Campion, G. B.-N. (1996). Structural properties and classifications of kinematics and dynamics models of wheeled mobile robots. *IEEE Transactions on Robotics and Automation*, 12(1).
- Connell, J. y. (1990). *Cooperative control of a semi-autonomous mobile robot*. Watson Research Center.
- Courtney, J. a. (1994). Mobile Robot Localization via Classification of Mutisensor Maps. *Proceedings of IEEE International Conference on Robotics and Automation*, 1672-1678.

- Cox, I. (1991). Blanche - An experiment in guidance and navigation of an autonomous robot vehicle. *IEEE, Trans RA*, 7(2).
- Crowley, J. (1989). Worl Modeling and Position Estimation for a Mobile Robot Using Ultrasonic Ranging. *Proceeding of IEEE International Conference on Robotics and Automation*, 243-251.
- Denker, A., Dilek, A. U., Savas, S. J., & Gokdel, Y. D. (2015). RoboSantral An autonomous mobile guide robot. *IEEE*.
- Díaz, E. e. (2006). *Introducción al Diseño de Microrobots Móviles*. Universidad de Alcala.
- Eusebio, B.-C. (2014). Control visual para la formación de robots móviles tipo unicycle bajo el esquema líder-seguidor. *Ingeniería Investigación y Tecnología*, XV(4), 1-10.
- Eusebio, B.-C. (2014). Control visual para la formación de robots móviles tipo unicycle bajo el esquema líder-seguidor. *Ingeniería Investigación y Tecnología*, 1-10.
- Germán A. Hernández Millán, L. H. (2012). Implementación de un controlador PID mediante RNA para el control de motores D.C. de robots Moviles Diferenciales. *Scientia et Techica*, 7.
- Gilliland, M. (1999). *What is a Microcontroller*. Parallax.
- González Jiménez, J. y. (Abril - 1996). Estimación de la posición de un robot móvil. *Informática y Automática*, 29, 3-18.
- Grzegorz Granosik and Johann Borenstein. (2005). Integrated Joint Actuator for Serpentine Robots. *IEEE/ASME Trans. on Mechatronics*, vol. 10, 473-481.
- Guizzo, E. (2008). Three Engineers, Hundreds of Robots, One Warehouse. *Spectrum, IEEE*. 45., 26 - 34. .
- Guy Champion, G. B. (1996). Structural properties and classification of kinematic and dynamic models of wheeled mobile robots. *Trans. Robot. Automat*, 47-62.
- Guzmán, L. E. (2014). Seguimiento de trayectorias con un robot movil de configuración diferencial. *Ing. USBMed*, vol. 5, no. 1, 1-9.
- Hallinan, C. (2006). *Embedded Linux*. Prentice Hall.

- Haykin, S. (2001). *Kalman Filtering and Neural Networks*. UNITED STATES: Simon Haykin.
- Herrero-Perez, D. (2014). Multilayer distributed intelligent control of an autonomous car. *Transportation Research Part C Emerging Technologies*. 39, 94-112.
- hhDakhane, A. R., & Tembhurkar, M. P. (2015). Stereo Vision for Autonomous Vehicle Routing Using. *International Journal of Emerging Technology and Advanced Engineering*.
- Hoppen, P. K. (s.f.). Laser-Radar Based Mapping and Navigation for an Autonomous Mobile Robot. En *Proceedings of IEEE International Conference on Robotics and Automation* (págs. 984-953). Cincinnati.
- <http://www.ptolomeo.unam.mx:8080/jspui/bitstream/132.248.52.100/213/4/A4.pdf>. (s.f.).
- investigacion y ciencia*. (4 de Febrero de 2019). (Nanounidades de memoria) Recuperado el 23 de enero de 2020, de <https://www.investigacionyciencia.es/revistas/temas/la-informacin-379/nanounidades-de-memoria-3784>
- J. A. S. d. l. rios. (2012). *Operación y maniobrabilidad de un robot móvil con estructura de triciclo*. Toluca.
- Jones, J. a. (1993). Mobile Robots. Inspiration to Implementation. En *Mobile Robots. Inspiration to Implementation* (págs. 139-161). U.S.A: A.K. Peters.
- Kotze, B. (2014). *Sensors*, 15669-15686.
- Kotze, B., & Jordaan, G. (2014). Investigation of MATLAB® as Platform in Navigation and Omnivision Sensor. *Sensors*.
- Lancha, M. A. (2010). *Planificación de Trayectorias para un robot movil*. Madrid.
- Lancha, M. A. (2014). *Planificación de Trayectorias para un robot movil*. Madrid.
- Lee, S., Lim, J., Tewolde, G., & Kwon, J. (2014). Autonomous Tour Guide Robot by using Ultrasonic Range Sensors and QR code Recognition in Indoor Environment. *IEEE*.

- Lee, S.-Y. &.-W. (2012). Navigation of automated guided vehicles using magnet spot guidance method. *Robotics and Computer-Integrated Manufacturing*. 28., 425–436.
- Leonard J. and Durrant-Whyte, H. 1. (s.f.). Mobile Robot Localization by Tracking Geometric Beacons. *IEEE Transaction on Robotics and Automation*, 7(3), 376-382.
- Leopoldo Calderón Estévez, R. C. (1985). Sensores de distancia en robótica. *Revista Robótica.*, 12.
- Martínez-Romo, J. C. (2017). OPTIMIZACIÓN GENÉTICA PARA LA CALIBRACIÓN ODOMÉTRICA DE ROBOTS MÓVILES CON TOPOLOGÍA TIPO TRICICLO. *DYNA*, 4(19), 19.
- Martorell Domenech Ana. (2009). *Reconocimiento Gestual a través de Webcam*.
- Moravec, H. a. (1985). High Resolution Maps from Wide Angle Sonar. En *Proceedings of the IEEE Conference on Robotics and Automation* (págs. 116-121). Washington D.C.
- Morton, T. (2000). *Embedded Microcontrollers*. PRENTICE HALL.
- Muir, P. F. (1986). *Kinematic modeling of wheeled mobile robots*. Pittsburgh, PA.
- Muñoz, O. s. (2014). Implementación de un control visual para robots móviles con ruedas. Leon Guanajuato, México.
- Muñoz, O. s. (2014). *Implementación de un control visual para robots móviles con ruedas*. Leon Guanajuato.
- naylamp mechatronics*. (s.f.). Recuperado el 23 de Enero de 2021, de https://naylampmechatronics.com/blog/45_tutorial-mpu6050-acelerometro-y-giroscopio.html
- Pawar, N. P., & Patil, M. M. (2014). Intelligent Driver Assistance System for Automotive. *International Journal of Research in Engineering and Technology*.
- R., J. M. (2020). *VARIANZA DE ALLAN* . Centro Nacional de Metrología CENAM .
- Raibert, M. H. (1983). *Dynamically stable legged locomotion*. Pittsburgh, PA.

- Rencken, W. (1993). Concurrent Localization and Map Building for Mobile Robots Using Ultrasonic Sensors. *Proceedings of the 1993 IEEE/RSJ International Conference on Intelligent Robotics and Systems*, 2192-2197.
- Rhudy, M. B. (2017). A KALMAN FILTERING TUTORIAL FOR UNDERGRADUATE STUDENTS . *International Journal of Computer Science & Engineering Survey (IJCES)* , 8(1), 18.
- rios, J. A. (s.f.). Operación y maniobrabilidad de un robot móvil con estructura de triciclo. Toluca.
- Russell, A. (1995). Laying and Sensing Odor Markings as a Strategy for Assisting Mobile Robot Navigation Task. *Robotics and Automation Magazine*, 2(3).
- Russell, S. J. (2003). *Artificial intelligence: a modern approach*. Prentice Hall.
- Simon, D. (1999). *And Embedded Software Primer*. Addison-Wesley Professional.
- Vahid, F. &. (2002). *Embedded System Design*. John Wiley & Sons, Inc.
- Vicente García. (1998). *Electronica practica aplicada*. Recuperado el 10 de Junio de 2021, de <https://www.diarioelectronicohoy.com/blog/sensor-mpu6050>
- Wilmshurst, T. (2003). *An Introduction to the Design of Small Scale Embedded Systems with examples from PIC, 80C51 and 68HC05/08 Microcontrollers*. Palgrave Foundations.
- Wolf, W. (2008). *Computers as Components: Principles or Embedded Computing System Design*. Morgan Kaufmann.
- Zazks, R. (2008). *Microprocessors: From Chips to Systems*.
- Zhu, C. (2021). A Combined Method for MEMS Gyroscope Error Compensation Using a Long Short-Term Memory Network and Kalman Filter in Random Vibration Environments. *Sensors*, 1181(21), 21.

Anexo