



**UNIVERSIDAD MICHOACANA DE
SAN NICOLÁS DE HIDALGO**



**DIVISIÓN DE ESTUDIOS DE POSGRADO DE LA
FACULTAD DE INGENIERÍA MECÁNICA**

**Diseño de un sistema para aplicar
técnicas de localización basadas en
sensores propioceptivos**

**TESIS QUE PARA OBTENER EL TÍTULO DE MAESTRO EN CIENCIAS EN
INGENIERÍA MECÁNICA**

Presenta:

Martín Ramos Hernández

Asesor:

Dr. Juan Felipe Soriano Peña

Morelia Michoacán; Agosto del 2016



Para Sonia y Jacinto



La dicha es uno de los tantos efectos inevitables que experimentamos al cosechar un logro, pero yo ya la sentía desde que ustedes me ayudaron a sembrarlo.

Gracias Leticia, Martín, Erik, abuelita Elena, Sandra, Álvaro, Diego, Sebastián, José David, José Pedro, Maribel, Daniel, Oskar, Julieta, Angélica, Orlando, Rocio, Mario, Sol, Laura, Samantha, Dr. Soriano, Dr. Pacheco, Dr. Crisanto, CONACYT, UMSNH, maestros sinodales, profesores en general, compañeros en general, el resto de mi incondicional familia, extraños que alguna vez cruzaron caminos conmigo y a mi hogar de por vida, Facultad de Ingeniería Mecánica.



Resumen

Se diseñará un sistema con las capacidades mecánicas, electrónicas y de telemetría para aplicar y experimentar con un sistema de localización basado en sensores propioceptivos, dicho sistema estará sujeto a una cantidad de características de diseño y limitaciones de diseño que competan con los reglamentos y estándares aplicados dentro del panorama de la robótica de competición, de manera que se desempeñe con las cualidades de movimiento de un robot velocista seguidor de línea.

La atención a este problema es crucial para nuestra época ya que la introducción de robots a ámbitos fuera de los industriales es una tendencia que seguirá en aumento. Gradualmente tendrán una participación cada vez más notoria en diversas tareas como, por ejemplo: jardinería, mantenimiento, vigilancia, asistencia de visitantes o incluso sillas de ruedas autónomas. La integración de estos móviles depende en medida extrema de la calidad con la que estos puedan navegarse, la destreza para leer su entorno y la accesibilidad de los componentes que competan esta tarea.

Se estudiarán las generalidades de cualquier sistema de navegación de acuerdo a la teoría, las características de los módulos en los que se divide, las particularidades de acuerdo al caso, los sub-problemas que se deben solventar para obtener una solución integral al problema de navegación, las tecnologías que se han usado dentro de la robótica móvil y las técnicas desprendidas a partir de estas tecnologías. Posteriormente se conjuntará el conocimiento adquirido para diseñar y construir dicho sistema, de manera que éste se desempeñe de manera correcta como un robot seguidor de línea y funcione como una unidad capaz registrar datos mientras se desplaza a través de su sistema sensorial.

Previendo que de esta investigación se pueda definir una plataforma móvil que no solo sirva para el fin de investigación de dichas técnicas, sino que también que este pueda ser construido a partir de elementos de carácter didáctico, con la intención de que dicha tecnología pueda ser replicada o imitada o usada para prácticas de alumnos en grados de licenciatura.

Palabras clave. Tecnologías de localización, robótica autónoma móvil, cinemática robot móvil diferencial, programación con Arduino.



Abstract

It will be design a system with mechanical, electronic and telemetry to implement and experiment with localization system based on proprioceptive sensors capabilities that system is subject to a number of design features and design limitations that are incumbent with the regulations and standards applied on panorama robotics competition, so as to play with the qualities of movement of a sprinter robot line follower.

Attention to this problem is crucial for our time since the introduction of robots outside industrial areas is a trend that will continue to increase. Gradually they will have an increasingly vocal participation in various tasks such as gardening, maintenance, monitoring, visitor attendance or even autonomous wheelchairs. The integration of these robots depends on extreme measure of the quality with which they can be navigated, the ability to read their environment and accessibility of components that fall this task.

It will be studied generalities of any navigation system according to the theory, the characteristics of the modules that are divided, the particulars according to the case, the sub-problems to be overcome for a comprehensive solution to the problem of navigation, technologies that have been used in mobile robotics and technical detached from these technologies. Subsequently it will be joined the knowledge gained and a design method to design and build the system. This system will perform properly as a line follower robot and function as a unit can record data while moving through their sensory system will coordinate.

Anticipating that this research can define a mobile platform that not only serve for the purpose of investigation of such techniques, but also that it can be built from elements of didactic character, with the intention that such technology can be replicated or imitated or used for practice of students in undergraduate degrees.

Keywords. Location technologies, mobile autonomous robotics, mobile robot kinematics differential, programming with Arduino



Contenido

Índice de figuras	7
Índice de tablas	9
Glosario	11
Nomenclatura.....	12
Objetivos.....	13
Objetivo general	13
Objetivos específicos.....	13
Definición del problema	15
Hipótesis	16
Justificación	16
Capítulo 1. Introducción.....	17
1.1. Sensorización	18
1.1.1 Sensores propioceptivos	18
1.1.2 Sensores exteroceptivos.....	22
1.2. Fusión sensorial	30
1.2.1. Fusión Bayesiana.....	32
1.2.2. Fusión basada en la teoría de Dempster-Shafer.....	34
1.3. Representaciones del entorno	35
1.4. Técnicas de localización	35
1.4.1. Localización relativa mediante medidas propioceptivas	35
1.4.2. Localización relativa mediante medidas exteroceptivos	37
Capítulo 2. Movimiento característico, cinemática y localización odométrica.....	46
2.1. Robótica a grandes rasgos.....	47
2.1.1 Clasificación general de los robots.....	47
2.1.2. Robótica móvil	48
2.2. Modelo cinemático	50
2.2.1. Modelo cinemático del robot diferencial.....	51
2.2.2. Análisis de los principales errores de localización odométrica en un robot diferencial	56
2.2.3 Determinación de la base de giro de un robot diferencial mediante el método directo 57	
2.2.4. Resultados para la localización mediante sensores de odometría	59
Capítulo 3. Diseño, selección de componentes y módulos	60



3.1.	Cualidades del diseño	61
3.1.1.	Características del robot marcadas por el reglamento	61
3.2.	Establecimiento de funciones	61
3.3.	Determinación de características	62
3.5.	Hardware del robot móvil de tracción diferencial	64
3.5.1.	Selección de los motores	64
3.5.2.	Selección de ruedas	66
3.5.3.	Elementos mecánicos	67
3.5.4.	Sistema de procesamiento	69
3.5.6.	Encoders	72
3.5.7.	Giroscopio	72
3.5.8.	Reflexivos.....	73
3.5.9.	Bluetooth	75
3.5.10.	Baterías	76
3.5.11.	Subidor de voltaje.....	78
Capítulo 4.	Diseño del sistema	79
4.1.	Diseño electrónico	80
4.1.1.	Diseño electrónico de seguidor de línea básico con elementos de competición y para fines didácticos	80
4.1.2.	Diseño electrónico de seguidor de línea con fines e investigación.	85
4.2.	Diseño mecánico.....	89
4.2.1.	Comprobación de la resistencia del chasis	90
4.2.2.	Otras cualidades del diseño mecánico	91
4.3.	Diseño de módulos virtuales.....	92
4.3.1.	Diseño de módulo de movimiento de ruedas.....	92
4.3.2.	Diseño de módulo de Lectura de encoders.....	93
4.3.3.	Diseño de módulo de seguimiento de línea, control PID	94
4.3.4.	Diseño de módulo de interacción con los sensores reflexivos	96
4.3.5.	Diseño de módulo de interacción con el MPU-6050.....	97
4.3.6.	Diseño de módulo de interacción con modulo bluetooth.	99
4.4.	Recuento de los componentes seleccionados en el diseño.....	101
Capítulo 5.	Conclusiones y trabajos futuros	102
5.1.	Conclusiones	103
5.2.	Trabajos futuros	104



Bibliografía.....	105
-------------------	-----

Índice de figuras

Figura 1 <i>Encoder</i> incremental o relativo[36]	19
Figura 2 Principio de codificador óptico de dos canales.....	20
Figura 3 Disco de codificador absoluto[37]	20
Figura 4 Acelerómetro de 3 ejes de la marca POLOLU, modelo MMA7361LC	21
Figura 5 Sensor de orientación UM7-LT de la marca POLOLU	22
Figura 6 Sensores de visión en configuración estéreo sobre un robot sencillo	23
Figura 7 Emisión y recepción de señales en un sensor ultrasónico.....	24
Figura 8 Interferencias (<i>crosstalk</i>) en el sónar: a) directa. b) indirecta.....	25
Figura 9 Sensor ultrasónico de carácter didáctico	25
Figura 10 Ejemplos de sensores láser.....	26
Figura 11 Principio de reflexión del haz infrarrojo incidente en un reflector <i>Lambertiano</i>	27
Figura 12 Sensor infrarrojo de la marca POLOLU utilizado comúnmente para detectar la presencia de objetos en rangos no mayores a 20 cm.	27
Figura 13 Izquierda y derecha: forma y vector del campo magnético de la tierra	28
Figura 14 Izquierda: componentes de salida de un compás magnético. Derecha: ecuación para determinar el ángulo de acimut	29
Figura 15 Ejemplo de fusión por intersección de elipses de covarianza.....	30
Figura 16 Arquitectura típica del sistema de percepción y fusión multisensorial de un robot móvil.....	31
Figura 17 Representación de un espacio muestra.....	32
Figura 18 Ejemplo de modelo inverso del sensor representado mediante una fusión de densidad e probabilidad. Los factores α ponderan el grado de confianza de la zona de medición [10].....	33
Figura 19 Fases de localización de un robot basada en Landmarks.....	37
Figura 20 Esquema de localización por trilateración. Este método se basa en la medición de las distancias (r_1 , r_2 , r_3) del móvil a las <i>Landmarks</i>	38



Figura 21 Esquema de localización por triangulación. Este método se basa en la medición de los ángulos de vista ($\theta_1, \theta_2, \theta_3$) del móvil a las <i>Landmarks</i>	39
Figura 22 Dos barridos láser tomados desde "o" y desde "x"; a la izquierda los mismos barridos mostrando los resultados del análisis de correspondencia[27].....	42
Figura 23 Red dinámica de Bayes que caracteriza la evolución del estado x , la medición z , y el control u en un robot móvil. El subíndice k indica la dependencia temporal del modelo	44
Figura 24 Robot manipulador fijo KR6 Figura 25 Ejemplo robot móvil.....	47
Figura 26 Robot terrestre Tlaloc Figura 27 Robot aéreo AR Drone II.....	48
Figura 28 Robot acuático Aqualelly.....	48
Figura 29 Ejemplo de configuración de robot de tracción diferencial	51
Figura 30 variables cinemáticas globales de un robot móvil diferencial	52
Figura 31 Desplazamientos relativo entre instantes de muestreo (derecha) y relaciones geométricas definidas por el movimiento (izquierda).	53
Figura 32 Esquema del sistema de medición de desplazamiento lineal mediante codificadores rotativos.....	54
Figura 33 Relaciones de transformación trayecto circular a cartesianas.....	55
Figura 34 Lugar geométrico puntos de apoyo de la base de giro en móvil diferencial.....	58
Figura 35 Lamborghini Viscacha. Seguidor de línea más rápido de América Latina	65
Figura 36 5:1 Micro Metal Gearmotor HPCB.....	65
Figura 37 Rueda SC-2008	66
Figura 38 Inspección de las dimensiones de la rueda.....	67
Figura 39 Robot Pumatron de la liga profesional de competición española	67
Figura 40 Piezas de pinza robótica impresas en 3D	68
Figura 41 Robot Velowii con chasis de fibra de carbono	69
Figura 42 Ejemplo de robot móvil usando Baby Orangutan.....	71
Figura 43 Encoders magnéticos de la marca POLOLU	72
Figura 44 Sensor MPU 6050	73
Figura 45 Sensor QTR-3A.....	73
Figura 46 Módulos seleccionados, HC-05 y HC-06	75



Figura 47 ejemplo de cargador de baterías Lipo	77
Figura 48 Batería seleccionada.....	77
Figura 49 Subidor de voltaje U3V50F6	78
Figura 50 Primer ensamble esquemático.....	80
Figura 51 Robot seguidor de línea diseñado para ser construido por estudiantes.....	82
Figura 52 Placa diseñada para robot seguidor de línea simple.....	83
Figura 53 Circuito impreso de seguidor de línea listo para imprimir	83
Figura 54 Versión B de circuito impreso para seguidor de línea	84
Figura 55 Esquema que muestra las conexiones entre los elementos previamente seleccionados para el seguidor de línea para investigación.....	85
Figura 56 Primera vista al circuito impreso del robot seguidor con fines de investigación	86
Figura 57 Ejemplo de mascara para soldar.....	87
Figura 58 ejemplo del uso de la máscara anisodonte	88
Figura 59 Chasis diseñado para robot seguidor de línea con fines de investigación	89
Figura 60 Aplicación de la malla sobre el modelo sólido	90
Figura 61 Resultados de la simulación.....	91
Figura 62 Soportes de motor que se usaran en el diseño.....	92

Índice de tablas

Tabla 1 Problemas de la localización	15
Tabla 2 Movimientos característicos típicos en robots móviles[30]	49
Tabla 3 Limite de dimensiones.....	61
Tabla 4 Clasificación de atributos	62
Tabla 5 Características de los motores	65
Tabla 6 Características de las ruedas.....	66
Tabla 7 Propiedades mecánicas de la baquelita.....	68
Tabla 8 Propiedades mecánicas del PLA.....	68
Tabla 9 Tabla comparativa de controladores para robots contra Arduino.....	70
Tabla 10 Características del puente H	71
Tabla 11 Propiedades de los encoders	72



Tabla 12 Características del MPU 6050.....	73
Tabla 13 Características sensores reflexivos.....	74
Tabla 14 Características de la batería Turnigy nano-tech.....	77
Tabla 15 Requisitos del subidor de voltaje.....	78
Tabla 16 Características del subidor de voltaje.....	78
Tabla 17 Propiedades del PLA empleadas en la simulación.....	90
Tabla 18 Propiedades de la malla usada para la simulación.....	91



Glosario

Robot: Los robots son máquinas en las que se integran componentes mecánicos, eléctricos, electrónicos y de comunicación, dotados de un sistema informático programado para control en tiempo real y percepción del entorno[1].

En el término robot confluyen las imágenes de máquinas para la realización de trabajos productivos, de imitación y comparación con seres vivos. Se trata de imitar los movimientos de otros organismos, pero también de demostrar técnicas de control remoto, incluyéndose en algunos casos funciones sensoriales primarias.

Robot móvil: Es una máquina automática que tiene la capacidad de trasladarse en un entorno.

Sistema de navegación: Otorga la inteligencia suficiente a un robot móvil como para reaccionar y tomar decisiones basándose en observaciones de su área de trabajo, sin suponer que ésta es perfectamente conocida.

Navegación: Se define como el método que permite guiar el curso de un robot móvil a través de un entorno con obstáculos.

SLAM: Del inglés *Simultaneous Localization And Mapping*, o en español; localización y mapeo simultáneos. Es una técnica usada por robots y vehículos autónomos para construir un mapa de un entorno en el que se encuentra, a la vez que estima su trayectoria al desplazarse dentro de ese entorno.

GPS: Sistema de posicionamiento global. Permite determinar la posición de un objeto, persona o vehículo alrededor del globo terráqueo por medio del uso de satélites.

Odometría: Método para estimación de posición basado en sensores de giro de rueda.

Landmark: Método para la localización haciendo uso de marcas u objetos pasivos inmersos en un entorno, cuya posición es conocida.



Nomenclatura

SDN	Sistema de navegación.
SDL	Sistema de localización.
DWR	<i>Differential wheeled robot</i> . Es un robot móvil cuyo movimiento se basa en dos ruedas accionadas por separado y colocadas en costados opuestos del robot.
A	Posición inicial del RMR.
B	Posición final preferible.
B'	Posición final obtenida.
E	Distancia entre el punto deseado y el obtenido. Precisión permisible por la navegación.
PCB	Circuito impreso.
EKF	Filtro extendido de Kalman.
dr	Distancia entre el móvil y cualquier otro objeto, medida mediante un sensor exteroceptivos.
v	Velocidad de desplazamiento.
w	Velocidad angular.
x	Posición sobre el eje horizontal en un plano cartesiano
y	Posición sobre el eje vertical en un plano cartesiano.
θ	Orientación del sistema.
S	desplazamiento del sistema.
N	Resolución del codificador rotativo.
C	Constante de conversión de pulsos en unidades de desplazamiento.
t	Tiempo.
d	Distancia nominal entre ruedas
B	Base de giro.
m	Distancia entre el punto de apoyo de la rueda al punto de medición de su lado.
φ	Diámetro del lugar geométrico.
R	Radio de las ruedas.



Objetivos

Objetivo general

Diseñar un sistema con las capacidades mecánicas, electrónicas y de telemetría para aplicar y experimentar con un SDL basado en sensores propioceptivos, tomando como características y limitaciones de diseño los reglamentos y estándares aplicados dentro del panorama de la robótica de competición, de manera que se desempeñe con las cualidades de movimiento de un robot velocista seguidor de línea.

Objetivos específicos

1. Estudiar los principales tipos de tecnologías, técnicas utilizadas y métodos de fusión dentro de la robótica móvil para generar conocimiento útil del entorno, localizarse y reconocer locaciones; de manera que la información consultada sirva en la ejecución de un juicio que defina las opciones podrían adicionarse en el diseño del SDL.
2. Desarrollar un sistema de localización únicamente basado en mediciones odométricas y características cinemáticas para un DWR, tal que pueda ser aplicado para generar datos de posición y orientación.
3. Seleccionar los elementos mecánicos que compondrán el sistema en base a cualidades y características.
4. Seleccionar el sistema de procesamiento de datos, para que forme parte de sistema de localización del robot.
5. Seleccionar los sensores de acuerdo a cualidades y características que sean útiles para ejecutar tareas de localización.
6. Seleccionar los componentes que ejecuten las tareas de telemetría en el sistema de localización.
7. Seleccionar los componentes que otorguen la autonomía energética al diseño del robot.
8. Completar el diseño mediante la creación de circuitos esquemáticos que integren los elementos seleccionados, utilizando el software Eagle 7.5.
9. Diseñar por medio de Autocad un chasis que albergue los elementos del sistema y probar sus cualidades de diseño mediante un análisis de elemento finito por medio del software COMSOL.



10. Diseñar módulos virtuales, programados en Arduino, que sirvan como subrutinas para operar los elementos seleccionados en relación al sistema de localización.



Definición del problema

Para estudiar y desarrollar sistemas de localización que puedan ser aplicados a robots móviles es necesario contar con una plataforma con la cual se puedan probar dichos sistemas, ya que la interacción con el entorno real ofrece una vista previa a como se desempeñará realmente un robot en tareas de navegación, sin prescindir de las limitaciones y variaciones que existen en el medio real y que no se pueden reproducir de manera fehaciente a través de la simulación.

El problema de localización consta de estimar la posición de un robot móvil en un mapa determinado por un sistema de coordenadas global, basándose únicamente en observaciones efectuadas por el mismo. Dichas observaciones pueden consistir en información de sensores propioceptivos y exteroceptivos.

Habrán tres instancias principales que se tomarán en cuenta para lograr una solución adecuada: sensorización, la representación del entorno, y los métodos o procedimientos para la combinación de los datos y el manejo de la incertidumbre.

Tabla 1 Problemas de la localización

Sensorización	...el robot debe ser capaz de...	Interpretar los datos que le suministran sus sensores para obtener información útil
Representación del entorno		Generar o reconocer un mapa de acuerdo a la información percibida.
Manejo de la incertidumbre		Hacer uso de un algoritmo que en base a la probabilística reduzca el grado de error lo más posible.

El modelo propuesto debe por consiguiente solventar estas problemáticas al constar de los elementos de sensorización adecuados, los módulos virtuales para la interpretación de los datos en forma útil y los módulos de comunicación para el manejo de la información.



Hipótesis

Es posible diseñar un sistema de localización para un robot tal cual se le puedan aplicar y experimentar técnicas de localización basadas en sensores interoceptivos partiendo de elementos electrónicos de carácter didáctico y haciendo uso de procedimientos de construcción accesibles para estudiantes de ingeniería.

Justificación

Es inevitable que paulatinamente las tecnologías inteligentes aborden todos los tipos de entorno, incluyendo lugares que el hombre no podría conocer si no fuera por la intervención de un autómatas (exploradores espaciales, por ejemplo). Cada uno de estos nuevos autómatas tendrá que valerse en un ambiente, tendrá que lidiar con condiciones del terreno, variables del medio y requerirá un sistema de control que incluya módulos bien estudiados de navegación.

La habilidad de navegarse adecuadamente es fundamental en la mayoría de animales y básicamente cualquier organismo inteligente. Esta necesidad se extrapola a las tecnologías inteligentes que hemos creado, dícese un robot. Es el aspecto con el cual una entidad se puede desenvolver en un medio espacial y es también el que otorga un propósito a su movilidad[2].

Al realizar un sistema de atributos que permitan la interacción entre un DWR y un SDL haciendo uso de sensores propioceptivos, incitando la participación, estudio y desarrollo de robots que se puedan ejecutar tareas en el mismo con autonomía de movilidad. Esto aportaría una solución dentro de las tecnologías móviles para entornos, los cuales están dotados de una serie de características tales que es necesario trabajar fuertemente con la parte de sensorización y a su vez dejaría un antecedente local a futuros proyectos que se lleven a cabo por parte de la UMSNH.

Siendo un artefacto ajustado a un número de características definidas, se espera que la solución tecnológica obtenida sea una opción fácil de reproducir y de bajo costo. Haciendo que el desarrollo del diseño esté al alcance de los estudiantes de licenciatura en la facultad de ingeniería mecánica.

De concebirse, el prototipo podría ser una herramienta de investigación y de uso para prácticas de laboratorio, a su vez que se podría trabajar sobre la misma para que haga presentaciones y representaciones dentro de la robótica de competición.

El contar esta herramienta formará un apoyo a la creación e integración de otras nuevas herramientas virtuales de navegación que ofrezcan conocimiento de entornos, como el campus universitario, y que funcionen en dispositivos diferentes a los robóticos, por ejemplo, en aplicaciones móviles.



Capítulo 1. Introducción



1.1. Sensorización

De manera análoga los robots cuentan, al igual del ser humano, con una cantidad de órganos que les permiten obtener conocimiento. A este tipo de instrumentaria se le conoce como sensores. Los sensores se valen de algunos principios físicos y químicos para interpretar, mediante un sistema transductor, una característica en el medio o en el mismo robot de manera cuantificable o de forma cualitativa. Actualmente existen en diferentes y muy variadas especies, las cuales no sólo dependen de la variable con la que se planea interactuar, sino también de la precisión que se necesita; el tamaño, el consumo de energía, el precio, la disponibilidad, la complejidad de uso y el tiempo de vida, son también características que intervienen en la selección de los sensores.

La manera en cómo se agrupan puede ir en función de las particularidades antes mencionadas, pero para el caso de la robótica móvil conviene clasificarlos por dos ramas principales: propioceptivos y exteroceptivos. El primer grupo recopila información dentro del robot; tal puede ser, velocidad, orientación, nivel de carga de las baterías, corriente consumida. Mientras el segundo grupo evalúa situaciones en el exterior del mecanismo; por ejemplo, temperatura, humedad, distancia, imagen.

Sobra decir que para que conocer la manera en como estos funcionan es la clave para tener un máximo aprovechamiento de ellos. Para el caso de la robótica móvil hay tendencias y constantes a la hora de hablar de sensores. En los párrafos siguientes se agrupan a grandes rasgos sus características y usos, así como ventajas y desventajas de los mismos.

1.1.1 Sensores propioceptivos

Son sensores que pertenecen a este tipo y que son de aplicaciones robóticas, están encaminados a medir condiciones y características cinemáticas de los móviles. Esto lo podemos ver al enunciar que los más comunes son los *encoders*, los acelerómetros y los giroscopios.

Ventajas: Se puede decir que su principal ventaja radica en la independencia a estímulos externos.

Desventajas: Las estimaciones producidas resultan no ser fiables en todos los casos, ya que los errores se acumulan continuamente al no disponer de referencias externas.

El contraste producido por la misma cualidad sólo se percibe si el tiempo de funcionamiento resulta ser lo suficientemente prolongado como para que el error acumulativo rebase los parámetros deseados o si existen alteraciones en el medio respecto al tiempo que obliguen a cambiar las referencias sobre las que trabajaría un sistema basado únicamente en sensores propioceptivos.

1.1.1.1 Codificadores rotatorios (*encoders*)

Estos son de los sensores obligados en robótica móvil. Su versatilidad ha definido una serie de aplicaciones que no sólo son exclusivas de este tema, sino que también tienen presencia en el área de manipuladores, los cuales recurren a codificadores que les den de la habilidad de ejecutar giros en las juntas de manera precisa. Los *encoders* son la fuente de información con la que trabaja uno de los métodos indispensables en robots auto-navegados, la odometría.



La odometría es una técnica que recurre a calificar el movimiento rotativo sobre un eje. Los más recurridos son los rotatorios, los cuales son dispositivos electromecánicos que convierten la posición angular de un eje en un código digital. El código generado puede ser particular para una serie de posición o una sola posición, de esta característica se desprenden dos tipos de codificadores, los relativos o incrementales y los absolutos, respectivamente.

Su naturaleza puede partir de una inspección magnética, mecánica u óptica, siendo la última la más recurrida, entre un elemento que girará junto con el eje y uno fijo que captará las condiciones del giro.

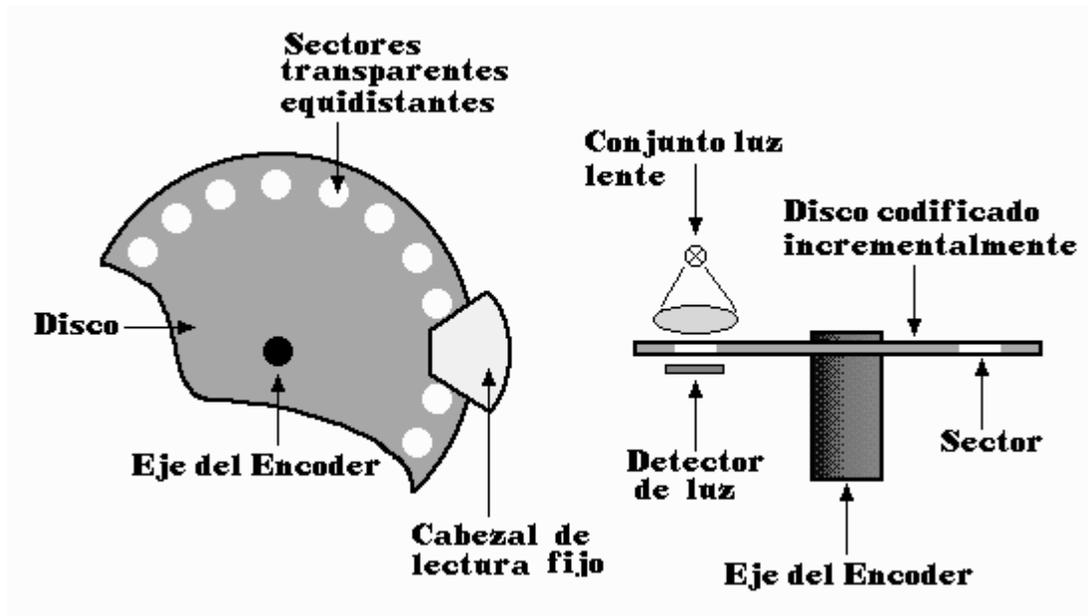


Figura 1 Encoder incremental o relativo[36]

En la figura 1, podemos observar los elementos que componen un codificador óptico. Acoplado al eje estará un disco ranurado, el cual permitirá el paso de luz de manera intermitente. La luz proviene generalmente de un led ubicado en un punto fijo del robot y de cara a las ranuras del disco. Del otro lado del disco se ubicará un elemento receptor que captará los pulsos, el cual puede ser un fotodiodo o un fototransistor. Los discos empleados pueden tener ranuras en forma circular, elíptica o rectangular. Su simplicidad queda a consideración de las aplicaciones para las que se pretende usar, ya que casi de manera única, sólo se recomienda para estimar la velocidad de rotación en un eje. Otra clara limitación de esta configuración es que no pueden detectar el cambio en el sentido de giro, aunque esta problemática queda resuelta al agregar adicionalmente un segundo canal, el cual está desplazado 90° del primero.

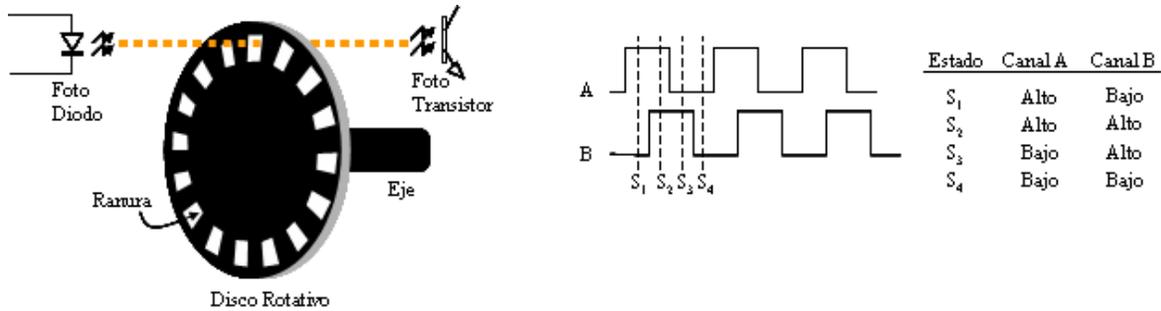


Figura 2 Principio de codificador óptico de dos canales

La figura 2, muestra cómo sería la configuración que nos permite tener hasta cuatro estados diferentes, de manera que el salto entre un estado y otro nos dé información sobre el sentido de giro.

Los *encoders* de tipo absoluto discernen entre una cantidad de posiciones en relación a la cantidad de combinaciones en las ranuras de un disco, este disco se ranura por secciones y las ranuras varían en cuanto a sus dimensiones.

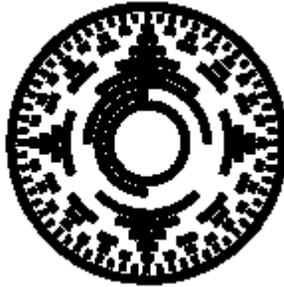


Figura 3 Disco de codificador absoluto[37]

La figura 3, muestra un ejemplo de cómo se compone un disco usado en un codificador absoluto. Se aplican principalmente para determinar condiciones de posición angular.

Ventajas: Son muy comunes, existen en una variedad de tamaños muy extensa, son fáciles de usar debido a que son de señal digital, la precisión que otorgan es lo suficientemente aceptable para que estos mismos se usen en máquinas de control numérico, su fabricación resulta relativamente sencilla, lo suficiente como para armar un sensor de este tipo con características personalizadas.

Desventajas: Las configuraciones relativas carecen de muchas virtudes con respecto a las absolutas, los codificadores absolutos son más difíciles de aplicar, debido a la cantidad de elementos que los pueden componer.



1.1.1.2. Acelerómetros

Los acelerómetros detectan los cambios en la velocidad de un móvil. Muchos de estos sensores no son capaces de medir la velocidad a la cual se está desplazando un objeto, únicamente pueden hacer toma de lectura cuando este acelera o desacera. Su costo fue una de las características que hizo que este sensor se mantuviera lejos de las aplicaciones robóticas y más cerca de las aplicaciones industriales. El creciente desarrollo tecnológico permitió que los acelerómetros no solo bajaran sus precios, sino que también participaran en dispositivos de la vida cotidiana, como son: teléfonos celulares, computadoras portátiles, juguetes o instrumentos médicos.

En la robótica móvil, los acelerómetros funcionan más bien como una manera de detectar movimiento. Su uso llega a complementar los sensores anteriores, al permitir conocer las características de un movimiento deslizante, en el cual un codificador rotativo no podría arrojar datos fieles acerca del movimiento. Sirven también para detectar movimientos cuando las ruedas están detenidas. Otros posibles usos son la detección de colisiones (las cuales se hacían tradicionalmente con sensores de contacto) o la teleoperación robótica. Un ejemplo consiste en colocar acelerómetros en los pies de una persona y en base a las lecturas que estos captan mientras la persona camina, se replica el movimiento en un robot bípedo.

Los acelerómetros existen en una gran variedad, la cual va de acuerdo a los elementos que se usen para su construcción. Existen en el mercado los que son: piezo-eléctricos, piezo-resistivos, térmicos, ópticos, capacitivos, basados en inducción magnética, etc. Su aplicación y su disposición serán los factores que imperen en el ejercicio de decisión de un acelerómetro sobre otro. Existen unos sencillos que, únicamente detectan aceleración en una dirección, mientras que hay unos más complejos que pueden llegar a funcionar hasta con 3 direcciones en monitoreo. Se puede hacer combinaciones de 3 unidireccionales para obtener el mismo resultado que no uno tridimensional.



Figura 4 Acelerómetro de 3 ejes de la marca POLOLU, modelo MMA7361LC

La figura 4, muestra un sensor comercial de tipo didáctico, dicho sensor es apenas tan largo como una pulgada y su precisión va desde 1.5 G hasta 6 G.

Ventajas: Portabilidad, disponibilidad y precio.

Desventajas: Es complicado para estos sensores detectar los cambios de velocidad de magnitud pequeña y son sensibles a las irregularidades en el suelo[3].



1.1.1.3. Giroscopios

Sirven para medir o mantener la orientación en relación a un sistema de referencia fijo. Se usan para mantener estabilizadas las embarcaciones, los telescopios espaciales, dentro del piloto automático de los aviones, entre otras utilidades. Se suelen utilizar en conjunto a los acelerómetros para dar pie a lo que se conoce como sensores de orientación.



Figura 5 Sensor de orientación UM7-LT de la marca POLOLU

Ventajas: Portabilidad, disponibilidad, precio y vienen en presentaciones conjugadas con acelerómetros.

Desventajas: Irregularidades en el terreno pueden provocar lecturas imprecisas.

1.1.2. Sensores exteroceptivos

Este apartado recoge la descripción de algunos de los sensores exteroceptivos más populares en robótica: sensores de visión (cámaras), ultrasónicos, infrarrojos y laser. En contraposición a los propioceptivos, los sensores exteroceptivos solamente detectan lo que ocurre en el exterior del móvil. Las medidas de este tipo de sensores normalmente son interpretadas por el robot para extraer características del entorno y construir un modelo del mismo

1.1.2.1. Sensores de visión

Los sensores de visión son muy importantes, debido a que proporcionan una enorme cantidad e información acerca del entorno, probablemente más que ningún otro tipo de sensor [3]. Mediante la visión, se pueden realizar una gran cantidad de tareas diferentes, por ejemplo, reconocer patrones, detectar movimientos, estimar la posición, extraer características, etc. En las máquinas, los sensores de visión tratan de emular a la vista humana por medio de sensores que sean capaces de detectar la luz del mismo modo que nosotros la detectamos. Actualmente existen dos tipos de tecnologías diferentes para construir sensores de visión: CCD y CMOS, siendo ambas más limitadas que la visión humana.

Los CCD (*Charge-Coupled Devices*) están formados por una matriz de varios millones de diminutas células fotoeléctricas sensibles a la luz denominadas píxeles (picture elements). Cada píxel se puede entender como un condensador sensible a la luz, con un tamaño de entre $5\ \mu\text{m}$ y $25\ \mu\text{m}$. Para medir la luz, en primer lugar, los condensadores se cargan y después comienza el periodo de integración. Cuando los fotones de la luz llegan a cada píxel, liberan



electrones, que son capturados y retenidos por campos eléctricos. A medida que pasa el tiempo, cada píxel acumula un nivel de carga, basado en el número total de fotones que ha almacenado. Una vez terminada la acumulación de fotones, se procede a leer la carga de cada píxel. Este proceso se efectúa en una esquina del chip CCD. Los píxeles de la fila inferior se transportan hasta la esquina y se leen, y a continuación las células de la fila superior se transportan hasta la fila inferior y se repite el proceso. Este proceso es muy complejo, y es crucial que las cargas sean transportadas correctamente a través del chip.

Los sensores de visión CMOS (*Complementary Metal Oxide Semiconductor*), al igual que los CCD, también están compuestos por una matriz de píxeles, con la diferencia de que al lado de cada píxel se encuentran varios transistores específicos para ese píxel. Los píxeles acumulan carga durante el periodo de integración, como en los CCD, pero a la hora de leer los valores de cada píxel, no hace falta leerlos por filas, sino que se pueden leer todos de una sola vez, gracias a los transistores colocados en cada píxel. Además, extraer características visuales para la navegación no es una tarea fácil, más aún si en lugar de una cámara de visión se utilizan dos (visión estéreo), para emular la visión humana [3]. A pesar de que la visión por computador lleva estudiándose desde hace décadas, existen pocos algoritmos robustos. Los métodos existentes suelen funcionar muy bien bajo determinadas circunstancias, pero si por ejemplo se modifica la iluminación o la textura del fondo, su rendimiento disminuye.

Ventajas: Los CMOS, respecto a los CCD, tienen un menor consumo de energía y su costo es menor. Ambos sensores proporcionan millones de datos.

Desventajas: Procesar los datos requiere un gasto computacional importante, aplicar técnicas basadas en estos sensores requiere una formación en el área de la informática.



Figura 6 Sensores de visión en configuración estéreo sobre un robot sencillo

1.1.2.2. Sensores ultrasónicos

Este tipo de sensores se conocen tradicionalmente como de tecnología sonar. El vocablo sonar es el acrónimo de “*sound navigation and ranging*”, o “navegación y localización por sonido”, en español. Los sensores de tipo sonar son los más empleados habitualmente en robots móviles de interiores [3]. La razón de su popularidad es que son fáciles de obtener, tienen un costo muy bajo, son muy sencillos de controlar, y además existe una gran cantidad de artículos de investigación en los que se hace uso de este sensor, con lo cual disponer de algoritmos de control está al alcance de la mano. Debido a su bajo precio, es común combinar



de estos dispositivos para equipar una plataforma robótica con cinturón de sónares, cubriendo 360°.

El principio de funcionamiento del sónar es utilizar energía acústica para efectuar medidas de distancia. Un dispositivo emisor envía un paquete de ondas medidas de distancia. Un dispositivo emisor envía un paquete de ondas de presión ultrasónicas, y posteriormente se registra el tiempo que tardan las ondas en reflejarse y volver a un receptor. El tiempo de ida y vuelta se denomina tiempo de vuelo (*time of flight*). La distancia (dr) al objeto que causó la reflexión se puede calcular basándose en la velocidad de propagación del sonido (c), y el tiempo de vuelo (t):

$$dr = \frac{c * t}{2} \quad (1.1)$$

La velocidad del sonido en el aire (c) viene dada por:

$$c = \sqrt{\gamma RT} \quad (1.2)$$

Donde γ es el coeficiente adiabático, R es la constante universal de los gases, y T es la temperatura absoluta en grados Kelvin. En el aire, en condiciones normales de presión y temperatura, la velocidad del sonido es aproximadamente $c = 343 \text{ m/s}$.

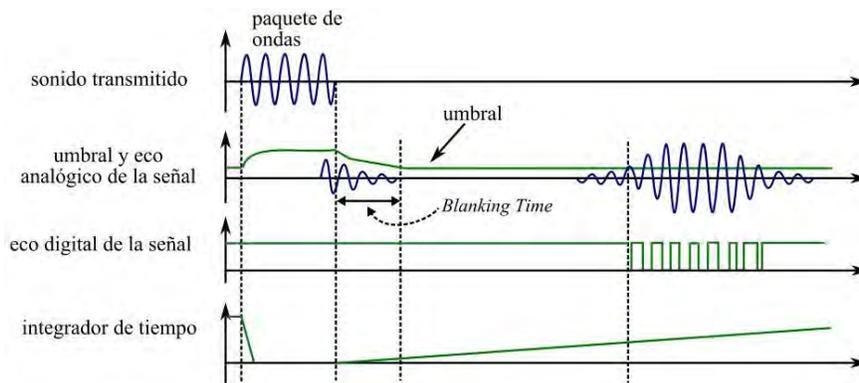


Figura 7 Emisión y recepción de señales en un sensor ultrasónico

La figura 7 muestra el funcionamiento de envío y recepción de señales de un sónar ultrasónico. Primero, se emiten una serie de pulsos de sonido conocidos como paquete de ondas. Cuando la transmisión de ondas termina, un integrados comienza a incrementar su valor, con el objeto de medir el tiempo desde la trasmisión de las ondas hasta la detección de un eco. En el receptor se restablece un valor umbral que, en caso de ser superado al recibir una onda de sonido, se considera que la onda es un eco de entrada válido. Normalmente en el sónar, el emisor y el receptor son el mismo dispositivo, pero no al mismo tiempo. La transformación de uno en otro no puede ser instantánea, porque la membrana electrostática que se usa para emitir los pulsos ultrasónicos no puede usarse como receptor hasta que no haya dejado de vibrar por completo. El tiempo que tarda en detenerse la membrana se denomina *blacking time* (tiempo borrado). Debido a esto, durante la transmisión del pulso de sonido y durante el tiempo de borrado, el umbral se deja con un valor más alto de lo normal



para eliminar la posibilidad de que el receptor detecte las vibraciones de onda emitida. Una vez pasado el tiempo de borrado, si se recibe una onda que supere el umbral, el sónar producirá una señal digital y calculará la distancia utilizando el valor del integrador.

Desventaja: la propagación de la energía acústica se ve afectada por varios factores, como puede ser la reflexión acústica de los materiales del entorno. Algunos materiales pueden no reflejar la onda con la intensidad necesaria para ser detectada por el sónar. Por ejemplo, la espuma y las prendas de piel, pueden absorber las ondas de sonido. Otro problema, que se produce cuando se tiene un robot móvil con sónares cubriendo los 360°, son las interferencias (*crosstalk*), representadas en la figura 8.

Ventajas: Tecnología accesible, bajo costo, fácil aplicación, datos fácil interpretación.

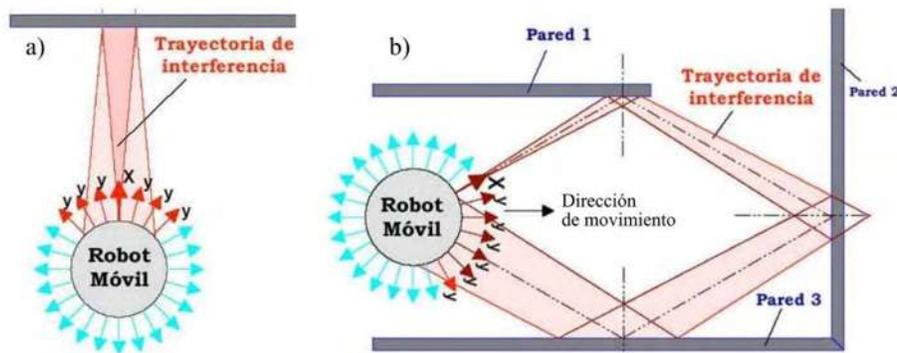


Figura 8 Interferencias (*crosstalk*) en el sónar: a) directa. b) indirecta

Las interferencias directas se dan cuando el sónar x emite un pulso de sonido y al reflejarse directamente en una pared u otro objeto, la recibe el sónar x , junto con los sónares y y adyacentes. Las interferencias indirectas se dan cuando el sónar x envía un pulso de sonido y mediante una serie de reflexiones, lo reciben los sónares y , pero no el x . Para eliminar las interferencias hay que evitar utilizar todos los sónares simultáneamente.



Figura 9 Sensor ultrasónico de carácter didáctico

En la figura 9 se muestra lo que sería el sensor más popular de tecnología sónar, conocido como HC-SR04. Se puede encontrar en casi cualquier parte y su precio no supera los \$50 MXN.



1.1.2.3. Láser

La palabra láser responde a las siglas *Light Amplification by Stimulated Emission of Radiation* (amplificación de luz mediante la emisión inducida de radiación). Al igual que el sónar, utiliza el principio de tiempo de vuelo para medir distancias (ec. 1.1). Entre sónar y láser existe una diferencia fundamental: la velocidad de propagación. Para el sonido es 0.3 m/ms, mientras que para las señales electromagnéticas es 0.3 m/ns, es decir, 1 millón de veces más rápido. Por ejemplo, en una distancia de 3 m. un sónar tardaría 10 ms. Mientras que un láser mediría la distancia en 10 ns. Es evidente que para medir el tiempo de vuelo de señales electromagnéticas se necesita una tecnología más avanzada que para medir el tiempo de vuelo de un sónar.

Ventajas: En la mayoría de los casos, las medidas pueden ser consideradas como instantáneas. La precisión de un láser es mucho mayor que la de un sónar. Por ejemplo, existen modelos que tienen una desviación estándar de error en la medida de tan sólo 1 mm, la distancia máxima que pueden medir es de 80 m, y la resolución angular es de 0.25° . Los datos de un escáner láser pueden interpretarse directamente, puesto que representa distancias a objetos. Con respecto a los sensores de visión, el tiempo de procesamiento de los datos de un láser es mucho menor.

Desventajas: Precio elevado, algunos objetos de cristal son invisibles al sensor. Normalmente los datos de un láser se limitan a un plano, si bien esto se puede solventar colocando el sensor en un dispositivo *pan & tilt*, de modo que se puedan obtener datos en 3D balanceando y girando el láser en diferentes planos.

Debido a sus ventajas con respecto al resto de sensores, el uso de escáneres láser de medición de distancia está muy difundido últimamente y es un campo muy activo de investigación hoy en día.



Figura 10 Ejemplos de sensores láser

1.1.3. Sensores infrarrojos

El transductor de infrarrojos es sensible a la intensidad de una señal luminosa. Por eso uno de los principales inconvenientes en las aplicaciones de medición por infrarrojos es el efecto de otras señales luminosas que no son propiamente las que se quieren medir (luz solar, luces artificiales). Un sensor de distancia por infrarrojos está compuesto típicamente por un emisor



LED y un fotodiodo encerrado en una cápsula que filtra la radiación visible. En su funcionamiento más básico el emisor es excitado mediante un pulso de corriente continua, emitiéndose así un haz de luz del espectro infrarrojo que, al ser reflejado por algún objeto, entonces es captado por el fotodiodo el cual produce un pequeño voltaje de salida que depende de la cantidad de luz que alcance el receptor (figura 11).

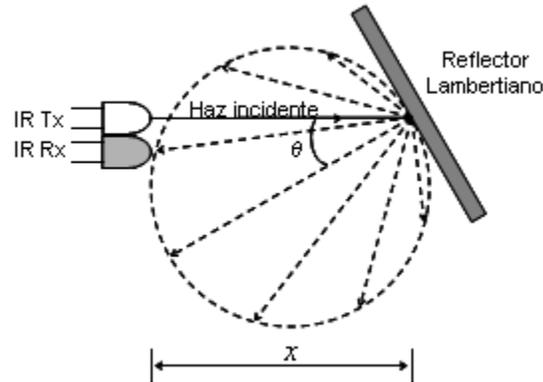


Figura 11 Principio de reflexión del haz infrarrojo incidente en un reflector *Lambertiano*

La energía infrarroja que refleja un objeto posee las siguientes características:

- 1 Decae aproximadamente en forma cuadrática con la distancia entre el reflector y el receptor.
- 2 Es mayor cuando más refractivo sea el objeto. El coeficiente de reflexión depende principalmente del color, brillo y textura del objeto.
- 3 Es mayor cuanto más cercano sea a 0° el ángulo que forma el sensor con el normal al objeto reflector.

Ventajas: Precio, disponibilidad, variedad dependiendo de la aplicación (rango de distancias).

Desventajas: Las características de la reflexión dependen de las cualidades del objeto sobre el que se proyecta, existe interferencia por otras señales luminosas provenientes de fuentes como el sol.

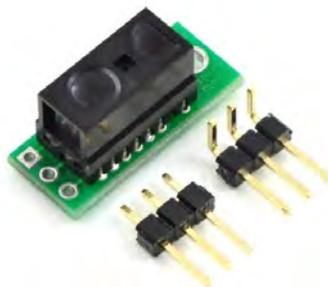


Figura 12 Sensor infrarrojo de la marca POLOLU utilizado comúnmente para detectar la presencia de objetos en rangos no mayores a 20 cm.

1.1.4. Compas magnéticos

La integración del compás en la robótica móvil es una posibilidad ampliamente considerada por los investigadores [4]. Sin embargo, en la práctica son pocos los robots equipados con compás que han sido reportado en la literatura, ya que su uso en ambientes interiores tiene una importante desventaja: la señal de salida del compás puede ser fácilmente contaminada por fuentes electromagnéticas (i.e. estantería metálica). Esto hace imposible usar las brújulas electrónicas como sensores de orientación confiables en ambientes interiores. Debido a lo anterior, en general el compás es usado más que nada como sensor de apoyo a otros sistemas sensoriales, como por ejemplo cuando se usa para compensar el efecto de los derrapes en las ruedas de odometría [5]. Por otra parte, de los tres estados (x, y, θ) usualmente controlamos en un robot móvil, el más significativo que es la orientación θ debido a la gran influencia que tiene en la acumulación de errores de estimación de la posición. Por esta razón, los sensores que proveen una medición de la orientación absoluta o de la velocidad angular relativa son extremadamente importantes en la resolución de los problemas de navegación de una plataforma móvil autónoma. De aquí que el compás siga siendo de interés en la comunidad científica en la robótica móvil.

El compás se basa en la medición de las componentes de campo magnético de la tierra para determinar la orientación del móvil donde se encuentra instalado. Los instrumentos sensibles al campo magnético son conocidos como magnetómetros, y para aplicaciones de navegación en robótica móvil, son sólo de interés aquellos que son capaces de medir el campo magnético de la tierra, y expresarla a través de una señal eléctrica. Este tipo de instrumentos se les conoce como compás electrónico. La intensidad del campo magnético cuyas líneas de campo se originan en un punto cerca del sur geográfico de la tierra y terminan en un punto cercano al norte geográfico de la tierra. La dirección e intensidad de campo magnético se representa por un vector de tres componentes (H_x, H_y, H_z) según se puede ver en la figura 13.

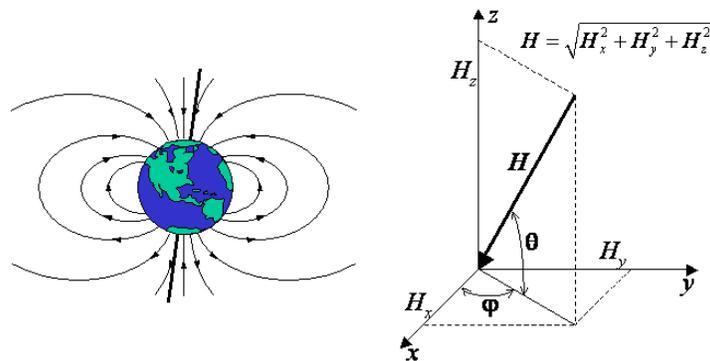


Figura 13 Izquierda y derecha: forma y vector del campo magnético de la tierra

Un tipo de compás electrónico muy usado es el basado en transductores magneto resistivos, cuya resistencia eléctrica varía con los cambios del campo magnético aplicado. Estos tipos de sensores presentan sensibilidades por debajo de 0.1 miliGauss con tiempos de respuesta menor a 1 μ s, lo que permite su uso confiable en vehículos que se mueven a altas velocidades [6].



El compás eléctrico más simple está compuesto por dos transductores magneto resistivos alineados verticalmente y desfasados 90° entre sí. Estos transductores presentan su máximo valor de salida cuando su respectivo eje se encuentra alineado con el polo norte de la tierra. De esta manera, si el compás se encuentra en una zona abierta, despejada y suficientemente plana, las componentes de salida del magneto-metro (H_x , H_y) pueden ser modeladas como $\cos \varphi$ y $\sin \varphi$ respectivamente, donde φ representa el acimut referido al norte magnético. Ya que $\varphi = \arctan (H_y/H_x)$, entonces en el caso más simple, se puede determinar la orientación de un compás mediante la medición de las componentes de campo paralelas a la superficie terrestre. En este modelo hay que tomar en cuenta que la tangente así calculada es válida sólo para el intercalo $-90^\circ < \varphi < 90^\circ$, y que además no se permite la división por $H_x = 0$. Por eso, un conjunto de ecuaciones que puede ser usado para determinar la orientación de un compás de este tipo son las mostradas en la figura 14.

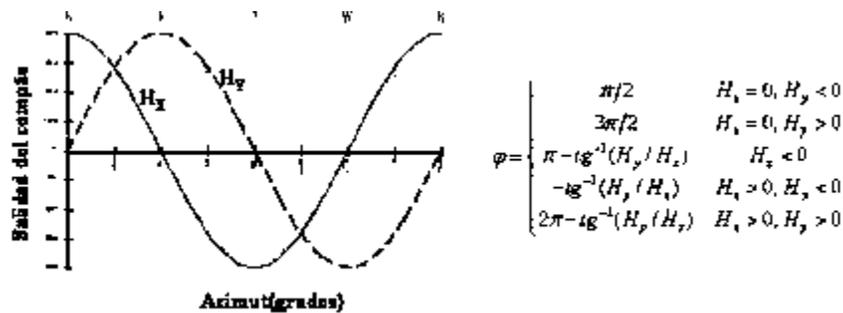


Figura 14 Izquierda: componentes de salida de un compás magnético. Derecha: ecuación para determinar el ángulo de acimut

1.2. Fusión sensorial

La fusión sensorial se refiere a la combinación de diferentes datos provenientes de un mismo sensor a diferentes instantes de tiempo (fusión en tiempo) o provenientes de múltisensores al mismo tiempo (fusión multisensorial). La fusión de datos persigue mejorar la exactitud y realizar inferencias más específicas que las que se podrían obtener usando un simple dato o un solo sensor. En este sentido, la fusión de datos multisensoriales, además de proporcionar la ventaja del filtrado estadístico que se gana con la combinación de los datos provenientes de la misma fuente (observaciones redundantes), puede incrementar la exactitud del resultado del proceso gracias a las características individuales de cada sensor. Por ejemplo, un sónar tiene buena capacidad para determinar con exactitud la distancia a la cual se encuentra un determinado objeto, sin embargo, tiene capacidad limitada para determinar su posición angular. Por otra parte, un sensor de infrarrojos puede determinar con buena exactitud la dirección angular en la que se encuentra el objeto, pero al mismo tiempo es menos preciso al momento de determinar la distancia al cual se encuentra el mismo. Si estas dos observaciones son combinadas adecuadamente, entonces el estimado de la posición del objeto puede ser mejorado. La idea de este proceso se muestra en la figura 15.

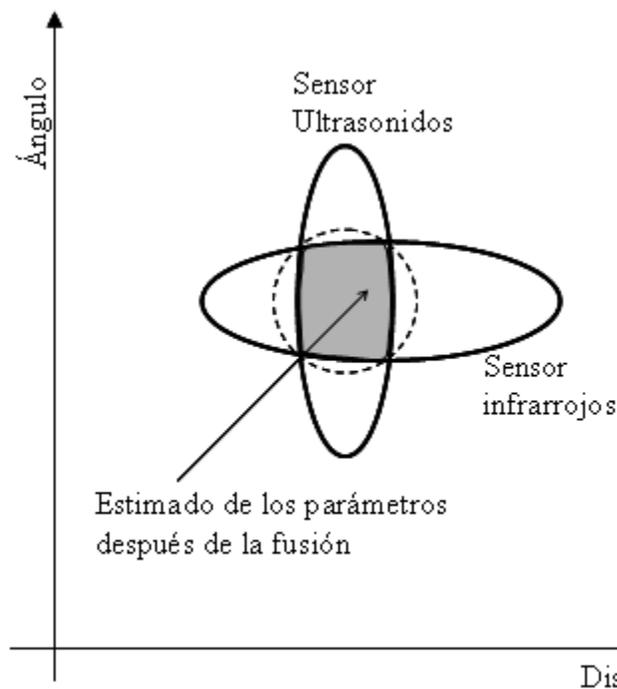


Figura 15 Ejemplo de fusión por intersección de elipses de covarianza

Una forma de clasificar la fusión sensorial está basada en el nivel de señal al cual se realice el proceso[7]: fusión a nivel de observación o de señal. A nivel de vector de estado o característica, o a nivel de decisión.

Señal. Los datos a nivel de señal pueden ser combinados directamente siempre que los datos de los distintos sensores estén conmensurados. Es decir, que los sensores estén midiendo el mismo fenómeno físico, como por ejemplo dos sensores de imagen o dos sensores



ultrasonicos. La idea consiste en promediar los datos para obtener información más precisa. Ejemplo de este método es el filtro de Kalman.

Características. A este nivel la fusión consiste en combinar varias características, tales como esquinas y líneas, en un solo mapa de características que pueda ser utilizado en procedimientos de más alto nivel. Las características combinadas pueden provenir de varias fuentes de datos o de la misma fuente. Ejemplo de este método son los algoritmos de agrupamiento.

Decisión. La fusión a nivel de decisión incluye la fusión de información sensorial después de que cada sensor ha hecho la determinación preliminar de la localización, atributos e identificación de cada instancia. Ejemplos de métodos de fusión a nivel de decisión incluyen el método del voto, métodos estadísticos, métodos basados en lógica difusa, etc.

Una de las áreas de aplicación más importantes dentro de la fusión e integración multisensorial es la robótica móvil. La fusión e integración de datos de sensores de visión, táctiles, térmicos, ultrasonicos, láser e infrarrojos; desempeña un papel muy importante para los sistemas robóticos, sobre todo cuando operan en ambientes dinámicos o en ambientes no del todo conocidos. En estos casos, la fusión e integración multisensorial le permite al robot obtener una rápida percepción del ambiente para propósitos de navegación o evitar obstáculos (ver figura 16).

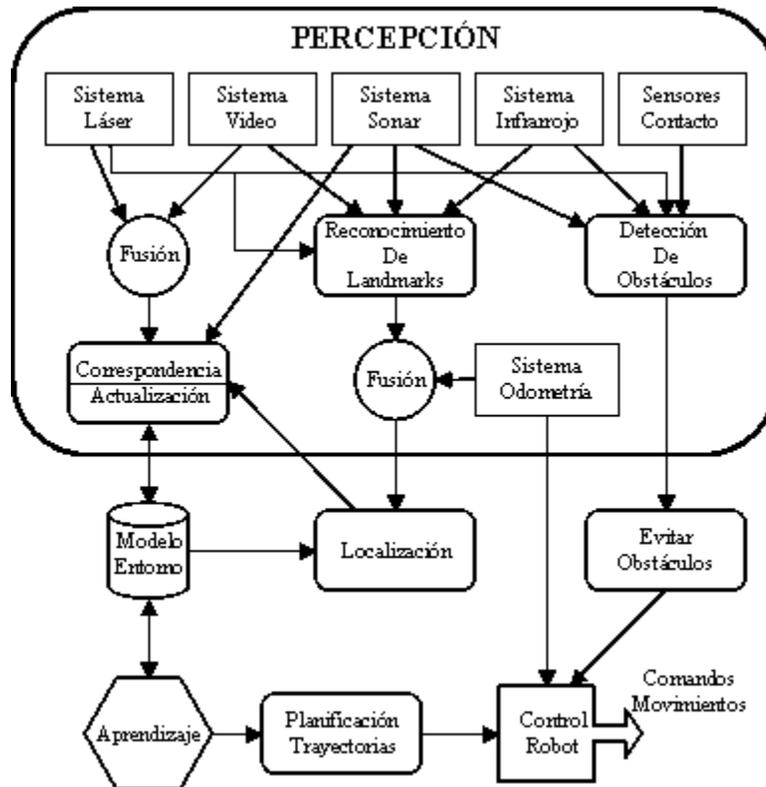


Figura 16 Arquitectura típica del sistema de percepción y fusión multisensorial de un robot móvil

A lo largo de esta tesis se usa más que nada la fusión a nivel de señal, razón por la cual a continuación se introduce algunas de las estrategias de fusión utilizadas a este nivel.



1.2.1. Fusión Bayesiana

El teorema de Bayes determina la probabilidad de una causa sabiendo el efecto que ha producido [8]. Sea un espacio muestra E , compuesto de un determinado número de sucesos disjuntos A_i , de tal manera que:

$$E = \bigcup_{i=1}^n A_i \quad (1.3)$$

La probabilidad de que ocurra un suceso $B \mid B \subseteq E$, se puede escribir como:

$$P(B) = P\left(\bigcup_{i=1}^n B \cap A_i\right) = \sum_{i=1}^n P(B \cap A_i) \quad (1.4)$$

Aplicando la ecuación de reducción del espacio muestra se obtiene la siguiente ecuación, que es el enunciado del teorema de la partición.

$$P(B) = \sum_{i=1}^n P(A_i)P(B|A_i) \quad (1.5)$$

$P(B)$ probabilidad *a posteriori* de que ocurra el suceso B

$P(A_i)$ probabilidad *a posteriori* de que ocurra la causa A_i

$P(B|A_i)$ probabilidad condicional de que ocurra el suceso B , dado que existe la causa A_i

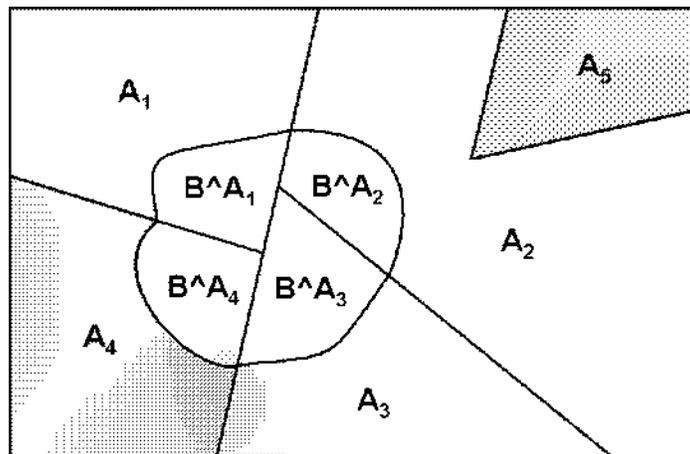


Figura 17 Representación de un espacio muestra

La fusión de datos consiste en combinar varias piezas de información de un mismo fenómeno, pero emanadas de diferentes fuentes, con la finalidad de tomar la mejor decisión respecto a este mismo fenómeno. La información extraída de cada sensor es representada mediante números reales como el grado de confianza en cierto evento, tomando así en cuenta la imprecisión, la incertidumbre y la naturaleza incompleta de la información [9]. En el caso de la fusión Bayesiana, el grado de confianza está representado por probabilidades (*a priori*,



condicional y *a posteriori*). Las decisiones son usualmente tomadas a partir de una probabilidad *a posteriori*.

Sea A el evento a ser calculado, y x_1, x_2 la información proveniente de dos sensores distintos. En este caso cada sensor asigna una probabilidad al evento A dada cierta medida x (por ejemplo, ver figura 19).

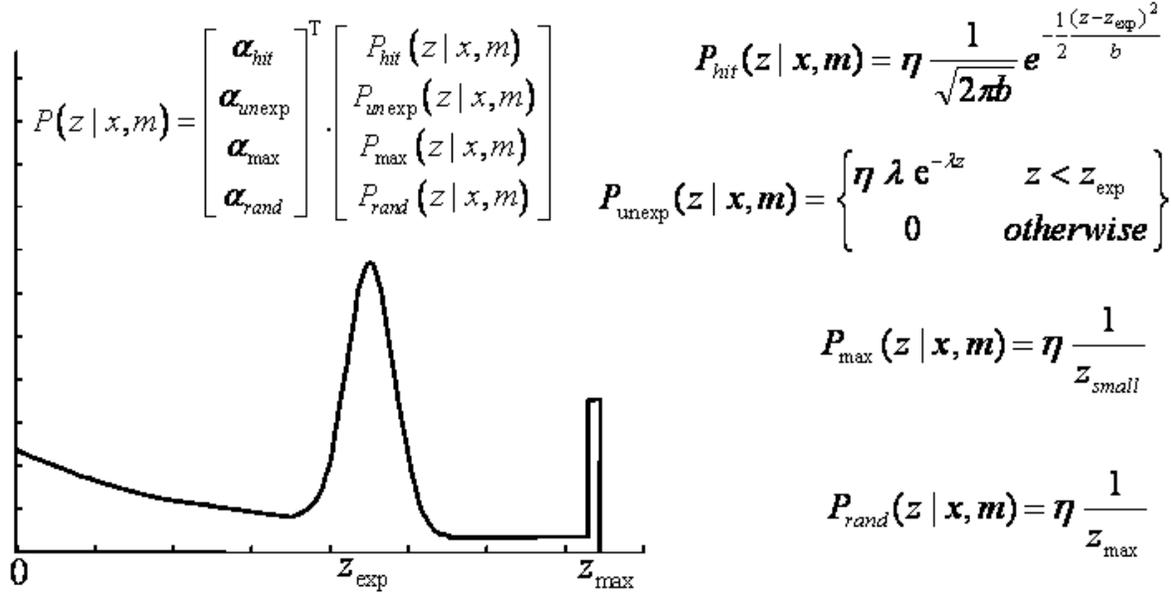


Figura 18 Ejemplo de modelo inverso del sensor representado mediante una fusión de densidad e probabilidad. Los factores α ponderan el grado de confianza de la zona de medición [10].

Así, dada una cierta medida x_1 , la misma es convertida a un valor de probabilidad mediante el modelo inverso del sensor $P(x_1|A)$. Este modelo se relaciona con las propiedades del sensor mediante la regla de Bayes según:

$$P(A|x_1) = \frac{P(x_1|A)P(A)}{P(x_1)} \quad (1.6)$$

De la misma forma, el modelo inverso del sensor 2 estaría relacionada con el evento A a través de la ecuación:

$$P(A|x_2) = \frac{P(x_2|A)P(A)}{P(x_2)} \quad (1.7)$$

La probabilidad de observación del evento A dado las mediciones x_1 y x_2 , se puede derivar de la regla del producto:

$$P(A|x_1, x_2) = \frac{P(x_1|A, x_2)P(A|x_2)}{P(x_1|x_2)} \quad (1.8)$$



Asumiendo la independencia de las mediciones x_1 y x_2 , entonces $P(x_1|A, x_2)$ y $P(x_1|x_2)$ corresponden a $P(x_1|A)$ y $P(x_1)$ respectivamente, transformando la probabilidad conjunta anterior en:

$$P(A|x_1, x_2) = \frac{P(x_1|A)}{P(x_1)} P(A|x_2) \quad (1.9)$$

Aplicando nuevamente la regla de Bayes a la probabilidad condicional $P(x_1|A)$, se obtiene la densidad de probabilidad conjunta:

$$P(A|x_1, x_2) = \frac{P(A|x_1)P(A|x_2)}{P(A)} \quad (1.10)$$

La función anterior representa los estimados y los valores de certidumbre del evento A a partir de la fusión de las medidas x_1 y x_2 de dos sensores diferentes.

1.2.2. Fusión basada en la teoría de Dempster-Shafer

El razonamiento evidencial es un método de describir la creencia de eventos inciertos soportándose en la existencia de ciertas evidencias [11]. En este enfoque, la mejor representación de la probabilidad usa una función de confianza (*Belief*), en lugar de la distribución de la probabilidad usada en los métodos Bayesianos. Aquí se le asigna valores de probabilidad a conjuntos de posibilidades en lugar de asignársela a un simple evento. Así, uno de sus atractivos reside en el hecho de que las funciones de confianza codifican la evidencia en favor de las proposiciones, de una manera inherentemente natural.

La teoría de *Dempster-Shafer* (D-S) envuelve la idea común de asignar valores entre 0 y 1 para indicar el grado de certeza de algunas hipótesis, pero en lugar de enfocar sobre como son determinados estos valores, ella se interesa por los grados de confianza de las mismas.

La teoría D-S de la evidencia está caracterizada por el marco de discernimiento (FOD: *Frame of Discernment*), la función de confianza (Bel: *Belief*), la función de verosimilitud (Pls: *plausibility*) y por la combinatoria de *Dempster*.

El marco de discernimiento, denotado como Θ , está compuesto por un conjunto finito de etiquetas que representan exhaustivamente los eventos y sus posibles combinaciones. Por ejemplo:

$$\Theta = \{A_1, A_2, A_3, (A_1, A_2), (A_1, A_3), (A_2, A_3), (A_1, A_2, A_3),\} \quad (1.11)$$

La asignación básica de probabilidad es la función $m: \Psi \rightarrow [0,1]$, donde Ψ es el conjunto de todos los subconjuntos de Θ , o también conocido como el conjunto potencia de Θ , $\Psi = 2^\Theta$. La función m puede ser interpretada como una distribución de probabilidad de todos los miembros de Ψ , donde se deben satisfacer los siguientes criterios:

$$\sum_{\forall A \subseteq \Psi} m(A) = 1 \quad (1.12)$$



$$m(\Phi) = 0 \quad (1.13)$$

De esta manera, a la etiqueta A le es asignada una probabilidad básica $m(A)$ que describe el grado de confianza que se le asigna exactamente a A . Sin embargo, la evidencia o confianza total se le atribuye a A es la suma de todas las probabilidades asignadas a A y sus subconjuntos.

1.3. Representaciones del entorno

Existen varias formas de representar la manera como el robot percibe el entorno. El caso más general es el que clasifica los tipos de mapas como:

Topológicos. Representan el entorno como un gráfico no dirigido, en donde los nodos corresponden a lugares de marcada importancia y las interconexiones o ramas representan la forma como se puede ir de un nodo a otro. El concepto de Mapa topológico fue introducido por Benjamin Kuipers a finales de los 70. Un mapa topológico es una descripción concisa de la estructura de los grandes espacios, ya que describe en forma compacta una colección de lugares o adyacencias interrelacionadas por medio de los trayectos que las enlazan o unen entre sí.

Métricos. Son representaciones de las propiedades métricas o de las coordenadas de los objetos del entorno (áreas, distancias, tamaño, localización, orientación, etc.). Este tipo de representación suele realizarse en el mismo sistema de coordenadas 2D en el cual se representa al robot, lo que facilita la fusión de los datos propioceptivos y exteroceptivos que se obtienen mediante los sensores del robot. Este tipo de representación tiene dos casos particulares que son los mapas Geométricos y los mapas de Ocupación.

Los mapas geométricos representan el entorno mediante un conjunto de características geométricas tales como puntos, líneas y esquinas, los cuales pueden ser parametrizados en función de su longitud, diámetro, color, etc.[12]

Los mapas de ocupación representan el entorno mediante celdas cuadrículadas. Cada celda representa una pequeña área o volumen del entorno donde opera el robot. Normalmente se les asigna a estas celdas un valor que expresa la probabilidad de que la parte del entorno que ella representa esté o no ocupada

1.4 Técnicas de localización

Las técnicas de localización son los medios matemáticos y geométricos que relacionan las cualidades físicas de los móviles a localizar y sus movimientos característicos con los parámetros monitoreados por los sensores para obtener un dato que funcione como medida de posición, tal medida puede ser en relación a una referencia o absoluta sobre un plano virtual.

1.4.1. Localización relativa mediante medidas propioceptivas

La localización relativa, en general es aquella que está basada en la sola observación del robot y sus sensores de abordaje; es decir, aquella donde no se usa información externa al robot. Una forma simplista de estimar la localización relativa de un robot consiste en monitorear las variables de estado del robot (i.e. $v(t)$ y $w(t)$), ver codificadores rotativos.



Este tipo de localización se conoce como *dead reckoning*, lo que originalmente es el proceso de estimar la posición de un avión o de una embarcación, basándose en la velocidad y dirección del vehículo, y en el tiempo transcurrido desde la última posición conocida hasta la actual. De aquí que este tipo de medidas de posición sólo da información referida al punto desde donde se inició la navegación del robot. Un inconveniente que surge de inmediato es que el error en el estimado de posición se incrementa con el tiempo ya que el mismo está basado en la estimación anterior.

Una de las técnicas más usadas para llevar a cabo la localización relativa de un robot se basa en la integración a través del tiempo de la información proveniente de sensores de odometría acoplados al cuerpo o a las ruedas del robot. Esta técnica, conocida como localización odométrica, está sujeta a grandes fuentes de error como por ejemplo el derrape y patinaje de las ruedas de odometría. Otra desventaja de la odometría es su sensibilidad al tiempo de terreno, ya que estos sistemas no son capaces de detectar si se trata de una superficie plana o de una con muchas irregularidades [13]. A pesar de estas desventajas, la localización odométrica sigue siendo usada porque proporciona buena exactitud, a bajos costos, para trayectos cortos.

1.4.1.1. Odometría

Un instrumento odométrico muy común es el codificador óptico rotativo, el cual se acopla directamente al eje del motor o al de las ruedas de tracción. El principio básico del codificador óptico está centrado en un haz de luz que alcanza una foto detectora. Este haz de luz es interceptado por áreas opacas y transparentes que se alternan en un disco rotativo que está sólidamente unido al eje de interés.

Existen dos tipos básicos de codificadores rotativos. Los incrementales y los absolutos. La versión incremental mide la velocidad de rotación y puede inferir la posición relativa. Por su parte los modelos absolutos miden directamente la posición angular y pueden inferir la velocidad.

El proceso más básico para la localización de un robot móvil se basa en el modelo cinemático del sistema de propulsión. Este sistema es el que permite al robot moverse alrededor de un determinado entorno. El sistema más usual se basa en el uso de simples ruedas de tracción, ya que el mismo es adecuado para la navegación en los entornos de desarrollo típico de las actividades del ser humano, y además es un sistema poco complejo. De las distintas configuraciones basadas en ruedas de tracción, la más popular ha sido la configuración diferencial. La posición de un robot de este tipo puede ser estimada a partir de las ecuaciones geométricas que surgen de la relación entre los componentes del sistema de propulsión y de la información de los codificadores rotativos que usualmente llevan acoplados a sus ruedas.

Desventaja. Por pequeña que sea la constante de error en el sensor, el error de localización del robot crecerá sin límites porque el mismo se va integrando a través del tiempo.

Ventajas: Además de producir resultados aceptables para la localización de trayectos cortos, también ayudando a reducir los costos de otros sistemas de localización ya que hacen que la actualización de la posición absoluta del robot sea menos frecuente. Los resultados de la



localización odométrico pueden ser fusionados con mediciones de posición absoluta para producir así mejores resultados en la localización global del vehículo.

1.4.2. Localización relativa mediante medidas exteroceptivos

Se basan completamente en el conocimiento que un robot puede percibir del entorno que lo rodea, siendo estos métodos mayormente apoyados en sensores de distancia.

1.4.2.1. Localización absoluta basada en hitos o landmark

Los hitos o landmark son elementos del entorno que poseen características distintas, especiales que, mediante los sensores, el robot puede detectar. Una vez que estas Landmarks son detectadas, se contrastan con la información a priori que se tiene del entorno (correspondencia o matching), o también se pueden aplicar técnicas de triangulación, para determinar la posición del robot. Así, este método de localización puede dividirse en 4 fases según se ve en la figura 19.

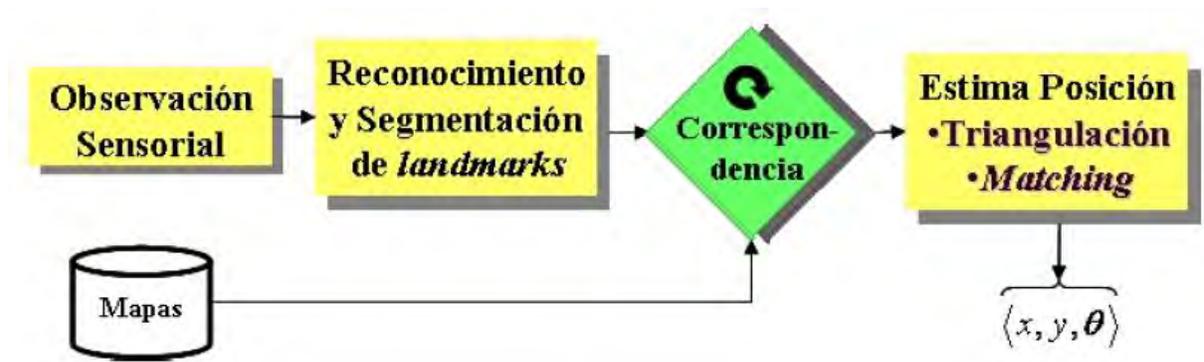


Figura 19 Fases de localización de un robot basada en Landmarks

La clasificación más usual de estas Landmarks es la que las divide en naturales y artificiales. Las Landmarks naturales son aquellas que de antemano forman parte del entorno donde se mueve el robot. Ejemplo de estas en ambientes interiores son las puertas, ventanas y lámparas de techos; mientras que para ambientes exteriores se pueden mencionar a los árboles, caminos y las señales de tráfico. El problema principal con este tipo de Landmarks es que son difíciles de reconocer y en consecuencia el robot generalmente necesitará un número mayor de observaciones para poder determinar inequívocamente su posición.

Las Landmarks artificiales son colocadas intencionalmente en el entorno donde se mueve el robot, de forma tal que sean bien visibles a los sensores del mismo[14]. Este tipo de Landmarks se puede clasificar en activas o pasivas. Las Landmarks activas, también conocidas como faros, son aquellas que emiten algún tipo de señal que informa sobre su localización. Ejemplo de ellas son los satélites de los sistemas GPS, los faros ultrasónicos, los radiofaros y dipolos magnéticos, etc. Las principales desventajas de este tipo de hitos activos, es que la señal que emiten puede verse perturbada por las condiciones geográficas o también por las condiciones atmosféricas del entorno. Otra desventaja es que en la práctica estas Landmarks no pueden enviar señal en forma omnidireccional y por lo tanto el robot no las puede ver desde cualquier lugar. Otra importante desventaja es que su costo de

construcción y mantenimiento puede ser excesivamente elevado. Por su parte, las Landmarks pasivas son las que no emiten activamente ningún tipo de señal, y en consecuencia el robot tiene que buscarlas activamente mediante sus sensores para poder ejecutar el proceso posterior de autolocalización. Ejemplo de este tipo de Landmarks son las balizas, las figuras geométricas coloreadas, y los códigos de barra. La principal desventaja de estas Landmarks es que mientras más alejado se encuentre el robot de ellas, menos ajustado y preciso será la estimación de estado. También, al compararlas con Landmarks activas, se pone de manifiesto que las pasivas son más difíciles de detectar y que requieren de mayor cantidad de proceso para poder identificarlas.

Los métodos comúnmente usados para determinar la posición de un móvil a partir de la detección de Landmarks (activas y pasivas) y del conocimiento previo de su localización (x_i, y_i) son dos tipos: Trilateración, el cual se basa en las distancias desde el móvil a cada una de las tres o más Landmarks, y Triangulación, el cual se basa en los ángulos de vista a tras o más Landmarks[15].

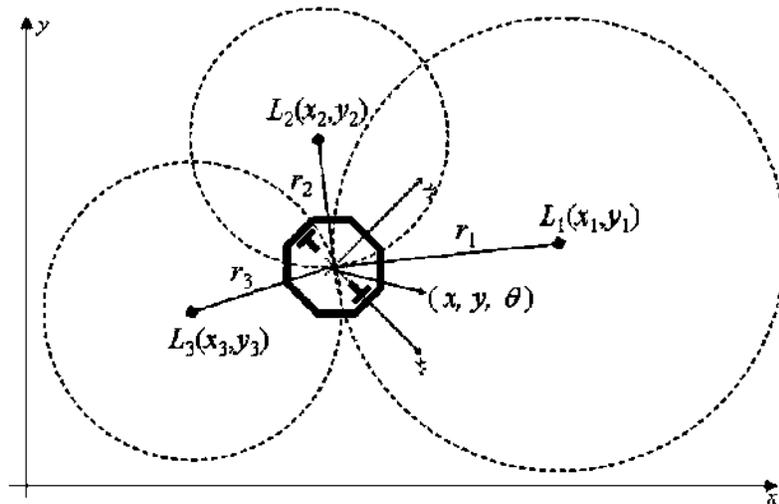


Figura 20 Esquema de localización por trilateración. Este método se basa en la medición de las distancias (r_1, r_2, r_3) del móvil a las Landmarks

La trilateración es la determinación de localización (x, y) de un móvil basado en la medición de las distancias (r_1, r_2, r_3) a 3 o más Landmarks cuyas coordenadas $\{(x_1, y_1), (x_2, y_2), (x_3, y_3)\}$ son conocidas [16] (ver figura 20). En el plano 2D, la trilateración se puede definir como el problema de encontrar la intersección de tres circunferencias. Esto es encontrar la solución al sistema de ecuaciones cuadráticas expresadas en la ecuación:

$$(x - x_1)^2 + (y - y_1)^2 = r_1 \quad (1.13)$$

$$(x - x_2)^2 + (y - y_2)^2 = r_2 \quad (1.14)$$

$$(x - x_3)^2 + (y - y_3)^2 = r_3 \quad (1.15)$$



La implementación práctica de la trilateración, usualmente consiste en instalar 3 o más transistores (faros) en lugares conocidos del entorno, e instalar un único receptor a bordo del móvil. A pesar de la simplicidad y buena precisión que puede ofrecer el método de la triangulación, en robótica móvil este esquema de localización es insuficiente, debido a que el mismo solo determina las coordenadas cartesianas (x, y) del robot; y como es sabido la orientación del robot respecto al sistema de coordenadas global es de muy alta importancia en las tareas de predicción de la posición durante la navegación.

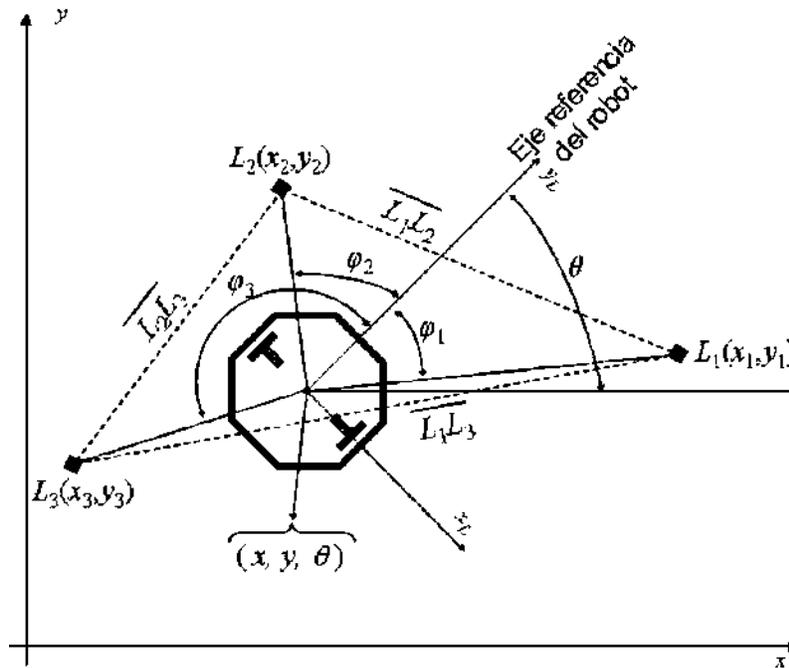


Figura 21 Esquema de localización por triangulación. Este método se basa en la medición de los ángulos de vista $(\theta_1, \theta_2, \theta_3)$ del móvil a las *Landmarks*.

Por su parte, la triangulación es el proceso de determinar la posición del robot (localización + orientación: (x, y, θ)) a partir de tres *Landmarks* (L_1, L_2, L_3) cuyas respectivas localizaciones en el espacio cartesiano (x_i, y_i) son conocidas. En general el proceso requiere que el robot tenga capacidad sensorial para detectar estas *Landmarks* y par a medir la orientación o Angulo de vista hasta cada una de ellas $(\varphi_1, \varphi_2, \varphi_3)$, con respecto al sistema de referencia de él mismo [17](ver figura 21). Aplicando la ley de coseno, la distancia entre dos *Landmarks* cualquiera se puede expresar como:

$$r_1^2 + r_2^2 - 2r_1r_2\cos\varphi_{12} = \overline{L_1L_2} \quad (1.16)$$

$$r_1^2 + r_3^2 - 2r_1r_3\cos\varphi_{13} = \overline{L_1L_3} \quad (1.17)$$

$$r_2^2 + r_3^2 - 2r_2r_3\cos\varphi_{23} = \overline{L_2L_3} \quad (1.18)$$



Aquí, (r_1, r_2, r_3) son distancias desconocidas desde el robot hasta las *Landmarks*, y $\{L_1L_2, L_1L_3, L_2L_3\}$ son las distancias entre *Landmarks* que se consideran conocidas. Igualmente, $\{\varphi_{12}, \varphi_{13}, \varphi_{23}\}$ surgen de sustraer los ángulos $\{\varphi_1, \varphi_2, \varphi_3\}$ entre sí, por lo que también se consideran conocidos.

Una vez obtenidos las distancias, se pasa a resolver la localización (x, y) del móvil usando el procedimiento descrito para las ecuaciones. Una vez obtenida la localización del móvil, se puede computar el Angulo del segmento que va desde el punto (x, y) hasta cualquiera de las *Landmarks* conocidas (x_i, y_i) ; y luego calcular el ángulo θ de orientación del móvil con la ecuación,

$$\theta = \angle_{(x_i, y_i)}^{(x, y)} - \varphi_i \quad (1.19)$$

1.4.2.2. Localización Basada En Mapas

Esta metodología se basa en la búsqueda de la correspondencia entre un mapa local que el robot construye mediante sensores, y un mapa global del entorno que el robot conoce con anterioridad, o que el mismo va construyendo mientras explora el entorno. De aquí el nombre general que recibe esta técnica: correspondencia entre modelos (*Model Matching*).

Partiendo de que el robot dispone de un mapa de su entorno, el cual puede ser obtenido por el mismo robot en una fase previa de exploración, o pueden ser suministrados externamente en una fase de inicialización (mapa tipo CAD), o pueden ser construidos simultáneamente mientras el robot navega (SLAM); el procedimiento general que se sigue con esta técnica de localización se divide en los siguientes pasos:

- Predicción de la posición basa en el estimado previo y en los datos de Odometría recogidos mientras el robot esta en movimiento.
- Predicción de observaciones basada en el estado estimado de la posición y en el mapa del entorno el cual es conocido con anticipación.
- Observación del entorno mediante los sensores del robot.
- Búsqueda de correspondencia entre las observaciones y el mapa.
- Actualización de la posición real del robot.

Los distintos métodos de localización basados en mapas se diferencian en la manera de caracterizar el mapa del entorno, y en la forma como se maneja la incertidumbre en la posición que el robot ocupa dentro del mismo.

1.4.2.3. Localización con mapas métricos

Se refiere a la técnica que usa mapas que describen el entorno en término métricos referidos a un sistema global de coordenadas, como lo son los mapas de rejillas o los de características geométricas.



Moravec & Elfes [18] proponen técnicas para encontrar la posición de un robot realizando la correlación entre un mapa local, construido por el robot con ayuda de sus sensores, y un mapa de *grid* que representa el entorno global. Esta técnica la aplican con mapas de diferentes resoluciones a fin de reducir los costos computacionales, son embargo aún reportan que la rutina de actualización del mapa es computacionalmente pesada. Borensteirn & Koren proponen una técnica de actualización más simple donde solamente las celdas que están a lo largo del eje acústico son actualizadas. Con esta técnica se reduce significativamente la carga computacional, manteniéndose una distribución de probabilidad realista.

Schiele & Crowley [19] evalúan diferentes métodos para correlacionar el mapa de *grid* obtenido con los sensores del robot un mapa global del entorno. Los métodos bajo consideración son: a) celdas locales vs. Celdas globales, b) líneas extraídas del mapa local vs. Celdas globales, c) celdas locales vs. Líneas extraídas del mapa global, d) líneas extraídas del mapa local vs. Líneas extraídas del mapa global. El resultado fue que la mejor localización se consigue cuando se correlacionan los mapas a igual nivel de abstracción. Es decir, celdas vs. Celdas, o lineal vs. Líneas.

Los mapas de características están formados por los elementos que caracterizan un determinado ambiente como por ejemplo líneas, esquinas y columnas; los cuales puede ser parametrizados en función de su longitud, diámetro, color, etc. Para la localización se usan algoritmos de correspondencia (*matching*) entre los elementos característicos del intonso que va observando el robot con sus sensores, y un mapa del entorno que se conoce con anterioridad[12]; o un mapa que se construye simultáneamente mientras el robot explora el ambiente [20]. Una tendería actual es la localización basada en mapas que incluyen *Landmarks* naturales como paredes y esquinas[21]. Existen muchas técnicas para llevar a cabo este tipo de localización, pero usualmente todas necesitan de métodos de extracción de mucha precisión o de sensores que suministren datos de alta densidad[22]. De aquí que, en la actualidad, las características extraídas a partir de los datos de sensores laser las más usadas[23]. Por otra parte, la localización basa en características geométricas extraídas con sensores de ultrasonido es menos común. Esto se debe a la falta de precisión en la información angular proporcionada por el sonar y a la dificultad que existe para clasificar con él las distintas características del entorno.

1.4.2.4. Localización con mapas topológicos

Se refiere a la técnica que usa mapas topológicos, con los cuales se sigue una estrategia de localización no-métrica, ya que estos mapas indican la conectividad de los lugares, pero no necesariamente las relaciones métricas o cuantitativas entre ellos. Para que un robot se auto localice en un mapa topológico basta con que identifique en cual nodo se encuentra, por lo que el problema principal en la localización topológica lo constituye el reconociendo fiable de los distintos nodos. Otra forma de resolver el problema de localización es basándose en el reconocimiento no de uno, sino de una secuencia de nodos. Estos sistemas, aunque son efectivos, requieren que el robot haga varios movimientos con la sola finalidad de determinar su posición en el entorno. Un caso particular de localización usando mapas topológicos y sensores de bajo costo es el propuesto por Duckett [24]. Estos investigadores presentan un método unificado, basado en un mapa topológico aumentando con características métricas (topológico-métrico), para la localización global y el seguimiento de la posición de un robot



móvil. El método combina técnicas de *Scan-Matching* sobre mapas de densidad de ocupación locales extraídos con un sonar, con técnicas de localización multihipotesis aplicadas sobre una representación tipológica del mapa global. De esta manera el método es eficiente por su baja carga computacional en la operación de navegación por el entorno global, y es preciso al momento de corregir su posición en lugares particulares.

1.4.2.5. Localización usando mapas de alta densidad de muestreo

Dentro de este tipo de técnicas, el de análisis de correspondencia o *Scan-Matching* es uno de los más usados. Aquí, el robot necesita un mapa del entorno, que puede ser suministrado previamente o puede ser una lista de exploraciones o barridos previamente ejecutados desde posiciones conocidas. En general, el estado del robot sigue una hipótesis única que se va corrigiendo a medida que se encuentran datos validos (*Pose tracking*). Para determinar la posición real de robot el sistema de localización analiza dos mapas para calcular la distancia y la rotación que existe entre ellos. Esto es, el mapa adquirido en la posición actual del robot es comparado con el mapa conocido que este más cercano a la posición actual. Si el proceso de correspondencia se lleva a cabo con éxito, el resultado es la diferencia real entre la posición desde donde fue tomado el mapa actual y la posición del mapa de referencia, lo cual es entonces usado para actualizar el estimado en la posición del robot.

Los algoritmos de análisis de correspondencia más usados son los de correlación (*CCA-Cross Correlation Algorithm*)[25] y el de correspondencia dual iterativa (*IDC-Iterative dual correspondence*)[26]. Para obtener buenos resultados a partir del algoritmo de *Scan-Matching* es fundamental que el algoritmo sea robusto frente a los cambios del entorno, porque si no, el algoritmo confiara en que se trata de mediaciones ruidosas en los sensores y como resultado entregara malos resultados. También es importante que el algoritmo tenga la capacidad para juzgar sus propios resultados, es decir, que pueda indicar al robot, mientras que los resultados defectuosos serán descartados y no habrá ningún cambio en la estimación de la posición del mismo. Tamien es deseable que el algoritmo sea rápido y a la vez preciso en el sentido que pueda realizar más análisis por cada metro recorrido y que mantenga el error en la estimación de la posición dentro del error estándar del sensor utilizado.

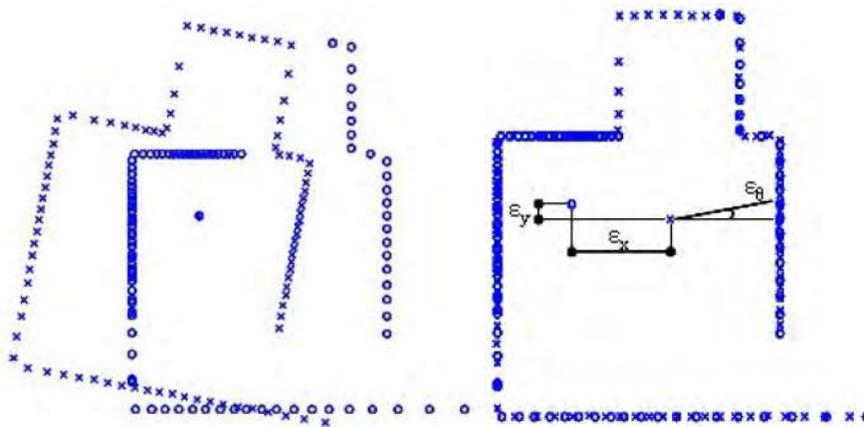


Figura 22 Dos barridos láser tomados desde "o" y desde "x"; a la izquierda los mismos barridos mostrando los resultados del análisis de correspondencia[27]



1.4.2.6 Localización y mapeo: enfoque probabilístico

En los métodos clásicos utilizados en robótica, el éxito de los algoritmos depende en gran medida del uso de sensores precisos y de la obtención de modelos exactos, tanto del robot como de su entorno. Sin embargo, estas dos condiciones no son suficientes garantía de pleno éxito, ya que los errores e incertidumbres siempre estarán presentes en cualquier sistema robótico real. Dentro de los principales factores que causan incertidumbres destacan los siguientes:

- Ruido en los sensores. Las observaciones realizadas por los sensores son normalmente ruidosas y la distribución estadística de este ruido no suele ser sencilla de modelar[10].
- Ruido en la detección de la posición. Los movimientos del robot no suelen ser exactos, ni tampoco detectados de forma precisa mediante odometría. Los errores de Odometría son además acumulativos. Pequeños errores en la rotación del robot pueden tener efectos importantes en la estimación de los movimientos de translación y en la determinación de su posición final.[13]
- Entorno complejos y dinámicos. Los entornos interiores en los que se desplazan el robot suelen ser complejos y dinámicos, haciendo casi imposible mantener modelos consistentes de los mismos.

Ante esta problemática, los métodos de enfoques probabilísticos han demostrado que ofrecen resultados más robustos que los métodos clásicos. Esto último se le atribuye al hecho de que los métodos probabilísticos se apoyan en modelos que representan la información a través de funciones de probabilidad, lo que los hace más robustos de cara a las limitaciones del sensor y al ruido mismo en la cinemática del robot y el modelo del entorno. Por otra parte, las limitaciones más citadas de los algoritmos probabilísticos son dos: la ineficiencia computacional al tener que considerar todas las distribuciones de probabilidad del espacio de posiciones del robot, y la inherente necesidad de tener que aproximar discretamente la relación continua del contexto del robot.

El principio básico subyacente en todo algoritmo probabilístico es la regla de Bayes. Si A es una variable que queremos inferir a partir de otra variable B , la probabilidad $P(A)$ se conoce como la distribución de probabilidad a priori y la variable B es conocida como los datos. La distribución $P(A/B)$ es conocida como la distribución de probabilidad a posteriori. La regla de Bayes provee una manera conveniente de computar la probabilidad a posteriori en función de la probabilidad inversa $P(B/A)$ y la probabilidad a priori $P(A)$, de acuerdo a

$$P\left(\frac{A}{B}\right) = \eta P\left(\frac{B}{A}\right) P(A) \quad (1.20)$$



Donde η es un factor normalizador que asegura que la expresión se encuentra en el rango $[0,1]$.

La robótica probabilista busca estimar la posición de un robot $s = (x, y, \theta)$ y el modelo del entorno que lo rodea (m). En adelante, la composición de la posición del robot y del mapa, forma y se conoce como el estado $x = (s, m)$. La evolución del estado, el control y la sensorización de un robot móvil puede ser modelado mediante una red dinámica de Bayes. De acuerdo a la ilustración 14, el estado x_k es estocásticamente dependiente del estado anterior x_{k-1} y del control u_k , mientras que la medición z_k depende estocásticamente del estado x_k .

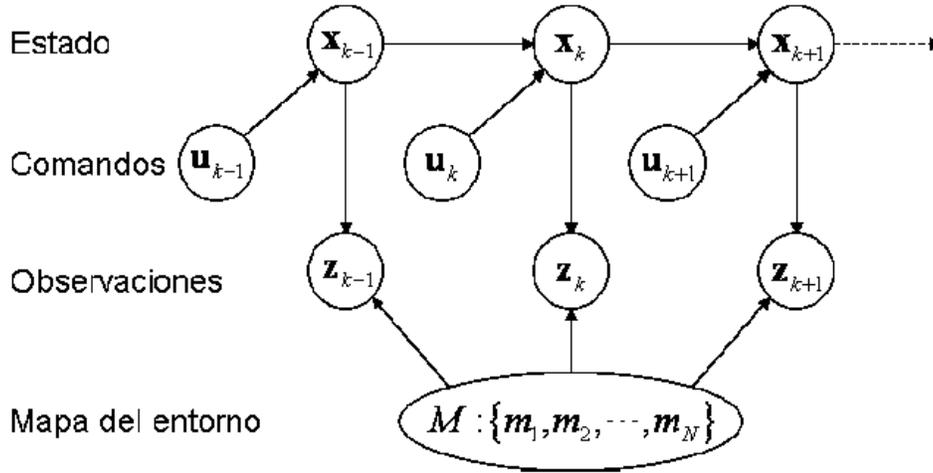


Figura 23 Red dinámica de Bayes que caracteriza la evolución del estado x , la medición z , y el control u en un robot móvil. El subíndice k indica la dependencia temporal del modelo

El filtro de Bayes extiende la regla de Bayes de la ecuación $P(A/B) = \eta P(B/A) P(A)$ para tratar problemas de estimación en el dominio del tiempo. Este filtro es un estimado recursivo que permite computar la secuencia de una distribución de probabilidad posterior que no puede ser observada directamente. Así, con el filtro de Bates se puede calcular la probabilidad posterior sobre el estado x_k por medio de la siguiente ecuación.

$$p(s_k, m_k | z_k, u_k) = \eta(z_k | s_k, m_k) \int \int p(s_k, m_k | u_k, s_{k-1}, m_{k-1}) p(s_{k-1}, m_{k-1} | z_{k-1}, u_{k-1}) ds_{k-1} dm_{k-1} \quad (1.21)$$

Asumiendo que el mapa del entorno es estático y que el movimiento del robot es independiente del mapa, la ecuación se transforma en:

$$p(s_k, m_k | z_k, u_k) = \eta(z_k | s_k, m) \int p(s_k | u_k, s_{k-1}) p(s_{k-1}, m | z_{k-1}, u_{k-1}) ds_{k-1} \quad (1.22)$$

Es la ecuación, la distribución $p(s_k, m | z_k, u_k)$ se conoce como la confianza o certidumbre del estado x_k . La misma se denota como $Bel(x_k)$ y representa la probabilidad de que el sistema se encuentre en el estado x_k , dado las mediciones $z_{1:k}$ y comandos de control $u_{1:k}$ ejecutados hasta el estado k . en la misma ecuación, el segundo factor dentro de la integral representa la confianza del estado anterior $Bel(x_{k-1})$, por lo que esta condición le da un



carácter recursivo al filtro de Bayes. Una forma más compacta de expresar la ecuación, utilizando la forma recursiva del filtro de Bayes es:

$$Bel(x_k) = \eta p(z_k | x_k) \int p(x_k | x_{k-1}, u_k) Bel(x_{k-1}) dx_{k-1} \quad (1.23)$$

Para implementar o poner en funcionamiento este estimador, se deben especificar dos densidades generativas: la densidad $p(z_k | x_k)$ conocida como el modelo perceptual, y la densidad $p(x_k | x_{k-1}, u_k)$ conocido como el modelo de transición. La densidad $p(z_k | x_k)$ describe en termino probabilísticos la manera como se generan las distintas observaciones z para diferentes localización y mapas en del entorno. La densidad $p(x_k | x_{k-1}, u_k)$ describe cómo afecta el control u a la evolución de la posición del robot. Así, esta densidad describe la probabilidad de que estando el robot en el estado x_{k-1} , alcance el estado x_k si le es aplicado el control u_k . Por eso este modelo también se conoce como el modelo de actuación.



Capítulo 2. Movimiento característico, cinemática y localización odométrica



2.1. Robótica a grandes rasgos

La robótica es la actividad intelectual y práctica que abarca el estudio sistemático de la aplicación del conocimiento científico asociado a la concepción, producción, teoría y aplicación de los robots.

La palabra robot proviene de la obra *Rossum's Universal Robots* de Karel Capek, la cual se deriva de “robota” que significa trabajo duro en el idioma checo y lenguas eslavas.

Un robot es una máquina compuesta de sensores, con una unidad de procesamiento y actuadores. Un robot capturaré información de su ambiente, usará esta información para procesarla y realizar un trabajo. El instituto de robótica de América (RIA por sus siglas en inglés) define a un robot como “Un manipulador multifuncional reprogramable, diseñado para mover materiales, partes, herramientas o dispositivos, por medio de varios movimientos programados con el propósito de completar diferentes tareas.

Los robots surgieron desde la antigüedad como entes mecánicos que pretendían imitar los movimientos humanos. Posteriormente los robots fueron diseñados para sustituir al hombre en la industria en tareas que son repetitivas. Al principio los robots realizaban tareas repetitivas invariables porque su reprogramación representaba un alto costo dado que solo un experto lo puede hacer.

2.1.1 Clasificación general de los robots

De acuerdo a su grado de movilidad los robots se clasifican en:

- Robots manipuladores, cuya base posee un enlace mecánico con la base de referencia fija.
- Robots móviles, son los robots que no poseen un enlace mecánico con la base de referencia fija.



Figura 24 Robot manipulador fijo KR6



Figura 25 Ejemplo robot móvil



2.1.2. Robótica móvil

En años recientes el estudio de la robótica móvil se ha incrementado debido a la necesidad a la necesidad de crear máquinas capaces de moverse en su entorno y porque su invención es más reciente que los robots manipuladores [28]. Las aplicaciones de los robots móviles incluyen misiones de búsqueda, misiones de explicación, rescate de personas, automatización de la industria, vigilancia, etc.

El robot móvil se puede clasificar de acuerdo a diferentes criterios [29]:

Por el ambiente en el que se desplazan:

- Robots terrestres
- Robots aéreos
- Robots acuáticos



Figura 26 Robot terrestre Tlaloc



Figura 27 Robot aéreo AR Drone II

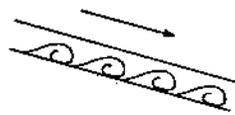
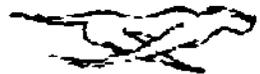
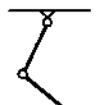


Figura 28 Robot acuático Aqualley



Por su sistema de locomoción ver tabla 2:

Tabla 2 Movimientos característicos típicos en robots móviles[30]

Type of motion	Resistance to motion	Basic kinematics of motion
Flow in a Channel 	Hydrodynamic forces	Eddies 
Crawl 	Friction forces	Longitudinal vibration 
Sliding 	Friction forces	Transverse vibration 
Running 	Loss of kinetic energy	Oscillatory movement of a multi-link pendulum 
Jumping 	Loss of kinetic energy	Oscillatory movement of a multi-link pendulum 
Walking 	Gravitational forces	Rolling of a polygon (see figure 2.2) 

Por su nivel de autonomía:

- Autónomo
- Semiautónomo

Por su aplicación:

- Robots educativos
- Robots de servicio
- Robots militares
- Etc.



En particular el robot móvil con llantas se puede clasificar como holonómicos o no holonómicos. Los robots móviles no holonómicos son los que poseen restricciones que limitan su movimiento, como por ejemplo un carro que no puede moverse lateralmente. Por otro lado, un robot holonómico es aquel que no posee restricciones de movimiento, también conocido como robot omnidireccional porque puede moverse en cualquier dirección sin ninguna restricción, además de las que el medio supone.

2.1.2.1. Elementos que componen un robot móvil

Un robot móvil está conformado por cuatro grandes sistemas [29]:

1. Sistema mecánico
2. Sistema sensorial
3. Sistema de control
4. Fuente de energía

El sistema mecánico de un robot móvil es aquel que comprende el conjunto de elementos o dispositivos cuya función es la transmisión de movimiento desde su fuente de origen a través de transformación de energía[31]. Los sensores son la parte que permite al robot obtener datos.

Por otra parte, el sistema de control es el que se encarga de gobernar los movimientos y comportamientos del robot. La capa de control de bajo nivel es la que se encarga del control de cada uno de los actuadores; la capa intermedia de la coordinación de los actuadores y la de más alto nivel se encarga del control de comportamientos complejos del robot. Es importante mencionar sistema de control depende directamente del sistema sensorial del robot.

Por último, la fuente de energía es el sistema que suministra la energía necesaria a los demás sistemas para que el robot pueda funcionar correctamente. Por lo general, el robot depende de una fuente de poder eléctrica la cual consiste en baterías y circuitos electrónicos de protección para la correcta alimentación eléctrica de los demás sistemas.

2.2. Modelo cinemático

El proceso más básico de localización de un robot móvil se basa en el modelo cinemático del sistema de propulsión. Este sistema es el que permite al robot moverse alrededor de un determinado entorno. El sistema más usual se basa en el uso de simples ruedas de tracción, ya que el mismo es adecuado para la navegación en los entornos de desarrollo típico de las actividades del ser humano, y además es un sistema poco complejo. De las distintas configuraciones basadas en ruedas de reacción, la más popular ha sido la configuración diferencial. La posición de un robot de este tipo puede ser estimada a partir de las ecuaciones geométricas que surgen de la relación entre los componentes del sistema de propulsión y de la información de los codificadores rotativos que usualmente llevan acoplados a sus ruedas. El problema de este método es que por pequeña que sea la constante de error en el sensor, el error de localización del robot crecerá sin límites porque el mismo se va integrando a través del tiempo. Varios investigadores han estudiado este problema, tratando de reducir el error



en la localización por medio de mejoras en el modelo del mismo. uno de los primeros trabajos es el de Borenstein y feng quienes analizaron las fuentes de error y propusieron un modelo de error para un robot de tipo diferencial [32]. Otra forma de atacar el problema de la acumulación del error es ir corrigiendo la posición del robot a intervalos regulares. Esto último usualmente se consigue mediante la colocación de sistemas de Landmarks necesarias o de sistemas de señalización activos [3]. Estos sistemas tienen altos costos de instalación y quizás de mantenimiento, y su costo incrementa con el número de Landmarks necesarias. Por esta razón, los sistemas de localización en trayectos cortos, también ayudan a reducir los costos de otros sistemas de localización ya que hacen que la actualización de la posición absoluta del robot sea menos frecuente. Además, los resultados de la localización odométrica pueden ser fusionados con mediciones de posición absoluta para producir así mejores resultados en la localización global de vehículo.

2.2.1. Modelo cinemático del robot diferencial

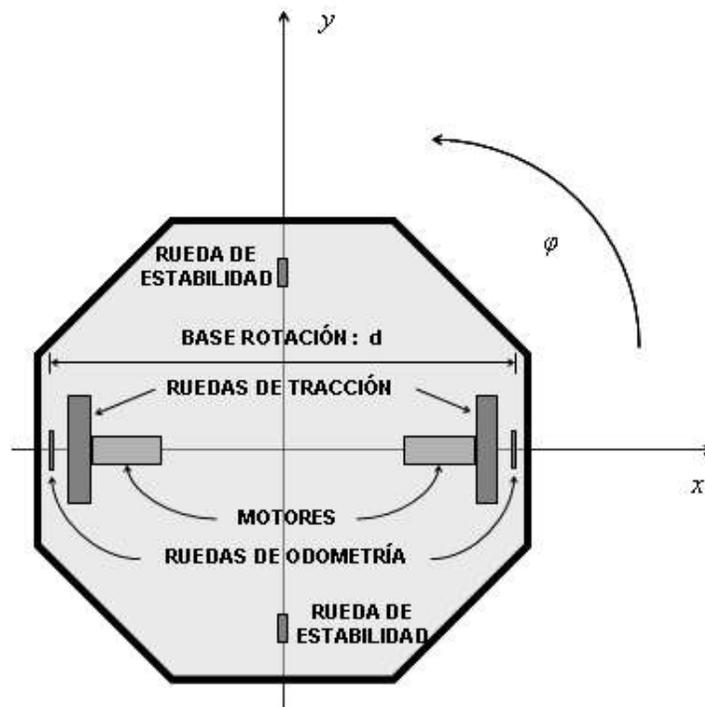


Figura 29 Ejemplo de configuración de robot de tracción diferencial

Típicamente una plataforma móvil de tracción diferencial cuenta con dos pares de ruedas: dos ruedas de tracción que tiene acoplados motores DC, dos ruedas locas o una, para brindar estabilidad.

La traslación y la rotación de este tipo de plataformas diferenciales están determinados por el movimiento independiente de cada una de las ruedas de tracción. Así, las ecuaciones cinemáticas del punto medio del eje entre las ruedas activas, referidas a un sistema de referencia global son:

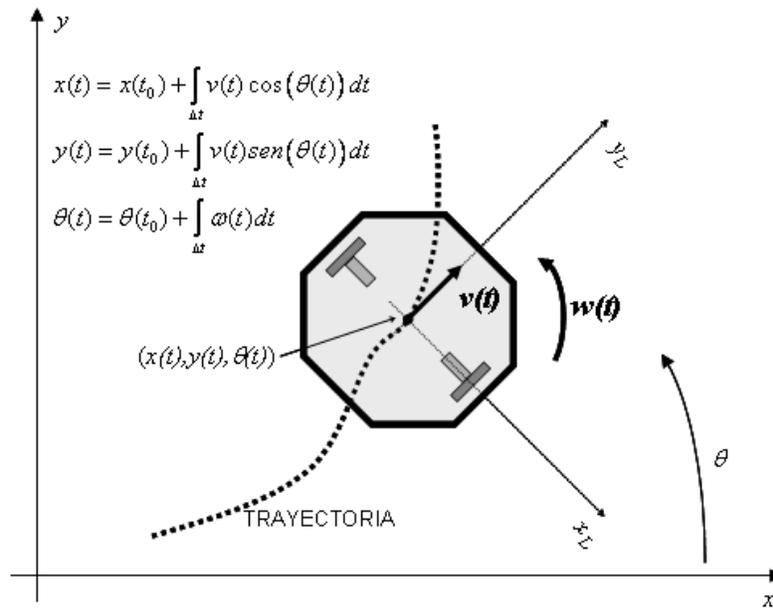


Figura 30 variables cinemáticas globales de un robot móvil diferencial

$$\dot{x} = v(t) \cos(\theta(t)) \quad (2.1)$$

$$\dot{y} = v(t) \sin(\theta(t)) \quad (2.2)$$

$$\dot{\theta} = \omega(t) \quad (2.3)$$

Donde \dot{x} , \dot{y} y $\dot{\theta}$ se refieren a la derivada respecto al tiempo de x , y y θ respectivamente. Es decir, se refiere a la velocidad lineal en la dirección de x e y , y a la velocidad angular del robot respectivamente.

La posición y orientación del móvil se obtienen integrando las velocidades del móvil en un periodo de tiempo Δt .

$$x = x_0 + \int_{\Delta t} v(t) \cos(\theta(t)) dt \quad (2.4)$$

$$y = y_0 + \int_{\Delta t} v(t) \sin(\theta(t)) dt \quad (2.5)$$

$$\theta = \theta_0 + \int_{\Delta t} \omega(t) dt \quad (2.6)$$

Si el periodo de observación tiende a ser infinitesimalmente pequeño (Δt tiende a 0), entonces las integrales anteriores pueden ser remplazadas por los desplazamientos diferenciales. Esto es:

$$\lim_{\Delta t \rightarrow 0} \left[\int_{\Delta t} v(t) \cos(\theta(t)) dt \right] \approx \Delta x \quad (2.7)$$



$$\lim_{\Delta t \rightarrow 0} \left[\int_{\Delta t} v(t) \sin(\theta(t)) dt \right] \approx \Delta y \tag{2.8}$$

$$\lim_{\Delta t \rightarrow 0} \left[\int_{\Delta t} \omega(t) dt \right] \approx \Delta \theta \tag{2.9}$$

De la ecuación 2.4 a 2.6 se pueden escribir como:

$$x = x_0 + \Delta x \tag{2.10}$$

$$y = y_0 + \Delta y \tag{2.11}$$

$$\theta = \theta_0 + \Delta \theta \tag{2.12}$$

Por lo que se mantiene una frecuencia de muestreo constante y elevada sobre la odometría de un móvil, se puede estimar la posición y orientación del mismo mediante las siguientes ecuaciones en diferencia:

$$x_k = x_{k-1} + \Delta x_k \tag{2.13}$$

$$y_k = y_{k-1} + \Delta y_k \tag{2.14}$$

$$\theta_k = \theta_{k-1} + \Delta \theta_k \tag{2.15}$$

Índices de tiempo discreto

Por otra parte, dado que se está considerando una alta frecuencia de muestreo, entonces también se puede considerar que la velocidad angular en cada una de las ruedas se mantiene constante durante el periodo de intermuestreo. Esto lleva a que el movimiento del robot pueda ser aproximado a través de pequeños desplazamientos de curvatura constante, según se muestra en la figura 32.

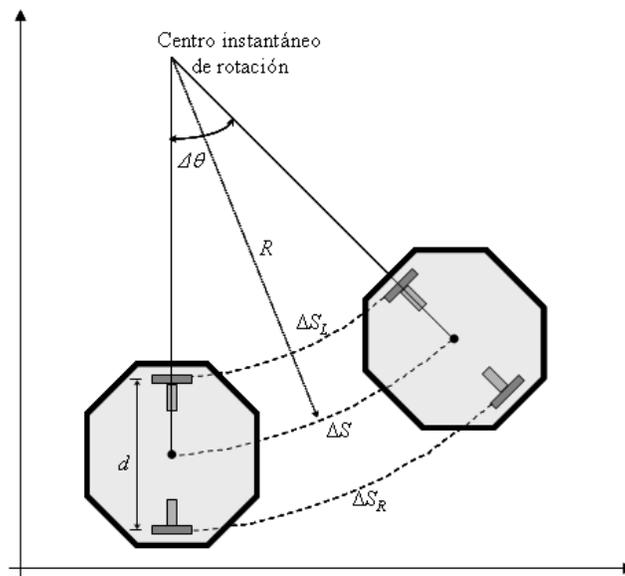


Figura 31 Desplazamientos relativo entre instantes de muestreo (derecha) y relaciones geométricas definidas por el movimiento (izquierda).



En la figura 32, ΔS_k es la distancia recorrida por el centro del robot en el intervalo $\{k-1, k\}$, mientras que $\Delta\theta_k$ es el cambio de orientación durante el mismo intervalo. Estas relaciones se pueden expresar según:

$$\Delta\theta_k = \frac{\Delta S_{k,R} - \Delta S_{k,L}}{d} \quad (2.16)$$

$$\Delta S_k = \frac{\Delta S_{k,R} + \Delta S_{k,L}}{2} \quad (2.17)$$

Donde $\Delta S_{k,R}$ y $\Delta S_{k,L}$ son los arcos recorridos en el intervalo $\{k-1, k\}$ por las ruedas derecha e izquierda respectivamente. Este arco puede ser calculado a partir de la cadena cinemática ruedas-sensor según como se muestra en la figura 33.

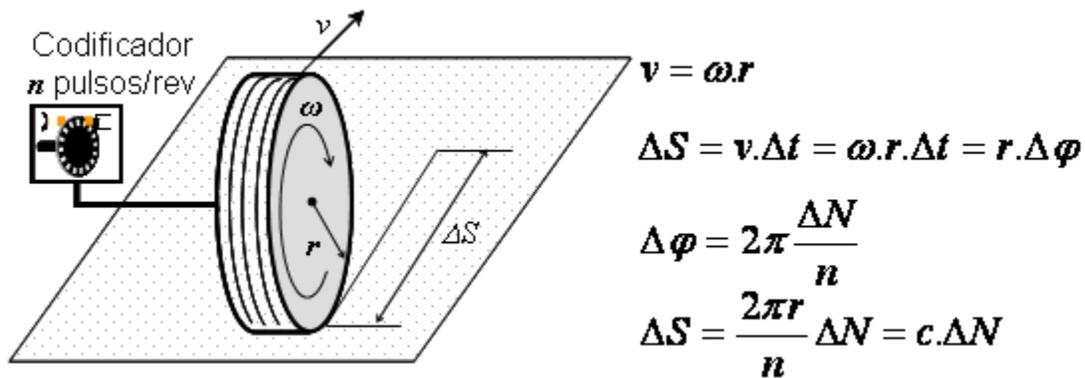


Figura 32 Esquema del sistema de medición de desplazamiento lineal mediante codificadores rotativos

En el caso de que el sensor sea un codificador rotativo incremental, el arco recorrido por la rueda se calcula según:

$$\Delta S = \frac{2\pi r}{n} \Delta N = C \cdot \Delta N \quad (2.18)$$

Donde:

R es el radio de la rueda

N resolución del codificador o número de pulsos por vuelta

C es la constante de conversión de pulsos de codificador a unidades de longitud

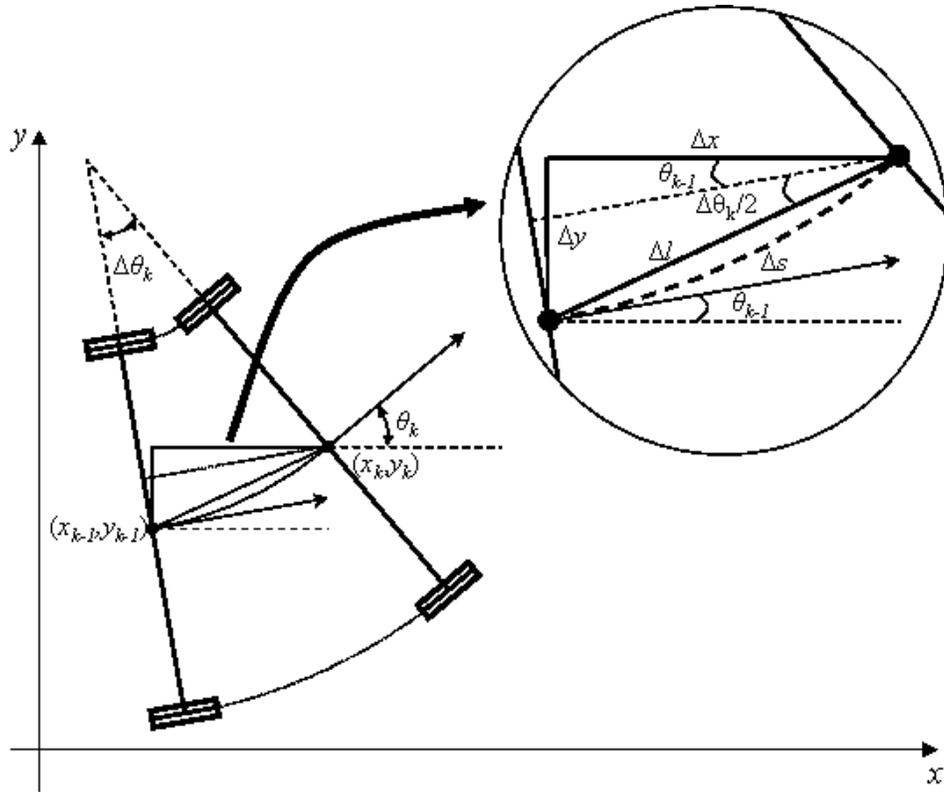


Figura 33 Relaciones de transformación trayecto circular a cartesianas

ΔN es la cantidad de pulso observados durante el periodo Δt

Δs es el desplazamiento lineal de la rueda durante el periodo Δt (mm)

considerando que los pulsos suministrados por cada sensor odométrico representa la única entrada del sistema, el vector de excitación o de comandos se puede escribir como:

$$u_k = \begin{bmatrix} u_{k,R} \\ u_{k,L} \end{bmatrix} = \begin{bmatrix} \Delta N_{k,R} \\ \Delta N_{k,L} \end{bmatrix} \quad (2.19)$$

Ahora bien, el diferencial de giro $\Delta\theta$ que sufre el robot en el intervalo $\{k-1, k\}$ es exactamente igual al calculado en la ecuación 2.16, mientras que los desplazamientos cartesianos Δx y Δy deben calcularse a partir de las relaciones mostradas en la figura 34. De donde resulta:

$$\Delta x_k = \Delta S_k \cos\left(\theta_{k-1} + \frac{\Delta\theta_k}{2}\right) \quad (2.20)$$

$$\Delta y_k = \Delta S_k \sin\left(\theta_{k-1} + \frac{\Delta\theta_k}{2}\right) \quad (2.21)$$

Considerando que en un robot diferencial las ruedas motrices suelen tener las mismas características físicas, y por lo tanto la constante de conversión C es igual para las dos, entonces las ecuaciones que modelan el movimiento de un robot diferencial en el plano se pueden reescribir según:



$$x_k = x_{k-1} + \left(\frac{C}{2}\right)(u_{k,R} + u_{k,L}) \cos\left(\theta_{k-1} + C \cdot \frac{u_{k,R} - u_{k,L}}{2d}\right) \quad (2.22)$$

$$y_k = y_{k-1} + \left(\frac{C}{2}\right)(u_{k,R} + u_{k,L}) \sin\left(\theta_{k-1} + C \cdot \frac{u_{k,R} - u_{k,L}}{2d}\right) \quad (2.23)$$

$$\theta_k = \theta_{k-1} + \left(\frac{C}{d}\right)(u_{k,R} - u_{k,L}) \quad (2.24)$$

2.2.2. Análisis de los principales errores de localización odométrica en un robot diferencial

El método de localización relativa más ampliamente utilizado es la odometría. Esta consiste en medir las revoluciones en las ruedas para estimar el desplazamiento de móvil a partir de una posición conocida. Ya se mostró en el punto 2.2.1. que el desplazamiento horizontal y los cambios de orientación de un robot diferencial no holonómico, pueden ser calculados en función de los desplazamientos diferenciales en cada una de las ruedas motrices. Este procedimiento de estimación de posición conocido como *odometric dead-reckoning*, tiene validez limitada debido a que el proceso se ve influenciado por varios factores que afectan sobre todo a la conversión de las lecturas e los codificadores en desplazamiento lineal del móvil. La clara distinción de los errores que afectan al sistema es muy importante en la corrección de errores en la odometría de un vehículo. Estas fuentes de error se dividen en forma general en dos categorías: errores sistemáticos y errores no sistemáticos [32]. Para reducir los errores de odometría es necesario distinguir los errores sistemáticos de los no sistemáticos, ya que el efecto y los procedimientos de medicación y corrección de los mismo son distintos.

2.2.2.1. Errores sistemáticos

Los errores sistemáticos son debidos a imperfecciones en el diseño y en la implementación mecánica del sistema mismo. son especialmente graves, y aquí se acumulan constantemente. Dentro de los agentes causantes de los errores sistemáticos se encuentran:

1. Distinto diámetro de las ruedas
2. Diferencia entre el diámetro teórico y el diámetro real de las ruedas
3. Diferencia entre la distancia entre ruedas teórica y la real
4. Resolución finita de los codificadores
5. Frecuencia finita en el muestreo de datos de los codificadores
6. Mala alimentación de las ruedas

Las dos fuentes principales de error sistemático son:

La incertidumbre sobre el diámetro real de las ruedas, que influyen predominantemente sobre los largos trayectos rectilíneos, aunque también es un factor que afecta directamente a la estimación de la orientación móvil.



La incertidumbre en la base de giro, la cual influye directamente en la estimación del Angulo de orientación móvil.

2.2.2.2. Errores no sistemáticos

Son aquellos que no son causados directamente por las propiedades cinemáticas del vehículo, sino por la interacción del móvil con características impredecibles del entorno por donde este se desplaza. No se producen constantemente (depende de la aplicación y del entorno) por lo que son más difíciles de medir y corregir. La principal fuente de error no sistemático en los sistemas odométricos es el deslizamiento de las ruedas, lo cual causa que las revoluciones de las ruedas no se correspondan al desplazamiento real del móvil. Los errores no sistemáticos están relacionados con:

1. Suelos irregulares. Muchas superficies de hormigón o asfalto son rugosas y además pueden tener baches, grietas, y cualquier otro tipo e irregularidades dificultan el correcto acoplamiento de las ruedas.
2. Choque con objetos o desplazamiento a través de ellos.
3. Deslizamiento de ruedas debido a:
 - Superficies resbaladizas.
 - Sobre aceleración o giros rápidos
 - Falta de contacto con la superficie de desplazamiento
 - Bloqueos externos (obstáculos) o internos (ruedas estabilizador).

2.2.3 Determinación de la base de giro de un robot diferencial mediante el método directo

En un robot diferencial, la distancia entre los dos puntos de contacto de las ruedas d odometría con el plano de desplazamiento, se conoce como base de giro B. las ecuaciones cinemáticas para la estimación de la posición mediante odometría, se basa en el caso ideal donde las ruedas se apoyan solo en su punto de tangencia con el ya mencionado plano de deslizamiento. Sin embargo, debido a que normalmente lo que las ruedas ofrecen es un área de contacto, entonces la distancia entre los puntos de apoyo suele variar ligeramente dependiendo de factores como el desbalance dinámico a causa de movimiento giratorios, a la deformación electica que sufren las ruedas pro el peso muerto sobre el robot, etc. De aquí que sea necesario determinar el valor practico de esta distancia

La incertidumbre en el diámetro de las ruedas [33], aquí se propone un me ‘todo que minimiza este efecto ya que se trata de una trayectoria de excitación estrictamente angular, la cual se muestra en la figura 35.



- O : Origen posición inicial
- θ : Angulo de giro
- r_R : Radio rueda derecha
- r_L : Radio rueda izquierda
- d : Distancia nominal entre ruedas
- B : Base de giro
- S_R : Trayectoria rueda derecha
- S_L : Trayectoria rueda izquierda
- ϕ_R : Diámetro lugar geométrico rueda derecha
- ϕ_L : Diámetro lugar geométrico rueda izquierda
- ϕ_O : Diámetro lugar geométrico centro robot

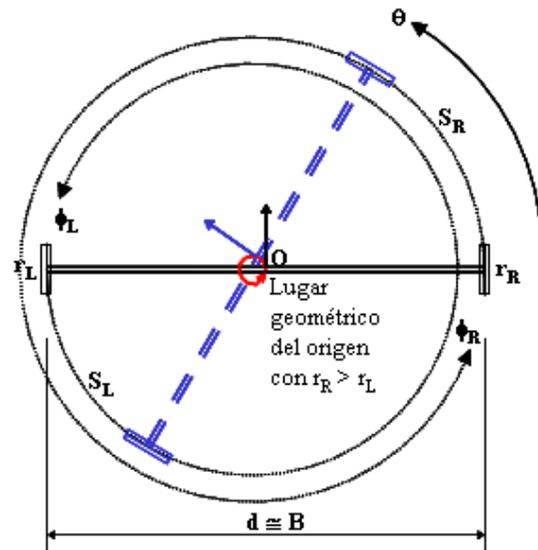


Figura 34 Lugar geométrico puntos de apoyo de la base de giro en móvil diferencial

En la figura expuesta se muestra el lugar geométrico que describen los puntos de apoyo y el centro del robot cuando el móvil se hace girar sobre su mismo eje a velocidad constante y con diámetros de ruedas distintos entre sí. En el caso de la figura anterior, $-V_L=V_R$ y $r_R > r_L$.

Un primer detalle que se observa es que el centro del robot describe una circunferencia cuyo diámetro está relacionado a su vez con los diámetros de los lugares S_R y S_L mediante la siguiente formula:

$$\phi_{origen} = \frac{\phi_R - \phi_L}{2} \quad (2.25)$$

Donde ϕ_R y ϕ_L son los diámetros de la circunferencia descrita por el punto de contacto de la rueda derecha e izquierda respectivamente. De lo anterior, y después de algunas operaciones, se puede deducir que la base de giro B es el promedio de los diámetros de los lugares geométricos S_R y S_L .

$$B = \frac{\phi_R + \phi_L}{2} \quad (2.26)$$

Un segundo detalle es que a pesar de que el centro del robot hace un giro excéntrico los puntos de apoyo mantiene la misma trayectoria consistentemente ciclo tras ciclo. Esto permite afirmar que el término de un giro de 360° , el centro del robot y los puntos de apoyo coincidirán con el punto del cual partieron originalmente. Esto es especialmente conveniente porque permitirá entonces determinar las coordenadas (x, y) de los puntos de medición dentro de un marco de referencia local. Así el procedimiento para determinar S_R y S_L se inicia programando el robot para que gire aproximadamente 360° sobre sí mismo eje, al tiempo que se registran los pulsos (N_R, N_L) proveniente de los codificadores derecho e izquierdo respectivamente. Entonces, conociendo el Angulo de giro total θ , se determinan los diámetros derecho e izquierdo de acuerdo a lo siguiente.

$$\phi_R = 2 \frac{C \Delta N_R}{\theta} \quad (2.27)$$



$$\phi_L = 2 \frac{C \Delta N_L}{\theta} \quad (2.28)$$

C corresponde a la sensibilidad odométrica. ΔN_R y ΔN_L corresponden al número de pulsos registrados por los codificadores rotativos a través de los trayectos S_R y S_L respectivamente, y θ es el giro total que se determina mediante la ecuación:

$$\theta = 2\pi + \sin^{-1} \left[\frac{(x_{L1} - x_{R1})(y_{R0} - y_{L0}) + (y_{L1} - y_{R1})(x_{L0} - x_{R0})}{(d + m_1 + m_2)} \right] \quad (2.29)$$

Aquí d es la distancia nominal entre los puntos de apoyo de las ruedas de odometría, (x_{R0}, y_{R0}) y (x_{L0}, y_{L0}) son las coordenadas de los puntos de medición derecho e izquierdo respectivamente, medidas al inicio del movimiento. (x_{R1}, y_{R1}) y (x_{L1}, y_{L1}) son las coordenadas de los puntos de medición derecho e izquierdo respectivamente medidas al finalizar el movimiento. Igualmente, m_1 y m_2 corresponden a la distancia que existe entre el punto de apoyo de la rueda derecha e izquierda respectivamente y el punto de medición de su correspondiente lado.

Finalmente, al sustituir las dos anteriores en la anteriores, se obtiene la base de giro según:

$$B = \frac{C \Delta N_R + C \Delta N_L}{\theta} \quad (2.30)$$

2.2.4. Resultados para la localización mediante sensores de odometría

El modelo de estimación del estado de localización odométrica puede expresarse como:

$$\begin{bmatrix} x_k \\ y_k \\ \theta_k \end{bmatrix} = \begin{bmatrix} x_{k-1} \\ y_{k-1} \\ \theta_{k-1} \end{bmatrix} + \begin{bmatrix} \Delta S_k \cos \left(\theta_{k-1} + \frac{\Delta \theta_k}{2} \right) \\ \Delta S_k \sin \left(\theta_{k-1} + \frac{\Delta \theta_k}{2} \right) \\ \Delta \theta_k \end{bmatrix} \quad (2.31)$$

$$\Delta S_k = \frac{1}{2} (C_R \Delta N_{k,R} + C_L \Delta N_{k,L}) \quad (2.32)$$

$$\Delta \theta_k = \frac{1}{B} (C_R \Delta N_{k,R} - C_L \Delta N_{k,L}) \quad (2.33)$$



Capítulo 3. Diseño, selección de componentes y módulos



3.1. Cualidades del diseño

El robot a diseñar estará sujeto a los requisitos que marca la asociación mexicana de mecatrónica para la competencia de robot seguidor de línea velocista. De manera que la selección de piezas y su fabricación quedará limitada por el reglamento que marca esta competencia.

3.1.1. Características del robot marcadas por el reglamento

El robot debe ser completamente autónomo. No puede ser controlado y/o calibrado remotamente en su recorrido por ninguna clase de dispositivo ya sea radiocontrol, bluetooth, wifi, infrarrojo, o por cualquier otro medio inalámbrico conocido.

Las dimensiones máximas del robot son:

Tabla 3 Limite de dimensiones

Largo	21 cm
Ancho	19 cm
Alto	Sin limite
Peso máximo	500 gr

Se puede emplear cualquier tipo de micro-controlador, microprocesador, tarjetas tipo Arduino, PICAXE, PLC'S, BASIC STAMP, etc. Al igual que utilizar cualquier tipo de componentes electrónico básicos (transistores, compuestas, etc.), sin embargo, nos debe utilizar robots comerciales.

La parte motriz deberá incluir motores de corriente directa de cualquier tipo de alimentación por baterías.

El voltaje máximo que debe existir en el circuito es de 17 V.

3.2. Establecimiento de funciones

De acuerdo a la literatura los robots móviles tienen una capacidad de funciones que determinan su autonomía enlistada a continuación:

Un robot móvil totalmente autónomo tiene la función de[30]:

- Obtener información sobre el medio ambiente con el que interactuar.
- Trabajar por un periodo prolongado sin intervención humana.
- Moverse todo o parte de sí mismo a lo largo de su entorno operativo sin asistencia humana.
- Evitar situaciones que son perjudiciales para personas, los bienes, o el mismo, a menos que esté dentro de sus especificaciones de diseño.

Un robot móvil también puede aprender o adquirir nuevas capacidades como el ajuste estratégico para el cumplimiento de su tarea(s) o la adaptación a un entorno cambiante. Estas funciones se encuentran en el robot móvil, pero son propias del sistema de navegación.



Las funciones clave que se pretende abordar para este diseño en particular son:

- Capacidad de desplazamiento por medio de motores CD.
- Capacidad de identificar mediante sensores reflexivos una pista de línea y fondo de contraste (cualidad de seguidor de línea).
- Captar datos odométricos por cada rueda de manera individual.
- Captar cambios en la orientación mediante un giroscopio.
- Enviar datos de manera inalámbrica sobre dichas lecturas.

La realización de estas actividades es suficiente para experimentar con la plataforma robótica, siempre y cuando los datos obtenidos sean recibidos en un valor útil o fácil de interpretar.

3.3 Determinación de características

A partir de los atributos que debe cumplir nuestro robot determinamos un orden jerárquico de los mismos, de manera que podamos identificar como se compone cada uno desglosándolo en un conjunto de atributos de menor nivel y a su vez en características de los mismos.

Tabla 4 Clasificación de atributos

Atributos primarios	Atributos secundarios	Atributos terciarios
Fácil de operar	Control manual simple	Botones e interruptores siempre a la vista
		Botos e interruptores con etiquetas para identificarlos
		Indicadores a la vista que muestren el estado o el avance de las operaciones (leds, display)
	Control a distancia simple	Interfaz de usurario limpia (número de comandos y funciones limitado a funciones básicas)
		Compatibilidad del software con los dispositivos más usados
		Indicadores en pantalla que muestren el estado del robot.
Buen desempeño	Seguimiento correcto de la trayectoria	Comprobar posición mediante un sistema de localización
		Comprobar mediante sensores internos si ha habido algún deslizamiento en las ruedas
		Utilizar las rampas



		Dividir los tramos a recorrer en secciones, de manera que pueda determinar cuándo cambiar la velocidad de acuerdo a las condiciones del terreno o antes de llegar a curvas.
	Interacción con el usuario	Indicar cuando el robot esté listo para avanzar
		Indicar cuando el robot tenga alguna dificultad para seguir una trayectoria
		Indicar cuando el robot se encuentre perdido o deje de reconocer referencias del entorno
		Indicar cuando el robot haya llegado a su destino
		Ofrecer la posición del robot en caso de ser requerida a un dispositivo móvil
	Arquitectura mecánica	Carcasa impermeable para soportar salpicaduras o lluvia
		Sujeción de las piezas para que conserven su posición
		Fácil de desarmar
		Acomodo compacto de los elementos para reducir espacio
		Conexiones mínimas visibles
		Bajos peso. Utilizar materiales de baja densidad para crear la carcasa que resguardará los elementos electrónicos
		Carcasa sencilla de reproducir
		Usar materiales baratos para construir la carcasa. Plásticos y metales de uso común
		Resistencia a impactos repentinos
		Resistencia a cargas aplicadas que se puedan dar
Buena apariencia	Vista superficial	Cableado y circuitería ocultos bajo la carcasa
		Acabado liso



		Fácil de limpiar
		Puertas, tornillos y juntas ocultos
	Ajuste	Espacio uniforme entre elementos que tengan vista al exterior (sensores)
		Simetría en su carcasa

3.4 Aportación en el diseño

El presente trabajo muestra el diseño y desarrollo de un robot de tracción diferencial, cuya arquitectura respeta las medidas y las características de un robot de competición seguidor de línea.

Se desarrollan las ecuaciones cinemáticas que describen el movimiento del robot. Además, se realiza un diseño electrónico de control para el sistema sensorial, abordando instancias para cada tipo de sensor, y de locomoción del robot. Posteriormente, se desarrollará un software de localización basado en el seguimiento de posición que funcione para dicho robot y los elementos seleccionados, de manera que el robot pueda aplicar el sistema de localización para determinar su posición en una pista de línea blanca sobre fondo negro, a la vez que sirva para pruebas más avanzadas de investigación basándonos en sistemas de localización de tipo propioceptivos. El robot de ser realizado será propiedad de la UMSNH y servirá para prácticas, concursos y presentaciones que la Facultad de ingeniería mecánica considere prudentes.

3.5 Hardware del robot móvil de tracción diferencial

El robot móvil de tracción diferencial fue diseñado para poder ser utilizado en competencias de alto desempeño, a la vez que tenga componentes que permitan utilizarlo para aplicar características de localización y navegación basadas en sistemas propioceptivos.

El fin último es que dicha plataforma robótica represente en competencias nacionales e internacionales a la UMSNH y que a la vez permanezca en constante actividad como una herramienta de investigación que motive a los estudiantes a abordar temas de robótica móvil.

3.5.1. Selección de los motores

Los motores seleccionados son los 5:1 Micro Metal Gearmotor HPCB 6V con eje extendido. Tienen la característica de contar con un reductor de velocidad diseñado especialmente para dicho motor. Estos motores son fabricados por la marca POLOLU. Se recurre a ellos en casi todos los casos de construcción de robot velocistas. Su tamaño, su torque y la velocidad tope que tienen, hacen que sean ideales para competición.

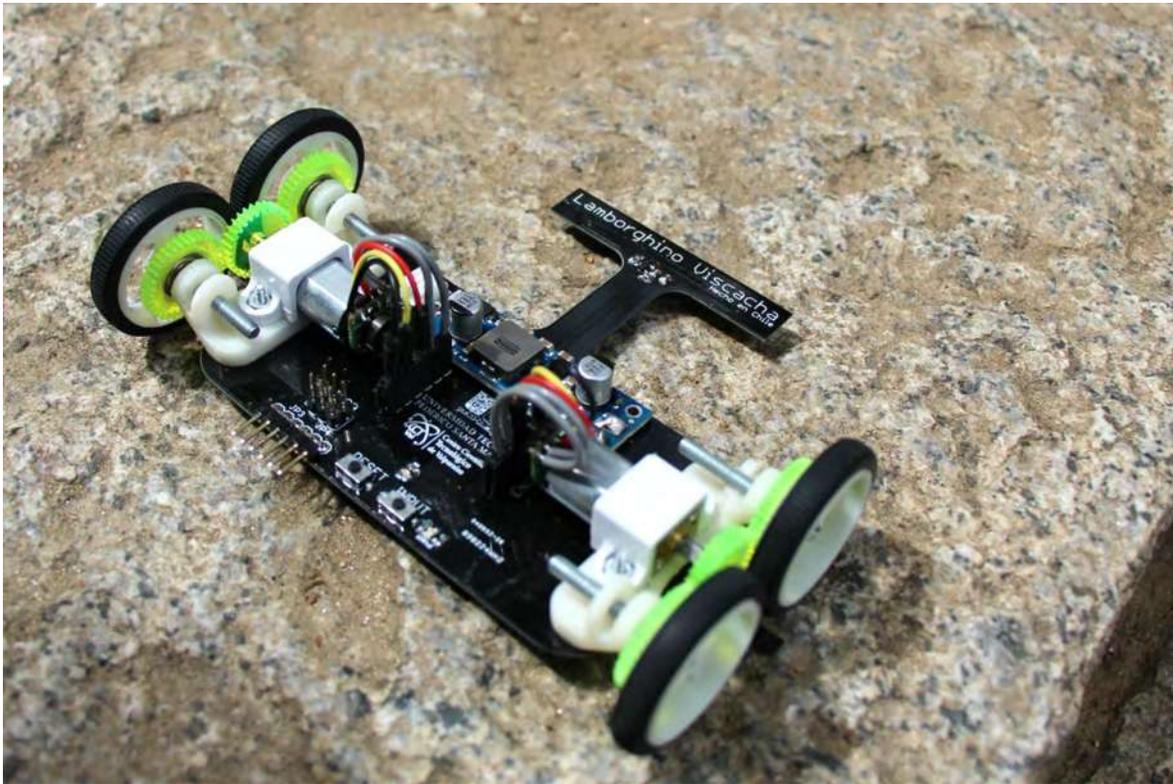


Figura 35 Lamborghini Viscacha. Seguidor de línea más rápido de américa latina

El Lamborghini Viscacha utiliza un par de motores de la misma marca y características, pero de una reducción en su tren de engranajes que sacrifica la velocidad por el torque, ya que este diseño contempla el uso de dos ruedas por eje motriz.

Las características de los motores son:

Tabla 5 Características de los motores

Potencia	6 V
Corriente máxima demandada	1600 mA
RPM max	6000
Torque	2 oz-in
Precio por motor	18.95 dll

Cuenta con un eje extendido que permite instalar un codificador rotatorio de lado contrario a donde se ubican las ruedas. También está equipado con escobillas de larga vida.



Figura 36 5:1 Micro Metal Gearmotor HPCB



3.5.2. Selección de ruedas

Existen una serie de criterios a la hora de escoger ruedas. Algunos de los criterios que se tomaron en cuenta son:

- Dimensiones
- Material

Las dimensiones deben ser tales que el diámetro de la rueda no separe mucho el chasis del robot del suelo, con la intención de mantener lo más cercano posible el centro de gravedad al suelo, de esta manera se estará aumentando el efecto de tracción y por tanto podrá desarrollar mayor velocidad sin haber deslizamiento. También influye el ancho de la rueda, ya que, a mayor ancho, mayor será la tracción de la misma.

Los materiales disponibles para ruedas son:

- Goma
- Espuma

La goma se usa principalmente en terrenos de características irregulares, ya que tienen mayor resistencia al desgaste que las ruedas de espuma, sin embargo, la espuma se adhiere mejor a las superficies, aumentando el efecto de tracción. Para el caso que nos comete, las superficies de pruebas serán limpias y lisas, por lo que no será necesario usar goma y podemos usar en su lugar un material de mejor adherencia. La espuma también presenta menos irregularidades en la rueda, ya que normalmente las ruedas de goma guardan una cavidad hueca entre la llanta y el rin.

Las ruedas seleccionadas fueron las SC-2008 de la marca Scaleauto.

La rueda tiene las siguientes características:

Tabla 6 Características de las ruedas

Dimensiones	20.5 x11 m, eje 3mm, llanta 15mm
Material del rin	Magnesio
Material de la rueda	Foam (espuma)
Precio	8.49 EUR



Figura 37 Rueda SC-2008



Estas ruedas son usadas en competencias de radiocontrol, pero también en el mundo de la robótica de competición, principalmente en las ligas profesionales europeas. Sobra decir que el barrero de la rueda está diseñado para albergar ejes como el del motor que fue previamente diseñado, siendo una medida estándar de 3 mm.

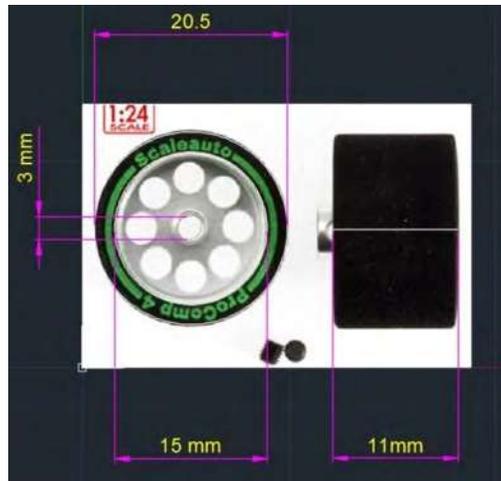


Figura 38 Inspección de las dimensiones de la rueda

3.5.3 Elementos mecánicos

Los elementos mecánicos constan de la serie de piezas que conforman el chasis, es decir, toda la indumentaria sobre la cual se ubicaran los elementos electrónicos.

Para este tipo de robots es muy común ver como todo se monta sobre la misma baquelita, la cual es cortada y diseñada en geometría con la arquitectura típica de un robot velocista.



Figura 39 Robot Pumatron de la liga profesional de competición española

En la figura 40 podemos ver cuál es la apariencia de un robot que utiliza la baquelita como único elemento de arquitectura mecánica. Tiene por inconveniente que al romperse se perdería todo el circuito electrónico, ya que son prácticamente la misma pieza. La baquelita tiene un grosor estándar de 0.062". sus propiedades de material son:

**Tabla 7** Propiedades mecánicas de la baquelita

Tensión a la rotura perpendicular	120 Mpa
Modulo aparente de elasticidad a flexión	7x10.3
Resistencia a la tensión	120 Mpa
Resistencia a la hendidura paralela de las capas	10 Mpa
Densidad	1.3-1.4 g/cm ³

**Figura 40** Piezas de pinza robótica impresas en 3D

Otro material que se emplea mucho en este tipo de funciones es el conocido como PLA (ácido profiláctico). Este material es principalmente usado mediante impresoras 3D y su mayor ventaja radica en eso. Dado que la impresora 3D puede duplicar cualquier tipo de prototipo que tengamos en mente, podemos hacer ensayar previos y simulaciones que garanticen la integridad del chasis antes de concebirlo físicamente. Una ventaja adicional a comparación de trabajar sobre la baquelita es que la circuitería quedaría respalda por el armazón a su alrededor, pudiendo sobrevivir a un impacto del móvil.

Sus propiedades mecánicas son[34]:

Tabla 8 Propiedades mecánicas del PLA

Resistencia a la tensión	40-60 Mpa
Módulo de Tensión	3-4 Gpa
Densidad	1.3 g/cm ³

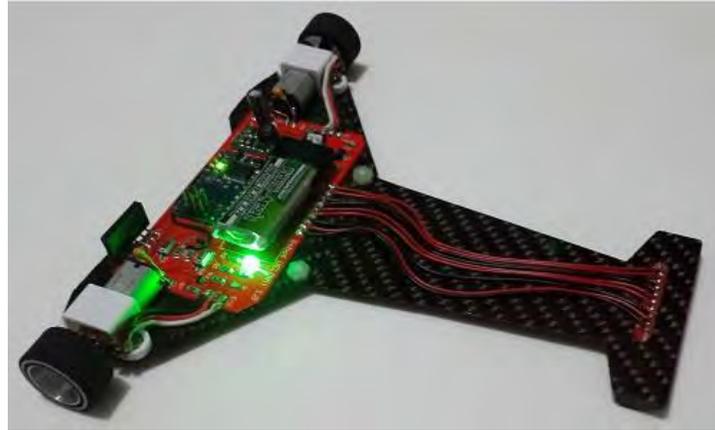


Figura 41 Robot Velowii con chasis de fibra de carbono

Los materiales pueden mostrar ciertas diferencias en sus propiedades, pero en la práctica no ha demostrado que uno sea significativamente mejor que otro, teniendo cada material sus pros y sus contras, sin embargo, la fibra de carbono ha demostrado tener mejores propiedades mecánicas que los antes mencionados, al ser fuerte y liviano a la vez. Su mayor inconveniente es su accesibilidad, ya que se considera un material especial y tiene poca difusión en usos fuera de las industrias aeronáuticas.

Por las ventajas que supone, el diseño del robot será a través de impresiones 3D, con esto garantizamos que su reproducción o reposición puede ser más sencilla, así como también su estudio en cuestiones mecánicas de tipo estructural.

3.5.4 Sistema de procesamiento

Actualmente el impacto que refiere entre la comunidad mecatrónica la introducción de Arduino ha hecho que trabajar con estas tarjetas sea casi un estándar dentro de la robótica de investigación. Arduino es una plataforma sencilla y el acceso libre en la cual podemos programar de manera sencilla y practica un microcontrolador, a la vez que las tarjetas ya están equipadas para ser conectadas con facilidad a diferentes módulos de sensores y a equipos de cómputo. La popularidad de las mismas ha hecho que los desarrolladores que trabajan con microcontroladores de la marca Atmel diseñen plataformas similares a arduino o maneras de programar estos controladores a través del lenguaje de arduino.

En la robótica móvil de carácter didáctico se frecuenta el uso de Arduino UNO y Arduino Nano, siendo estos los que tienen más presencia en el mercado, sin embargo, eso no quiere decir que sean las mejores opciones o que sean opciones obligadas a la hora de armar un robot.

Existen controladores dedicados a robótica, los cuales cuentan además de la sección de procesamiento, algún extra que se pueda aplicar a esta área.



Tabla 9 Tabla comparativa de controladores para robots contra Arduino

	 Arduino Uno	 Baby Orangutan B-328	 Orangutan SV-328	 Orangutan SVP-1284
Supply voltage	7 – 12 V	5 – 13.5 V	6 – 13.5 V	6 – 13.5 V
5V regulator	0.25 – 0.8 A (dep. on VIN)	0.1 A	3 A	3 A
Secondary regulator	50 mA @ 3.3 V			3 A, adjustable (2.5 V – 85% VIN)
Size	2.95 × 2.10 in	1.20 × 0.70 in	2.15 × 1.90 in	3.70 × 2.20 in
Processor	ATmega328P	ATmega328P	ATmega328P	ATmega1284P
Program memory	31.5 KB	32 KB	32 KB	128 KB
RAM	2 KB	2 KB	2 KB	16 KB
Clock	16 MHz resonator	20 MHz resonator	20 MHz resonator	20 MHz resonator
Digital I/O	20	16	8	17
Analog inputs	6	8	8	12
User UARTs	0	1	1	2
User LEDs	1	1	2	2
User pushbuttons	0	0	3	3
User trimpot		✓	✓	✓
Built-in programmer	✓			✓
USB	✓			✓
Reset button	✓		✓	✓
Power button			✓	✓
Self-shutdown capability			✓	✓
Motor drivers		2 × 1A (continuous)	2 × 1A (continuous)	2 × 2A (continuous)
Motor current sensing				✓
LCD			8×2 character	16×2, backlit
Buzzer			On PWM output	On PWM output, MOSFET-amplified
Dedicated servo hardware				8 channels
Quadrature encoder inputs				✓
Price	\$29.95	\$19.95	\$64.95	\$99.95



De la siguiente tabla comparativa podemos destacar una cosa muy importante, la placa Arduino no es la mejor opción para controlar robots, ya que existen placas que incluyen en su estructura mayor número de herramientas y a pesar de que se basan en la misma tecnología de procesamiento y programación, claramente están mejor equipadas para manejar un robot. Una de las inclusiones en las placas Orangutan con respecto a las Arduino es la inclusión de un puente H, el cual es de mucha ayuda para controlar motores.

Otra observación que podemos resaltar es que, para nuestros fines prácticos, la cantidad de puertos digitales y analógicos no será un número elevado, por lo que los controladores grandes quedarían sobrados.

El controlador que cuenta con las mejores ventajas prácticas es el Baby Orangutan, ya que como se verá en el capítulo siguiente, tiene los puertos necesarios, el puente H es muy útil y su inclusión reduce el tamaño del circuito impreso que se va a realizar, además de que su tamaño representa una ventaja muy grande con respecto a los otros. Estamos hablando de un controlador que cuenta con el tamaño de un Arduino micro, pero con agregados de valor que lo convierten en una opción excelente para este proyecto.

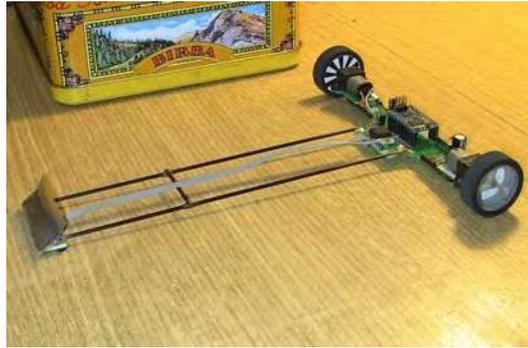


Figura 42 Ejemplo de robot móvil usando Baby Orangutan

Podemos ver este controlador en robots súper livianos, como es el caso de la figura 44, en la cual podemos observar como casi toda la parte electrónica queda sustituida únicamente por este controlador. Dicho controlador también es muy socorrido en la robótica de competición por sus cualidades, aunque muy de cerca compete el Arduino Nano, el cual es más fácil de conseguir.

3.5.5 Puente H

El puente H (llamado también Driver de motores) seleccionado es por consiguiente el integrado en la placa Baby Orangutan, conocido como TB6612FNG. El cual tiene las siguientes especificaciones[35]:

Tabla 10 Características del puente H

Voltaje máximo admisible	15 V
Corriente de salida	1.2 A promedio, hasta 3.2 A en pico



Los motores seleccionados trabajan con una corriente máxima de 1.6 A, lo cual sugiere que el Driver de motores queda limitado, pero hay que contemplar que en la práctica no se usarán corrientes mayores o cercanas al 1.6 A, ya que muy difícilmente un robot de este tipo puede manejar su velocidad máxima sin notar deslizamiento.

3.5.6. Encoders

Los encoders seleccionados son los diseñados por POLOLU para sus Micro motores. Estos encoders funcionan por marcas magnéticas y se ajustan perfectamente al eje extendido de los motores seleccionados.

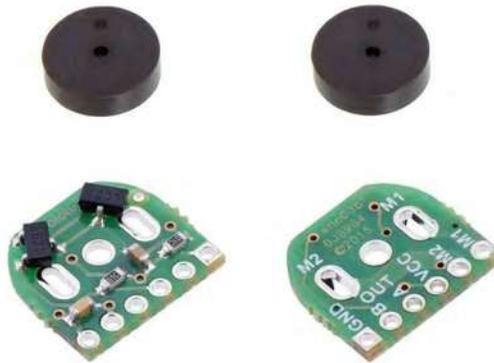


Figura 43 Encoders magnéticos de la marca POLOLU

Sus características principales son:

Tabla 11 Propiedades de los encoders

Resolución	12 cuentas por revolución
Voltaje de operación	2.7 – 18 V
Precio	8.95 dll por par

Estos encoders apenas comienzan a tener relevancia, ya que anteriormente los entusiastas de la robótica móvil diseñaban sus propios encoder de tipo óptico usando sensores reflexivos.

3.5.7 Giroscopio

Este elemento pertenece a una serie de sensores denominados IMU, los cuales básicamente se denominan así por su capacidad de medir aceleración y velocidad. Cabe resaltar que un IMU no puede medir ángulos de manera directa, pero su aplicación se utiliza para determinar estos cambios de orientación a través de modelos matemáticos.

El giroscopio seleccionado es el MPU-6050. Este sensor es muy popular y fácil de conseguir. El MPU-6050 es un IMU de 6DOF (se lee “6 Degrees Of Freedom”). Esto significa que lleva un acelerómetro y un giroscopio, ambos de 3 ejes (3+3=6DOF). Hay IMUs de 9DOF, en este



caso también llevan un magnetómetro. Otras pueden tener 5DOF, en cuyo caso el giroscopio sólo mide dos ejes, etc.



Figura 44 Sensor MPU 6050

Características:

Tabla 12 Características del MPU 6050

Voltaje de operación	3.3 V
Tipo de comunicación	Protocolo I2C
Precio	120 MXN

3.5.8 Reflexivos

Los sensores reflexivos son la parte más característica de un seguidor de línea, siendo estos los que dotan al robot de la habilidad de percibir contraste entre negro y blanco al medir valor de luz reflejada sobre la superficie que se pretende observar.



Figura 45 Sensor QTR-3A



La figura 47 muestra el sensor seleccionado, el cual pertenece a la marca POLOLU. Este sensor emite señales analógicas individuales a través de sus 3 componentes reflexivos distribuidos sobre su superficie.

Características del sensor:

Tabla 13 Características sensores reflexivos

Voltaje de operación	5 V
Corriente típica de operación	50 mA
Rango de la señal analógica	0 V hasta el voltaje de operación
Distancia de censado óptico	3 mm
Distancia máxima recomendada de censado	6 mm
Peso	0.6 g
Precio	5 DLL

Como característica agregada, la distribución de los sensores está estandarizada para competencias de seguidor de línea en grosor típico de línea de 1.5 cm con variación de hasta 0.5 mm.

La selección del componente se llevó a cabo tomando en cuenta dos factores importantes que existen en las piezas previamente seleccionadas y los diseños de robot previamente estudiados:

- Los puertos con los que cuenta el controlador
- En la práctica raras veces se utilizan más de 3 sensores, usando el central para detectar presencia sobre la línea y los laterales para corregir movimientos al despegarse del contorno e invadir la línea. Un buen ejercicio de control debe ser suficiente para no depender de más sensores laterales, ya que se prevé que no deba hacer correcciones muy intempestivas. Los robots de tipo didáctico llegan a usar arreglos de hasta 8, pero también su sistema de control es apenas de tipo proporcional y no gestiona bien la velocidad del móvil.

El fabricante sugiere para interactuar con el sensor hay tres maneras diferentes:

- Utilizar el convertidos analógico-digital (ADC) de un microcontrolador para medir las tensiones.
- Usar un comparador con un umbral ajustable para convertí cada tensión analógica en una señal digital (es decir, comprar la reflexión sobre negro y blanco) que puede ser leído por la línea 1/0 digital de un microcontrolador.



- Conectar cada salida directamente a una línea E/S digital de un microcontrolador y confiar en el agente de comparación interna.

Para el diseño que pretendemos, la manera de interactuar con el sensor será diferente a las propuestas por el fabricante. Se utilizará la señal analógica en un puerto analógico y se calibrará en función de lo que se quiera observar, asociando rangos de reflexión al negro y al blanco para distinguirlos.

3.5.9. Bluetooth

Para entender porque se usará un módulo de bluetooth hay que considerar varios puntos en robótica, uno de esos puntos es que el diseño contempla la comunicación de datos, es decir, el diseño debe incluir un medio que le permita no solo adquirir datos, sino también comunicarlos. La razón principal por la que los datos serán comunicados es para tener lecturas que puedan ser estudiadas, tratadas y posteriormente usadas con fines de investigación.

Esta manera de comunicar un móvil lleva un nombre que se repite mucho dentro de la robótica, telemetría. La telemetría es la técnica automatizada de las comunicaciones que ayuda a tomar mediciones y recopilación de datos de manera remota. Esta técnica utiliza comúnmente transmisión inalámbrica, aunque originalmente se llevaba mediante la inclusión de sistemas alámbricos.

Los módulos seleccionados para las tareas de telemetría son los HC-06 y el HC-05. Gracias a sus puertos TxD y RxD nos permite realizar comunicaciones inalámbricamente a 10 mts. Max. Resultando muy fácil de manejar gracias a su SPP (*Serial Port Protocol*) de bajo consumo, ya que trabaja a 3.3 V. los módulos de bluetooth HC resultan populares para aplicaciones con microcontroladores PIC y Arduino. Sacando ventaja de su tamaño: 12.7*27mm.

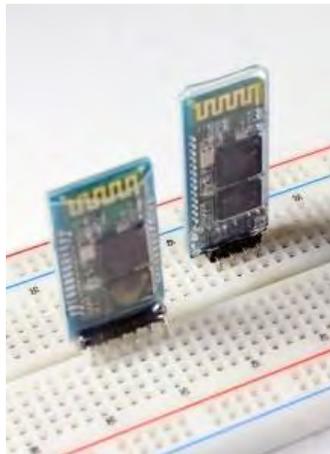


Figura 46 Módulos seleccionados, HC-05 y HC-06

El módulo HC-06 solo opera como esclavo, a diferencia del 05, que ofrece una mejora con respecto al precio y características, ya que es un módulo maestro-esclavo, esto quiere decir



que además de recibir conexiones desde una PC o Tablet, también es capaz de generar conexiones hacia otros dispositivos bluetooth.

Esto no permite, por ejemplo, conectar dos módulos de BT y formar una conexión punto a punto, para transmitir datos entre dos microcontroladores o dispositivos.

Su valor está entre los 150-200 MXN y son relativamente fáciles de conseguir.

Para nuestra configuración el HC-05 irá sobre el robot, ya que es el que debe transmitir datos, mientras que el HC-06 se conectará a un Arduino UNO que funcionará como receptor y el cual será a su vez un vínculo entre el robot y una computadora. Esta parte será explicada a mayor detalle en el capítulo siguiente.

3.5.10. Baterías

De acuerdo a la práctica, las baterías que están tomando más relevancia en el mundo de la robótica son las de tipo Lipo. Son baterías de última generación dentro del mundo de radio control, generalmente usadas en drones. Son baterías con una excelente relación entre capacidad, peso, volumen y voltaje.

Estas baterías al ser comparadas con sus antecesoras, las de Ni Mhi o Ni Cd, presentan puntos destacables. Donde antes teníamos un pack del tipo 7.2V y 3000 ma de Ni Mhi o Ni Cd, ahora tenemos uno de 7.4 V y 6000 ma de Lipo, pensando tan solo la mitad, y con incluso mejor capacidad de descarga. Esto hace que al instalar esta nueva tecnología se obtengan solo mejoras en los diseños.

Las baterías Lipo están formadas por elementos de 3.7 V, a estos elementos se les conoce como celdas o células. En radio control se suele utilizar baterías hasta de 8 celdas, esta cualidad se denomina con la letra S. Las más habituales son las de 1, 2 y 3 celdas. Para saber el voltaje de una batería basta con multiplicar el voltaje de celda por el número de celdas, ya que se conectan en serie, de ahí que se denominen con 2S, 3S, por ejemplo.

Descarga de una batería Lipo. Para escoger una batería de esta tecnología se debe tener en cuenta el amperaje de descarga que tendrá el modelo en el que se va a usar. La batería tiene una referencia de descarga máxima que viene expresada con un número seguido de una "C" (25C, por ejemplo), donde C es el amperaje de la batería. Para saber que amperaje descarga una batería tenemos que multiplicar los mA de esta por el número delante de la C. Por ejemplo, si tenemos una batería de 4000mA y 30C, la descarga máxima a la que podemos someter esta batería sería 120 A. es decir que la podemos montar en un prototipo que consuma un máximo de 100-110 A, manteniendo un margen de seguridad.

En algunas baterías Lipo, aparece la referencia de descarga indicada por dos valores (20-30C, por ejemplo), esto nos indica que el fabricante está diciendo que la batería se puede descargar a 20C de manera continua y que la batería puede darnos hasta 30C por un tiempo limitado, unos segundos.

Carga de una batería Lipo. Estas baterías utilizan un cargador específico por lo que, si el cargador no especifica claramente que sirve para cargar baterías Lipo, no se debe utilizar nunca con este propósito. Las baterías Lipo nunca se cargan por encima de su voltaje nominal. Es decir, si la batería es de 3500 mA no se puede cargar a 4000 mA.



Por otro lado, la conexión entre la batería y el cargador se lleva a cabo por diferentes entradas de cables, dependiendo de la cantidad de celdas de la batería.



Figura 47 ejemplo de cargador de baterías Lipo

La batería seleccionada fue la Turnigy nano-tech de 300mA 1S 45-90C. Sus características principales son:

Tabla 14 Características de la batería Turnigy nano-tech

Capacidad	300 mA
Voltaje	1S/3.7 V
Constante de descarga	45C continua/ 90C limite
Peso	9 g
Dimensiones	43*17*6 mm

Nuestro diseño contempla una demanda de corriente de 1.6 A por motor, lo que equivale a un máximo de 3.2 A. La capacidad de descarga de la pila sería igual a 25×300 , obteniendo un valor de 7.5 A, un valor que quedaría perfecto para el prototipo. También hay que acotar que los motores funcionan con 6 V, y la batería es de 3.7 V. Será necesario utilizar un subidor de voltaje de acople ambos valores, por consiguiente, la corriente necesaria subirá por el valor de conversión y eficiencia del subidor de voltaje.



Figura 48 Batería seleccionada



3.5.11. Subidor de voltaje

Para la selección de este componente se tomó en cuenta el voltaje de entrada, el voltaje de salida que se necesita y la corriente que se planea manejar. Los datos que manejamos son:

Tabla 15 Requisitos del subidor de voltaje

Voltaje de entrada	3.7 V
Voltaje de salida	6 V
Corriente de trabajo	Mínimo 3.2 A

Se seleccionó el subidor de voltaje U2V50F6. Este regulador tiene las siguientes características:

Tabla 16 Características del subidor de voltaje

Voltaje de entrada	2.9-6 V
Corriente de operación	5 A
Eficiencia típica	80-95%
Dimensiones	48*15*10.5 mm

Si tomamos en cuenta un valor de eficiencia del 80%, podemos decir que la corriente máxima que el robot va a demandar de la batería es igual a 4.41 A, lo que es menor a la corriente máxima que maneja este regulador.



Figura 49 Subidor de voltaje U3V50F6



Capítulo 4. Diseño del sistema



La lista de elementos que comprende este diseño se limita a:

- Batería Lipo de 3.7 V
- Subidor de voltaje BOOST de tipo ajustable
- Regulador de voltaje 785TV
- Leds indicadores de estado
- Pulsador
- Interruptor de dos canales
- Módulo HC-05
- Controlador B. Orangutan
- Arreglo de sensores reflexibles QTR-8A

Estos elementos se consiguieron en manera física para poder hacer un software con el cual se pudiera lograr que el robot siga una línea de manera autónoma e incorporé un sistema de control del tipo PID, a la vez que el apartado de telemetría quedaría relegado para una posterior intromisión en modelos que puedan extra mayor cantidad de datos, dejando esta conexión para un módulo HC-06 que pudiera dotar al móvil de funciones de radio control.

Este primer acercamiento al diseño que se pretende es una opción que se conserva y que se planea desarrollar en forma para prácticas de licenciatura, claro, prescindiendo de elementos que para estudiantes pudieran ser caros y reemplazándolos por otros de menos calidad y que puedan cumplir las mismas funciones.

En este primer acercamiento se usaron los puertos analógicos para captar las señales de los sensores reflexivos, y los puertos digitales para registrar la información instruida por el pulsador y los interruptores. La idea de los interruptores es poder cambiar la velocidad del móvil sin tener que entrar a la programación, simplemente se introducen instrucciones que moderen una velocidad máxima en relación a un número binario formado por los dos interruptores, tal número puede ir de 0 a 3, dando un total de 4 opciones. Los leds indicadores pueden interactuar con el usuario de manera que muestren algún estado en específico, el cual puede ser en relación a la velocidad, la energía o algún tipo de contador.

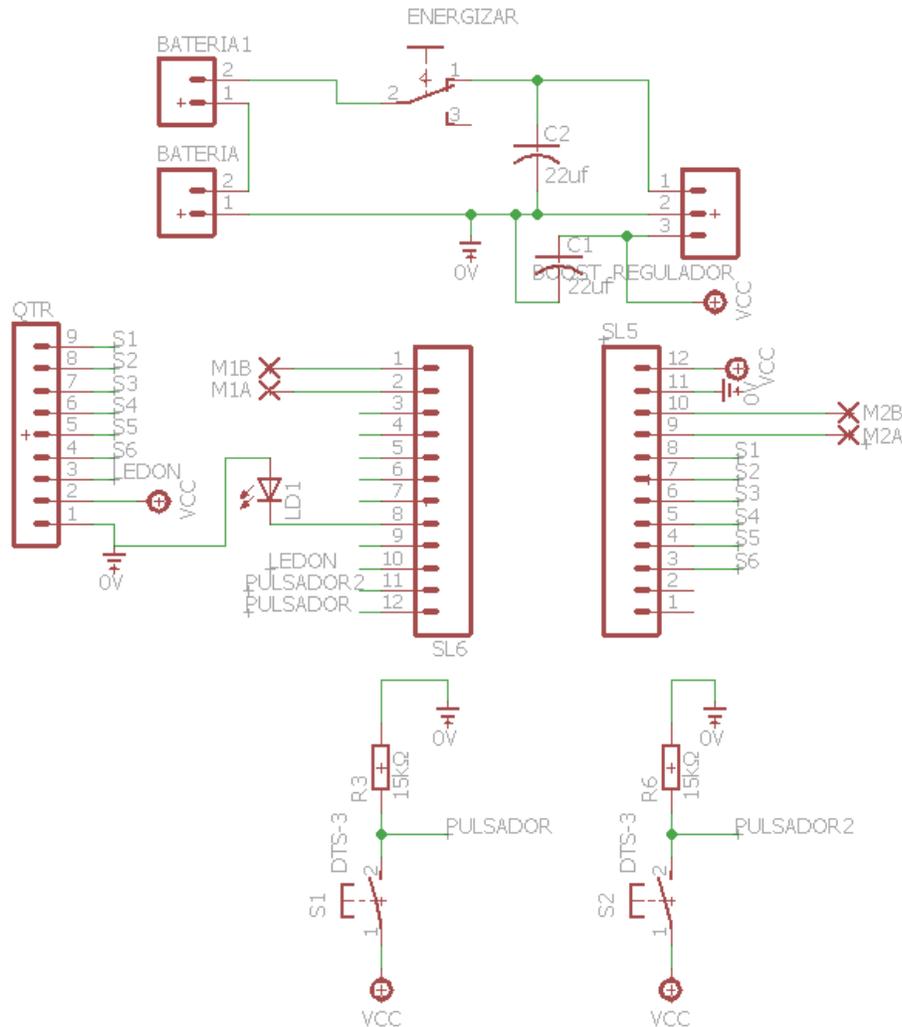


Figura 51 Robot seguidor de línea diseñado para ser construido por estudiantes

La aproximación al diseño que se muestra en la figura 53 reduce el número de elementos en relación con el mostrado con la figura 52. En este modelo se suprimió el regulador BOOST ya que es relativamente caro, y se solucionó el problema al usar dos pilas en serie y un bajador de voltaje de bajo precio. También se suprimió el uso de un módulo de bluetooth y de algunos interruptores y leds. Dejando únicamente dos pulsadores para dar instrucciones (se pretende por conteo de pulsaciones) y un led indicador para interacción con el usuario.

Este diseño se trabajó hasta una etapa en la que se pudiera un circuito impreso.

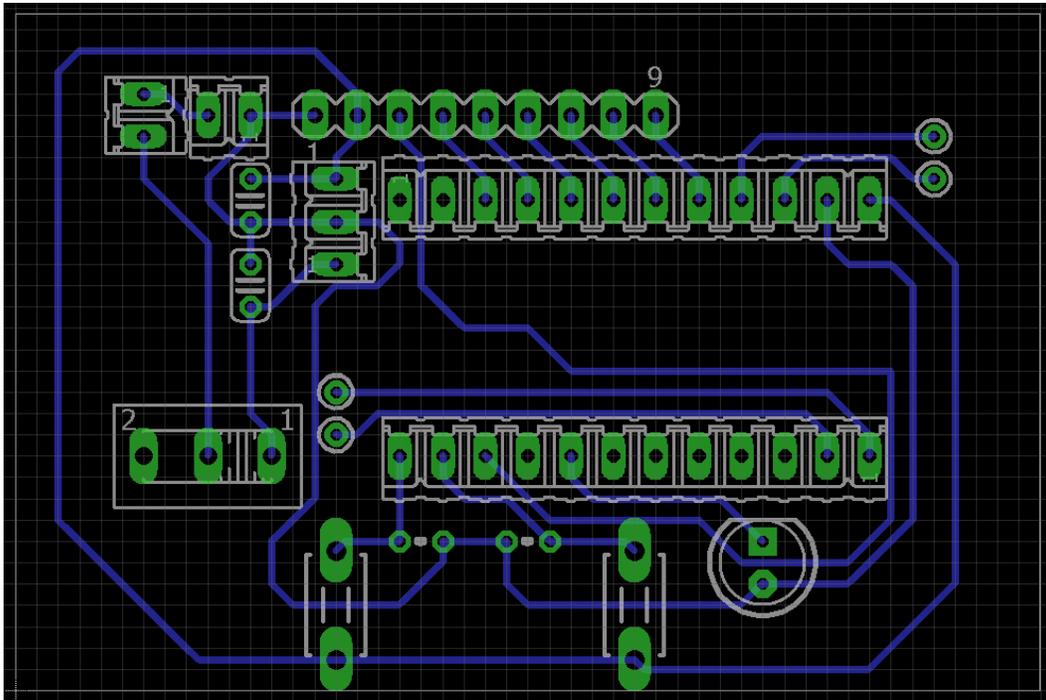


Figura 52 Placa diseñada para robot seguidor de línea simple

Debido a la cantidad reducida de elementos, se pudo concebir el diseño en una placa de dimensiones 4*5cm. En la figura 54 se observan los elementos que conforman el circuito, sus dimensiones en escala y su distribución.

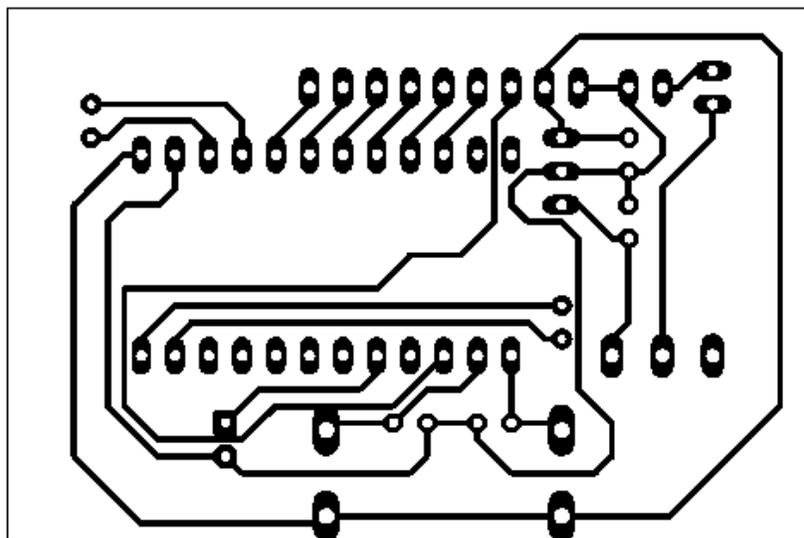


Figura 53 Circuito impreso de seguidor de línea listo para imprimir

La figura 57 muestra como quedarían las pistas del circuito y los barrenos sobre los que se soldaran os elementos.



De este mismo circuito se hizo una versión B, la cual pretendía separar los componentes de manera que fueran más fáciles de instalar y soldar, también acerca más las conexiones con los motores en un modelo físico, ya que en la primera placa las conexiones quedaban un tanto escondidas.

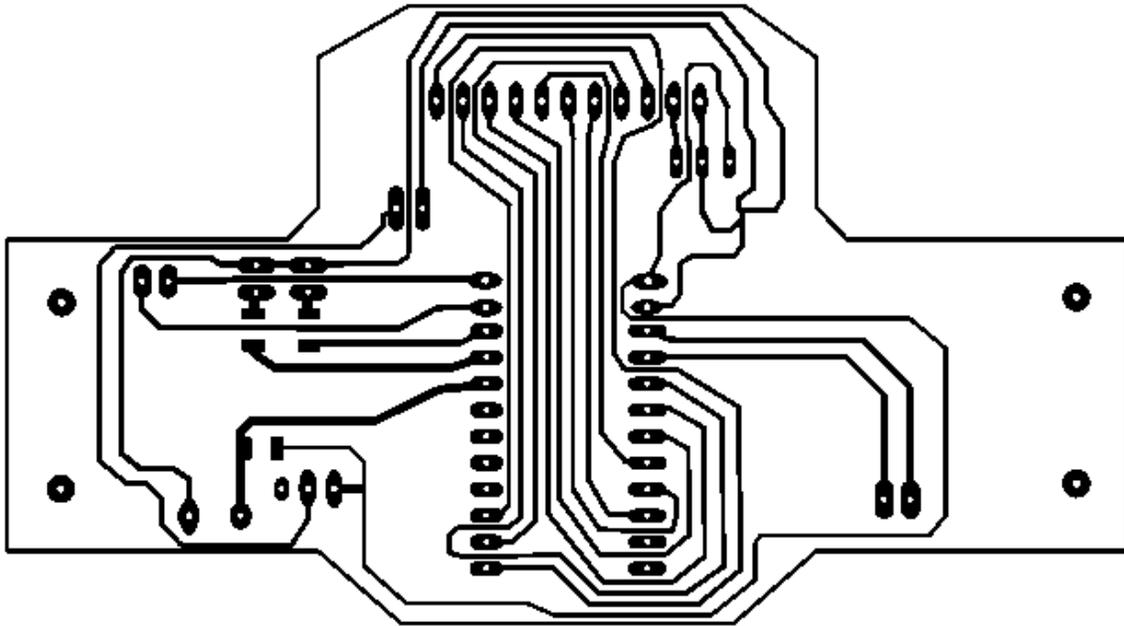


Figura 54 Versión B de circuito impreso para seguidor de línea



4.1.2. Diseño electrónico de seguidor de línea con fines e investigación.

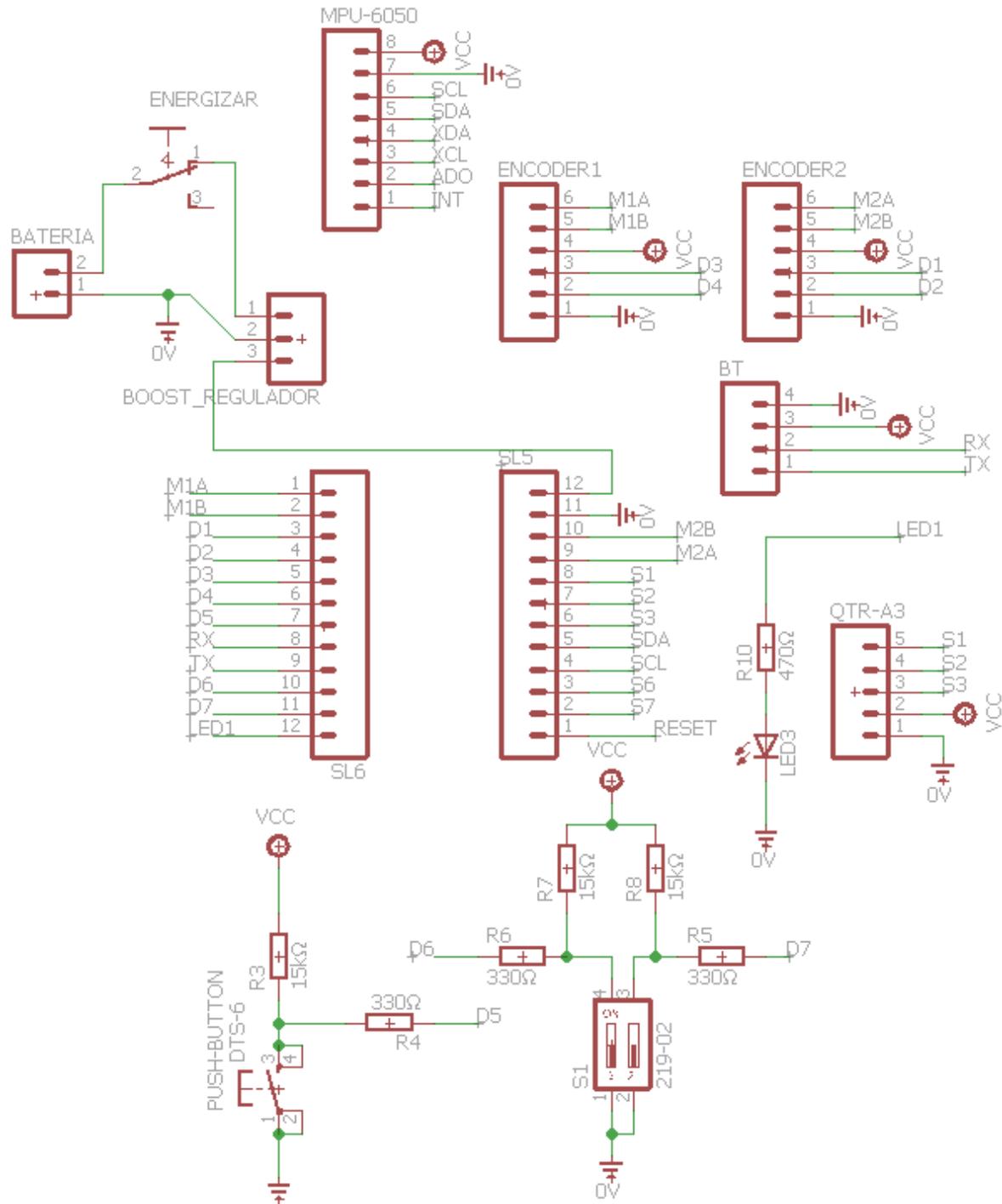


Figura 55 Esquema que muestra las conexiones entre los elementos previamente seleccionados para el seguidor de línea para investigación



La lista de elementos que comprende este diseño se limita a:

- Batería Lipo de 3.7 V
- Subidor de voltaje U3V50F6
- Led indicador de estado
- Pulsador
- Interruptor de dos canales
- Módulo HC-05
- Controlador B. Orangutan
- Arreglo de sensores reflexivos QTR-3A
- Encoders magnéticos
- MPU-5060

En este diseño se emplearon los elementos seleccionados a través de la investigación. El uso del sensor QTR-3A facilita la inclusión de más módulo que trabajen con señales analógicas, como es el caso del MPU-5060.

Se destaca que casi todos los puertos del controlador fueron ocupados, por lo que es interesante ver como la selección de un controlador tan pequeño fue suficiente la labor que le fue encomendada, dejando claro que la selección del elemento fue la correcta. También se debe agregar que, a comparación del seguir con fines didácticas, la cantidad de elementos que conforman este circuito ha crecido, por lo que su conexión en sobre una placa impresa será mucho más compleja, como veremos a continuación.

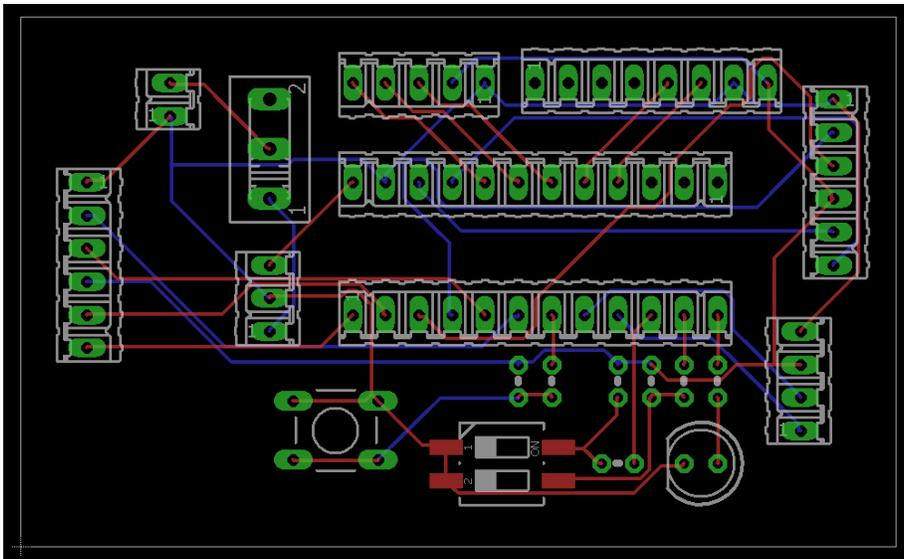


Figura 56 Primera vista al circuito impreso del robot seguidor con fines de investigación



En la figura 58 se puede observar cómo ha cambiado el circuito, mostrando en primera instancia una superficie mayor, la cual mide 7*4cm. Este diseño además muestra otras indudables diferencias, como el hecho de que ahora habrá pistas por las dos caras, lo que implica soldar algunos elementos en la cara superior. La complejidad para elaborar este diseño aumenta con dicho hecho, pero permanece dentro del rango de posibilidades que se pueden realizar y es una alternativa viable para el diseño que se comete completar.

Ante los problemas que las elaboraciones de la placa puedan sugerir existen una serie de alternativas. La soldadura en cara superior suele ser el punto que la placa muestra como más complicado, pero ya existen técnicas que pueden solucionar esto con demasiada facilidad, como es el uso de pasta de estaño en vez de usar caudín y alambre. El uso de la pasta de estaño nos permite vaciar sobre el área que se va a soldar el estaño suficiente y luego, sin meter puntas calientes de manera complicada, usamos una pistola de aire caliente que realice la unión. Esta podría ser una alternativa, no siendo la única. Otra de las alternativas posibles es el uso de un horno para fundir los puntos de soldadura, los cuales también se preparan previamente con una pasta de estaño.

El cuidado de la placa y las características finales que la van a acompañar está dado por el material, pero también por el recubrimiento que se pretende usar para preservar la integridad de las piezas, es por eso que para el diseño se seleccionó un recubrimiento muy sencillo de usar, el cual reacciona por medio de la luz.



Figura 57 Ejemplo de mascara para soldar.

El recubrimiento protector se conoce comúnmente como mascara para soldar o anisodonte, es un recubrimiento que generalmente encontramos en una presentación de color verde y que cubre todo el circuito a excepción de los nodos donde se soldan los componentes. Cabe mencionar que su nombre se debe debido a que uno de sus principales usos no es el de proteger, sino el de limitar los puntos de soldadura a un espacio, de manera que no se crucen con otros puntos o pistas.

El producto tiene un precio de 100 MXN y se presenta en forma de jeringa. Su aplicación consiste en distribuir una capa ligera sobre el circuito que se planea proteger. Se debe imprimir sobre una hoja transparente los pads o nodos donde irá la soldadura. Después de aplicar sobre la pieza el producto, se emparejan el circuito impreso con la impresión de los



nodos, de manera que la impresión cubra los nodos, y se hace incidir una luz ultravioleta o luz solar. El resultado es que la zona cubierta permanecerá limpia de solución y la zona sobre la que se incidió la luz estará seca.



Figura 58 ejemplo del uso de la máscara anisodonte



4.2 Diseño mecánico

Haciendo uso del software **AutoCAD** y el software **SketchUp** se realizaron varios prototipos de chasis, los cuales contaban con diferentes características. El prototipo seleccionado resulto ser la opción de diseño, debido a la cantidad de atributos asociados al mismo.

El diseño concebido se basó en los límites impuestos por la sociedad nacional de mecatrónica para sus competiciones y no rebasa los límites de dicho reglamento.

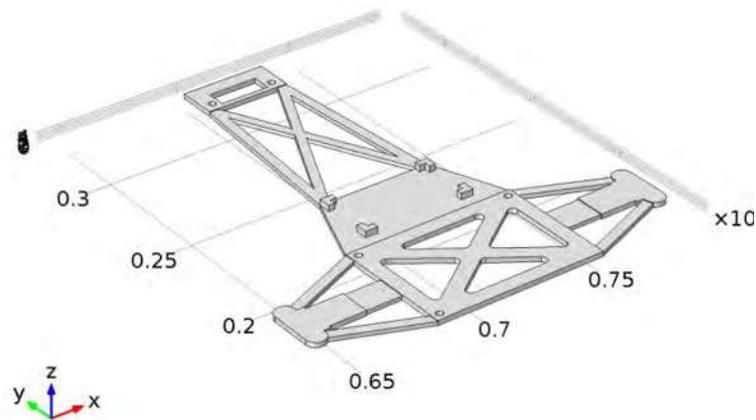


Figura 59 Chasis diseñado para robot seguidor de línea con fines de investigación

Características del diseño:

- El diseño fue pensado para llevarse a cabo a través de impresiones 3D y usando material PLA.
- Sus dimensiones son de 19*13*0.65cm, por lo que cumple con los requisitos de diseño.
- Cuenta con un compartimiento para la batería
- Cuenta con un espacio para el circuito impreso que incluye barrenos de sujeción pensados para tornillos M3.
- Está diseñado de manera que el centro de masa se acerque más a la línea en la que se desempeñan los ejes motrices. Esto para reducir el efecto de fuerza centrífuga en las curvas.



- Cuenta con un compartimiento para sensores reflexivos, el cual no es exclusivo para el sensor QTR-3 A, sino que mantiene dimensiones más grandes para poder ajustar otro tipo de arreglos.
- Mantiene muchos huecos entre su estructura, de esta manera se baja el peso considerablemente.
- Los huecos en su estructura están pensados de manera que los mismos sigan el contorno y las formas del diseño de la placa, dejando líneas que refuercen la estructura dentro de los contornos

4.2.1. Comprobación de la resistencia del chasis

Usando software **COMSOL** se probó que el chasis propuesto con las cualidades de material del PLA, de tal manera que se pudiera demostrar que dicho chasis puede resistir las cargas a las cuales será sometido sin presentar esfuerzos que sobrepasen las capacidades del material.

La geometría fue importada a través de un archivo de objetos 3D en formato. iges.

Las propiedades de material asignadas fueron las siguientes:

Tabla 17 Propiedades del PLA empleadas en la simulación

Nombre	Valor	Unidad
Módulo de Young	3.5e+9	Pa
Coefficiente de Poisson	.35	1
Densidad	1.3	kg/m ³

Se generó una malla con las siguientes características para llevar a cabo el cálculo

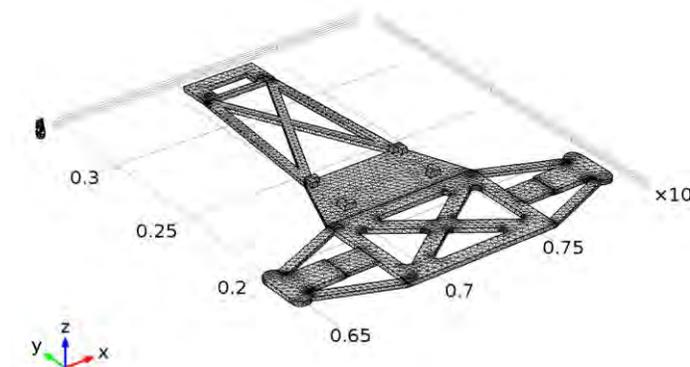


Figura 60 Aplicación de la malla sobre el modelo sólido



Las características de la malla generada son las siguientes:

Tabla 18 Propiedades de la malla usada para la simulación

Nombre	Valor
Tamaño máximo de elemento	0.0138
Tamaño mínimo de elemento	0.00172
Factor de curvatura	0.5
Resolución de regiones estrechas	0.6
Tasa de crecimiento máxima de elemento	1.45
Tamaño predefinido	Fina

Para comprobar la resistencia del chasis de aplico una fuerza mayor a la suma del peso de los componentes, el cual no rebasa los 100 gramos. La carga decidida fue de un Newton, y se aplicó de manera puntual sobre el centro de gravedad del chasis.

El resultado de la simulación arrojó los siguientes resultados de desplazamiento y esfuerzos.

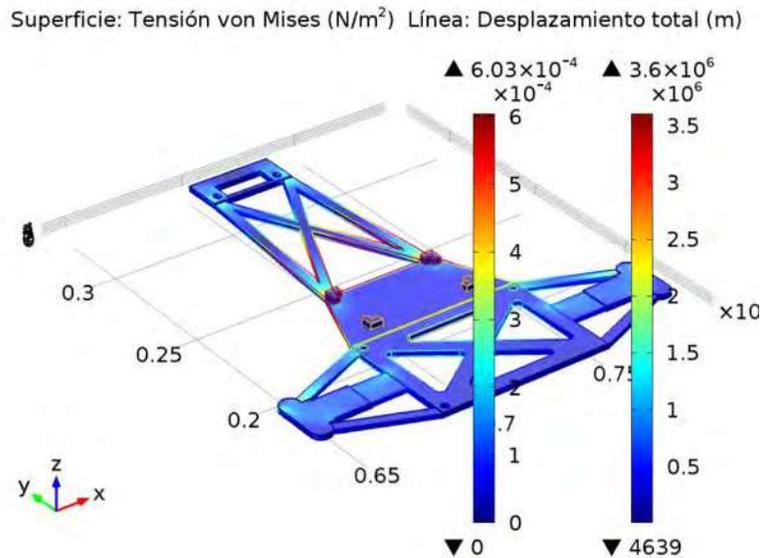


Figura 61 Resultados de la simulación

Se puede observar de acuerdo a la imagen producida que los esfuerzos máximos se encuentran en los vértices más agudo y no superan el valor de $3.6E6 \text{ N/m}^2$. Mientras que el desplazamiento no supera el valor 0.6 mm . Estos resultados resultan ser favorables, ya que demuestran que la integridad del chasis persistirá la aplicación de las cargas que representa la adición de los componentes.

4.2.2 Otras cualidades del diseño mecánico

En una impresora 3D del tipo PRUSA I3, el tiempo para producir esta pieza, de acuerdo al software CURA, sería de un aproximado de 3 horas a una calidad media de parámetros preseleccionados. Su elaboración consumiría casi 10 metros de filamento distribuido en un total de 32 capas de 0.2 mm . Estas capas incluyen un soporte sobre el que se crea la pieza, el cual evita que se genere directamente sobre la plancha de la impresora 3D, evitando que el diseño final guarde irregularidades de la superficie de la plancha.



Otros elementos que conforman el diseño mecánico es la selección de los elementos de sujeción, para los cuales se pretende usar tornillo de M3. Estos son un estándar y todas las piezas que se van a usar de carácter electrónico llevan este tipo de barreno para ubicarlas.

Uno de los elementos que hace falta mencionar es la inclusión de soportes que fijen los motores al chasis. Estos soportes se compran a medida o se pueden reproducir mediante impresión. Las dimensiones de los mismos ya fueron consideradas a la hora de realizar el diseño del chasis.



Figura 62 Soportes de motor que se usarán en el diseño.

4.3 Diseño de módulos virtuales

Los módulos virtuales son el subconjunto de rutinas que permiten al robot canalizar y tomar medidas de los sensores que lo acompañan. Estas subrutinas son las que también controlan y dan movimiento al robot. Teniendo entonces, las dedicadas al control motriz y las de captación de datos y comunicación. Estos módulos están programados en el lenguaje de Arduino.

4.3.1 Diseño de módulo de movimiento de ruedas

Este módulo es una subrutina empleada para comprobar el funcionamiento de los motores y su sentido de giro. De igual manera pone a prueba el puente H y la manera como se opera.

```

/*
 * El programa de test de movimiento de ruedas sirve
 * para comprobar que el driver de motores funciona
 * y para saber que las ruedas se mueven en el sentido
 * correcto.
 * Si alguna de las ruedas se mueve en el sentido
 * equivocado hay que indicarle al constructor de robot
 * la(s) rueda(s) que gira(n) en sentido contrario
 *
 * Si la rueda izquierda gira en sentido contrario:
 * Velocista robot = obtenerRobot(true, false);

```



```

*
* Si la rueda derecha gira en sentido contrario:
* Velocista robot = obtenerRobot(false, true);
*
* Si ambas ruedas giran en sentido contrario:
* Velocista robot = obtenerRobot(true, true);
*/

void setup()
{
}

void loop()
{
  /* Con led rojo encendido las ruedas se mueven hacia atras */
  robot.establecerVelocidad(-100,-100);
  robot.encenderLed(ROJO);
  robot.apagarLed(VERDE);
  delay(2000);
  /* Con leds apagados las ruedas no se mueven */
  robot.establecerVelocidad(0,0);
  robot.apagarLed(VERDE);
  robot.apagarLed(ROJO);
  delay(2000);
  /* Con led verde encendido las ruedas se mueven hacia delante */
  robot.establecerVelocidad(100,100);
  robot.encenderLed(VERDE);
  robot.apagarLed(ROJO);
  delay(2000);
  /* Con leds apagados las ruedas no se mueven */
  robot.establecerVelocidad(0,0);
  robot.apagarLed(VERDE);
  robot.apagarLed(ROJO);
  delay(2000);
}

```

En el desarrollo de este módulo se tomaron en cuenta dos leds, haciendo referencia al led que se incluye en el circuito impreso y al miniled que se ubica en el circuito integrado B. Orangutan, al cual se le puede tener acceso al interactuar con un pin especial, siendo esta una de sus adiciones útiles.

4.3.2 Diseño de módulo de Lectura de encoders

Es el modulo encargado de tomar datos a partir del codificador rotativo.

```

volatile unsigned int cuentaEncoderIzquierdo = 0; //!< Guarda la cuenta del
encoder izquierdo
volatile unsigned int cuentaEncoderDerecho = 0; //!< Guarda la cuenta del encoder
derecho
/**
 * Función que aumenta la cuenta del encoder izquierdo
 */

```



```

void aumentarCuentaIzquierda()
{
    cuentaEncoderIzquierdo++;
}
/**
 *Función que aumenta la cuenta del encoder derecho
 */
void aumentarCuentaDerecha()
{
    cuentaEncoderDerecho++;
}

```

Este módulo funciona mediante la capacidad de algunos pines para medir interrupciones de tipo digital, el controlador tiene pines que cuentan con esta característica. Arduino recomienda usar una variable entera(int) de tipo volátil para esta función, ya que es incremento. También hay que hacer mención de que este programa únicamente se usa una de las señales por cada encoder, dejando abierta la posibilidad a usar dos señales por encoder en futuras investigaciones.

4.3.3 Diseño de módulo de seguimiento de línea, control PID

Módulo de control que otorga al sistema las cualidades de seguidor de línea, de manera que no es necesario introducir funciones de velocidad, pero sí valores de velocidad tope. Estos valores generalmente se toman de acuerdo a la experimentación, tomando en cuenta el desempeño del sistema en velocidades altas y su límite de velocidad antes de experimentar deslizamiento aparente en las ruedas.

```

void Velocista::establecerVelocidad(int velocidadIzquierda, int
velocidadDerecha) const {
if (velocidadIzquierda >= 0) {
digitalWrite(Velocista::M_IZQ_DIR_PIN, motorIzquierdoAvanza_);
} else {
digitalWrite(Velocista::M_IZQ_DIR_PIN, !motorIzquierdoAvanza_);
}
//Ajuste de la velocidad máxima y mínima para el motor izquierdo
if (velocidadIzquierda > 255) {
velocidadIzquierda = 255;
} else if (velocidadIzquierda < -255) {
velocidadIzquierda = -255;
}
analogWrite(Velocista::M_IZQ_PWM_PIN, abs(velocidadIzquierda));

if (velocidadDerecha >= 0) {
digitalWrite(Velocista::M_DER_DIR_PIN, motorDerechoAvanza_);
} else {
digitalWrite(Velocista::M_DER_DIR_PIN, !motorDerechoAvanza_);
}
//Ajuste de la velocidad máxima y mínima para el motor derecho
if (velocidadDerecha > 255) {
velocidadDerecha = 255;
} else if (velocidadDerecha < -255) {

```



```
velocidadDerecha = -255;
}
analogWrite(VelocistaEdu::M_DER_PWM_PIN, abs(velocidadDerecha));
}
```

Control PD

```
int error = 0;
int errorAnterior = 0;

/* Ajustar kp y kd para que el robot no oscile */
int kp = x;
int kd = y;

/* Aumentar pwmParteComun para que el robot vaya más rápido */
int pwmParteComun = 25;
int pwmParteDiferencial = 0;

bool estoyEncendido = false;

Velocista robot = obtenerRobot();

void setup()
{
}

void loop()
{
/* Aun no usamos la cuenta de los encoders */
/*
int encoderDerecho, encoderIzquierdo;
robot.obtenerCuentaEncoders(encoderIzquierdo, encoderDerecho);
*/

/* Comprobar boton de encendido */
if (robot.botonLiberado())
{
estoyEncendido = !estoyEncendido;
}

/* Obtener el error para el PD */
robot.actualizarSensoresLinea();
error = robot.obtenerMedidaLinea();

/* Ejecutar el algoritmo de PD */
/*
* El pwm puede tomar valores enteros entre -255 y 255
* Teniendo en cuenta que el error absoluto máximo es de x,
* tendremos que dividir por lo menos entre 12 para tener una parte angular
razonable
* Además queremos tener algo de resolución para las Ks
* Vamos a poner una Kp en el orden de las centenas, con lo que tendremos que
aumentar en *100 el divisor
```



```

* Todo esto es un ejemplo. Hay que ajustar mejor estas operaciones
*/
pwmParteDiferencial = (kp * error + kd * (error - errorAnterior))/12;
errorAnterior = error;

/* Establecer velocidad en los motores */
if (estoyEncendido)
{
robot.establecerVelocidad(pwmParteComun - pwmParteDiferencial, pwmParteComun +
pwmParteDiferencial);
}
/* Si tenemos que estar parados ponemos la velocidad a 0 */
else
{
robot.establecerVelocidad(0, 0);
}
}

```

Los valores kp y kd son valores constantes y enteros, estos valores se incluirán una vez concebido el prototipo físico, ya que su forma de obtenerlos es a través de la experimentación. Una correcta selección de estos valores nos da como resultado un robot que sigue la línea sin despegarse casi nada de ella, corrigiendo de manera constante su trayectoria y sin mostrar oscilamiento para buscarla.

4.3.4 Diseño de módulo de interacción con los sensores reflexivos

Como se había comentado, los sensores seleccionados son de tipo analógico, por lo que se clasificara la señal y dividirá para que el robot pueda identificarla, para esto se programó una subrutina de calibración, la cual utiliza librerías propias del fabricante para relacionar los sensores con la placa B. Orangutan.

Las librerías son un conjunto de programas que nos facilitan el diseño de un software al comprimir información relevante en paquetes programados que pueden ser llamados con instrucciones sencillas.

```

#include <QTRSensors.h>
#include <OrangutanMotors.h>
QTRSensorsAnalog qtra((unsigned char[]) {0, 1, 2, 3, 4, 5},
    NUM_SENSORS, NUM_SAMPLES_PER_SENSOR, EMITTER_PIN);
unsigned int sensorValues[NUM_SENSORS];
int velcalibrate = 20;
OrangutanMotors motors;
digitalWrite(ledRojo, HIGH);
digitalWrite(ledVerde, HIGH); // Enciende el leds para indicar que se está
calibrando.
for (int counter=0; counter<21; counter++)
{
if (counter < 6 || counter >= 15)
OrangutanMotors::setSpeeds(-velcalibrate, velcalibrate);
else

```



```

OrangutanMotors::setSpeeds(velcalibrate, -velcalibrate);
qtra.calibrate();

delay(20);
}
digitalWrite(ledrojo, LOW);    // Apaga el led para indicar que se termino la
calibracion.
digitalWrite(ledverde, LOW);
OrangutanMotors::setSpeeds(0, 0);
delay(200);
digitalWrite(ledrojo, HIGH);
digitalWrite(ledverde, HIGH);
delay(200);

```

Los valores en esta subrutina son los sugeridos por el fabricante, por lo que este apartado queda sujeto a las especificaciones que el proveedor recomienda, para nuestro caso la marca POLOLU.

4.3.5. Diseño de módulo de interacción con el MPU-6050

El MPU-6050 dará unos raw values (valores en bruto) que después habrá que refinar (dividir por una constante) con el fin de conseguir un valor útil.

La plataforma Arduino dispone de un ejemplo para trabajar este módulo. Los valores asociados a este sensor se declaran de la siguiente manera:

```

#include <Wire.h>
#define MPU 0x68
//Ratios de conversion
#define A_R 16384.0
#define G_R 131.0
//Conversion de radianes a grados 180/PI
#define RAD_A_DEG = 57.295779

//MPU-6050 da los valores en enteros de 16 bits
//Valores sin refinar
int16_t AcX, AcY, AcZ, GyX, GyY, GyZ;

//Angulos
float Acc[2];
float Gy[2];
float Angle[2];

```

La librería Wire.h es necesaria para la interacción vía protocolo I2C.

Las ratios de conversión son los especificados en la documentación. Debemos dividir los valores que nos dé el Giroscopio y el Acelerómetro entre estas constantes para obtener un valor coherente. RAD_A_DEG es la conversión de radianes a grados.



La IMU da los valores en enteros de 16 bits. Como Arduino los guarda en menos bits, hay que declarar las variables que almacenarán los enteros provenientes de la IMU como un tipo de enteros especiales. `int16_t AcX, AcY, AcZ, GyX, GyY` son, pues, los `raw_values` de la IMU.

Finalmente tenemos tres arrays (`Acc[]`, `Gy[]`, `Angle[]`) que guardan el ángulo X, Y del Acelerómetro, el Giroscopio y el resultado del Filtro respectivamente. `[0]` se corresponde a X. `[1]` a Y.

La función `setup` quedaría de la siguiente manera:

```
void setup()
{
  Wire.begin();
  Wire.beginTransmission(MPU);
  Wire.write(0x6B);
  Wire.write(0);
  Wire.endTransmission(true);
  Serial.begin(9600);
}
```

Se inicia la comunicación por I2C con el dispositivo MPU, y se “activa” enviando el comando 0. También se inicia el puerto de serie para ver los resultados.

El `void loop` es un poco más complejo. Se leen y guardan los datos del sensor, se calcula el ángulo y se aplica el filtro complementario. Para simplificar el código, he supuesto que Δt es 0.01. Pero si se requiere ser más riguroso, se deberá calcular el tiempo con la función `millis()` de Arduino.

```
void loop()
{
  //Leer los valores del Acelerometro de la IMU
  Wire.beginTransmission(MPU);
  Wire.write(0x3B); //Pedir el registro 0x3B - corresponde al AcX
  Wire.endTransmission(false);
  Wire.requestFrom(MPU,6,true); //A partir del 0x3B, se piden 6 registros
  AcX=Wire.read()<<8|Wire.read(); //Cada valor ocupa 2 registros
  AcY=Wire.read()<<8|Wire.read();
  AcZ=Wire.read()<<8|Wire.read();

  //A partir de los valores del acelerómetro, se calculan los ángulos Y, X
  //respectivamente, con la fórmula de la tangente.
  Acc[1] = atan(-1*(AcX/A_R)/sqrt(pow((AcY/A_R),2)
  + pow((AcZ/A_R),2)))*RAD_TO_DEG;
  Acc[0] = atan((AcY/A_R)/sqrt(pow((AcX/A_R),2) + pow((AcZ/A_R),2)))*RAD_TO_DEG;

  //Leer los valores del Giroscopio
  Wire.beginTransmission(MPU);
  Wire.write(0x43);
  Wire.endTransmission(false);
  Wire.requestFrom(MPU,4,true); //A diferencia del Acelerometro, solo se piden 4
  registros
```



```

GyX=Wire.read()<<8|Wire.read();
GyY=Wire.read()<<8|Wire.read();

//Calculo del angulo del Giroscopio
Gy[0] = GyX/G_R;
Gy[1] = GyY/G_R;

//Aplicar el Filtro Complementario
Angle[0] = 0.98 *(Angle[0]+Gy[0]*0.010) + 0.02*Acc[0];
Angle[1] = 0.98 *(Angle[1]+Gy[1]*0.010) + 0.02*Acc[1];

//Mostrar los valores por consola
Serial.print("Angle X: "); Serial.print(Angle[0]); Serial.print("\n");
Serial.print("Angle Y: "); Serial.print(Angle[1]); Serial.print("\n-----
\n");

delay(10); //Nuestra dt sera, pues, 0.010, que es el intervalo de tiempo en
cada medida
}

```

4.3.6. Diseño de módulo de interacción con modulo bluetooth.

Para esta configuración es necesario que se alimente con 3.3V y GND no 5V. Para hacer la configuración KEY se ocupa un estado ALTO. Para elaborar esta configuración es necesario recurrir a la librería <SoftwareSerial.h>, con la cual podemos colorar a Tx y Rx en donde sea necesario. Recordando que Tx y Rx son los pines de comunicación serial que están incluidos en el controlador y en el módulo correspondiente a la telemetría.

Configuración de módulo HC-05 y comando asociados:

Regresar a opción Fabrica	AT+RESET
Saber el nombre del módulo:	AT+NAME?
Cambiarlo a “MINOMBRE”:	AT+NAME=MINOMBRE
Saber configuración:	AT+ROLE?
Cambiarlo al modo esclavo:	AT+ROLE=0
Cambiarlo al modo maestro:	AT+ROLE=1
Saber la contraseña:	AT+PSWD?
Cambiarlo a 0000: (max 4 char.)	AT+PSWD=1234---según la Contraseña a Conectar
Saber la velocidad (9600):	AT+UART?
Cambiarlo a 57600:	AT+UART=57600,0,0
Saber modo de Conexión:	AT+CMODE? (0 Especifico 1 General)



A cualquier dispositivo: AT+CMODE=1

Interacción por medio de Arduino:

```
#include <SoftwareSerial.h>
#define RxD 11
#define TxD 10
#define KEY 9
SoftwareSerial BTSerial(RxD, TxD);
void setup()
{
  pinMode(KEY, OUTPUT);
  digitalWrite(KEY, HIGH); // Como se mencionó en configuración colocar a KEY
  en estado alto.
  delay(500);
  BTSerial.flush();
  delay(500);
  BTSerial.begin(38400); // comunicación directa al Modulo.
  Serial.begin(9600); // comunicación directa al Monitor.
  Serial.println("Enter AT commands:");
  BTSerial.print("AT\r\n");
  delay(100);
}
void loop()
{
  if (BTSerial.available())
  Serial.write(BTSerial.read());
  if (Serial.available())
  BTSerial.write(Serial.read());
}
```

Bluetooth configuración maestro:

```
/*
Programa Modulo Bluetooth-HOST
El siguiente programa enviara información mediante el modulo bluetooth.
Dado que el programa del módulo Bluetooth SLAVE
espera que se le envíe '1' y '0' cada 2 segundos
así encienda y apague el LED
*/
#include <SoftwareSerial.h>
int sensorPin = A0; // select the input pin for the potentiometer
int sensorValue = 0; // variable to store the value coming from the sensor
// se define al Pin2 como RX, Pin3 como TX
SoftwareSerial mySerial(2,3);// RX,TX
void setup()
{ //inicializa la comunicación serial
  Serial.begin(9600);
  mySerial.begin(9600);
  Serial.println("Ready");
  mySerial.print("Ready");
}
void loop()
```



```
{ int U = analogRead(sensorPin); // escala el valor para usarlo con el servo
(valor entre 0 y 180)
mySerial.print(U); mySerial.print("."); Serial.println(U);
delay(15);
}
```

4.4 Recuento de los componentes seleccionados en el diseño.

Los materiales seleccionados para el diseño fueron los siguientes:

- Chasis impreso en 3D usando PLA.
- Motores 5:1 Micro Metal Gearmotor HPCB.
- Ruedas SC-2008.
- Sensor MPU-6050.
- Sensor QTR-3 A.
- Módulos de telemetría HC-05 y HC-06.
- Batería Lipo nano-tech 300 Ma 1S 3.7 V.
- Subidor de voltaje U3V50F6

La mayoría de estos elementos se adquirieron para poder hacer las pruebas y diseñar los módulos virtuales, sin embargo, no se contó con todos los elementos aquí enlistados para poder culminar la construcción.



Capítulo 5. Conclusiones y trabajos futuros



5.1 Conclusiones

El prototipo de robot móvil que se diseñó presenta un sistema de localización sencillo el cual sirva para experimentar con sensores y elementos de localización de tipo propioceptivo. La estructura del robot móvil es de un espacio tal que puede albergar en su superficie los componentes electrónicos y eléctricos, y con ayuda de exploración en algoritmos de control PD o PID se puede dar uso al sistema como robot de competición. El uso de esta arquitectura permite explorar una variedad de movimientos más amplia que en plataformas robóticas, ya que la dirección está sugerida a partir de la velocidad relativa entre las ruedas motrices. De acuerdo al análisis de movilidad se demostró que una técnica simple de localización puede partir de un análisis cinemático y relacionarlo con las cualidades de un codificador rotativo. Emplear solo dos motores y dos ruedas motrices independientes facilita la construcción del sistema y facilita su control. La arquitectura sugerida otorga la libertad de poder hacer cambios de orientación en el sistema sin producir cambios de desplazamiento.

Los elementos de telemetría escogidos son sencillos y fáciles de implementar, incluso pueden funcionar para tareas de teleoperación y a su vez, pueden ser relacionados con dispositivos móviles, tales como teléfonos celulares, de manera que se pueda visualizar la información captada por el móvil en equipos que no son propiamente de computo. Las características de comunicación que existen entre el robot y el usuario para trabajar en distanciamiento son un acercamiento importante a la hora de pensar en cualidades de autonomía de mayor complejidad, como la implementación de un sistema de navegación.

El módulo virtual diseñado puede ser usados de manera conjunta o independiente para la construcción y programación de robots móviles que recurran los elementos seleccionados en esta investigación. Al dar al sistema cualidades de robot seguidor de línea, prescindimos de la tarea de programar funciones de velocidad angular, ya que éstas mismas son instruidos por el control PD que mantiene al robot sobre la línea trazada. Para cualquier caso de localización basado en sensores propioceptivos, la localización dependerá de las variables manejadas por los sensores; en el caso de los codificadores rotativos, la localización constara del número de pulsos registrados por rueda, independiente a la velocidad con la que se registraron.

Las ecuaciones desprendidas del análisis cinemático pueden aplicarse en un sistema de procesamiento, por ejemplo, por medio de Matlab, para generar información de localización haciendo uso de los datos enviados por el sistema a través de la telemetría y los codificadores rotatorios.

El trabajo realizado puede contener áreas de oportunidad que pueden ser trabajadas a futuro, por otra parte, establece bases firmes para la realización de trabajos en la misma rama de estudio, que pueden ser enfocados a una aplicación de mayor ambición.

El costo de producción del sistema diseñado puede variar entre un rango de \$3000 MXN y \$5000 MXN, debido a que los componentes que constituyen el diseño son en su mayoría de origen extranjero, y la moneda que los precede tiene un valor cambiante con respecto al peso mexicano.



5.2 Trabajos futuros

Los conocimientos aquí presentados y los conocimientos adquiridos durante la investigación se pretenden retomar para continuar en un trabajo de investigación en el doctorado.

En la tesis doctoral que pretende continuar a ésta investigación, y se pretende lograr los siguientes objetivos:

1. Cualificar off-line los distintos mecanismos de fusión y filtrado de datos.
2. Desarrollar modelos sensoriales, así como un código para la simulación de la operación de estos sensores típicos empleados en robótica móvil para en un robot real.
3. Elaborar procedimientos y software que se usarán en un robot real para que este ejecute las acciones de navegación, de adquisición y filtrado de datos sensoriales, y la fusión pertinente a la localización y el mapeo en su entorno de operación.
4. Desarrollar modelos de sensores que permitan integrar los mismos a sistemas más avanzados de fusión sensorial, que sirvan para construir mapas de densidad de ocupación y de características geométricas del entorno.
5. Desarrollar al menos dos modelos que describan el sistemas de localización, uno mediante el procedimiento de seguimiento y otro integrando medidas de sensores de distancia, fundamentado en un filtro de Kalman extendido y corregir la posición del robot utilizando dos tipos de hitos naturales (columnas, paredes).
6. Aplicar los sistemas a una plataforma robótica multisensorial que cuente con un sistema impulsos de tracción diferencial.
7. Comprobar y evaluar los métodos propuestos, asegurándose que ambos puedan aplicarse en tiempo real, utilizando la plataforma diseñada y observando su comportamiento.
8. Determinar la factibilidad y ventajas prácticas de uno sobre otro en base a las diferencias de precisión y el alcance de los puntos objetivos.

La intención de los trabajos futuros es abordar temáticas de fusión sensorial e integrarlas en sistemas de localización más avanzados que combinen características propioceptivas y exteroceptivos.



Bibliografía

- [1] A. Ollero, “Robótica, manipuladores y robots móviles,” *Marcombo*. 2001.
- [2] G. Dudek and M. Jenkin, *Computational Principles of Mobile Robotics*, Second edi. Cambridge University Press, 2009.
- [3] J. Borenstein, H. R. Everett, and L. Feng, “Where am I? Sensors and methods for mobile robot positioning,” *Univ. Michigan*, vol. 119, 1996.
- [4] F. O. R. Mobile and R. Everett, *Sensors For Mobile Robots book*. .
- [5] R. W. Hogg, a. L. Rankin, S. I. Roumeliotis, M. C. McHenry, D. M. Helmick, C. F. Bergh, and L. Matthies, “Algorithms and sensors for small robot path following,” *Proc. 2002 IEEE Int. Conf. Robot. Autom. (Cat. No.02CH37292)*, vol. 4, no. May, pp. 3850–3857, 2002.
- [6] M. J. Caruso, “Applications of magnetic sensors for low cost compass systems,” *IEEE 2000. Position Locat. Navig. Symp. (Cat. No.00CH37062)*, pp. 1–8, 2000.
- [7] H. Jifei, L. Xiang, and Y. Dashun, “A Multisensor Fusing System on Ultrasonic Sensors,” pp. 140–143.
- [8] B. Olshausen, “Bayesian probability theory,” *Redw. Cent. Theor. ...*, no. November, pp. 1–6, 2004.
- [9] I. Bloch, “ation Combination Operators for Data Comparative Review with Classi,” *IEEE Trans. Syst. Man*, vol. 26, no. 1, pp. 52–67, 1996.
- [10] S. Thrun, “Probabilistic Robotics,” pp. 1999–2000, 2000.
- [11] J. Kohlas, P. Monney, and B. Functions, “Theory of Evidence - A Survey of its Mathematical Foundations , Applications and Computational Aspects Key Words 2 . The Theory of Evidence - Di ering Approaches and Interpretations,” no. 21, pp. 1–29.
- [12] J. J. Leonard and H. F. Durrant-Whyte, “Mobile robot localization by tracking geometric beacons,” *IEEE Transactions on Robotics and Automation*, vol. 7. 1991.
- [13] J. Borenstein, “UMBmark: a benchmark test for measuring odometry errors in mobile robots,” *Proc. SPIE*, vol. 2591, 1995.
- [14] E. Polit, “Sistema de Posicionamiento y Autolocalización para Sillas de Ruedas Autónomas,” p. 298, 2001.
- [15] C. C. And, C. Cohen, and F. V Koss, “A Comprehensive Study of Three Object Triangulation,” *Proc. 1993 Spie Conf. Mob. Robot.*, vol. 1831, 1992.
- [16] O. Wijk and H. Christensen, “Triangulation-based fusion of sonar data with application in robot pose tracking,” *Robot. Autom. IEEE ...*, vol. 16, no. 6, 2000.



- [17] M. Betke and L. Gurvits, "Mobile robot localization using landmarks," *IEEE Trans. Robot. Autom.*, vol. 13, no. 2, 1997.
- [18] H. Moravec and a. Elfes, "High resolution maps from wide angle sonar," *Proceedings. 1985 IEEE Int. Conf. Robot. Autom.*, vol. 2, 1985.
- [19] B. Schiele and J. L. Crowley, "A comparison of position estimation techniques using occupancygrids," *Proc. 1994 IEEE Int. Conf. Robot. Autom.*, 1994.
- [20] K. O. Arras, N. Tomatis, B. T. Jensen, and R. Siegwart, "Multisensor on-the-fly localization: Precision and reliability for applications," *Rob. Auton. Syst.*, vol. 34, 2001.
- [21] J. A. Meyer and D. Filliat, "Map-based navigation in mobile robots: II. A review of map-learning and path-planning strategies," *Cogn. Syst. Res.*, vol. 4, no. 4, pp. 283–317, 2003.
- [22] G. A. Borges and £. M.-J. Aldon, "A Split-and-Merge Segmentation Algorithm for Line Extraction in 2-D Range Images," pp. 0–3.
- [23] K. O. Arras, "Feature-Based Robot Navigation in Known and Unknown Environments," vol. 2765, 2003.
- [24] T. Duckett and U. Nehmzow, "Mobile robot self-localization using occupancy histograms and a mixture of Gaussian location hypotheses," *Rob. Auton. Syst.*, vol. 34, no. 2–3, pp. 117–129, 2001.
- [25] G. Weiß, C. Wetzler, and E. von Puttkamer, "Keeping Track of Position and Orientation of Moving Indoor Systems by Correlation of Range-Finder Scans," *Proc.~of IEEE/RSJ Int.~Conf.~on Intell. Robot. Syst.*, pp. 595–601, 1994.
- [26] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2D range scans," *IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 249–275, 1994.
- [27] T. Einsele, "Localization in indoor environments using a panoramic laser range finder," 2002.
- [28] J. R. de G. Pascual, "Robótica: Estado del arte," *Univ. Deusto*, 2010.
- [29] A. Y. Experiencias and G. Bermúdez, "Robots Móviles. Teoría, Aplicaciones Y Experiencias," *Tecnura 10*, vol. 5, no. December, pp. 6–17, 2002.
- [30] R. Siegwart and I. R. Nourbakhsh, *Introduction to Autonomous Mobile Robots*, vol. 23. 2004.
- [31] B.- Nisbett, "Mechanical Engineering," pp. 860–900, 2006.
- [32] T. Abbas, M. Arif, and W. Ahmed, "Measurement and correction of systematic odometry errors caused by kinematics imperfections in mobile robots," *2006 SICE-ICASE Int. Jt. Conf.*, vol. 12, no. 6, pp. 2073–2078, 2006.
- [33] J. Borenstein and L. F. L. Feng, "Correction of systematic odometry errors in mobile



robots,” *Proc. 1995 IEEE/RSJ Int. Conf. Intell. Robot. Syst. Hum. Robot Interact. Coop. Robot.*, vol. 3, pp. 569–574, 1995.

- [34] L. Serna C., A. Rodríguez de S., and F. Albán A., “Ácido Poliláctico (PLA): Propiedades y Aplicaciones,” *Rev. Ing. y Compet.*, vol. 5, no. 1, pp. 16–26, 2011.
- [35] Toshiba, “Driver for Dual DC motor TB6612FNG,” p. 11, 2007.
- [36] M. D. E. La and P. Con, “Desplazamiento y posición encoders 1.-,” pp. 1–10.
- [37] C. Fernandez, “Técnicas de navegación de robots basadas en sistemas de medición por laser,” Universidad De Salamanca, 2007.