



UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE GUAYAQUIL
CARRERA DE MECATRÓNICA

**DESARROLLO DE UN ROBOT AUTÓNOMO CON SISTEMA DE
MAPEO Y DETECCIÓN DE OBSTÁCULOS MEDIANTE SENSOR
LIDAR**

Trabajo de titulación previo a la obtención del
Título de Ingeniero en Mecatrónica

AUTORES: Isaac Andre García Ulloa
Washington Joel Nieves Guerrero
TUTOR: Ing. Tomás Santiago Gavilánez Gamboa, MSc.

Guayaquil - Ecuador
2024

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Nosotros, **Isaac Andre García Ulloa** con documento de identificación N° **0931474688** y **Washington Joel Nieves Guerrero** con documento de identificación N° **0803769603**; manifestamos que:

Somos los autores y responsables del presente trabajo; y, autorizamos a que sin fines de lucro la Universidad Politécnica Salesiana pueda usar, difundir, reproducir o publicar de manera total o parcial el presente trabajo.

Guayaquil, 7 de marzo del año 2024

Atentamente,



Isaac Andre García Ulloa
0931474688



Washington Joel Nieves Guerrero
0803769603

**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE TITULACIÓN A LA
UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, **Isaac Andre García Ulloa** con documento de identificación N° **0931474688** y **Washington Joel Nieves Guerrero** con documento de identificación N° **0803769603**, expresamos nuestra voluntad y por medio del presente documento cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del **Dispositivo Tecnológico: DESARROLLO DE UN ROBOT AUTÓNOMO CON SISTEMA DE MAPEO Y DETECCIÓN DE OBSTÁCULOS MEDIANTE SENSOR LIDAR**, el cual ha sido desarrollado para optar por el título de: Ingeniero en Mecatrónica, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo a final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana

Guayaquil, 7 de marzo del año 2024

Atentamente,



Isaac Andre García Ulloa
0931474688



Washington Joel Nieves Guerrero
0803769603

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, **Tómas Santiago Gavilánez Gamboa**, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **DESARROLLO DE UN ROBOT AUTÓNOMO CON SISTEMA DE MAPEO Y DETECCIÓN DE OBSTÁCULOS MEDIANTE SENSOR LIDAR**, realizado por **Isaac Andre García Ulloa** con documento de identificación N° **0931474688** y por **Washington Joel Nieves Guerrero** con documento de identificación N° **0803769603**, obteniendo como resultado final el trabajo de titulación bajo la opción **Dispositivo Tecnológico** que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 7 de marzo del año 2024

Atentamente,



Ing. **Tómas Santiago Gavilánez Gamboa**, MSc.

1802792646

DEDICATORIA

A mis padres, por su apoyo incondicional y creer en mí desde el primer día, quienes se alegraban de mis logros y supieron darme palabras de aliento cuando más las necesitaba, espero que me alcance la vida para agradecerles todo lo que hicieron por mí y por todo lo que han dado. A mi papá por demostrarme que nunca es tarde para empezar algo nuevo como algún deporte como el boxeo o algún emparedamiento como abrir un local de comida. A mi mamá por enseñarme a ser fuerte ante las adversidades y siempre seguir adelante. No puedo estar más feliz y orgulloso de los padres que tengo. A mi abuela por apoyarme durante no solo toda la carrera sino durante toda la vida, por su amor incondicional y por aconsejarme en cada oportunidad para que sea una persona de bien y pueda ser mejor cada día, por creer en mi potencial y en que puedo llegar muy lejos, por esto y mucho más le dedico esta tesis. A Catalina Ulloa, quien fue un pilar fundamental en mi vida, y a quien le tengo que agradecer por la persona que soy ahora, por siempre preocuparse por mí sin tener ninguna obligación, por estar siempre pendiente de mí, por preguntarme si ya comí, si ya llegué a la casa, si me va bien en la universidad, si tengo para el transporte. Espero que estés orgullosa de mí y que sepas que no hay día que no te extrañe. Sé que estarías junto a mí disfrutando este logro. A mi tía Malena Terán por siempre apoyarme, tenerme paciencia, recibirme con un plato de comida al llegar a casa y brindarme un amor de madre. A mis primos Andrés, Mathias y Urias Ulloa, con los cuales crecí, por ser como mis hermanos mayores y por apoyarme siempre de manera incondicional. A mi tío Mathias Ulloa, quien es mi ejemplo a seguir, por mostrarme lo que es un amor incondicional, por tratarme como a un hijo más, por siempre apoyarme en todo y no dejar que me faltara nada. Espero algún día llegar a ser el hombre que fuiste, sé que estarías orgulloso de mí, todos en la casa te extrañamos cada día. A mis hermanos José, Iker, Grecia, Debray y Helena por darme momentos de felicidad. Espero que en unos años mi hermana Helena, lea este documento y esté orgullosa de su hermano mayor. A Jose por hacerme ver que hay personas que siguen mis pasos y me ven como un ejemplo a seguir. Agradezco a Iker por darme momentos de alegría. A Debray por apoyarme siempre y estar conmigo en los buenos y malos momentos, dándome palabras de aliento y haciéndome creer en mí mismo. A mi hermana mayor, Grecia, por enseñarme a luchar por mis sueños y a no dejarme influenciar por las opiniones de los demás, espero que estés orgullosa de tu LU. A mi primer amigo de la universidad, Elias Jair Quinde Chamaidan, quien llegó a ser como mi hermano, por estar presente en los buenos momentos así como en los no tan buenos, por apoyarme y confiar en mí en toda la carrera universitaria y por presentarme a buenos amigos como a Meli, Tomás, Milena, Isaac, Nataly, entre otros, con los cuales he pasado buenos momentos. A mi tía Melissa Valencia, por apoyarme, por confiar y creer en mí, por enseñarme que la salud mental es igual de importante que la salud física y por ser uno de mis ejemplos a seguir. A los amigos que hice en la universidad, como Miguel Granados, Edwin Culqui, Kevin franco, Melanie Vásquez, Carlos Estrella y muchos más, por apoyarme durante toda la carrera universitaria, por darme momentos felices y por estar conmigo cuando los necesito. A Viviana Villon Vega por siempre apoyarme y estar conmigo en los momentos difíciles y siempre creer en mí.

A Dayra Mosquera y María de los Angeles Estrada por siempre apoyarme y confiar en mí, por darme palabras de aliento cuando más lo necesito y no dejarme caer. Espero que estén orgullosas de mí, así como yo lo estoy de ustedes.

Isaac Andre García Ulloa

Este trabajo de titulación está dedicado a mi padre Washington Nieves Intriago, por ser mi guía e inspiración en este emocionante viaje hacia la excelencia académica. Tu apoyo ha sido el motor que me ha impulsado a superar mis propios límites y a alcanzar nuevas alturas; a mi madre Grecia Guerrero Prado por su inmenso amor, apoyo y cuidado, gracias a sus oraciones que han sido mi roca en los momentos de incertidumbre y desafío.

Finalmente, agradezco a mi tía Fernanda Guerrero que en paz descansa y mi tío Gorki Guerrero por motivarme a ser mejor persona día a día y así mismo superarme; a mi abuela Mirian Prado por su apoyo incondicional y arduo cuidado.

Dedico este trabajo a mi familia, quienes me han enseñado que la educación es lo más valioso que me pudieron ofrecer. Es un camino que nos permite explorar y descubrir nuevos horizontes, donde los anhelos pueden convertirse en realidad.

Washington Joel Nieves Guerrero

AGRADECIMIENTO

Agradezco en primer lugar a Dios por guiarme durante todo este camino, a mis padres por apoyarme siempre, a mi abuela por estar siempre presente, a mis tías y tíos por ser mis pilares, a mis primos por darme ánimos, a mis hermanos por siempre confiar y a mis amigos por nunca dudar de mí. Agradezco infinitamente a la Universidad Politécnica Salesiana y al Ing. Jorge Fariño por haberme dado la oportunidad de representar al país en una competencia mundial, dándome así uno de los momentos más felices de mi vida y del cual me siento orgulloso. Agradezco a mi compañero de tesis Washington Joel Nieves Guerrero por ayudarme a culminar este proyecto, siendo su aporte indispensable y de gran ayuda, no cabe duda de que no podría haberlo terminado sin él, estoy contento de haberlo tenido como compañero de tesis. Deseo agradecer al Ing. Tomás Gavilanez por su guía durante la creación de este proyecto, por sus sugerencias y correcciones, por su orientación y consejos. Estoy muy agradecido por la confianza depositada en mí durante la culminación del proyecto, estoy muy contento de haberlo tenido como mi tutor de tesis, espero que pueda seguir impartiendo sus conocimientos y en algún momento poder llamarlo Colega.

Isaac Andre García Ulloa

Agradezco primero a Dios por ser mi principal guía, a mi madre y padre porque ustedes han sido mi mayor apoyo y fuente de inspiración. Han estado a mi lado en cada paso del camino, brindándome amor incondicional, orientación sabia y aliento constante. Sus sacrificios no han pasado desapercibidos para mí, y estoy eternamente agradecido por todo lo que han hecho por mí. Agradezco a la Ing. Michelle Cárdenas por ser una inspiración constante y una guía a lo largo de este viaje académico. Además, por enseñarme ese valor de perseverancia y formar mi carácter dentro del ámbito académico. Agradezco a mi compañero de tesis Issac Andre García Ulloa por su arduo esfuerzo en la realización de este proyecto, su colaboración y dedicación han sido fundamentales para alcanzar este logro. Gracias por compartir este camino conmigo. Quiero expresar mi profundo agradecimiento a la Universidad Politécnica Salesiana, así como a los profesores que han sido parte fundamental en el desarrollo y cumplimiento de este trabajo de investigación. Además, deseo agradecer de manera especial al Ing. Tomás Gavilánéz por su invaluable guía, sugerencias y confianza depositada en mí durante la realización de este proyecto, lo cual ha sido fundamental para su culminación con éxito.

Washington Joel Nieves Guerrero

RESUMEN

Los robots autónomos móviles han cobrado gran relevancia mundial en la industria asistiendo en tareas peligrosas y ejecución de trabajos brindando ventajas competitivas. El concepto de los robots viene siendo más popular gracias a los avances tecnológicos de la autonomía en robots permitiendo así optimizar tareas dentro de campos como lo son: minería, agricultura, transporte, logística, entre otros. El presente trabajo tiene como objetivo desarrollar un robot autónomo móvil con sistema de mapeo y detección de obstáculos en entornos controlados mediante un sensor LIDAR. Este sistema se implementó mediante un Raspberry PI 4 que controla el robot, ambas máquinas se comunican por medio de una red inalámbrica creada por ROS. Además se implementó el algoritmo SLAM para la creación de mapas y varios métodos de navegación con odometría. El robot fue capaz de generar mapas mediante algoritmos de mapeo mediante el uso del sensor LIDAR generando trayectorias con el algoritmo RRT de exploración con la finalidad de navegar desde un punto A inicial hacia un punto B de llegada de forma autónoma y evitando obstáculos en su trayectoria.

Palabras claves: LIDAR, ROS, SLAM, Gmapping mapping, Hector mapping, Cartographer mapping, RRT, PRM.

ABSTRACT

Autonomous mobile robots have gained worldwide relevance in the industry assisting in dangerous tasks and execution of jobs providing competitive advantages. The concept of robots has become more popular thanks to the technological advances in robot autonomy, allowing to optimize tasks in fields such as mining, agriculture, transportation, logistics, among others. The present work aims to develop a mobile autonomous robot with mapping and obstacle detection system in controlled environments by means of a LIDAR sensor. This system was implemented using a Raspberry PI 4 that controls the robot, both machines communicate through a wireless network created by ROS. In addition, the SLAM algorithm was implemented for the creation of maps and various methods of navigation with odometry. The robot was able to generate maps through mapping algorithms using the LIDAR sensor, generating trajectories with the RRT exploration algorithm in order to navigate from an initial point A to an arrival point B autonomously and avoiding obstacles in its trajectory.

Keywords: LIDAR, ROS, SLAM, Gmapping mapping, Hector mapping, Cartographer mapping, RRT, PRM.

ÍNDICE

| | | |
|-------------|--|----|
| I. | INTRODUCCIÓN | 14 |
| II. | PROBLEMA | 15 |
| III. | OBJETIVOS | 16 |
| III-A. | Objetivo general | 16 |
| III-B. | Objetivos Específicos | 16 |
| III-C. | Tabla de Objetivos | 17 |
| IV. | MARCO TEÓRICO | 18 |
| IV-A. | Introducción a los robots | 18 |
| IV-A1. | Tipos de robots | 18 |
| IV-A2. | Robótica en cifras | 18 |
| IV-B. | Robot autónomo móvil | 20 |
| IV-B1. | Aplicaciones de un robot móvil | 22 |
| IV-B2. | Elementos de un robot móvil | 23 |
| IV-C. | Sistema de navegación autónoma | 25 |
| IV-C1. | Localización y mapeo simultáneo (SLAM) | 25 |
| IV-C2. | Algoritmos de navegación | 26 |
| IV-D. | Control de trayectoria | 28 |
| IV-D1. | Planificadores de trayectorias | 30 |
| IV-D2. | Algoritmos de planificación de trayectorias | 33 |
| V. | MARCO METODOLÓGICO | 37 |
| V-A. | Diseño mecánico del robot | 37 |
| V-A1. | Chasis del robot | 37 |
| V-A2. | Motores | 39 |
| V-A3. | Ruedas | 41 |
| V-B. | Diseño electrónico del robot | 41 |
| V-B1. | Sensor LIDAR | 41 |
| V-B2. | Controlador | 42 |
| V-B3. | Placa de control de robot para Raspberry PI | 43 |
| V-B4. | Placa de expansión HUB USB.0 1 | 43 |
| V-B5. | Fuente de alimentación | 44 |
| V-C. | Software | 45 |
| V-C1. | Algoritmos de mapeo | 45 |
| V-C2. | Algoritmo de planificación de trayectorias | 48 |
| V-C3. | Algoritmo de navegación | 50 |
| VI. | RESULTADOS | 52 |
| VI-A. | Resultados de los algoritmos de mapeo | 52 |
| VI-A1. | Pruebas de mapeo con algoritmo Gmapping mapping | 52 |
| VI-A2. | Pruebas de mapeo con algoritmo Hector mapping | 53 |
| VI-A3. | Pruebas de mapeo con algoritmo Cartographer mapping | 55 |
| VI-B. | Pruebas de planificación de trayectorias con algoritmo RRT_exploration | 57 |
| VI-C. | Pruebas de un punto A hacia un punto B con algoritmo AMCL | 57 |
| VII. | CRONOGRAMA | 59 |

| | |
|---------------------------|----|
| VIII. PRESUPUESTO | 60 |
| IX. CONCLUSIONES | 61 |
| X. RECOMENDACIONES | 62 |
| Apéndice A: Planos | 66 |

ÍNDICE DE FIGURAS

| | | |
|-----|---|----|
| 1. | Instalación anual de robots industriales. | 19 |
| 2. | Stock operacional de robots industriales. | 20 |
| 3. | Robot móvil. | 21 |
| 4. | Componentes de un robot móvil. | 22 |
| 5. | Componentes de un robot móvil. | 23 |
| 6. | Diagrama de sistema eléctrico y electrónico. | 24 |
| 7. | Diagrama de flujo de la generación de los algoritmos de navegación. | 26 |
| 8. | Entorno mapeado utilizando algoritmo RRT_exploration. | 27 |
| 9. | Entorno mapeado utilizando algoritmo PRM. | 27 |
| 10. | Trayectoria de un robot móvil. | 29 |
| 11. | Localización del robot. | 29 |
| 12. | Posición del robot. | 30 |
| 13. | Localización de un robot móvil. | 31 |
| 14. | Planificación de tareas de un robot móvil. | 31 |
| 15. | Planificación de movimientos de un robot móvil. | 32 |
| 16. | Clasificación general de métodos de planificación de trayectorias. | 32 |
| 17. | Diagrama de Voronoi. | 33 |
| 18. | Algoritmo A*. | 34 |
| 19. | Planeación de trayectorias utilizando el algoritmo A*. | 34 |
| 20. | Algoritmo de Dijkstra. | 34 |
| 21. | Planeación de trayectorias utilizando el algoritmo de Dijkstra. | 35 |
| 22. | Grafo de visibilidad. | 35 |
| 23. | Algoritmo AMCL. | 36 |
| 24. | Diagrama general del robot. | 37 |
| 25. | Análisis estático - Tensión. | 38 |
| 26. | Análisis estático - Factor de seguridad. | 39 |
| 27. | Ruedas. | 41 |
| 28. | Placa de expansión "ROS" para robot. | 43 |
| 29. | Placa de expansión HUB USB.0 1 a 4 canales. | 44 |
| 30. | Algoritmo Hector mapping. | 46 |
| 31. | Algoritmo Gmapping mapping. | 46 |
| 32. | Algoritmo Cartographer mapping. | 47 |
| 33. | Diagrama de flujo de los distintos algoritmos de mapeo. | 48 |
| 34. | Algoritmo de planificación de trayectorias RRT_exploration. | 49 |
| 35. | Diagrama de flujo del algoritmo RRT_exploration. | 49 |
| 36. | Algoritmo de navegación AMCL. | 50 |
| 37. | Diagrama de flujo del algoritmo AMCL. | 51 |
| 38. | Prueba #1 algoritmo Gmapping mapping. | 52 |
| 39. | Prueba #2 algoritmo Gmapping mapping. | 52 |
| 40. | Prueba #3 algoritmo Gmapping mapping. | 53 |
| 41. | Prueba #1 algoritmo Hector mapping. | 53 |
| 42. | Prueba #2 algoritmo Hector mapping. | 54 |
| 43. | Prueba #3 algoritmo Hector mapping. | 54 |
| 44. | Prueba #1 algoritmo Cartographer mapping. | 55 |
| 45. | Prueba #2 algoritmo Cartographer mapping. | 55 |
| 46. | Prueba #3 algoritmo Cartographer mapping. | 56 |
| 47. | Prueba del algoritmo RRT. | 57 |
| 48. | Prueba del algoritmo AMCL. | 58 |
| 49. | Cronograma. | 59 |

| | | |
|-----|--|----|
| 50. | Plano de pieza: Suspensión | 66 |
| 51. | Plano de pieza: Rendija | 67 |
| 52. | Plano de pieza: Base media | 68 |
| 53. | Plano de pieza: Base inferior | 69 |
| 54. | Plano de pieza: Base inferior de la suspensión | 70 |
| 55. | Plano de pieza: Base superior | 71 |
| 56. | Entorno controlado #1 en software CAD | 72 |
| 57. | Entorno controlado #1 | 73 |
| 58. | Entorno controlado #2 en software CAD | 74 |
| 59. | Entorno controlado #2 | 75 |
| 60. | Entorno controlado #3 en software CAD | 76 |
| 61. | Entorno controlado #3 | 77 |

ÍNDICE DE TABLAS

| | | |
|-------|---|----|
| I. | Características de un robot móvil | 23 |
| II. | Métodos de navegación autónoma | 26 |
| III. | Algoritmos de navegación autónoma | 28 |
| IV. | Tabla de datos #1 | 39 |
| V. | Tabla de datos #2 | 39 |
| VI. | Tabla de datos #3 | 40 |
| VII. | Tabla de datos #4 | 40 |
| VIII. | Selección de motores | 40 |
| IX. | Selección de sensores LIDAR | 42 |
| X. | Selección de placas de control | 42 |
| XI. | Selección de tarjeta de potencia | 43 |
| XII. | Consumo energético de los componentes | 44 |
| XIII. | Selección de baterías | 45 |
| XIV. | Resultados del algoritmo Gmapping mapping | 53 |
| XV. | Resultados del algoritmo Hector mapping | 54 |
| XVI. | Resultados del algoritmo Cartographer mapping | 56 |
| XVII. | Presupuesto | 60 |

I. INTRODUCCIÓN

La importancia de los robots autónomos móviles se deriva de la necesidad de realizar tareas sin depender de una asistencia continua de un usuario operador, permitiendo así optimizar tareas [1]. Actualmente los robots autónomos están siendo aplicados en campos como: minería, agricultura, transporte, logística, exploración espacial, exploración marina, aplicaciones militares, construcción, entre otros. Sus características como la capacidad de moverse y navegar en diferentes entornos dinámicos donde se requiere una adaptabilidad constante, permiten interactuar y colaborar en las tareas que se requiere la combinación de destrezas tanto humanas y de capacidades robóticas ya que estos pueden recopilar datos, realizar mediciones y ayudar a los científicos a obtener información valiosa sin exponerse a riesgos potenciales [2].

Estos sistemas son ampliamente utilizados a nivel mundial, por ejemplo, en el estudio planteado por [3], se propone un robot autónomo enfocado a la agricultura logrando automatizar el proceso de sembrado de semillas, consecuentemente optimizando el proceso y liberando a las personas de largas jornadas de trabajo repetitivo, además reduciendo costos de producción. Otro de los ejemplos, es el robot autónomo propuesto por [4], el cual es capaz de evaluar los niveles de polución en el aire y el suelo, de acuerdo con las normas establecidas, de modo que se pueda acceder a la información desde una estación remota. En la investigación realizada por [5] se desarrolló un robot móvil equipado con un sensor LIDAR para evitar obstáculos midiendo la distancia siendo robusto frente a factores ambientales como la intensidad de luz. Además, otro robot implementado por [6] consiste en un robot móvil con la capacidad de detectar de metales realizando un mapeo de terreno y reconocer las irregularidades e interpretarlo como minas.

Los robots autónomos han logrado avances significativos; sin embargo; aún existen varias condiciones de mejora como, por ejemplo: reducir el tiempo de respuesta, usualmente presentan mediciones inusuales, necesitan una gran capacidad de cómputo, algoritmos de toma de decisiones complejos, entre otros. Los sistemas de LIDAR suelen tener inconvenientes como: limitada distancia, resolución espacial, y complejidad.

Como se mencionó anteriormente este campo de estudio aún presenta varias oportunidades, por esta razón, la investigación está enfocada en el desarrollo de un robot móvil capaz de navegar de forma autónoma en un entorno controlado con obstáculos, realizando una representación digital del mismo usando un sensor LIDAR. Con esto se busca fomentar el uso de tecnología propia en robots autónomos para en un futuro desarrollar aplicaciones que cubran necesidades específicas del país.

II. PROBLEMA

Los robots autónomos móviles han cobrado gran relevancia mundial en la industria [7]. Asisten en tareas peligrosas y aceleran la ejecución de trabajos, brindando ventajas competitivas y un crecimiento exponencial en diferentes sectores. Su desarrollo se impulsa mediante tecnologías que mejoran su autonomía, haciéndolos más eficientes y atractivos para las grandes industrias. Su despliegue se experimenta principalmente en países desarrollados, con infraestructura, recursos y apoyo económico para su fabricación y mantenimiento, y personal altamente capacitado. Los robots autónomos se han vuelto indispensables para incrementar la productividad y la seguridad en los procesos industriales a nivel global [8].

En Latinoamérica, existe una baja inversión en sistemas robotizados autónomos, posiblemente debido a la falta de recursos, tecnología e infraestructura. Consecuentemente, se incurre a la importación provocando dependencia tecnológica de países desarrollados. Superar estos obstáculos requiere inversión en investigación, desarrollo y políticas que promuevan la adopción tecnológica. La colaboración entre gobiernos, instituciones académicas y empresas impulsará la innovación y transferencia de conocimientos para cerrar la brecha tecnológica, fomentando el desarrollo industrial y la competitividad en el mercado global [9]. En Ecuador, un país con riqueza minera y agrícola no se han implementado tecnologías de robots autónomos a pesar de las múltiples ventajas que estos sistemas presentan [10]. La tendencia mundial apunta al desarrollo de dichos sistemas, el no aprovechar estas tecnologías tendría consecuencias negativas para el país.

Los robots autónomos han alcanzado un nivel significativo de desarrollo en diversos campos industriales. Estos sistemas robóticos poseen la capacidad de tomar decisiones y ejecutar tareas sin intervención humana directa, lo que ha mejorado la eficiencia, la precisión y la seguridad en diversas operaciones. Sin embargo, a pesar de sus logros, los robots autónomos aún enfrentan desafíos sustanciales. Los problemas incluyen la complejidad del entorno en el que operan, la necesidad de una detección y percepción más precisa, la interpretación adecuada de escenarios imprevistos y la adaptación a situaciones cambiantes [11]. Estos obstáculos señalan la necesidad de desarrollo para lograr una autonomía de robots autónomos móviles más completa y confiable en el futuro.

III. OBJETIVOS

III-A. Objetivo general

- Desarrollar un robot autónomo con sistema de mapeo y detección de obstáculos adaptable a diferentes entornos mediante sensor LIDAR.

III-B. Objetivos Específicos

- Implementar sistemas mecatrónicos para navegación autónoma del robot.
- Implementar un algoritmo de control de trayectoria mediante sensores LIDAR para esquivar obstáculos.
- Validar la navegación del robot entre dos puntos mediante un sistema de mapeo.

III-C. Tabla de Objetivos

| OBJETIVOS | PLANTEAMIENTO | META | INDICADOR |
|------------|---|---|----------------------------------|
| OBJETIVO 1 | Implementar sistemas mecatrónicos para la navegación autónoma del robot. | Obtener una trayectoria predefinida con un error máximo del 10 %. | Porcentaje de error. |
| OBJETIVO 2 | Implementar un algoritmo de control mediante sensores LIDAR para esquivar obstáculos. | Reconocer el entorno con una precisión del 90 %. | Porcentaje de error. |
| OBJETIVO 3 | Validar la navegación del robot entre dos puntos mediante un sistema de mapeo. | Lograr llegar de un punto A a un punto B esquivando 3 obstáculos. | Número de obstáculos a esquivar. |

IV. MARCO TEÓRICO

IV-A. *Introducción a los robots*

La robótica nació en el año 1980 como resultado del desarrollo de campos de la ingeniería en electrónica, ingeniería en telecomunicaciones, ingeniería en informática e ingeniería mecánica. La robótica ha experimentado avances científicos y tecnológicos fuera de la ingeniería, en áreas como las matemáticas y la física teórica. En particular, las teorías tradicionales desarrolladas por Lagrange, Newton y Euler en épocas pasadas fueron cruciales para optimizar y perfeccionar las ecuaciones que sustentan el intelecto y dinámica de los robots modernos. El desarrollo de robots que se acercan a los ideales deseables de automatismo y autonomía que persiguen los diseñadores de robots se ha visto impulsado sobre todo por los avances informáticos de las últimas décadas [12].

IV-A1. Tipos de robots: Existen muchas variedades de robots, cada uno destinado a realizar una tarea concreta. En las secciones siguientes se describen las distintas categorías, desde robots industriales hasta robots móviles.

- **Robots industriales:** La International Organization for Standardization (ISO) determina a los robots como “un manipulador programable, automáticamente controlado, reprogramable y multipropósito, en tres o más ejes, que puede estar o no fijo en un lugar, y es utilizado en aplicaciones industriales automatizadas” en su norma ISO 8373 [13]. La robótica, eje de la Industria 4.0, optimiza la producción industrial con precisión y costos reducidos. Esta tecnología potencia fábricas inteligentes, eficientes y seguras, preservando asociaciones y mejorando la calidad de productos. Estas instalaciones, fruto de la Industria 4.0, se construyen con robots avanzados, datos masivos, nube, seguridad sólida y sensores inteligentes para garantizar una producción de calidad y eficaz [14].
- **Robots colaborativos:** Un robot colaborativo, o cobot, es un tipo de robot diseñado para trabajar junto con humanos en un entorno compartido de forma segura. Incorporan sensores avanzados y sistemas de control para interactuar sin riesgo de causar daño, siendo útiles en industrias como la fabricación, logística y atención médica [15]. Los robots colaborativos han evolucionado en la Industria 4.0, siendo esenciales en la manufactura actual. Su versatilidad y seguridad permiten tareas variadas en entornos compartidos con humanos. Su programación accesible fomenta la cooperación humano-robot, fusionando decisiones humanas con eficiencia robótica, generando ahorro y eficacia [16].
- **Robots de servicios:** Los robots de servicio fusionan aspectos humanos y tecnológicos, reemplazando labores previamente manejadas por personas o máquinas. Representan un campo innovador al poder socializar con clientes, brindando experiencias adaptadas a cada individuo sin depender de la intervención humana. Mediante la inteligencia artificial, estos autónomos se ajustan a diversas situaciones, ofreciendo interacciones complejas que antes demandaban habilidades humanas [17].
- **Robots móviles:** Un robot móvil es un dispositivo automatizado capaz de moverse de manera autónoma en su entorno. Estos robots realizan tareas especializadas como exploración, inspección, entrega de mercancías y patrullas de seguridad utilizando una serie de sensores y sistemas de navegación para identificar obstáculos y trazar rutas [18]. La autonomía lograda les permite actuar sin dirección humana, con áreas de mejora como la planificación de trayectorias, evasión de obstáculos, localización y mapeo. Este proyecto continuará en el desarrollo de técnicas como el SLAM para mejorar la autonomía y capacidades de navegación de estos robots [19].

IV-A2. Robótica en cifras: La evolución de la robótica ha generado y seguirá provocando grandes cambios en la economía, en el trabajo e incluso en la sociedad y su forma de pensar y de vivir [20]. En la Figura 1, se puede observar desde el 2012 el incremento hasta la actualidad de los robots.



Figura 1. Instalación anual de robots industriales.

Fuente: [21].

En el transcurso del año 2023, se contabilizó un stock activo de robots industriales que alcanzó la cifra de 2.722.077 unidades, evidenciando un incremento del 12%. Sin embargo, es relevante destacar que durante ese mismo período se produjo una reducción del 12% en la instalación de estos robots, con un total de 373.240 unidades implementadas. Antes de la crisis económica mundial comprendida entre 2005 y 2008, la venta anual de robots se mantenía en torno a las 115.000 unidades, descendiendo abruptamente a 60.000 unidades en 2009. Posteriormente, se experimentó un repunte en las inversiones, reflejado en un notable aumento en las ventas de robots, con alrededor de 245.000 unidades comercializadas en 2015 y el hito histórico de superar las 400.000 unidades vendidas en 2018 [22].

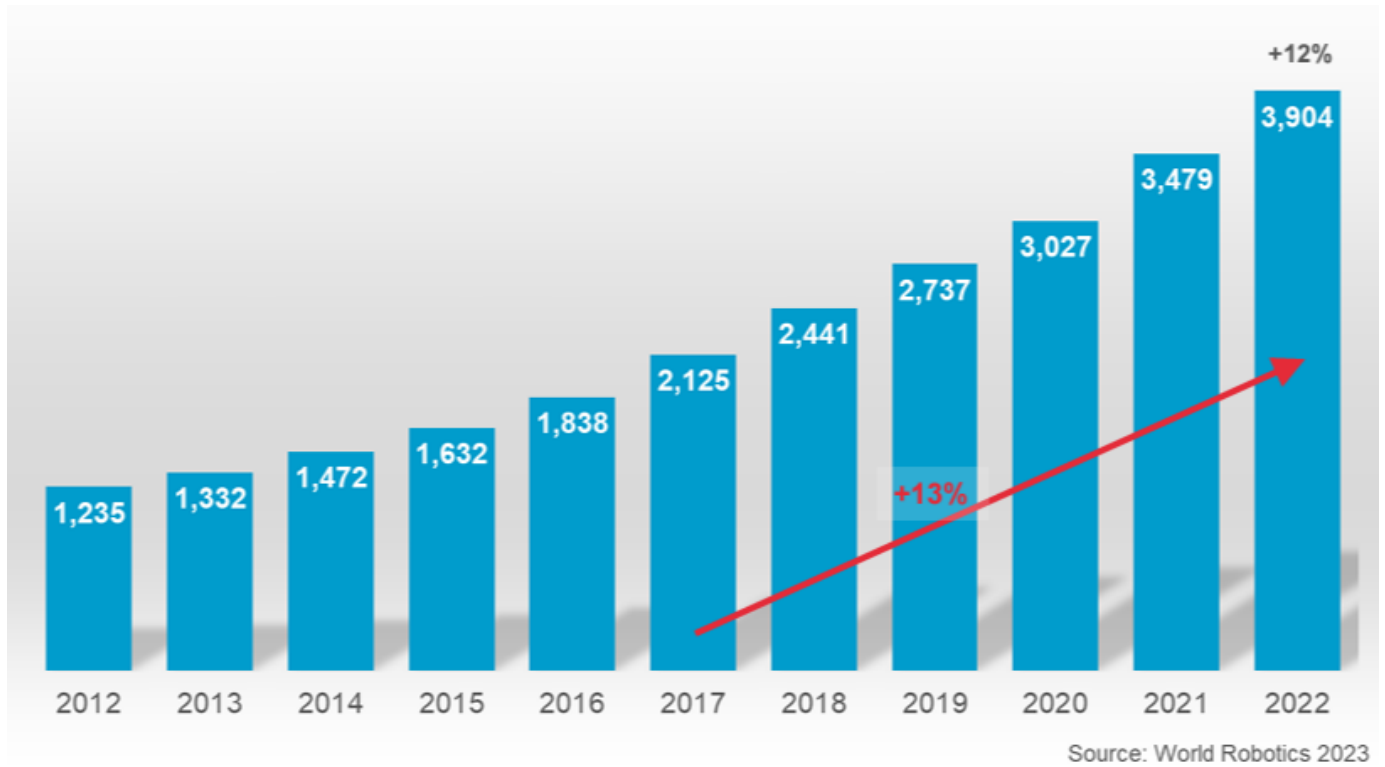


Figura 2. Stock operacional de robots industriales.
Fuentes: [21].

En el presente, Asia se rige como el principal mercado global de robots industriales, dos de cada tres robots desplegados en el mundo se destinaron a naciones asiáticas, alcanzando un total de 245,158 unidades. China, Japón y Corea del Sur destacan como los países más relevantes en este mercado, con 140.491, 49.908 y 27.873 unidades respectivamente, evidenciando descensos del 9 %, 10 % y 26 % en comparación con el año previo. En Europa, se posiciona como el segundo mercado más importante de robots industriales a nivel mundial. Igual que en Asia, se puso fin a seis años de crecimiento en la instalación de robots en fábricas en 2019, registrando una disminución del 5 % en la instalación de unidades, con un total de 71.932 [22].

Por otro lado, en América se experimentó una reducción del 13 % en la instalación de unidades, alcanzando un total de 47,908, luego de un período de crecimiento sostenido durante seis años. Es destacable que la distribución de robots industriales en las plantas no es uniforme, ya que el 73 % del total de robots instalados se concentra en cinco países [22].

Con el tiempo, se ha producido un notable aumento del uso de robots industriales en múltiples industrias. Estos sistemas han crecido considerablemente y abarcan desde la gestión logística hasta la automatización de cadenas de montaje. Estas tecnologías se adoptan cada vez con más frecuencia debido a su importancia para mejorar la productividad, la precisión y la eficiencia en entornos industriales. El avance continuo de los sistemas robóticos redefine la forma de abordar las dificultades de la producción contemporánea, al tiempo que impulsa la innovación.

IV-B. Robot autónomo móvil

En la actualidad, los robots móviles autónomos se están convirtiendo en una tecnología innovadora que está cambiando varios sectores, desde la exploración espacial hasta la logística. En esta revisión se examinará el desarrollo y las capacidades de estos robots, que están revolucionando la interacción entre el ser humano y su entorno sin

necesidad de intervención humana.

Los robots autónomos denotan sistemas robóticos capaces de realizar tareas en un entorno regulado sin requerir una intervención constante por parte de un operador humano. Estos sistemas pueden exhibir variados niveles de autonomía, manifestándose en diversas modalidades de independencia funcional. pueden tener diferentes grados de autonomía y ser autónomos en diferentes maneras [23].



Figura 3. Robot móvil.

Fuentes:[24]

Un robot de plena autonomía presenta diversos componentes característicos de un robot móvil, tal como se muestra en la Figura 4. Entre ellos se encuentran una computadora que actúa procesando datos y tomando decisiones en tiempo real. La presencia de una red inalámbrica permite la comunicación del robot con otros dispositivos y a su vez la transmisión de datos. El controlador es responsable de gestionar los movimientos y acciones del robot, asegurando su precisión. Las placas de control, por otro lado, proporcionan la interfaz necesaria entre los componentes mecánicos y electrónicos del robot, permitiendo su integración y funcionamiento coordinado. Finalmente, el software desempeña un papel crucial al proporcionar algoritmos de navegación, planificación de rutas, percepción del entorno y otras funciones necesarias para la operación autónoma del robot. En conjunto, estos componentes trabajan en armonía para dotar al robot de la capacidad de moverse de manera autónoma, realizar tareas específicas y adaptarse a diferentes entornos y situaciones.

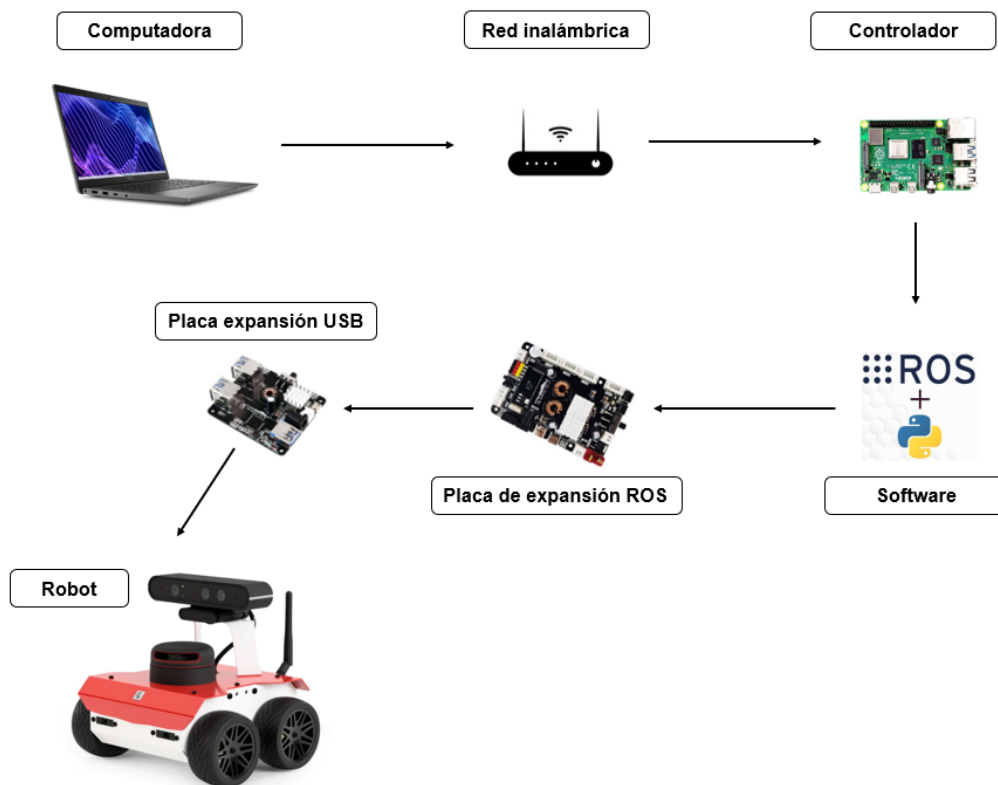


Figura 4. Componentes de un robot móvil.
Fuentes: Autores.

Los robots autónomos también pueden adquirir nuevas capacidades como lo es adaptar estrategias para el cumplimiento de tareas y adaptaciones de circunstancias cambiantes. Igualmente, los robots autónomos demandan un programa periódico de mantenimiento, en consonancia con el proceder común de otras máquinas [23].

IV-B1. Aplicaciones de un robot móvil: Los robots móviles, equipados con capacidad de desplazamiento autónomo, revolucionan diversas industrias. Su versatilidad y adaptabilidad los convierten en herramientas fundamentales para la logística, exploración espacial, medicina y más. Estos ingenios tecnológicos prometen transformar la manera en que interactuamos con el entorno. El robot presentado por [25] es un robot cortacésped capaz de distinguir y cortar el césped, explorar áreas designadas y moverse eficazmente en terrenos mapeados. Aunque cumplió la mayoría de sus metas; aún así, la navegación y evasión de obstáculos en tiempo real aún representan un desafío pendiente. Además, el robot planteado por [26] es un robot móvil diseñado para fumigar campos de maíz buscando reemplazar la exposición directa de los agricultores a productos químicos. Su propósito es evitar enfermedades a largo plazo y mejorar la eficiencia al reducir el tiempo en más de 4 horas empleado en la fumigación, así como los costos asociados, pudiéndose potenciar la sección mecánica del dispositivo de aspersión mediante la inclusión de más aspersores, lo cual aceleraría el proceso de fumigación. Además, se propone ajustar las boquillas de los aspersores para adaptarlas de manera regulable a las necesidades específicas de la fumigación del cultivo de maíz. Otra aplicación es la de [27] que consiste en un robot equipado con un algoritmo de seguimiento de individuos y una plataforma de videoconferencia. Estos elementos trabajan en conjunto para posibilitar la telepresencia, permitiendo además que el robot sea controlado a distancia por medio de un dispositivo remoto, considerando la posibilidad de mejorar los procesos utilizando un CPU más robusto, dado que el actual no resultó ser lo bastante eficaz. En el siguiente trabajo realizado por [28] es un robot de navegación que puede realizar un seguimiento a través de una trayectoria indicada por coordenadas geográficas, a su vez puede reconocer obstáculos en su camino y cambiar su rumbo para evitarlos midiendo las distancias con un sensor LiDAR y modificando sus parámetros. En las pruebas

realizarse pudo observarse que existe un error aproximado de 1.15 metros con respecto a la trayectoria indicada y al usar una red wifi de 2.4 GHz se crearon interferencias en el sistema de telemetría reduciendo la transmisión de datos.

Tabla I
CARACTERÍSTICAS DE UN ROBOT MÓVIL

| CARACTERÍSTICAS DE UN ROBOT MÓVIL | |
|-----------------------------------|--|
| Tipo de robot móvil | Características principales |
| Robot cortacésped | Explorar áreas. |
| | Moverse en terrenos mapeados. |
| | Navegación y evasión de obstáculos en tiempo real. |
| Robot fumigador | Reducir tiempo de fumigación. |
| | Reducir costos por fumigación. |
| | Potenciar el sistema de aspersión. |
| Robot de seguimiento | Seguimiento de individuos. |
| | Posibilitar la telepresencia. |
| | Plataforma de videoconferencia. |
| Robot de navegación | Seguimiento de trayectorias. |
| | Reconocer obstáculos y evitarlos. |

En resumen, la decisión de implementar la tecnología de robots móviles autónomos brinda una oportunidad para investigar y comprender los principios fundamentales de la robótica autónoma. Esta metodología promueve la creatividad y el pensamiento crítico en la resolución de problemas y la generación de soluciones innovadoras, al tiempo que proporciona un entendimiento profundo del funcionamiento del robot.

IV-B2. Elementos de un robot móvil: Los robots móviles se sustentan en componentes como se muestra en la Figura 5 que son esenciales tales como la estructura física, sistemas de control, sensores y software especializado, que son críticos para la efectiva navegación y ejecución de tareas en diversos entornos.

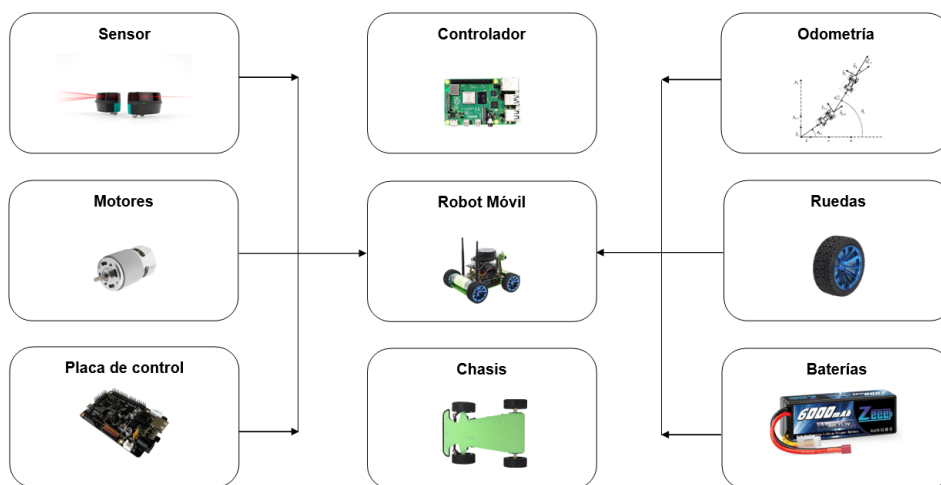


Figura 5. Componentes de un robot móvil.
Fuentes: Autores.

- Odometría:** En el campo de la robótica móvil, la odometría describe la técnica utilizada para rastrear y estimar la posición, orientación y desplazamiento de un robot dentro de su entorno. En la mayoría de los casos, el método se basa en la detección del giro de las ruedas o los actuadores del robot para calcular la distancia recorrida y los cambios de orientación. Los algoritmos de odometría utilizan información como el tiempo transcurrido y la velocidad lineal y angular para determinar la posición del robot. Aunque esta técnica es útil, es susceptible de acumular errores debidos a imprecisiones de las ruedas, irregularidades del terreno

y deslizamientos. Por este motivo, suele utilizarse junto con otras técnicas de localización para aumentar la precisión del posicionamiento del robot [29].

- **ROS:** Un marco de código abierto llamado Robot Operating System (ROS) está diseñado para facilitar y hacer más eficiente el desarrollo de software para robots móviles. Este entorno proporciona una amplia gama de herramientas y bibliotecas destinadas a facilitar las tareas relacionadas con la robótica. Estas tareas incluyen la configuración de comportamientos complejos, el intercambio de datos entre procesos, el control de hardware y la gestión de dispositivos. Gracias a su arquitectura adaptable y modular, los programadores pueden diseñar sistemas robóticos robustos y adaptables, lo que fomenta la reutilización de código y el trabajo en equipo entre la comunidad robótica [29].
- **Sistema mecánico:** La estructura mecánica de un robot móvil es un proceso minucioso que incluye construir y perfeccionar la estructura física del robot para garantizar su maniobrabilidad, estabilidad y funcionalidad. Este diseño tiene en cuenta factores como la elección de materiales ligeros y duraderos, la colocación cuidadosa de los componentes mecánicos y electrónicos, y la inclusión de sistemas de tracción, suspensión y dirección que permitan al robot moverse con precisión y eficacia por diversas superficies y entornos [30]. En conclusión, el chasis del robot es un componente crucial que no sólo ofrece estabilidad estructural, sino que también facilita la integración de dispositivos electrónicos al alojarlos en su interior. Por otro lado, los motores son cruciales para la locomoción, su selección se realiza mediante cálculos detallados que tienen en cuenta el peso total del robot, la carga máxima que puede soportar y la velocidad necesaria para garantizar su correcto funcionamiento. Por su parte, las ruedas son componentes esenciales que permiten el movimiento del robot; transfieren la potencia de los motores al terreno y ofrecen estabilidad y tracción en diversas circunstancias.
- **Sistema eléctrico y electrónico:** El diseño eléctrico-electrónico de un robot móvil es esencial para su funcionamiento eficaz. Implica diseñar y construir sensores, sistemas de control y circuitos eléctricos que permitan al robot percibir su entorno y realizar determinadas tareas [30].

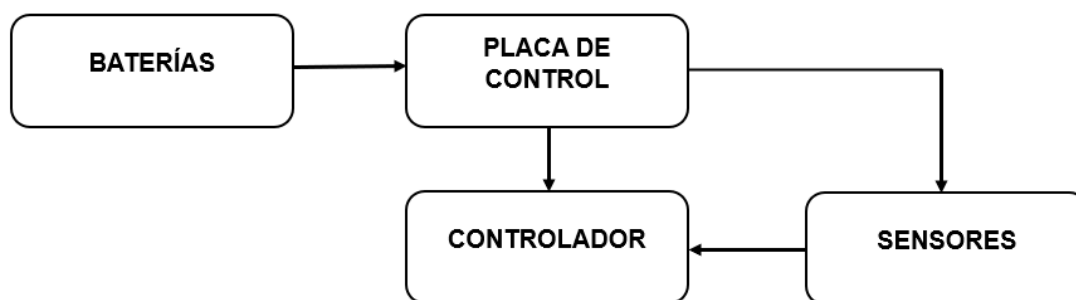


Figura 6. Diagrama de sistema eléctrico y electrónico.

Fuentes: Autores.

Como se muestra en la Figura 6 se incorporan placas de control para poder gestionar eficazmente la energía suministrada a los distintos actuadores del robot, además tiene una interfaz de bus CAN, controla los motores ya sean Ackerman, diferencial de 4 ruedas, orugas y omnidireccionales. Así mismo, las decisiones del robot son coordinadas por el controlador, que puede ser una CPU de propósito particular como Raspberry, Arduino, ESP32, entre otras. Para ello, utiliza los datos de los sensores ya sean infrarrojos, proximidad, distancia entre otros dependiendo del funcionamiento del robot. Por último, las baterías recargables, que pueden variar en capacidad dependiendo de los requisitos de diseño, se utilizan como fuente de energía por el sistema eléctrico, que es esencial para el funcionamiento del robot [31].

- **Software:** La programación del controlador de un robot móvil es esencial para su funcionalidad y comporta-

miento. Los controladores de estos robots pueden programarse en diversos lenguajes de programación, como Python, C++ y lenguajes exclusivos de la plataforma del robot. Los programadores tienen que crear un código que permita al robot moverse, navegar, manipular objetos e interactuar con su entorno. También debe ser capaz de interpretar los datos de los sensores, procesar la información del entorno, tomar decisiones y ejecutar acciones. Para conseguir el comportamiento deseado y una interacción segura y eficaz con el entorno, hay que crear algoritmos eficaces, poner en práctica técnicas de control y utilizar bibliotecas específicas [30].

IV-C. Sistema de navegación autónoma

La implementación de sistemas de navegación autónomos permiten la creación de diversas aplicaciones en campos como la agricultura, vigilancia, servicio de entrega, reconocimiento, construcción y diversas aplicaciones industriales [32]. La navegación autónoma se centra en la capacidad de un robot para moverse libremente por su entorno sin ayuda humana directa. Detectar y comprender el entorno, planificar rutas y realizar movimientos para seguir una ruta predeterminada son algunas de las tareas que implica este proceso. La navegación basada en mapas y la navegación reactiva o basada en el comportamiento son los dos tipos principales de técnicas de navegación autónoma. Con la navegación basada en mapas, el robot planifica y ejecuta una ruta precisa utilizando un mapa previamente creado de los alrededores. La navegación reactiva, en cambio, no requiere un mapa preestablecido, sino que el robot utiliza su entorno para reaccionar en tiempo real con el fin de evitar obstáculos y cumplir su objetivo [25]. Aunque se han logrado progresos en la navegación autónoma, persisten retos en la navegación en entornos que son desconocidos, cambiantes y complicados. Además, cuestiones como la incertidumbre relacionada con los sensores y el movimiento, la creación de mapas a gran escala y la planificación de rutas eficientes son temas de investigación activa en este campo.

IV-C1. Localización y mapeo simultáneo (SLAM): La técnica conocida como Localización y Mapeo Simultáneo, abreviada como SLAM por sus siglas en inglés (Simultaneous Localization And Mapping), representa un desafío esencial en el campo de la robótica móvil autónoma. Este problema se centra en la capacidad de un robot para crear un mapa de su entorno y determinar su posición dentro de este mapa, utilizando la información sensorial que recopila del entorno en tiempo real [33].

Existen diversas estrategias y algoritmos para abordar el problema de SLAM, los cuales pueden diferir en términos de eficiencia, robustez, precisión y complejidad.

- **Filtro de Kalman Extendido (EKF):** Esta técnica emplea un modelo probabilístico para representar el estado del robot y los puntos de referencia en el mapa, utilizando observaciones de sensores y comandos de movimiento para actualizar y corregir este modelo. Sin embargo, el EKF SLAM puede enfrentar desafíos relacionados con la consistencia y la escala, ya que se apoya en aproximaciones lineales y supone que el ruido sigue una distribución gaussiana.
- **SLAM basado en Filtros de Partículas:** Es una alternativa que emplea un método de muestreo de Monte Carlo para representar la incertidumbre en el proceso de estimación. A diferencia del EKF SLAM, FastSLAM es capaz de manejar modelos no lineales y no gaussianos. Sin embargo, requiere una gran cantidad de partículas para lograr una precisión aceptable, lo que puede resultar en un costo computacional elevado.
- **SLAM basados en gráficos:** Representan el problema de SLAM mediante un grafo que incluye las poses del robot y los puntos de referencia, donde las conexiones entre nodos reflejan las restricciones espaciales entre las poses y las observaciones de los puntos de referencia. Este método tiene la capacidad de manejar entornos extensos y es más resistente a la correlación a largo plazo y a la incertidumbre en la inicialización. Sin embargo, requiere técnicas de optimización avanzadas y puede enfrentar dificultades en cuanto a la robustez frente a los errores de asociación de datos.

Tabla II
MÉTODOS DE NAVEGACIÓN AUTÓNOMA

| Método de Navegación Autónoma | Ventajas | Desventajas | Aplicaciones |
|--------------------------------------|---|---|--|
| Filtro de Kalman Extendido (EKF) | Utiliza un modelo probabilístico para representar el estado del robot y los puntos de referencia, permitiendo una estimación precisa. | Puede enfrentar desafíos relacionados con la consistencia y la escala debido a las aproximaciones lineales. | Aplicaciones en vehículos autónomos, robots móviles en entornos controlados. |
| SLAM basado en Filtros de Partículas | Capaz de manejar modelos no lineales y no gaussianos, lo que lo hace más flexible en diversos entornos. | La convergencia puede ser lenta y depende de la cantidad de partículas y la calidad de las observaciones. | Aplicaciones en entornos dinámicos, entornos no estructurados donde las suposiciones del EKF no son válidas. |
| SLAM basados en gráficos | Capacidad para manejar entornos extensos y más robusto frente a la incertidumbre en la inicialización y la correlación a largo plazo. | Requiere técnicas de optimización avanzadas, lo que puede aumentar la complejidad computacional. | Aplicaciones en mapeo de entornos grandes y complejos, mapeo de interiores en tiempo real. |

Cada uno de estos métodos presenta tanto beneficios como limitaciones, y la selección entre ellos puede variar según el entorno, las capacidades de los sensores y del movimiento del robot, así como los requisitos particulares de la aplicación.

IV-C2. Algoritmos de navegación: A continuación, se muestra en la Figura 7 el diagrama de flujo que resume la estructura del sistema de navegación en su totalidad. Se ha dividido el sistema en cuatro subsistemas distintos: mecánico, visión, planificación de trayectorias y control. Estos subsistemas trabajan en conjunto para alcanzar el objetivo establecido de resolver el problema de la generación del camino.

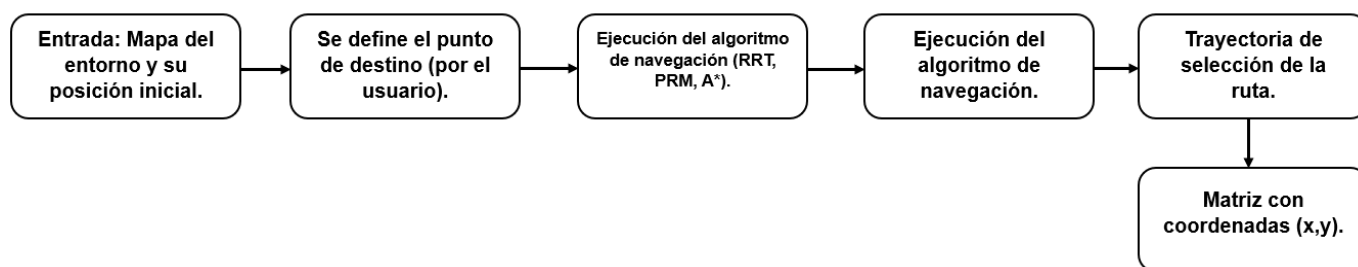


Figura 7. Diagrama de flujo de la generación de los algoritmos de navegación.

Fuente: Autores.

1. **Algoritmo RRT_exploration:** Se centra en casos de búsqueda individuales. Esencialmente, este algoritmo comienza en el punto inicial y construye iterativamente un árbol de caminos potenciales. El algoritmo comienza con una red que no tiene conexiones entre ninguno de los estados de los vértices, sólo el estado inicial. En cada paso se intenta establecer una conexión con un punto de muestra que cumpla unos criterios específicos predefinidos (como ser el más cercano al punto final o la distancia máxima entre puntos) [34].

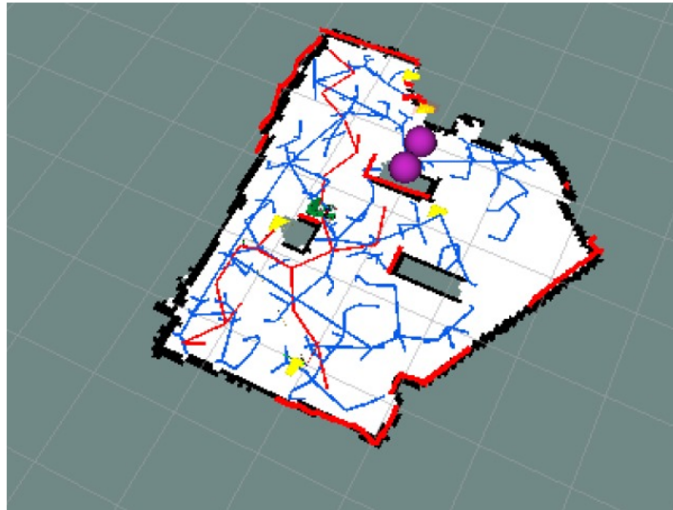


Figura 8. Entorno mapeado utilizando algoritmo RRT_exploration.
Fuentes: Autores.

2. **Algoritmo PRM:** Se centra en casos de búsquedas múltiples. Su forma fundamental consta de dos fases: una fase de preprocesamiento en la que se conectan puntos muestreados aleatoriamente en un espacio libre de obstáculos para crear una red de trayectorias, y una fase de consulta en la que se buscan trayectorias que conecten las condiciones iniciales y finales a través de esta red de trayectorias. Es un proceso fácil y rápido de llevar a cabo, y a medida que se realizan más puntos de muestreo e iteraciones, las trayectorias pueden optimizarse [34].

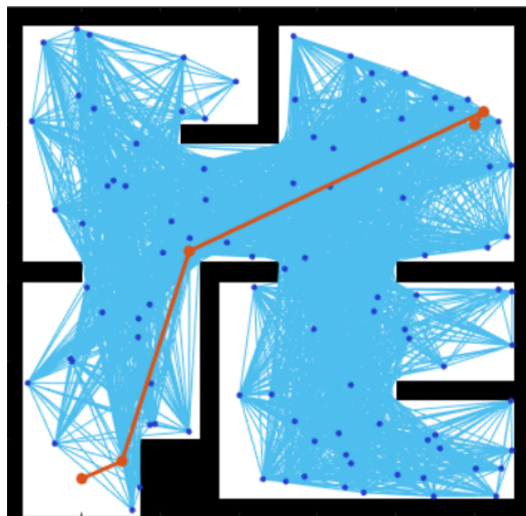


Figura 9. Entorno mapeado utilizando algoritmo PRM.
Fuentes: [35].

Tabla III
ALGORITMOS DE NAVEGACIÓN AUTÓNOMA

| Algoritmo de navegación autónoma | Ventajas | Desventajas | Aplicaciones |
|----------------------------------|--|---|--|
| Algoritmo RRT_exploration | Eficiente en entornos con obstáculos dinámicos o desconocidos. | Puede no encontrar soluciones óptimas. | Navegación de robots móviles en entornos con obstáculos cambiantes. |
| | No requiere conocimiento previo del espacio de búsqueda. | Puede requerir una gran cantidad de iteraciones para converger a una solución. | |
| Algoritmo PRM | Eficiente para la búsqueda de múltiples soluciones en un entorno estático. | Requiere una fase de preprocesamiento inicial que puede ser costosa en términos de tiempo y recursos computacionales. | Planificación de rutas para robots móviles en entornos con obstáculos estáticos. |
| | Proporciona soluciones de alta calidad y generalizadas. | La calidad de las soluciones puede depender significativamente de la densidad y distribución de los puntos de muestreo. | |

En síntesis, cada algoritmo de navegación presenta tanto aspectos positivos como negativos, y se destaca en diferentes aplicaciones específicas. La selección del algoritmo más adecuado se basa en el contexto particular de la situación, considerando factores como las características del entorno, la complejidad del área de búsqueda y las necesidades de optimización de la ruta.

IV-D. Control de trayectoria

El control de trayectoria y la odometría son componentes clave en el desarrollo de sistemas de navegación autónoma, donde el robot debe moverse de manera inteligente y segura en su entorno. La odometría proporciona información sobre la posición actual del robot, mientras que el control de trayectoria utiliza esta información para guiar al robot a lo largo de la ruta deseada, ajustando los comandos de control según sea necesario para mantener la trayectoria deseada y evitar obstáculos.

Odometría: Se podría describir como el método o proceso que posibilita estimar de forma relativa la ubicación de un vehículo en movimiento, utilizando el desplazamiento de sus ruedas (en las coordenadas (x,y) y el ángulo de orientación ϕ). Esta estimación se considera relativa porque se realiza dentro de un marco de referencia específico, donde se establece el origen según la situación real, como se muestra en la Figura 10.

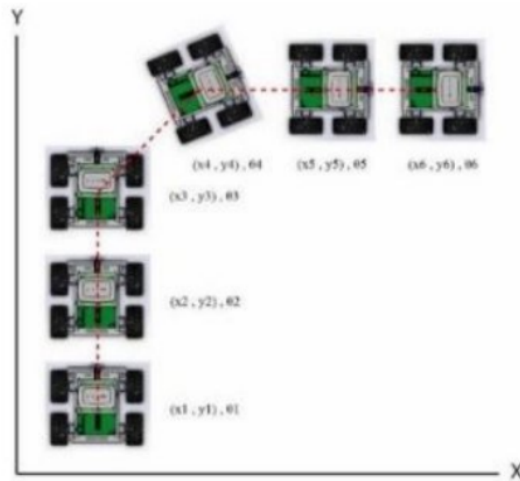


Figura 10. Trayectoria de un robot móvil.
Fuentes: [36].

La odometría implica la recopilación de datos provenientes de actuadores y codificadores para determinar la posición y la velocidad de un robot desde su punto de partida. Cada nueva posición se calcula basándose en la anterior, utilizando la ecuación 1, donde d representa el desplazamiento, v la velocidad y t el tiempo.

$$d = v * t \quad (1)$$

Sin embargo, es importante destacar que la odometría no proporciona una determinación general y precisa de la posición del robot, sino que la estima a partir de datos obtenidos de la lectura de los codificadores. Por lo tanto, la odometría puede introducir errores acumulativos en la medición del desplazamiento.

Localización del robot: Entre los diversos métodos empleados para lograr una correcta localización del robot, se destaca el método odométrico, el cual permite integrar la trayectoria mediante la estimación de la posición y orientación, basándose en el movimiento de las ruedas motrices. Sin embargo, es importante tener en cuenta los errores derivados de diversas fuentes, algunos asociados a fallos en los sistemas mecánicos y otros causados por deslizamientos. Estos errores pueden generar incertidumbre en la obtención de datos sobre la posición y orientación, lo que conlleva a que los datos proporcionados por el sistema odométrico no coincidan con la realidad [36].

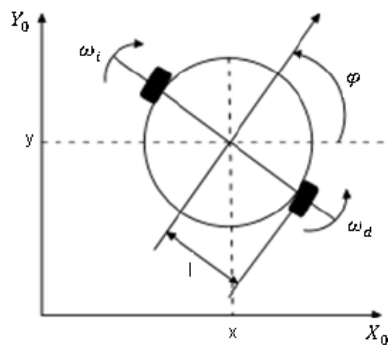


Figura 11. Localización del robot.
Fuentes: [37].

Cinemática del robot: La determinación de la ubicación del robot se logra a partir del modelo cinemático del sistema de tracción, el cual facilita el movimiento del robot dentro de un entorno específico. En la configuración diferencial, la posición del robot puede estimarse utilizando ecuaciones geométricas derivadas de la interacción entre los componentes del sistema de tracción y los datos proporcionados por los codificadores rotativos ubicados en las ruedas [36].

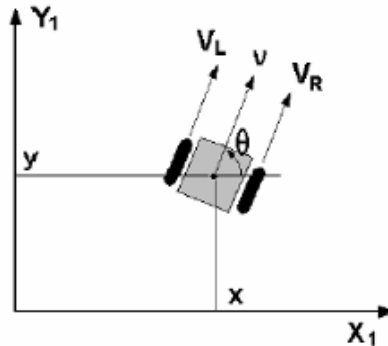


Figura 12. Posición del robot.
Fuentes: [38].

En la Figura 12, se observa la posición del robot en las coordenadas (x, y) , donde v representa la velocidad lineal del robot móvil, mientras que v_L y v_R denotan las velocidades tangenciales.

En el ámbito de los robots móviles industriales autónomos, un componente fundamental reside en la capacidad para la generación de trayectorias. Este proceso posibilita la toma de decisiones respecto a la selección del camino más apropiado para llevar a cabo una tarea, dado que se presenta una diversidad infinita de posibles rutas o movimientos que el robot puede ejecutar con el fin de desplazarse desde un punto A hasta un punto B. Se dispone de una amplia variedad de algoritmos diseñados para permitir al robot adoptar la ruta más eficiente, teniendo en consideración que la distancia a recorrer sea mínima y que se logre evadir obstáculos hasta alcanzar su destino [39].

IV-D1. Planificadores de trayectorias: Para que el robot pueda dar seguimiento a una trayectoria, debe abordar tres problemas fundamentales:

1. **Localización:** Este aspecto implica la determinación precisa de las coordenadas que describen la posición del robot en relación con un sistema de referencia en todo momento. Esto posibilita su ubicación en el entorno circundante de manera constante.

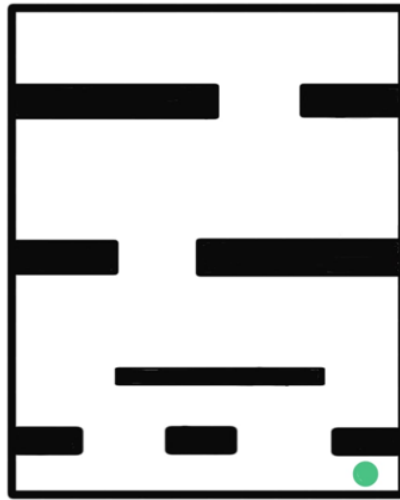


Figura 13. Localización de un robot móvil.
Fuentes: Autores.

2. **Planificación de tareas:** Consiste en la toma de decisiones con respecto a las acciones a ejecutar y la adquisición del conocimiento de las coordenadas del punto final para alcanzar la meta establecida.

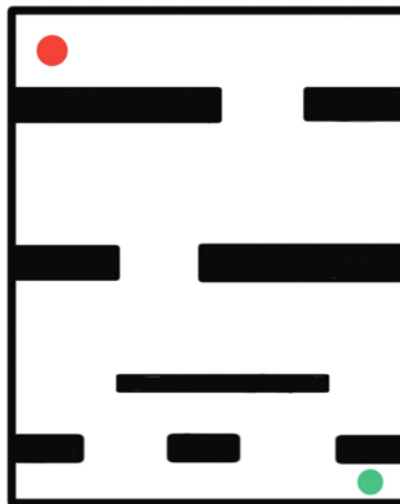


Figura 14. Planificación de tareas de un robot móvil.
Fuentes: Autores.

3. **Planificación de movimientos:** Se encarga de la generación automática de movimientos con el propósito de permitir que el robot se desplace eficazmente en su entorno, eludiendo cualquier obstáculo que pueda surgir en su trayectoria.

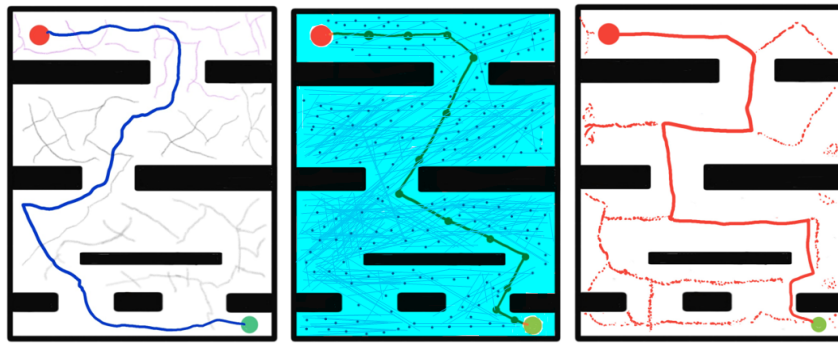


Figura 15. Planificación de movimientos de un robot móvil.
Fuentes: Autores.

El abordaje eficiente de estos tres problemas fundamentales resulta crucial para asegurar el desempeño óptimo de los robots móviles autónomos en la ejecución de tareas específicas en entornos dinámicos y complejos. Existen diversas formas de representar el entorno para planificar trayectorias, como el mapa de ruta (roadmap), la descomposición en celdas, y los grafos o nodos, dependiendo del algoritmo a implementar. Los planificadores pueden dividirse en globales o locales: los primeros tienen acceso a toda la información del entorno antes de iniciar, mientras que los segundos desconocen parte de la información del espacio de trabajo. En la siguiente Figura 16 ilustra una clasificación general de los algoritmos de planificación de trayectorias.

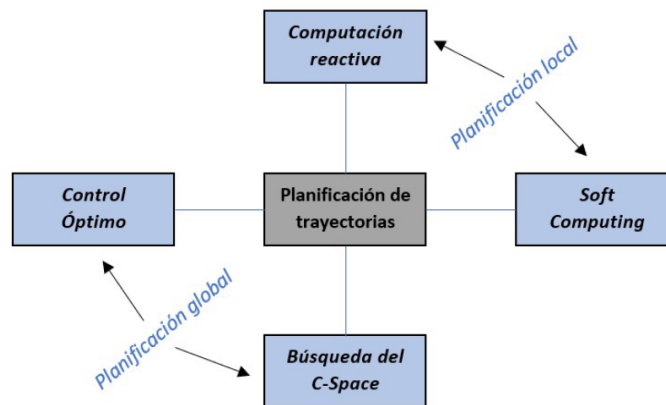


Figura 16. Clasificación general de métodos de planificación de trayectorias.
Fuentes: [39].

- Computación reactiva:** Los algoritmos de planificación de trayectorias de esta categoría se caracterizan por dividir el entorno en regiones ocupadas o libres con baja resolución y complejidad computacional. Se trata de un método de planificación local, ampliamente utilizado en entornos dinámicos debido a su capacidad para evadir obstáculos rápidamente. Los robots que emplean estos algoritmos necesitan estar equipados con sensores que detecten la presencia de obstáculos, para poder replanificar la trayectoria según sea necesario. Estos algoritmos se subdividen en maniobra reactiva, donde se realizan maniobras evasivas ante obstáculos, y métodos de optimización local, que ajustan la ruta planeada al detectar espacios ocupados.
- Soft Computing:** Estos algoritmos son utilizados en la planificación local y están diseñados para hallar soluciones que estén cerca de la óptima dentro de un margen de tolerancia establecido. Cuanto más cercana sea la solución a la óptima, mayor será el costo computacional implicado.

- **Búsqueda de C-Space:** Estos algoritmos se utilizan en la planificación global de trayectorias y tienen en cuenta todo el espacio accesible por el robot. Por esta razón, la mayoría de los trabajos de esta categoría se refieren al espacio de trabajo como el “C-Space”. Para implementar estos métodos, se discretiza el espacio y se realiza la búsqueda visitando los nodos que lo componen. El camino resultante es la unión de todas las muestras encontradas, y en ocasiones es necesario realizar un post-procesamiento para suavizar la trayectoria resultante. Esta categoría se divide en dos grupos de algoritmos según la discretización utilizada del “C-Space”: los métodos de búsqueda en grafos y los basados en el muestreo. Estos últimos pueden continuar buscando incluso después de encontrar una solución, con el objetivo de encontrar rutas mejores.
- **Control Óptimo:** Estos algoritmos se utilizan en la planificación global de trayectorias y se fundamentan en la creación de una función de control que guía al robot desde un estado inicial en el “C-Space” hasta su destino. A diferencia de los métodos de Soft Computing, estos enfoques se centran en una planificación de trayectorias óptima y se distinguen por carecer de parámetros configurables. Dentro de esta categoría, hay dos subcategorías: en la primera, se resuelven ecuaciones diferenciales parciales mediante el principio de programación dinámica, mientras que en la segunda se emplea la optimización numérica considerando las restricciones cinemáticas y dinámicas del robot.

IV-D2. Algoritmos de planificación de trayectorias: Existen varios algoritmos empleados para la planificación de trayectorias, cada uno tiene una ventaja sobre el otro según varios parámetros como la capacidad de procesamiento del robot o el tipo de entorno en el que vaya a realizar las tareas, los algoritmos más conocidos son:

- **Diagrama de Voronoi:** Es un método de planificación global que busca y encuentra trayectos no suaves que logran evadir los obstáculos, el entorno se lo representa por medio de la construcción de figuras geométricas que encierran los obstáculos [39]. Se utiliza el algoritmo de búsqueda A* para encontrar el mejor camino entre los puntos de partida y destino predeterminados teniendo en cuenta las distancias mínima y máxima permitidas entre nodos, se realiza un proceso de filtrado de los nodos redundantes en el camino.

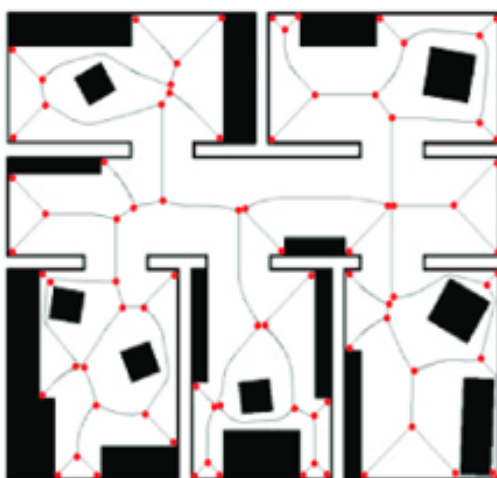


Figura 17. Diagrama de Voronoi.

Fuentes: [40].

- **Algoritmo A*:** Es una herramienta común en la planificación de rutas, especialmente en navegación basada en mapas. Utiliza una función heurística para estimar el costo de alcanzar el objetivo desde un nodo, priorizando aquellos que están más cerca del destino. En entornos de mapa, representa el terreno como una cuadrícula y busca la ruta más corta desde el punto de partida del robot hasta su objetivo, explorando las celdas según la heurística definida [41].

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|
| 7 | 6 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | | 19 | 20 | 21 | 22 | |
| 6 | 5 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | | 18 | 19 | 20 | 21 | |
| 5 | 4 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | 17 | 18 | 19 | 20 | |
| 4 | 3 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | 16 | 17 | 18 | 19 | |
| 3 | 2 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | 15 | 16 | 17 | 18 | |
| 2 | 1 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | | 14 | 15 | 16 | 17 | |
| 3 | 2 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | | 13 | 14 | 15 | 16 | |
| 4 | 3 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | | 12 | 13 | 14 | 15 | |
| 5 | 4 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | 10 | 11 | 12 | 13 | 14 |
| 6 | 5 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | |

Figura 18. Algoritmo A*.
Fuentes: [42].

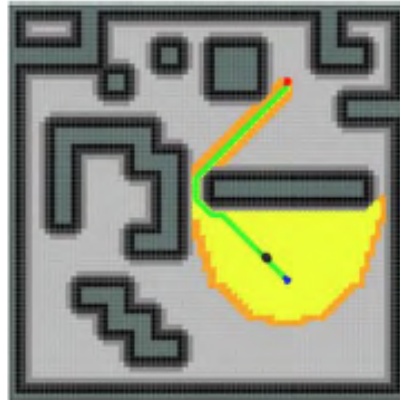


Figura 19. Planeación de trayectorias utilizando el algoritmo A*.
Fuentes: [43].

- Dijkstra:** El algoritmo de Dijkstra resuelve la ruta más corta en grafos finitos con complejidad temporal $O(n \log(n))$, donde n es el número de nodos. Ordena los nodos por costo creciente, visitando desde el punto de partida y expandiéndose hacia los más cercanos. Garantiza encontrar la ruta más corta sin aristas de costo negativo, explorando iterativamente los nodos no visitados. Su eficacia depende de estructuras de datos adecuadas y es fundamental en la optimización de rutas en redes de transporte y comunicación [41].

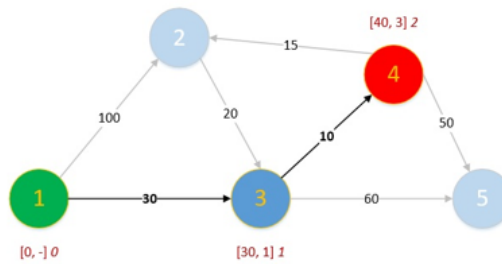


Figura 20. Algoritmo de Dijkstra.
Fuentes: [44].



Figura 21. Planeación de trayectorias utilizando el algoritmo de Dijkstra.

Fuentes: [43].

- Grafo de visibilidad:** Se representa el entorno mediante figuras geométricas, donde cada vértice visible de los obstáculos se guarda en el grafo junto con las aristas que son conexiones entre cada nodo. Este algoritmo ayuda a encontrar el camino óptimo en distancia entre vértices garantizando de esta manera la evasión del obstáculo presentado [39].

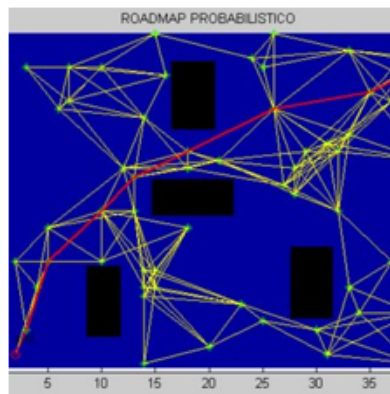


Figura 22. Grafo de visibilidad.

Fuentes: [45].

- **Algoritmo AMCL:** El algoritmo Adaptive Monte Carlo Localization (AMCL) es un algoritmo básico de robótica que se utiliza para determinar la orientación y posición de un robot dentro de un entorno previamente mapeado. Mediante una técnica de filtro de partículas, el AMCL utiliza una colección de partículas para representar la pose del robot. Estas partículas se modifican continuamente en función de las percepciones y el modelo de movimiento del robot, convergiendo finalmente a su posición verdadera. Su capacidad para modificar dinámicamente la cantidad y distribución de partículas en respuesta a la incertidumbre del movimiento y la retroalimentación sensorial explica su naturaleza “adaptativa”, que garantiza una localización precisa incluso en entornos intrincados y cambiantes. El AMCL se utiliza ampliamente en aplicaciones de mapeo, localización y navegación autónoma [46].

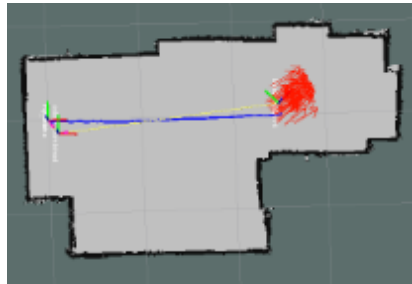


Figura 23. Algoritmo AMCL.

Fuentes: Autores.

V. MARCO METODOLÓGICO

El enfoque del proyecto se centra en desarrollar un sistema robótico móvil con la capacidad de realizar la cartografía de un entorno dado y de planificar de manera autónoma una ruta desde un punto A hasta un punto B. Esta planificación de ruta se lleva a cabo de manera que el robot evite de manera efectiva cualquier obstáculo presente en el trayecto. En la Figura 24, se presenta el diagrama general de un robot móvil autónomo, detallando los componentes esenciales que se emplearán en su construcción. Este robot está diseñado con un chasis fabricado en una aleación de aluminio anodizado, lo que garantiza una estructura robusta y ligera. Además, integra un sensor LIDAR, basado en tecnología láser infrarrojo, destinado a mapear con precisión los entornos circundantes. Asimismo, cuenta con placas de expansión que facilitan la conexión entre los componentes electrónicos y mecánicos del robot, permitiendo una integración eficiente y sin problemas. En términos de movilidad, el robot está equipado con ruedas diferenciales de goma y motores con encóder, proporcionando no solo tracción adecuada, sino también la capacidad de determinar con precisión la ubicación del robot en su entorno. Este diseño meticulosamente elaborado garantiza un rendimiento óptimo y una operación autónoma efectiva del robot móvil en una variedad de situaciones y entornos.

V-A. Diseño mecánico del robot

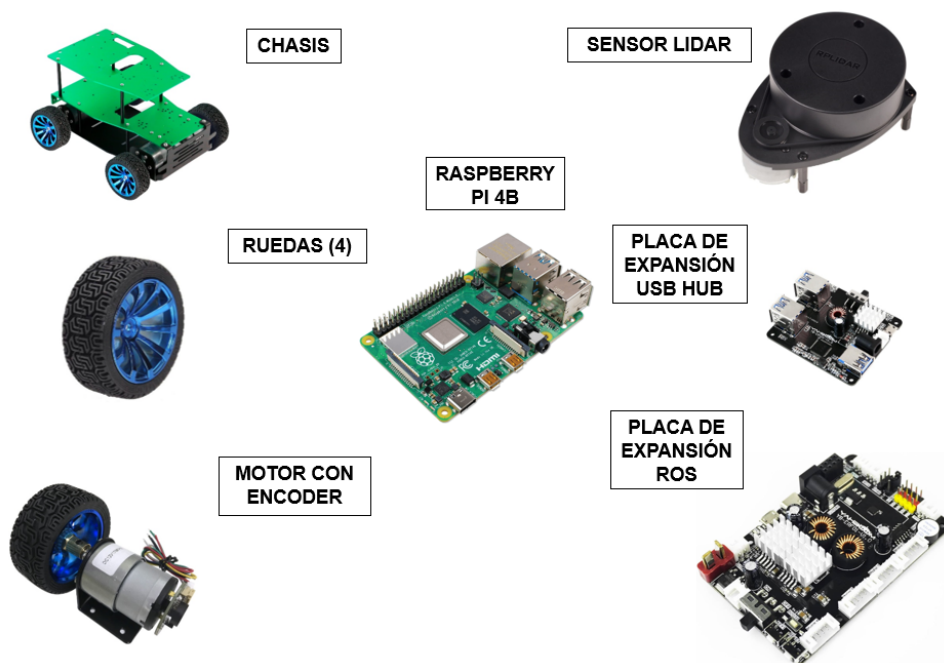


Figura 24. Diagrama general del robot.

Fuentes: Autores.

V-A1. Chasis del robot: Para el diseño del chasis se ha optado en emplear un diseño con tres niveles, con el propósito de lograr una distribución de los elementos electrónicos, garantizando así un desempeño sin contratiempos en sus respectivas funciones. Para el material del chasis, se ha elegido utilizar de aleación de aluminio anodizado, debido a su alta rigidez y resistencia, con el objetivo de asegurar una durabilidad superior.

La estructura mecánica del robot debe cumplir condiciones importantes que permitan el correcto desplazamiento de este, además de soportar el peso total de sus componentes como sus placas de expansión, batería, controlador y motores.

Considerando lo mencionado previamente, se realizarán análisis estáticos de la estructura mecánica del robot, garantizando un diseño seguro y óptimo. Estos análisis se realizarán mediante un software de modelado CAD, en

el cual se podrán comprobar la resistencia de la estructura tomando en cuenta las deformaciones y tensiones.

En primer lugar, se debe obtener la carga que soportará la estructura del robot móvil tomando en cuenta todos sus componentes. Se obtuvo un peso de $3200g$ o $3,2[kg]$, con esto se puede calcular la fuerza que ejerce sobre la estructura del robot.

$$N = W$$

$$W = m * g$$

$$W = (3,2) * (9,81)$$

$$W = 31,40N$$

■ Análisis estático - Tensión

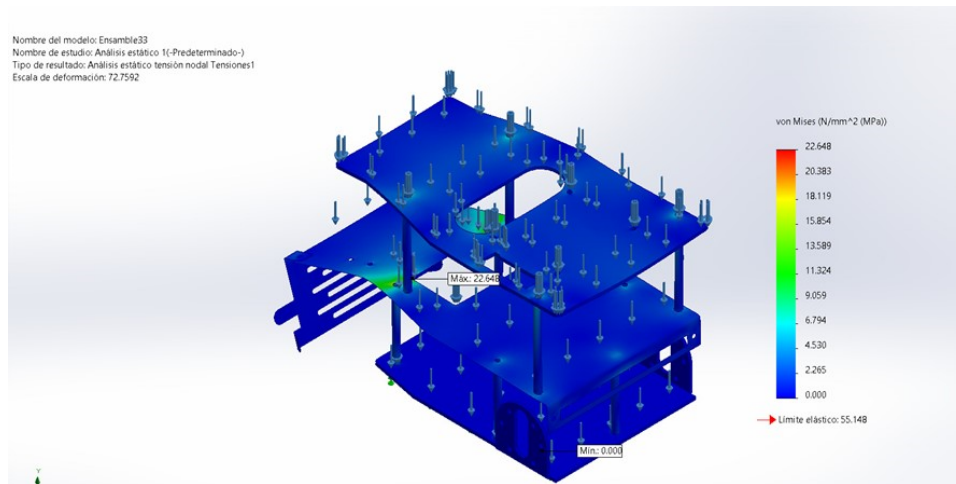


Figura 25. Análisis estático - Tensión.

Fuentes: Autores.

En la Figura 25, se puede observar que el valor de von Mises máximo es menor al límite elástico del material lo que garantiza su seguridad. Se puede apreciar que el análisis muestra un valor de $22,65[MPa]$ del esfuerzo de von Mises lo que es menor al límite elástico del material usado en este caso aleación de aluminio anodizado, siendo de $55,14[MPa]$, cumpliendo así con la condición de que el esfuerzo de von Mises máximo es menor que el límite elástico del material.

■ Análisis estático - Factor de seguridad

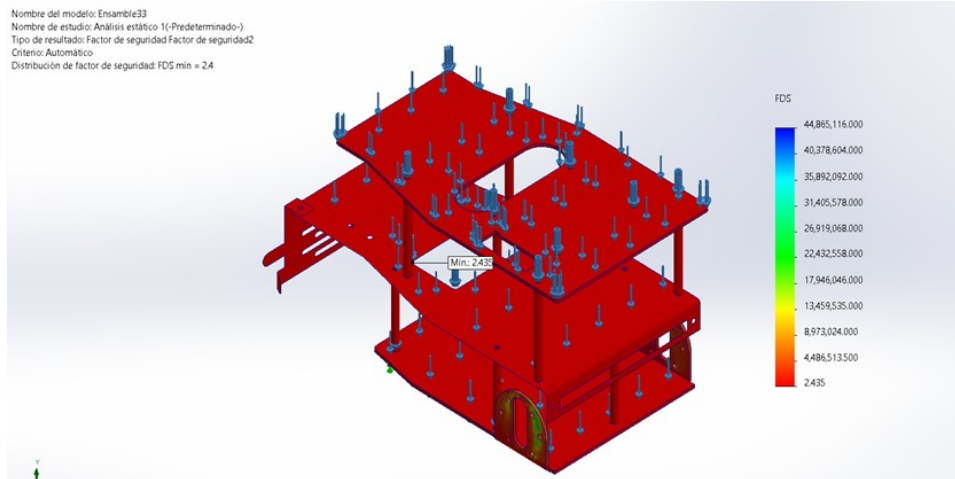


Figura 26. Análisis estático - Factor de seguridad.
Fuentes: Autores.

En la Figura 26, se aprecia el factor de seguridad de la estructura del robot la cual tiene un valor de 2,4 siendo mayor a 1, indicando la seguridad del mismo, este valor se obtiene dividiendo el límite elástico de 55,14[MPa] para el valor máximo del esfuerzo de von Mises siendo este de 22,65[MPa].

V-A2. *Motores:* Para determinar y seleccionar el motor adecuado para el robot, se llevaron a cabo una serie de pasos los cuales se fundamentan en el cálculo de motores para el robot móvil. Estos procedimientos implican tener en cuenta el peso total del robot, su carga máxima y la velocidad nominal [47].

Tabla IV
 TABLA DE DATOS #1

| Parámetros | Datos del robot |
|-------------------------|-----------------|
| Peso del robot [kg] | 1.328 |
| Factor de seguridad (%) | 50 |
| Carga máxima [kg] | 1.872 |
| Velocidad nominal [m/s] | 0.1 |

A continuación, es necesario determinar las revoluciones por minuto (RPM) del motor teniendo una velocidad específica y el diámetro de la rueda, empleando la siguiente ecuación 2.

$$\frac{60 * v}{\pi * D} = \frac{60 * 0,1}{\pi * 0,065} = 31,83098RPM \quad (2)$$

Tabla V
 TABLA DE DATOS #2

| Parámetros | Datos del robot |
|--------------------------|-----------------|
| Diámetro de llanta [m] | 0.065 |
| Velocidad nominal [m/s] | 0.1 |
| RPM necesarios | 31.83098 |
| Gravedad $\frac{m}{s^2}$ | 9.81 |

A continuación, se calcula la cantidad de fuerza requerida para desplazar el motor, y se emplea la ecuación 3.

$$F_T = g * (m_R + m_L) = 9,81 * (1,992 + 1,872) = 31,392N \quad (3)$$

Tabla VI
TABLA DE DATOS #3

| Parámetros | Datos del robot |
|-------------------|-----------------|
| Carga máxima [kg] | 1.872 |
| Fuerza [N] | 31.392 |

Finalmente, es crucial comprender el torque generado por el motor para impulsar el robot. Para lograrlo, se utiliza la fuerza calculada en el paso 3 junto con el diámetro de la llanta, lo que permitirá obtener el torque empleando la ecuación 3 en $Kg * cm$ ya que es el sistema comúnmente utilizado al buscar las especificaciones de un motor.

$$T_T = \frac{D}{4} * F_T = 0,47088N * m \quad (4)$$

Tabla VII
TABLA DE DATOS #4

| Parámetros | Datos del robot |
|-------------------------|-----------------|
| Diámetro de llanta [m] | 0.065 |
| Fuerza [N] | 31.392 |
| Torque [N * m] | 0.47088 |
| Torque [kg * cm] | 4.7088 |

Una vez obtenidos los cálculos y parámetros necesarios para la selección de los motores, se procede a crear una tabla de especificaciones que incluya varios modelos de motores. El propósito de esta Tabla V-A2 es facilitar la elección del motor que mejor se adapte a los resultados previamente calculados. Esto permite una visualización clara de los parámetros que se necesitan evaluar y así seleccionar el correcto.

Tabla VIII
SELECCIÓN DE MOTORES

| EVALUACIÓN | JGB37-520 | CAL | Motor RF 130 | CAL | Motor FK 370 | CAL |
|---------------------|---------------|-----|-----------------|-----|-----------------|-----|
| Potencia | 4 [w] | 4 | 1 - 5 [w] | 3 | 5 - 20 [w] | 5 |
| Eficiencia | 50 % | 5 | 20 % | 2 | 20 % | 2 |
| Velocidad con carga | 330 RPM | 3 | 4500 - 1500 RPM | 4 | 6000 - 4500 RPM | 5 |
| Torque | 4.4 [kg * cm] | 5 | 10 [g * cm] | 2 | 40 [g * cm] | 2 |
| TOTAL | 17 | | 11 | | 14 | |
| APROBACIÓN | Si | | No | | No | |

V-A3. *Ruedas:* En el contexto de este robot móvil, se presentan diversas opciones para la selección de sistemas de propulsión, como las ruedas omnidireccionales, que confieren al robot la capacidad de movimiento en múltiples direcciones, o la alternativa de reemplazar las ruedas por orugas para una tracción mejorada. Sin embargo, en este caso particular se ha optado por la adopción de ruedas de goma. La justificación reside en que, al tratarse del primer prototipo, las ventajas inherentes a las alternativas mencionadas resultarían irrelevantes en relación con las funciones específicas que el robot ejecutará. Esto deja las otras configuraciones de ruedas como posibles mejoras a considerar en etapas posteriores.



Figura 27. Ruedas.
Fuentes: [48].

V-B. *Diseño electrónico del robot*

En este apartado se hablará sobre los sistemas del diseño eléctrico y electrónico del robot ilustrados en la Figura 24 vista antes en los fundamentos teóricos.

V-B1. *Sensor LIDAR:* Se dispone de diversas categorías de sensores LIDAR (Laser Imaging Detection and Ranging) destinados al escaneo y teledetección utilizando la emisión laser para mapear espacios en tres dimensiones, sin embargo, en la concepción de este robot se ha determinado la incorporación del sensor denominado SLAM A1 LIDAR. Esta decisión se fundamenta en la característica de mayor alcance intrínseca al sensor SLAM A1 LIDAR en contraposición con otras variantes de sensores LIDAR. Un atributo adicional que respalda esta elección es la capacidad que ostenta el sensor SLAM A1 LIDAR para ejecutar sus operaciones de manera eficiente en contextos con niveles considerables de iluminación, contrariamente a la limitación que presentan otras alternativas de sensores en tales circunstancias [49].

Tabla IX
SELECCIÓN DE SENSORES LIDAR

| EVALUACIÓN | SLAM A1M8 LIDAR | CAL | YDLIDAR X3 Pro LIDAR | CAL | C1 LIDAR | CAL |
|------------------------------|-----------------|-----|----------------------|-----|--------------|-----|
| Frecuencia de muestreo | 8000 times/S | 5 | 4000 times/S | 3 | 5000 times/S | 4 |
| Frecuencia de escaneo | 10Hz | 5 | 5Hz | 3 | 8Hz | 4 |
| Resolución angular | $\leq 1^\circ$ | 5 | 0.6° | 4 | 0.72° | 3 |
| Distancia mínima de medición | 0.15m | 3 | 0.12m | 4 | 0.05m | 5 |
| Radio de medición | 12m | 4 | 8m | 3 | 12m | 4 |
| TOTAL | 22 | | 17 | | 20 | |
| APROBACIÓN | Si | | No | | No | |

El sensor SLAM A1M8 LIDAR combina la funcionalidad de mapeo y localización simultánea (SLAM, por sus siglas en inglés). Como se especifica en la Tabla V-B1 este LIDAR utiliza pulsos láser para escanear y mapear entornos tridimensionales con una rapidez y precisión impresionantes con una frecuencia de muestreo de 8000 times/S, frecuencia de escaneo de 10Hz y una resolución angular de igual o menos a 1° . Su capacidad para generar mapas detallados y localizar la posición en tiempo real lo convierte en una herramienta fundamental en aplicaciones de robots autónomos en entornos complejos. Con su amplio rango de detección desde 0.15 metros hasta 12 metros y su capacidad para operar en diversas condiciones ambientales.

V-B2. Controlador: Este elemento desempeña un papel esencial al posibilitar la comunicación y la gestión eficiente entre diversos componentes de un sistema, ya sea en el ámbito del hardware de computadoras personales o en entornos más complejos y automatizados.

Tabla X
SELECCIÓN DE PLACAS DE CONTROL

| EVALUACIÓN | RASPBERRY PI 4B | CAL | ARDUINO | CAL | ESP32 | CAL |
|------------------------|--------------------------------------|-----|---|-----|--|-----|
| Procesador | Quad-Core ARM Cortex-A72 a 1.5GHz | 5 | ATmega328P (16 MHz) | 2 | Doble núcleo Xtensa LX6 (240Mz) | 4 |
| Memoria | 4GB de LPDDR4 SDRAM | 5 | 2KB de memoria SRAM y 32KB de memoria flash | 1 | 520KB de SRAM y 448KB memoria ROM | 4 |
| Número I/O digitales | Cabezal GPIO de 40 pines | 4 | 14 pines/6 salidas PWM | 5 | 36 pines/PWM, I2C, SPI | 4 |
| Compatibilidad con ROS | Robot Operating System | 5 | Mínima (con librerías) | 1 | Mínima (mayor trabajo y configuración) | 3 |
| Alimentación | 5V/3A vía USB-C, 5V vía cabezal GPIO | 4 | 7-12 V | 5 | 5V y 12V | 4 |
| TOTAL | 23 | | 14 | | 19 | |
| APROBACIÓN | Si | | No | | No | |

Como se detalla en la Tabla V-B2 el Raspberry Pi 4 sobresale con un procesador quad-core ARM Cortex y 4GB de memoria, superando al Arduino UNO y al ESP32 en potencia de cálculo y capacidad para ejecutar sistemas operativos completos. Sus 40 pines de E/S digitales ofrecen versatilidad para conectar dispositivos, superando la capacidad de pines del Arduino UNO y el ESP32. Esta plataforma permite aplicaciones más complejas como es la

compatibilidad con el sistema ROS y una mayor flexibilidad en entornos de desarrollo.

V-B3. Placa de control de robot para Raspberry Pi: Esta placa de control presentada en la Figura 28 integra la unidad de control central STM32, el sensor MPU9250 que sirve para realizar rastreo de movimiento de 9 ejes combinando un giroscopio, un acelerómetro y un magnetómetro de 3 ejes cada uno, esto hace que el robot pueda saber su posición exacta en la que se encuentra en un entorno de trabajo.

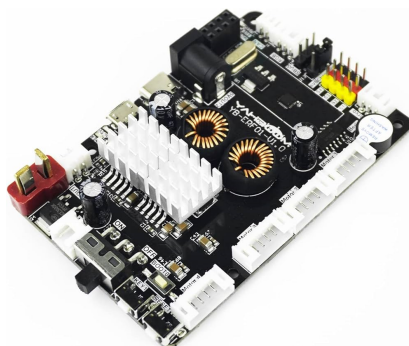


Figura 28. Placa de expansión “ROS” para robot.
Fuentes: [50].

La placa permite una comunicación con el Raspberry Pi 4 y a su vez permite suministrar energía de 5V a 12V, cuenta con comunicación por bus CAN y lo más importante cuenta con 4 canales de comunicación para motores con encoder lo que ayuda a minimizar las conexiones con el Raspberry Pi directamente.

Tabla XI
SELECCIÓN DE TARJETA DE POTENCIA

| EVALUACIÓN | SABERTOOTH 2.5 | CAL | ROS Robot Control Board | CAL | Driver H-Bridge Dual L9110S | CAL |
|------------------------------|---------------------------|-----|--|-----|-----------------------------|-----|
| Potencia | Corrientes máximas de 50A | 4 | 12V DC y 5V DC | 4 | 2.5V a 12V | 2 |
| Integración de sensores IMU | No posee | 2 | Navegación, control del robot o la estimación del estado | 4 | No posee | 1 |
| Número de canales de control | 2 canales de control | 5 | 4 canales de control de 12V | 4 | 2 canales de control | 3 |
| Eficiencia energética | 90 % | 4 | 95 % | 5 | 80 % | 2 |
| TOTAL | 15 | | 17 | | 8 | |
| APROBACIÓN | No | | Si | | No | |

El módulo ROS Robot Control Board, compatible con ROS, integra sensores IMU para detectar movimiento y orientación. Destaca por su potencia para cargas demandantes, eficiencia energética y capacidad de ejecutar algoritmos complejos como se especifica en la Tabla V-B3, siendo ideal para aplicaciones robóticas exigentes en rendimiento y autonomía.

V-B4. Placa de expansión HUB USB.0 1: La presente placa de concentrador USB está concebida para ser empleada por fabricantes, entusiastas tecnológicos y desarrolladores en el ámbito de la robótica. Esta placa se encuentra en consonancia con las placas de desarrollo pertenecientes a la serie Raspberry Pi. Su diseño incorpora el chip VL817 HUB y ofrece 4 puertos de salida USB3.0, esto hace que la velocidad de transmisión pueda alcanzar 500

MB/S. Además, dispone de interfaces de fuente de alimentación que incluyen micro USB, DC5.5*2.5 y XH2.54, lo que permite admitir una variedad de fuentes de alimentación externa con un amplio rango de voltajes. En términos de capacidad, esta placa es capaz de manejar una corriente total máxima de hasta 5A. En comparación con los dispositivos USB HUB convencionales disponibles en el mercado, esta placa tiene la ventaja de proporcionar una transmisión de datos más constante, evitando interrupciones en la transmisión ocasionadas por una insuficiente corriente eléctrica.

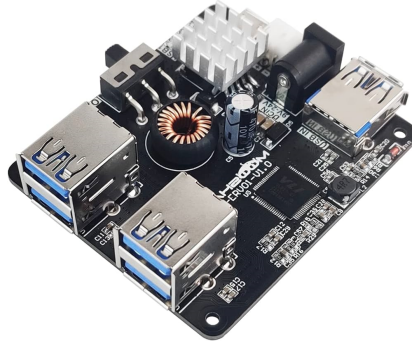


Figura 29. Placa de expansión HUB USB.0 1 a 4 canales.

Fuentes: [50].

V-B5. Fuente de alimentación: Considerando los elementos que compondrán el robot móvil, se hace imperativa la disponibilidad de una fuente de energía que pueda ofrecer tanto un voltaje como una corriente suficiente para garantizar el funcionamiento óptimo de los diversos componentes eléctricos, con especial énfasis en los motores. La entrega de corriente adecuada es crucial para asegurar la potencia necesaria que permita el desplazamiento del robot. A raíz de este análisis, se ha llegado a la determinación de que una batería de litio con una capacidad de 12 V y 6000mAh sería la elección idónea para el funcionamiento fluido del robot, sin que se presenten contratiempos debido a deficiencias en el suministro energético.

Tabla XII
CONSUMO ENERGÉTICO DE LOS COMPONENTES

| CONSUMO ENERGÉTICO | | |
|--------------------|----------------------------|---------------------------|
| Cantidad | Componentes | Corriente consumida (A/h) |
| 1 | Placa de expansión ROS | 0.05 |
| 1 | Placa de expansión USB HUB | 0.09 |
| 1 | Sensor LIDAR A1M8 | 1.2 |
| 1 | Raspberry PI 4 | 3 |
| 4 | Motor JGB37-520 | 1.2 |
| TOTAL | | 5.54 |

Tabla XIII
SELECCIÓN DE BATERÍAS

| EVALUACIÓN | Paquete de baterías de Litio | CAL | Batería LIPO 3S Turnigy | CAL | Batería LIPO 3S Tattu | CAL |
|----------------------|------------------------------|-----|-------------------------|-----|-----------------------|-----|
| Capacidad | 6000 mA/h | 5 | 450 mA/h | 3 | 450 mA/h | 3 |
| Peso | 300g | 5 | 51g | 4 | 42g | 3 |
| Voltaje | 12.6V | 5 | 11.1V | 4 | 11.1V | 4 |
| Tamaño | 68x55x37mm | 4 | 56x31x17mm | 5 | 62x17x19mm | 4 |
| Conector de descarga | T-plug +4017 female | 5 | XT30 | 5 | XT30 | 5 |
| TOTAL | 24 | | 21 | | 19 | |
| APROBACIÓN | Si | | No | | No | |

El paquete de baterías de litio de 6 amperios (6A) es el que se implementará debido a que es liviano y de dimensiones pequeñas por lo cual se acoplan perfectamente dentro de la estructura del robot autónomo. Este paquete de baterías ofrece una alta densidad de energía, lo que significa que pueden proporcionar una mayor potencia en un tamaño más pequeño en comparación con las otras baterías analizadas anteriormente.

V-C. Software

V-C1. Algoritmos de mapeo: Esta sección tratará de los distintos algoritmos SLAM que se emplean para mapear el entorno, así como de las pruebas que se realizan para evaluar el algoritmo de navegación autónoma. Debido a su versatilidad, el software desempeña un papel vital tanto en su función como en su aplicación. Permite el procesamiento de datos y la automatización de procesos mediante la ejecución de algoritmos e instrucciones que maximizan el rendimiento del robot. A continuación se mencionará los distintos algoritmos que se pueden aplicar, en este caso son: Hector mapping, Gmapping mapping y Cartographer mapping. Sin embargo, para garantizar la precisión y eficacia en estas tareas, es esencial seleccionar el algoritmo de mapeo adecuado. En este marco, esta idea pretende comparar diferentes algoritmos de mapeo utilizados por robots autónomos que se enfrentan a obstáculos visuales comparables pero que poseen características únicas y debilidades intrínsecas.

- **Hector mapping:** Este sistema se emplea para mapear un entorno integrando el sensor LIDAR. La determinación del estado actual del sistema se basa en los resultados de exploraciones previas; sin embargo, en su fase inicial, enfrenta desviaciones significativas. Estas desviaciones no afectan la continuación del proceso de mapeo, pero sí interfieren con la estimación de la posición futura del robot móvil. Hector SLAM compara las trayectorias que genera con las trayectorias de marcos de referencia anteriores, con el objetivo de calcular las rotaciones y traslaciones que surgen del desplazamiento del robot móvil desde su punto inicial [51].

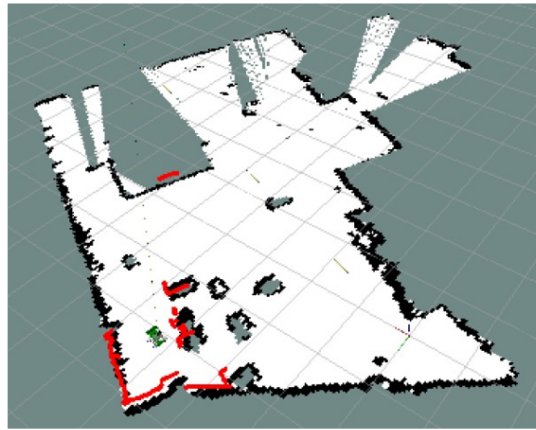


Figura 30. Algoritmo Hector mapping.

Fuentes: Autores.

- Gmapping mapping:** Su método se basa en la creación de mapas con cuadrículas y gráficos de ocupación RBPF. Además, este enfoque proporciona adaptabilidad en la configuración de los parámetros de creación de mapas, como el umbral mínimo de muestreo, la cantidad de partículas con Rao-Blackwellización (RBPF) [52], el cual realiza cálculos para determinar la posición real del robot directamente a partir de la probabilidad ofrecida por los datos recopilados previamente, junto con la ayuda de la postura actual del robot y los mapas previamente generados y los pasos de desplazamiento, por lo que puede personalizarse para satisfacer los requisitos y objetivos únicos de la aplicación solicitada [53].



Figura 31. Algoritmo Gmapping mapping.

Fuentes: Autores.

- **Cartographer mapping:** Este algoritmo se apoya en la fusión de dos enfoques de SLAM 2D distintos: uno opera a nivel local y el otro a nivel global, ambos empleando un sensor LiDAR y optimizados de manera independiente. El SLAM local se centra en la recopilación y construcción de submapas, entre los cuales se incluye la recopilación y alineación de múltiples escaneos en relación con la posición inicial. Estos submapas se configuran como una malla de puntos con una resolución específica y se les asigna una probabilidad que indica la posibilidad de que uno de sus puntos esté ocupado. Dicha probabilidad se basa en mediciones anteriores y se mantiene a medida que se generan más submapas. Una vez que se crea un submapa, se somete a un algoritmo que busca la posición óptima para que coincida con los demás submapas, y luego se extrapolan los restantes [51].

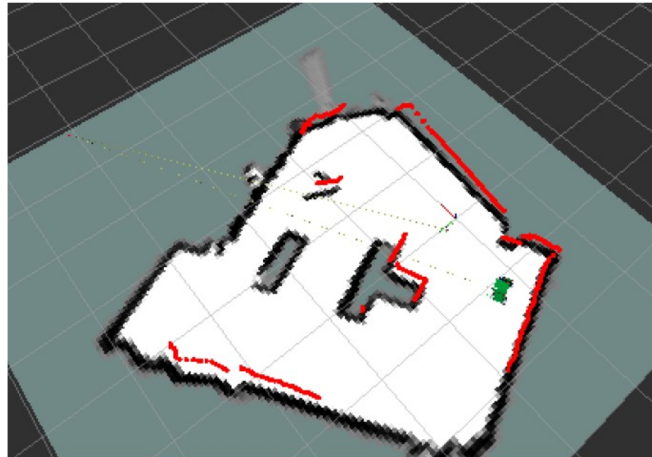


Figura 32. Algoritmo Cartographer mapping.
Fuentes: Autores.

Diagrama de flujo de los distintos algoritmos de mapeo

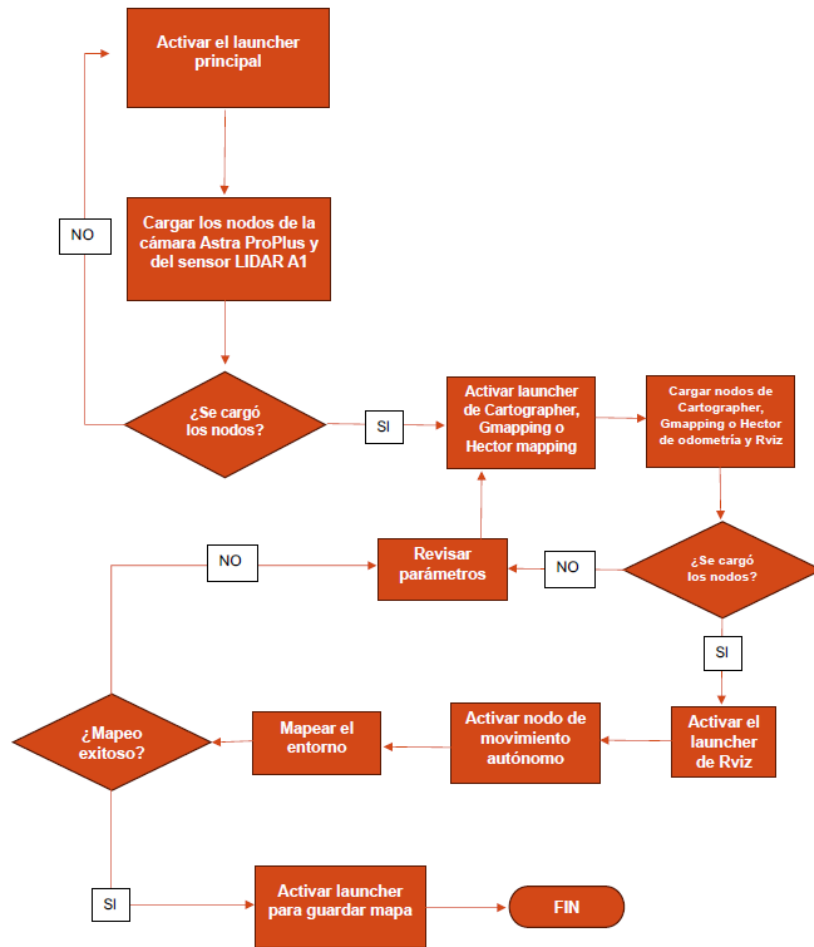


Figura 33. Diagrama de flujo de los distintos algoritmos de mapeo.

Fuentes: Autores.

V-C2. Algoritmo de planificación de trayectorias:

- **Algoritmo RRT_exploration:** Se elige el algoritmo RRT debido a su propensión a explorar áreas desconocidas. En este contexto, el algoritmo RRT se emplea principalmente para generar puntos límite, lo cual resulta muy útil para explorar estos puntos. Se define un punto límite como el punto de unión entre las áreas exploradas y las desconocidas. En esta definición, todas las áreas exploradas sin obstáculos se registran como 0, las áreas con obstáculos como 1 y las áreas desconocidas como -1. Inicialmente, se registra toda el área como -1.

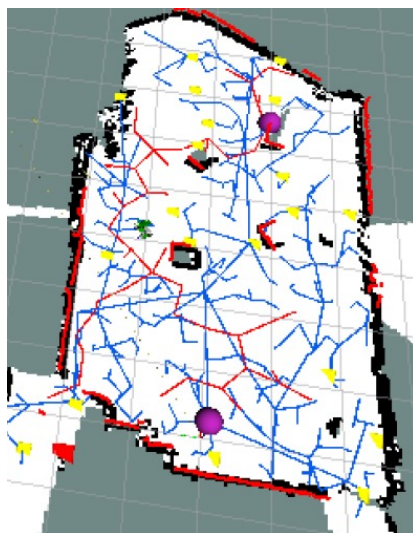


Figura 34. Algoritmo de planificación de trayectorias RRT_exploration.
Fuentes: Autores.

En la siguiente Figura 35 se muestra el diagrama de flujo del algoritmo RRT el cual ha sido implementado para que el robot móvil planifique rutas de navegación.

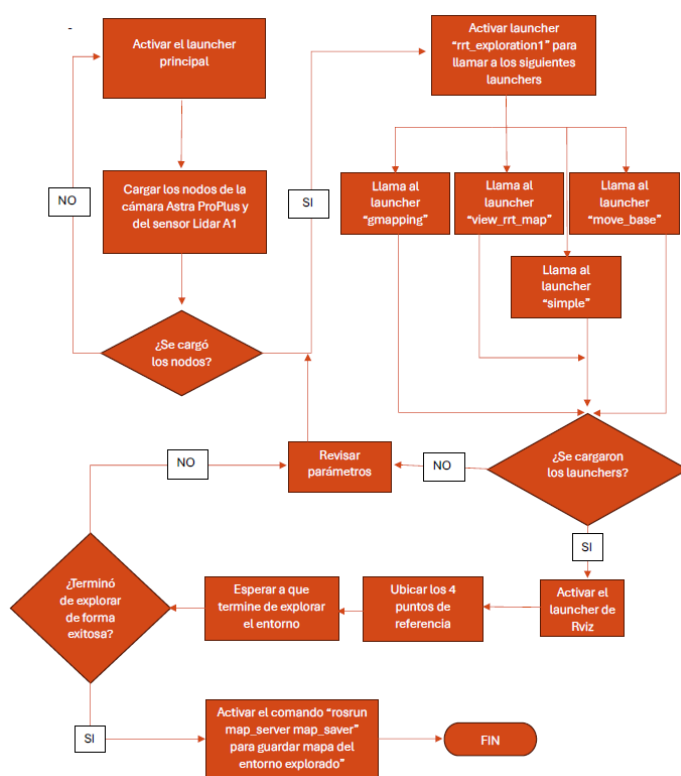


Figura 35. Diagrama de flujo del algoritmo RRT_exploration.
Fuentes: Autores.

V-C3. *Algoritmo de navegación:*

- **AMCL:** El Algoritmo de Monte Carlo Adaptativo constituye un método probabilístico de localización para un robot móvil bidimensional. Este emplea un enfoque adaptativo KLD para actualizar partículas y utiliza filtros de partículas para seguir la posición del robot en base a un mapa que se genera anteriormente. En su implementación actual, este nodo solo admite escaneos y mapas láser, aunque puede ser ampliado para procesar otros datos de sensores. AMCL recibe mapas basados por medio del sensor LIDAR (escaneo láser) e información de transformación, generando estimaciones de la postura del robot. Al iniciarse, AMCL inicializa su filtro de partículas conforme a los parámetros proporcionados. Es importante tener en cuenta que, por configuración predeterminada, si no se establecen parámetros, el filtro se inicia con una nube de partículas de tamaño mediano centrada en x,y,z (0,0,0).

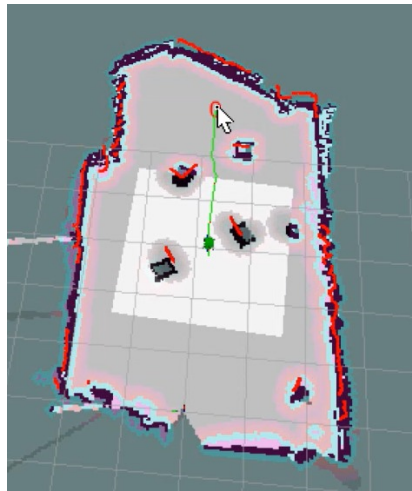


Figura 36. Algoritmo de navegación AMCL.

Fuentes: Autores.

Como se puede observar en la Figura 37 se presenta el diagrama de flujo del algoritmo AMCL el cual ha sido implementado con el propósito de permitir que el robot móvil se mueva de un punto A hacia un punto B.

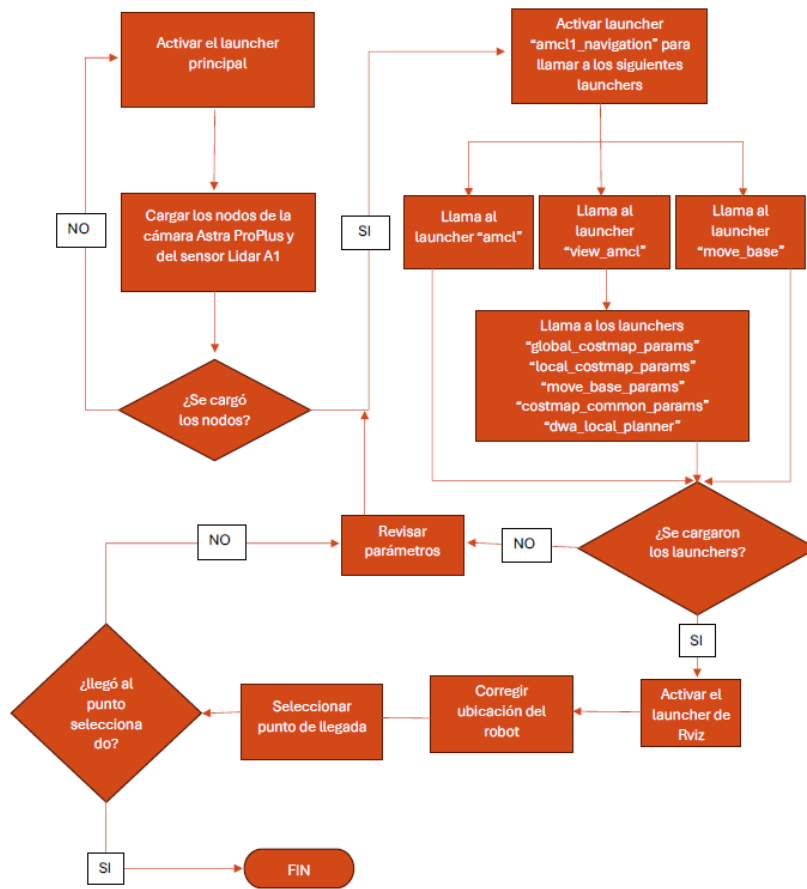


Figura 37. Diagrama de flujo del algoritmo AMCL.
Fuentes: Autores.

VI. RESULTADOS

En esta sección se mostrarán los resultados de las pruebas que se realizaron para medir el rendimiento de cada tipo de mapeo según su velocidad y la distribución de obstáculos en el área de prueba. Se comparó el tiempo que le tomó a cada algoritmo en completar el entorno propuesto, de igual manera se hizo una comparación con los mapas obtenidos visualizando cual dio un mejor resultado.

VI-A. Resultados de los algoritmos de mapeo

Se presentan los resultados obtenidos de las pruebas realizadas sobre cada uno de los algoritmos de mapeo. Dichos resultados abarcan mediciones de velocidad lineal y angular, así como el tiempo requerido para generar el mapa del entorno. Además, se registran los errores detectados y se asigna una calificación en una escala del 0 al 5, donde un puntaje de 5 indica un desempeño óptimo sin errores en el proceso de mapeo, completado en un tiempo adecuado. El área del entorno controlado es de $32m^2$.

VI-A1. Pruebas de mapeo con algoritmo Gmapping mapping:

1. Prueba #1 en entorno controlado #1

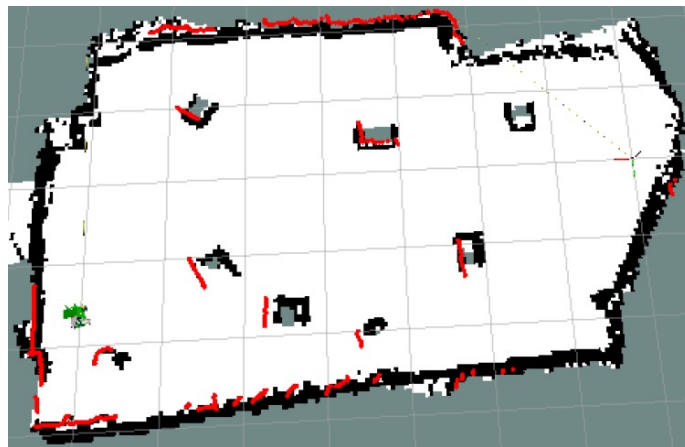


Figura 38. Prueba #1 algoritmo Gmapping mapping.
Fuentes: Autores.

2. Prueba #2 en entorno controlado #2

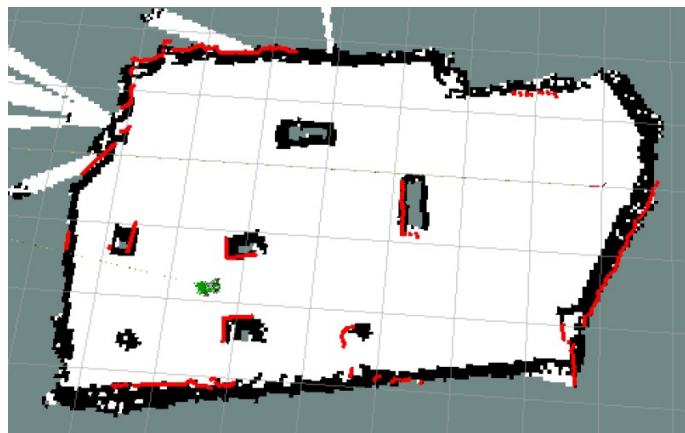


Figura 39. Prueba #2 algoritmo Gmapping mapping.
Fuentes: Autores.

3. Prueba #3 en entorno controlado #3

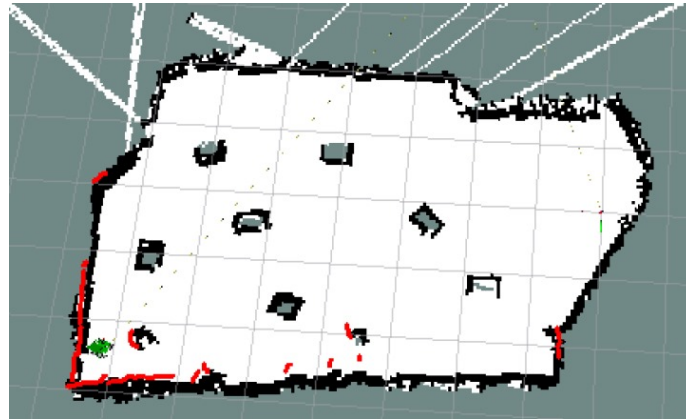


Figura 40. Prueba #3 algoritmo Gmapping mapping.
Fuentes: Autores.

Tabla XIV
RESULTADOS DEL ALGORITMO GMAPPING MAPPING

| GMAPPING MAPPING | Velocidad lineal [m/s] | Velocidad angular [rad/s] | Tiempo [S] | Errores | Calificación |
|------------------|------------------------|---------------------------|------------|---------|--------------|
| Prueba #1 | 0.1 | 0.15 | 1:56.65 | No | 4 |
| Prueba #2 | 0.2 | 0.25 | 1:27.60 | No | 5 |
| Prueba #3 | 0.3 | 0.35 | 0:40.36 | No | 4 |

En la evaluación de las pruebas llevadas a cabo con el algoritmo Gmapping, se han registrado los tiempos que el robot requirió para generar un mapa preciso del entorno. Además, se han observado las velocidades angulares y lineales utilizadas en cada prueba, así como la puntuación obtenida en cada una de ellas. En la primera y tercera prueba, se identificaron pequeños espacios casi despreciables que el algoritmo no mapeó. A pesar de esto, no se los considera errores significativos, ya que no afectan las ubicaciones de los obstáculos en el mapa. Sin embargo, debido a estas anomalías, no se les asignó la calificación más alta. En contraste, en la segunda prueba, se obtuvo un mapa sin errores y se logró completar el mapeo del entorno en un tiempo notablemente corto. Por lo tanto, esta prueba se ponderó con una calificación sobresaliente de 5/5.

VI-A2. Pruebas de mapeo con algoritmo Hector mapping:

1. Prueba #1 en entorno controlado #1

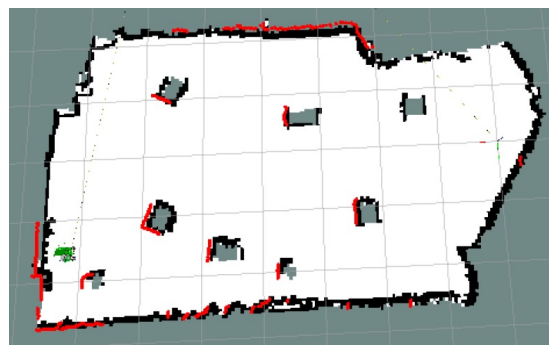


Figura 41. Prueba #1 algoritmo Hector mapping.
Fuentes: Autores.

2. Prueba #2 en entorno controlado #2

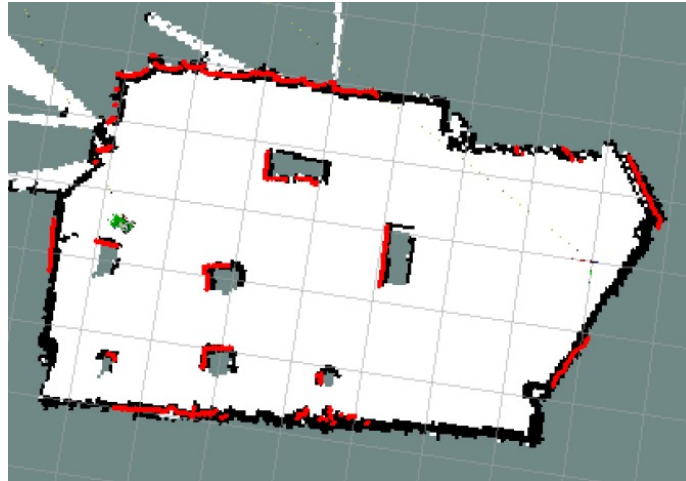


Figura 42. Prueba #2 algoritmo Hector mapping.
Fuentes: Autores.

3. Prueba #3 en entorno controlado #3

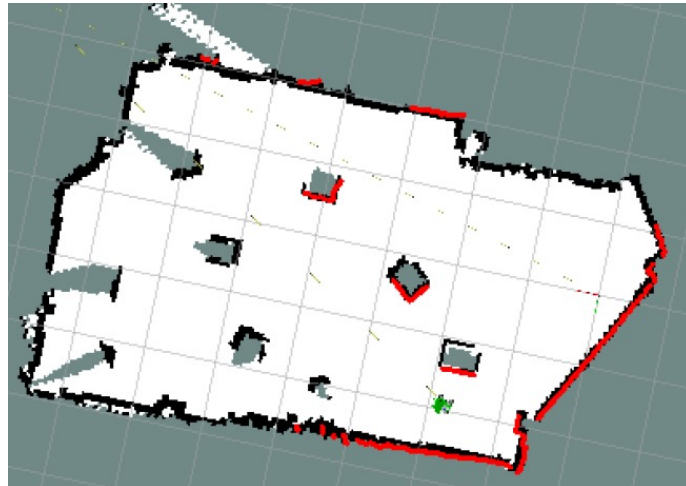


Figura 43. Prueba #3 algoritmo Hector mapping.
Fuentes: Autores.

Tabla XV
RESULTADOS DEL ALGORITMO HECTOR MAPPING

| HECTOR MAPPING | Velocidad lineal [m/s] | Velocidad angular [rad/s] | Tiempo [S] | Errores | Calificación |
|----------------|------------------------|---------------------------|------------|---------|--------------|
| Prueba #1 | 0.1 | 0.15 | 1:56.65 | No | 4 |
| Prueba #2 | 0.2 | 0.25 | 1:27.60 | No | 5 |
| Prueba #3 | 0.3 | 0.35 | 0:40.36 | No | 4 |

En el análisis de las pruebas realizadas al algoritmo de mapeo Hector, se han registrado los tiempos que el algoritmo requirió para generar un mapa del entorno, así como las distintas velocidades utilizadas en cada prueba. El mapa obtenido mediante este algoritmo no presentó errores, reproduciendo con precisión las ubicaciones de obstáculos y paredes tal como se encontraban en el entorno real del robot. Destaca especialmente la tercera prueba,

ya que el tiempo necesario para completar el mapeo del entorno fue significativamente menor en comparación con las otras dos pruebas. Por esta razón se le dio una calificación de 5/5 a la tercera prueba.

VI-A3. Pruebas de mapeo con algoritmo Cartographer mapping:

1. Prueba #1 en entorno controlado #1

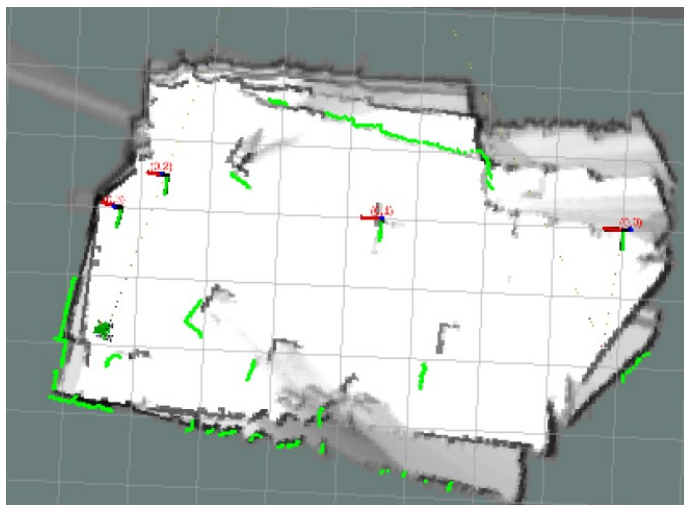


Figura 44. Prueba #1 algoritmo Cartographer mapping.

Fuentes: Autores.

2. Prueba #2 en entorno controlado #2

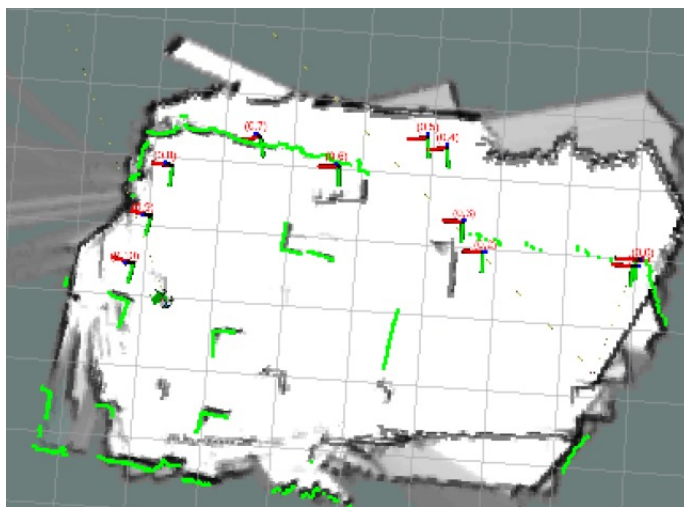


Figura 45. Prueba #2 algoritmo Cartographer mapping.

Fuentes: Autores.

3. Prueba #3 en entorno controlado #3

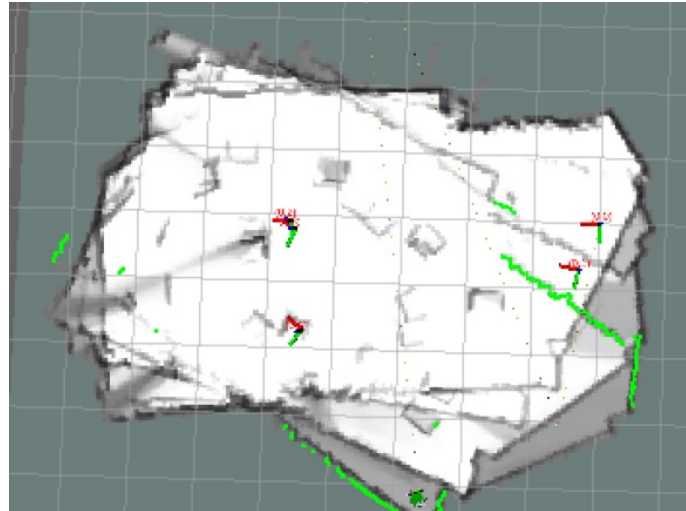


Figura 46. Prueba #3 algoritmo Cartographer mapping.

Fuentes: Autores.

Tabla XVI
RESULTADOS DEL ALGORITMO CARTOGRAPHER MAPPING

| CARTOGRAPHER MAPPING | Velocidad lineal [m/s] | Velocidad angular [rad/s] | Tiempo [S] | Errores | Calificación |
|----------------------|------------------------|---------------------------|------------|---------|--------------|
| Prueba #1 | 0.1 | 0.15 | 1:56.50 | Si | 1 |
| Prueba #2 | 0.2 | 0.25 | 1:42.24 | Si | 1 |
| Prueba #3 | 0.3 | 0.35 | 0:41.35 | Si | 1 |

Al analizar los resultados del algoritmo de mapeo Cartographer, se registraron los tiempos que el robot requirió para mapear el entorno. Además, se documentaron las velocidades lineales y angulares utilizadas en cada prueba, y se asignó una calificación a cada una de ellas. Lamentablemente, este algoritmo de mapeo demostró ser el menos eficiente para este proyecto. Los mapas generados presentaron numerosos errores al ubicar obstáculos y paredes. Por esta razón, se otorgó una calificación de 1/5 a cada prueba. A pesar de ello, la tercera prueba se destaca por su tiempo reducido para completar el mapeo, aunque el mapa resultante sigue siendo deficiente.

Conclusión: Tras evaluar los resultados de cada prueba, se ha llegado a la conclusión de que el algoritmo más adecuado para este robot es el Gmapping. Este algoritmo genera mapas sin errores ni anomalías, además, el tiempo necesario para obtener dichos mapas es reducido. Su eficiencia y precisión hacen que sea el algoritmo elegido para este robot móvil.

VI-B. Pruebas de planificación de trayectorias con algoritmo RRT_exploration

Se presenta prueba con el algoritmo RRT como se aprecia en la Figura 47, donde se generan trayectorias en un ambiente controlado. El propósito principal de esta prueba es analizar el desempeño del algoritmo en la creación de rutas favorables para que el robot pueda planificar movimientos.

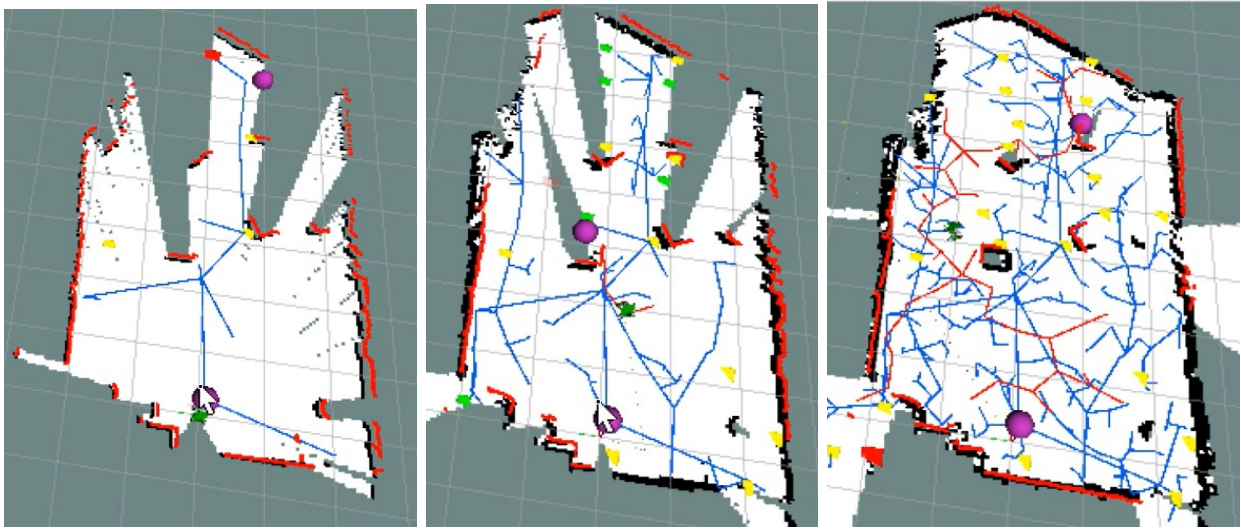


Figura 47. Prueba del algoritmo RRT.
Fuentes: Autores.

Los resultados obtenidos de las pruebas han demostrado que el algoritmo RRT_exploration es altamente efectivo en la generación de trayectorias en entornos complejos y dinámicos. Se ha observado que este algoritmo es capaz de encontrar soluciones viables en un tiempo razonable, incluso en situaciones con múltiples obstáculos y condiciones difíciles. Además, las pruebas realizadas con el algoritmo RRT confirman su versatilidad en la generación de trayectorias en entornos controlados. Estos resultados respaldan su aplicación en una amplia gama de aplicaciones prácticas, desde la robótica móvil hasta los sistemas de navegación autónoma, donde la planificación de movimientos precisa y eficiente es esencial.

VI-C. Pruebas de un punto A hacia un punto B con algoritmo AMCL

Se presenta la prueba realiza con el algoritmo de AMCL como se muestra en la Figura 48 en las que se simula la trayectoria en un entorno controlado. El propósito principal de estas pruebas es analizar el desempeño del algoritmo con los distintos parámetros establecidos y preconfigurados para su correcta función.

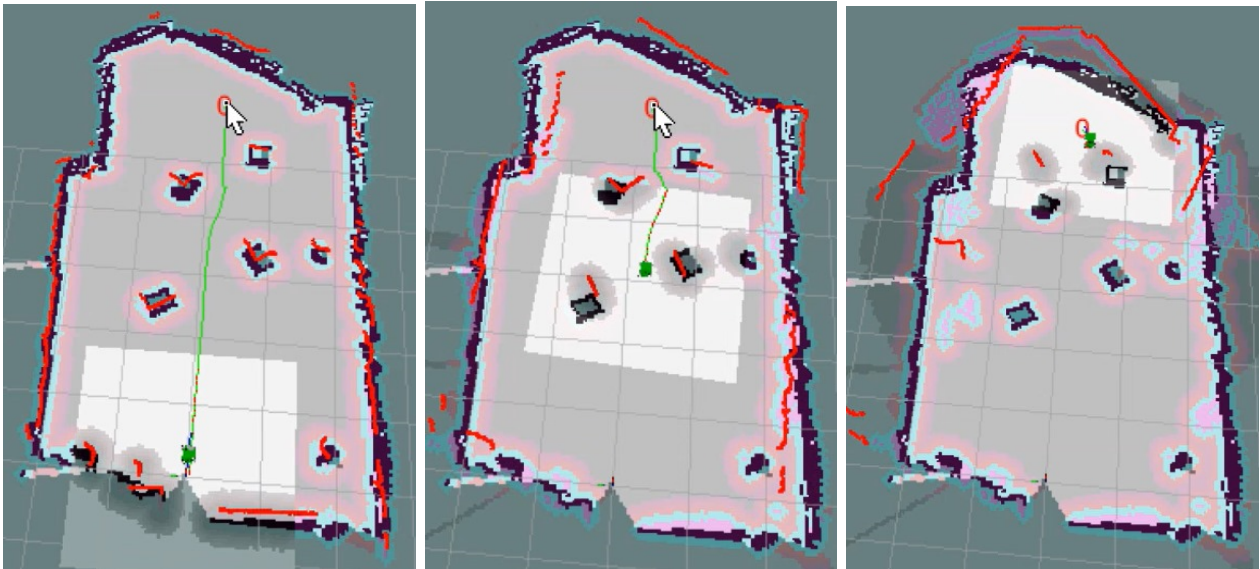


Figura 48. Prueba del algoritmo AMCL.
Fuentes: Autores.

Se ha optado por la implementación de este algoritmo debido a su capacidad para operar con diversos sensores, como el sensor LIDAR, además de emplear odometría, lo que lo convierte en una opción adecuada para robots diferenciales. Este algoritmo es compatible con ROS, lo cual simplifica su integración. Uno de los principales desafíos al utilizar este algoritmo reside en la configuración de sus parámetros, ya que estos varían considerablemente según el entorno y el tipo de robot. La carga computacional puede resultar significativa dependiendo de la configuración de los parámetros, dado que se basa en un enfoque probabilístico basado en partículas. Esta carga se puede mitigar ajustando la cantidad de partículas utilizadas, aunque esto aumenta la susceptibilidad del algoritmo a fallos.

VIII. PRESUPUESTO

Tabla XVII
PRESUPUESTO

| Nombre del elemento | Descripción | Cantidad | Valor total |
|---------------------|--------------------------------|------------|---------------|
| Chasis | Aleación de aluminio anodizado | 1 | \$80 |
| Motor | JGB37-520 | 4 | \$40 |
| Cámara | Astra ProPlus | 1 | \$200 |
| Sensor LIDAR | A1M8 SLAM LIDAR | 1 | \$110 |
| Placa de expansión | ROS robot | 1 | \$80 |
| Placa de expansión | USB HUB 0.1 a 4 | 1 | \$30 |
| Raspberry PI 4B | Controlador | 1 | \$110 |
| Ruedas | Ruedas de goma | 4 | \$20 |
| Batería | Batería de litio (6A) | 1 | \$40 |
| Horas de trabajo | | 3000 horas | \$3000 |
| TOTAL COSTOS | | | \$3710 |

IX. CONCLUSIONES

La implementación y evaluación de algoritmos de mapeo en este robot móvil autónomo demostraron ser indispensables para este proyecto. A través del uso del sensor LiDar, se logró obtener un mapa 2D de un entorno controlado, con un margen de error dentro de lo propuesto en un inicio como se muestra en la Figura 38, esto es crucial para el desplazamiento del robot desde un punto A hasta un punto B.

Se concluye que de los tres algoritmos de mapeo probados, el que obtuvo un mejor desempeño fue del algoritmo Gmapping debido a su capacidad para generar mapas del entorno del robot sin errores en un corto periodo de tiempo. Esto permitió obtener un mapa 2D de un área de $32m^2$ en un tiempo de 40 segundos con una velocidad lineal de 0.3 m/s y una velocidad angular de 0.35 rad/s , como se muestra en la Tabla XIV.

Para la obtención de trayectorias se implementaron dos algoritmos, el `rrt_exploration` y el AMCL, los cuales permiten trazar posibles rutas en un entorno cada uno con sus diferencias. El algoritmo de exploración `rrt_exploration` permite realizar un mapeo autónomo creando un árbol de rutas representando las trayectorias que puede tomar en el entorno como se muestra en la Figura 34. El algoritmo permite que el robot siga las trayectorias del árbol de rutas hasta finalizar la exploración del entorno como se muestra en la Figura 47. Se concluye que este algoritmo permite al robot adaptarse a un entorno dinámico modificando sus trayectorias.

El algoritmo AMCL permitió la navegación autónoma del robot de un punto inicial hasta un punto específico trazando una trayectoria la cual se adapta de acuerdo a la presencia de los obstáculos del entorno, para usar este algoritmo es necesario haber mapeado el entorno con anterioridad dado que este algoritmo realiza la navegación mediante un mapa ya establecido, como se evidencia en la sección VI-C. Al realizar las pruebas de este algoritmo, se colocaron 4 obstáculos en el camino, según seguía la trayectoria evadió todos los obstáculos hasta acercarse al punto especificado como se muestra en la Figura 48. Se comprobó que el robot culmina la trayectoria en un radio de 25cm del punto final indicado lo está dentro del porcentaje de error propuesto en un inicio.

X. RECOMENDACIONES

- Para lograr un mapeo exitoso es necesario ajustar las velocidades angulares y lineales ya que de esto depende en gran medida el resultado final de los algoritmos de mapeo, dado que si las velocidades son muy rápidas la carga computacional se eleva lo que puede ocasionar falsas lectura de datos del sensor LIDAR.
- Se debe prestar especial atención a la configuración del sensor LIDAR y su distancia de detección. En el caso especificado, se sugiere establecer una distancia mínima de 4 metros hasta los obstáculos para garantizar que el sensor pueda captar con precisión la información necesaria del entorno. Esta distancia proporciona al robot el espacio adecuado para procesar la información del sensor y tomar decisiones de navegación de manera efectiva.
- Es crucial tener en cuenta las limitaciones del sensor LIDAR en relación con obstáculos de colores oscuros, vidrios o elementos transparentes, ya que estos pueden afectar la detección y el mapeo del entorno.
- Las configuraciones de los parámetros de los algoritmos influyen en gran medida el resultado final, una muestra de esto es el algoritmo AMCL, al cual hay que reconfigurarle los parámetros según el entorno lo que puede representar un gran desafío.
- Se recomienda ubicar todos los paquetes en un mismo espacio de trabajo para evitar problemas con ROS ya que puede que no reconozca los launchers que están en espacios de trabajos distintos. Los nombres de los nodos deben ser diferentes para no tener errores al momento de llamarlos mediante ROS.

REFERENCIAS

- [1] Y. Lu, C. Li, B. Li y W. Qiao, «Detección de bloques fronterizos basada en muestras para la exploración de robots autónomos,» en *2021 4th International Conference on Intelligent Robotics and Control Engineering (IRCE)*, IEEE, 2021, págs. 73-78.
- [2] S. Songmin Jia, Y. He, X. Li, X. Zhang y M. Li, «Mejora de la tecnología de exploración autónoma de robots móviles en entornos de edificios desconocidos,» en *2018 Chinese Automation Congress (CAC)*, IEEE, 2018, págs. 1615-1620.
- [3] D. F. Quiroga Guerra y B. L. Rubio Amaya, «Robot móvil autónomo para la siembra de semillas en el campo,» B.S. thesis, 2023.
- [4] E. W. Rodríguez De La Cruz, «Estudio para el diseño de un robot móvil autónomo para medición de contaminantes del aire y suelo en el corredor minero del sur,» 2020.
- [5] D. Dony Hutabarat, M. Rivai, D. Purwanto y H. Hutomo, «Evitación de obstáculos basada en Lidar para el robot móvil autónomo,» en *2019 12th International Conference on Information & Communication Technology and System (ICTS)*, IEEE, 2019, págs. 197-202.
- [6] C. M. Peña González, «Diseño y construcción de un robot móvil para un equipo detector de metales capaz de realizar búsqueda de minas antipersonal. Bucaramanga, Colombia,» 2020.
- [7] V. Villani, F. Pini, F. Leali y C. Secchi, «Estudio sobre la colaboración entre humanos y robots en entornos industriales: Seguridad, interfaces intuitivas y aplicaciones,» *Mechatronics*, vol. 55, págs. 248-266, 2018.
- [8] F. A. S. Jaesub y S. B. J. HyeonSeok, «Desarrollo de criterios de rendimiento de navegación autónoma y métodos de prueba relacionados para robots móviles autónomos en entornos exteriores,» en *2021 21st International Conference on Control, Automation and Systems (ICCAS)*, IEEE, 2021, págs. 1653-1656.
- [9] P. A. Garzón Espitia y A. M. Rodríguez Padilla, «Análisis sobre marcos regulatorios internacionales sobre en la evolución de la inteligencia artificial (2008-2018),» *Punto de Vista*, vol. 13, n.º 20, págs. 127-144, 2022.
- [10] W. Montalvo, C. A. Garcia, J. E. Naranjo, A. Ortiz y M. V. Garcia, «Sistema de Tele-operación para Robots Móviles en la industria del Petróleo y Gas,» *Revista Ibérica de Sistemas e Tecnologias de Informação*, n.º E29, págs. 351-365, 2020.
- [11] A. J. S. Chinchilla y J. M. Murillo, «Desarrollo de robot autónomo de localización y mapeo simultáneo utilizando un sensor Lidar 2D,» 2023.
- [12] F. Sánchez-Martín, J. Salvador-Bayarri et al., «Historia de la robótica: desde Arquitas de Tarento hasta el robot Da Vinci.(Parte II),» *Actas urológicas españolas*, vol. 31, n.º 3, págs. 185-196, 2007.
- [13] «Robotics-Vocabulary ISO 8373:2021(E),» *ISO*, 2021.
- [14] M. Javid, A. Haleem, R. P. Singh y R. Suman, «Capacidades sustanciales de la robótica para potenciar la implantación de la industria 4.0,» *Cognitive Robotics*, vol. 1, págs. 58-75, 2021.
- [15] L. D. Evjemo, T. Gjerstad, E. I. Grøtli y G. Sziebig, «Tendencias en fabricación inteligente: El papel de los humanos y los robots industriales en las fábricas inteligentes,» *Current Robotics Reports*, vol. 1, págs. 35-41, 2020.
- [16] F. Sherwani, M. M. Asad y B. S. K. K. Ibrahim, «Robots colaborativos y revolución industrial 4.0 (ir 4.0),» en *2020 International Conference on Emerging Trends in Smart Technologies (ICETST)*, IEEE, 2020, págs. 1-5.
- [17] D. Belanche, L. V. Casalo, C. Flavián y J. Schepers, «Implantación de robots de servicio: marco teórico y programa de investigación,» *The Service Industries Journal*, vol. 40, n.º 3-4, págs. 203-225, 2020.
- [18] J. G. Guarnizo Marin, D. Bautista Díaz y J. S. Sierra Torres, «Una revisión sobre la evolución de la robótica móvil,» 2021.
- [19] C. G. Miguélez Machado, I. O. Benítez González, A. M. Rivera Rivera y V. Moreno Vega, «Implementación de sistema operativo robótico en una plataforma de robot móvil,» *Ingeniería Electrónica, Automática y Comunicaciones*, vol. 41, n.º 3, págs. 79-92, 2020.
- [20] J. G. Corvalán, «El impacto de la inteligencia artificial en el trabajo,» *Revista de Direito Econômico e Socioambiental*, vol. 10, n.º 1, págs. 35-51, 2019.
- [21] IFR. «Internatio Federation of Robotics.» (2024), dirección: <https://ifr.org/worldrobotics/>.

- [22] J. Deiro Ferre, «Robótica aplicada, un paso ya presente en el futuro. Análisis de mercado consecuencias de implantación e introducción en las PYMES,» Tesis doct., Industriales, 2021.
- [23] D. H. Rubino, «Robot autónomo,» Tesis doct., Universidad de Belgrano. Facultad de Ingeniería, 2010.
- [24] D. Nowak. «ROSBOT 2R.» (2018), dirección: <https://medium.com/husarion-blog/rosbot-2-0-a-new-open-source-autonomous-robot-platform-640af0b95a8d>.
- [25] P. Cano García, «Mapeado y navegación aplicado a un robot cortacésped,» 2023.
- [26] J. J. Crisostomo Poma, «Diseño de un robot móvil de servicio para aplicaciones de fumigación del cultivo de maíz en la provincia de concepción de la región Junín,» 2020.
- [27] M. R. Mendieta Molina y P. A. Quichimbo Plaza, «Diseño, construcción y programación de un prototipo de robot móvil para aplicaciones de telepresencia,» B.S. thesis, 2022.
- [28] A. M. Mera Chamorro y D. A. Ruano González, «Implementación de un robot de navegación autónoma terrestre para evasión de obstáculos usando el dispositivo Pixhawk,» B.S. thesis, Quito: Universidad de las Américas, 2019, 2019.
- [29] T. A. Roza Manrique y B. S. Pallares Olivares, «Diseño y construcción de un robot móvil autónomo para localización y mapeo simultáneos (SLAM) en ambientes cerrados domésticos,» Tesis doct., Universidad Santo Tomás, 2020.
- [30] A. A. Sosa Navarro y M. A. Pucllas Palacios, «Diseño de un robot móvil de dos hileras para trasplante de plántulas de arroz en sembríos de la costa norte peruana,» 2020.
- [31] K. De La Torre Muña y J. G. C. Yufra Torres, «Diseño de un robot móvil recolector y compactador de botellas de plástico utilizando redes neuronales en playas con arena fina,» 2020.
- [32] N. A. K. Zghair y A. S. Al-Araji, «A one decade survey of autonomous mobile robot systems,» *International Journal of Electrical and Computer Engineering*, vol. 11, n.º 6, pág. 4891, 2021.
- [33] A. Rubio Coteló, J. A. Placed Perales y J. Á. Castellanos Gómez, «Evaluación de Estrategias de Exploración Robótica Autónoma,» 2022.
- [34] I. Palacios Serrano, C. Cruz Ulloa y M. Barraza Rodríguez, «Análisis de los algoritmos de planificación de trayectorias RRT, PRM y Voronoi en la solución de un laberinto modular controlado por una plataforma de dos GDL,» *Ingeniare. Revista chilena de ingeniería*, vol. 30, n.º 1, págs. 157-170, 2022.
- [35] MathWorks. «Hojas de ruta probabilísticas (PRM),» (2023), dirección: <https://la.mathworks.com/help/robotics/ug/probabilistic-roadmaps-prm.html>.
- [36] S. J. Ramos Romero, «Nodo de odometría de un robot móvil a través del sistema operativo de robots (ros),» B.S. thesis, 2023.
- [37] J. E. Jara Jimbo y J. P. Uruchima Calle, «Levantamiento de mapas de ambientes para la navegación autónoma de robots móviles,» B.S. thesis, 2021.
- [38] W. A. Vaca Paredes, «Robot móvil para investigación en algoritmos de planeamiento de rutas: sistema de Odometría,» B.S. thesis, 2018.
- [39] E. Grisales Ramírez, «Planificación de trayectorias óptimas de robots móviles empleando el mapeo de celdas,» Tesis doct., Universidad Nacional de Colombia, 2023.
- [40] J. O. Wallgrün, «Hierarchical voronoi-based route graph representations for planning, spatial reasoning, and communication,» en *Proceedings of the 4th International Cognitive Robotics Workshop (CogRob-2004)*, 2004, págs. 64-69.
- [41] F. Morales Martín, J. Garrido Jurado, F. Vázquez Serrano y M. L. Ruz Ruiz, «Herramienta de simulación para el análisis de algoritmos de planificación de trayectorias,» en *XLIII Jornadas de Automática*, Universidade da Coruña. Servizo de Publicacións, 2022, págs. 777-784.
- [42] S. Mahadevi, K. Shylaja y M. Ravinandan, «Memory Based A-Star Algorithm for Path Planning of a Mobile Robot,» *International Journal of Science and Research (IJSR)*, vol. 3, n.º 6, págs. 1351-1355, 2014.
- [43] S. Rajapakshe y R. Hettiarachchi, «Design and development of a research oriented low cost robotics platform with a novel dynamic global path planning approach,» en *2022 8th International Conference on Control, Automation and Robotics (ICCAR)*, IEEE, 2022, págs. 71-76.

- [44] E. C. Navone. «Algoritmo de la ruta más corta de Dijkstra - Introducción gráfica y detallada.» (2022), dirección: <https://www.freecodecamp.org/espanol/news/algoritmo-de-la-ruta-mas-corta-de-dijkstra-introduccion-grafica/>.
- [45] A. I. Yandún Torres, «Planeación y seguimiento de trayectorias para un robot móvil,» B.S. thesis, QUITO/EPN/2011, 2011.
- [46] E. Martín Arias, «Evaluación de un algoritmo de localización híbrida para un robot móvil dotado de láser y cámara omnidireccional,» Tesis doct., Industriales, 2020.
- [47] A. Naranjo Zapata y J. R. Londoño Sánchez, «Desarrollo de un robot móvil capaz de desplazarse por terrenos irregulares inclinados,» 2021.
- [48] DfRobot. «D65mm Rubber Wheel Pair - Blue (Without Shaft).» (2016), dirección: <https://www.dfrobot.com/product-652.html>.
- [49] E. Cabrera-Bustamante, R. Moreno-Leal, M. Torres-Rivera et al., «Integración de tecnología lidar en vehículo para escaneo con ros,» *Pädi Boletín Científico de Ciencias Básicas e Ingenierías del ICBI*, vol. 10, n.º Especial6, págs. 42-45, 2022.
- [50] Yahboom. «ROSMaster X1 ROS Robot for Jetson NANO 4GB/RaspberryPi 4B.» (), dirección: <https://category.yahboom.net/products/rosmaster-x1>.
- [51] D. I. Soque León, M. G. Guerra Pintado, D. A. Plaza Guingla et al., «Implementación de técnica de mapeo y localización simultáneo (SLAM) en vehículo autónomo,» Tesis doct., ESPOL. FIEC, 2020.
- [52] R. Yagfarov, M. Ivanou e I. Afanasyev, «Map comparison of lidar-based 2d slam algorithms using precise ground truth,» en *2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV)*, IEEE, 2018, págs. 1979-1983.
- [53] K. Trejos, L. Rincón, M. Bolaños, J. Fallas y L. Marín, «2D SLAM Algorithms Characterization, Calibration, and Comparison Considering Pose Error, Map Accuracy as Well as CPU and Memory Usage,» *Sensors*, vol. 22, n.º 18, pág. 6903, 2022.

APÉNDICE A
PLANOS

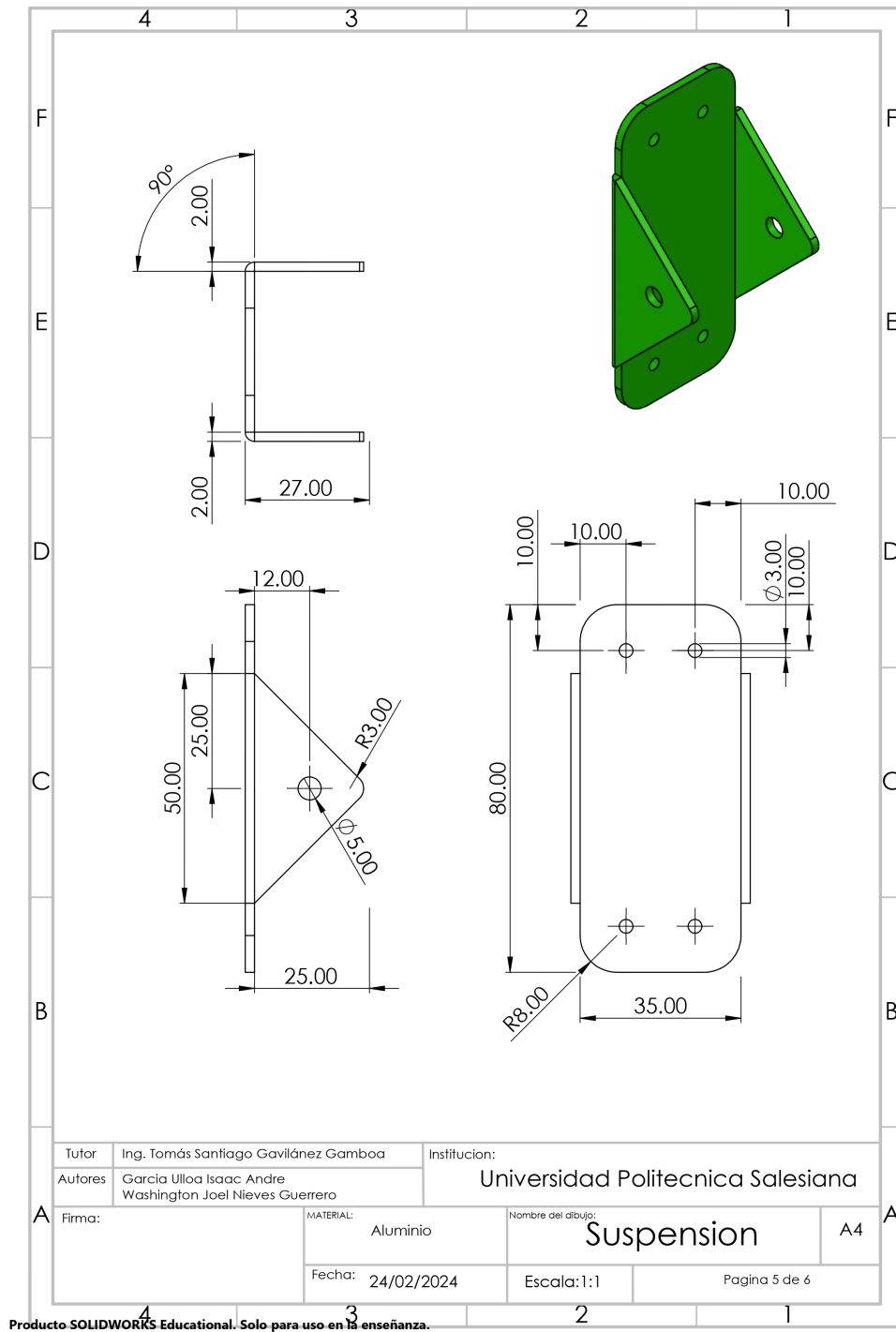


Figura 50. Plano de pieza: Suspensión

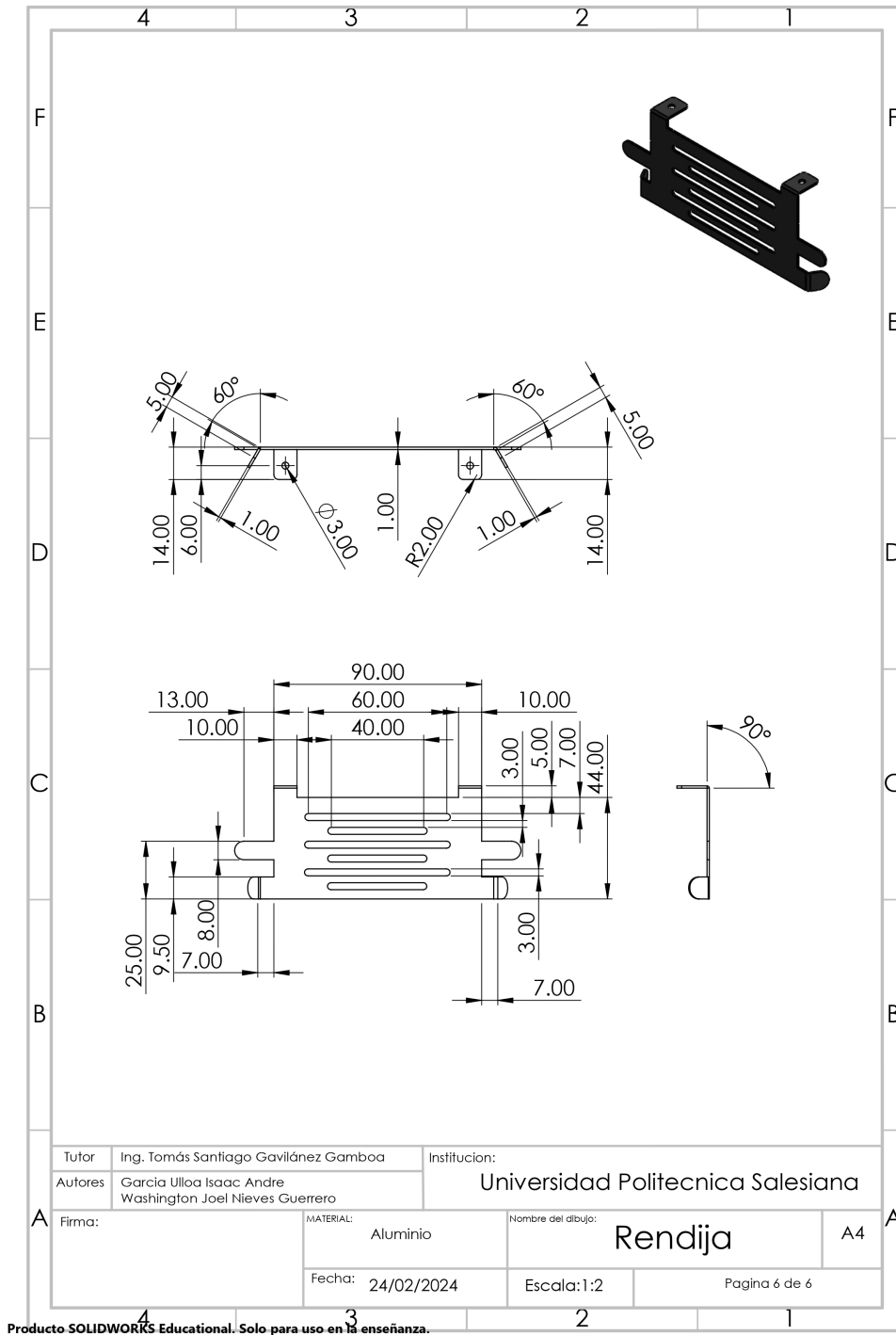


Figura 51. Plano de pieza: Rendija

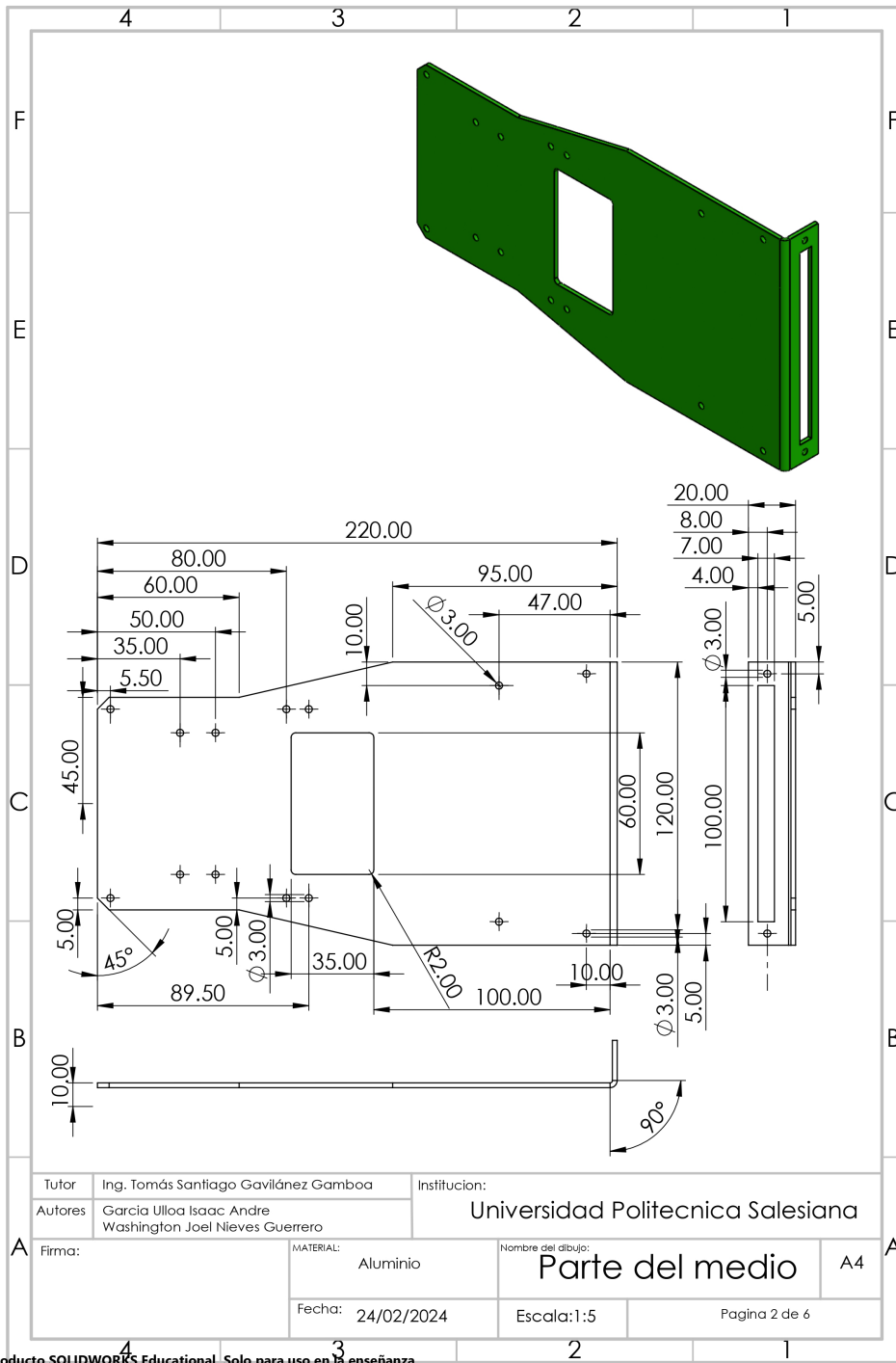


Figura 52. Plano de pieza: Base media

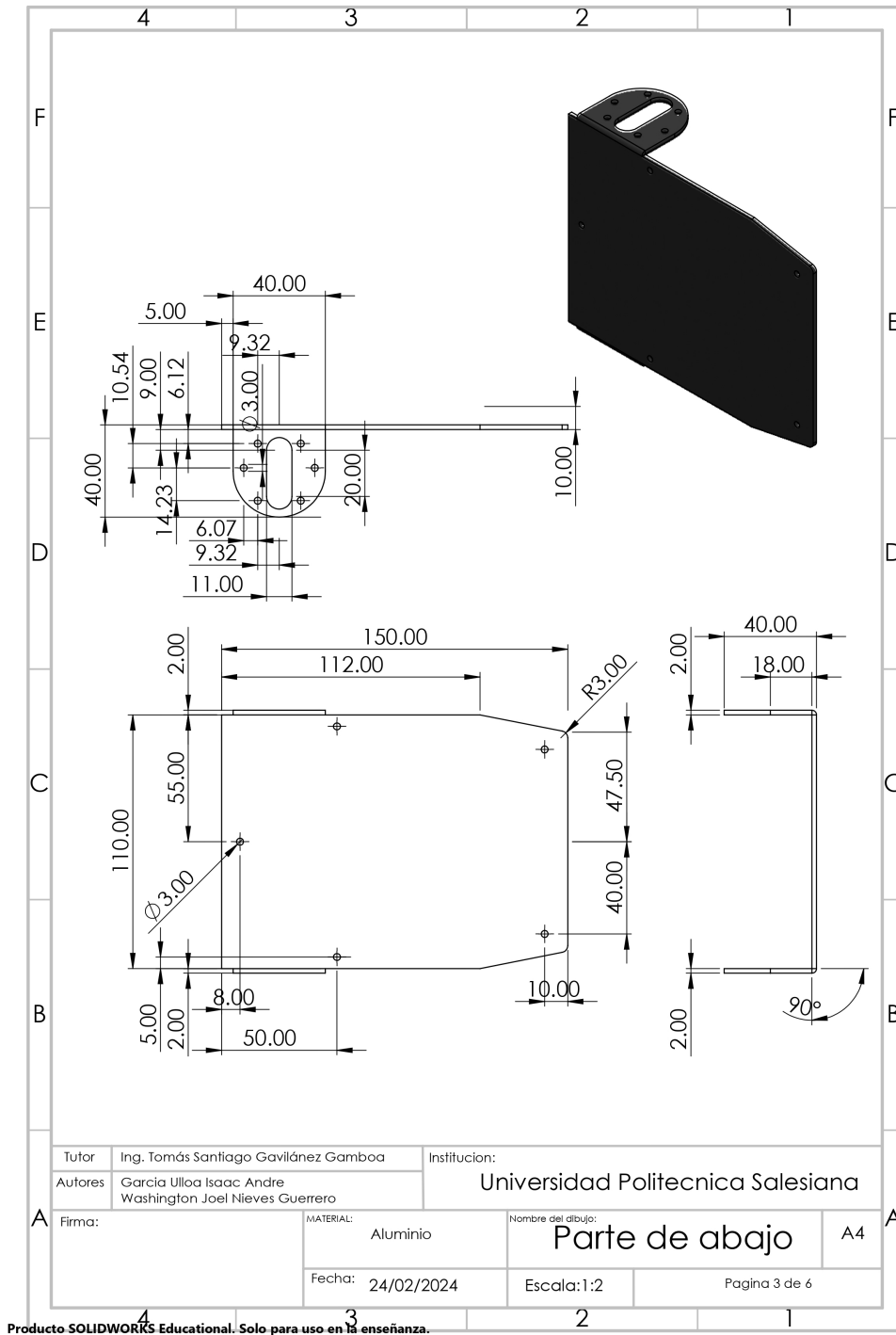


Figura 53. Plano de pieza: Base inferior

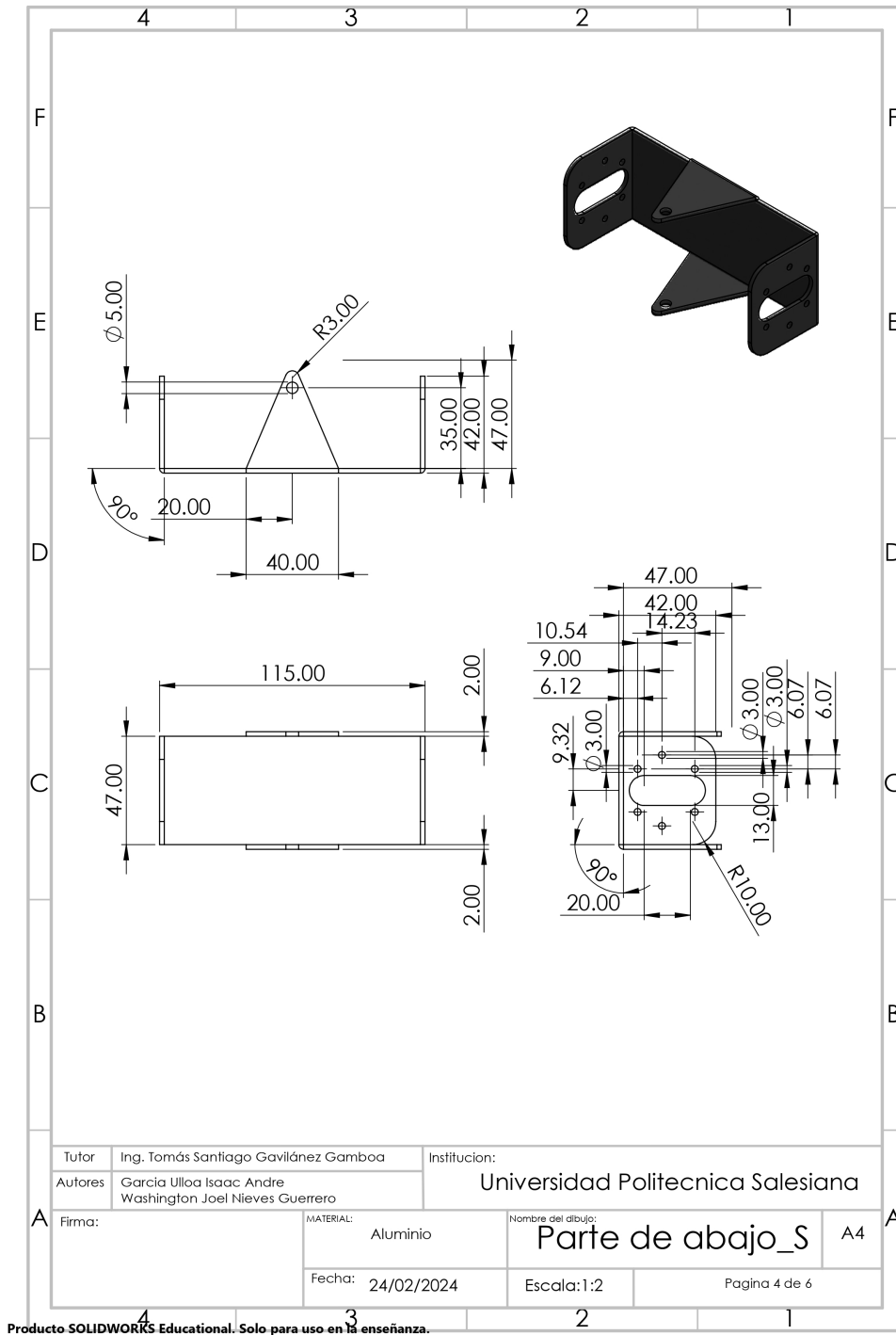


Figura 54. Plano de pieza: Base inferior de la suspensión

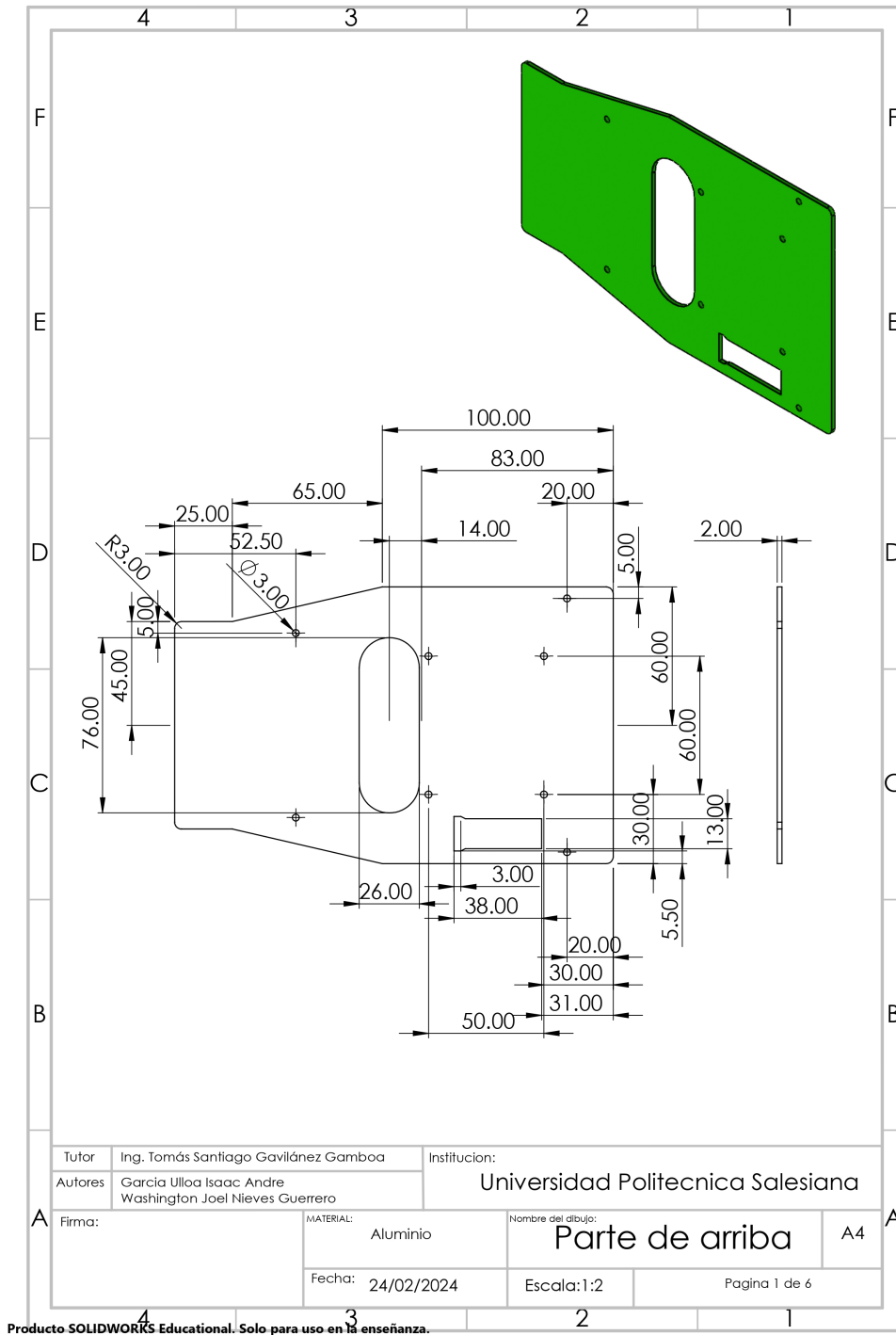


Figura 55. Plano de pieza: Base superior

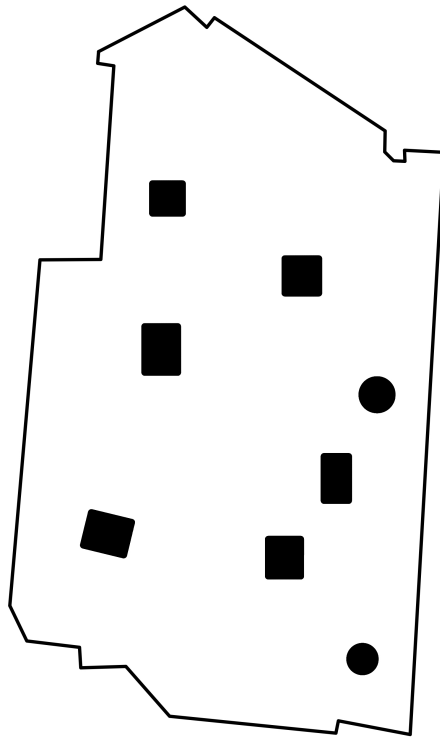


Figura 56. Entorno controlado #1 en software CAD

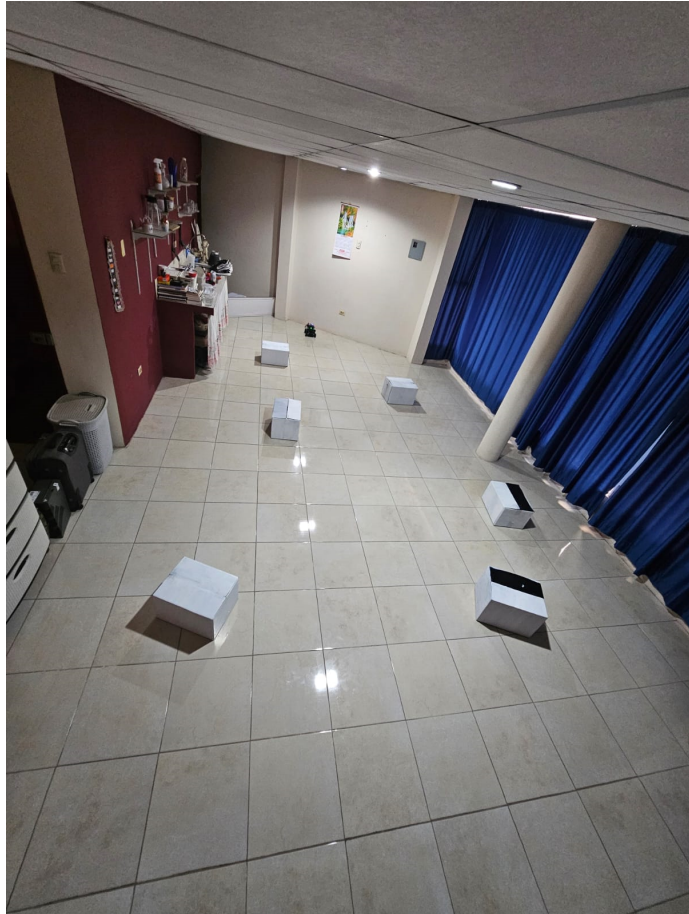


Figura 57. Entorno controlado #1

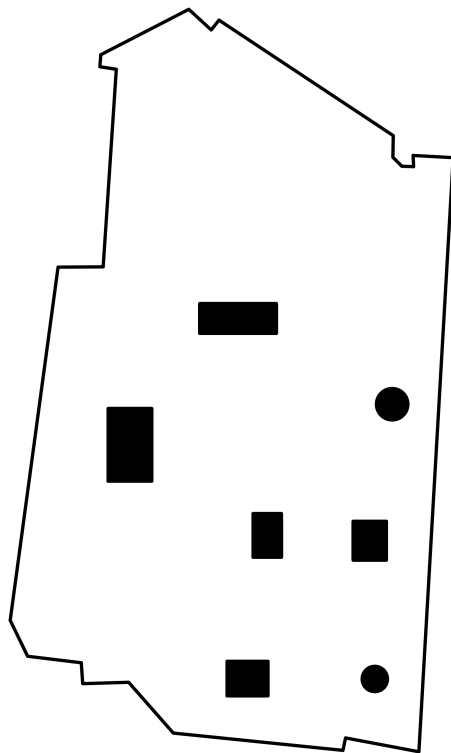


Figura 58. Entorno controlado #2 en software CAD

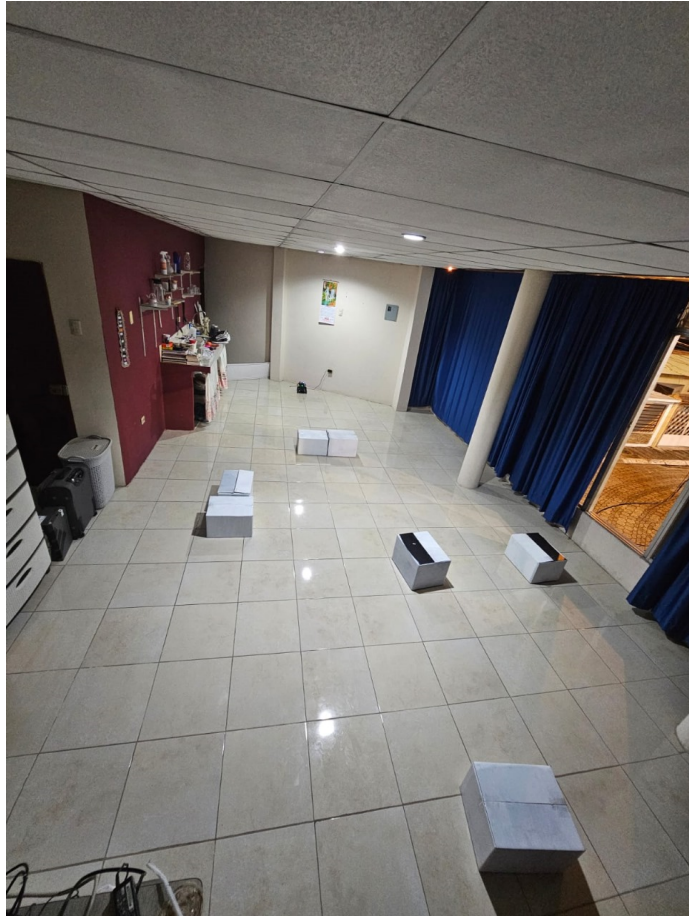


Figura 59. Entorno controlado #2

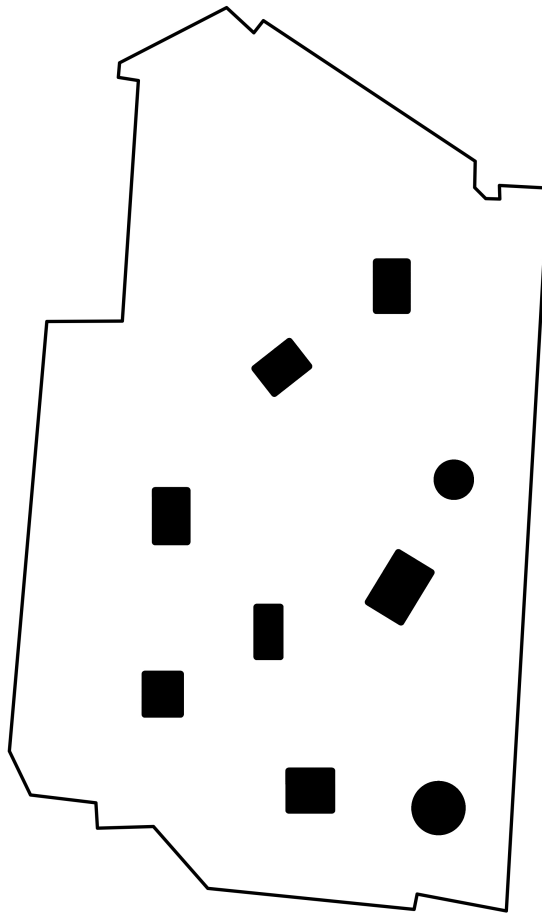


Figura 60. Entorno controlado #3 en software CAD

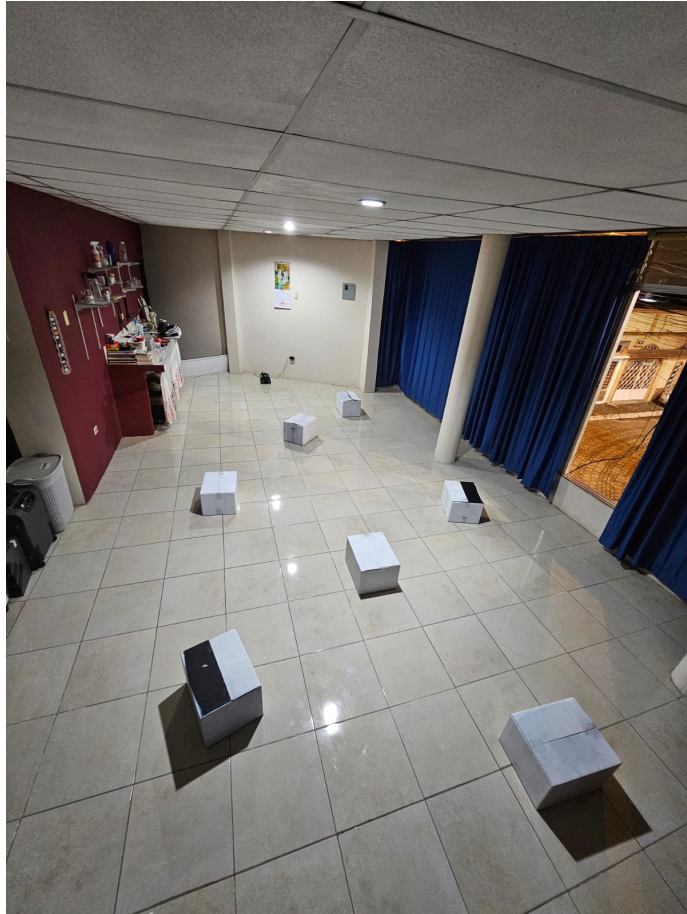


Figura 61. Entorno controlado #3