

SIMULACIÓN Y VALIDACIÓN DEL DESEMPEÑO DE NAVEGACIÓN DE UN PROTOTIPO
ROBÓTICO PARA LIMPIEZA DE PANELES SOLARES EN TECHOS Y FACHADAS URBANAS
DE MEDELLÍN

MARGARITA SOFIA FONSECA BAYONA

Trabajo de Grado

Asesor

Davinson Castaño Cano

UNIVERSIDAD EAFIT
ESCUELA DE CIENCIAS APLICADAS E INGENIERÍA
MAESTRÍA EN INGENIERÍA
MEDELLÍN
2025

CONTENIDO

pág.

0.	INTRODUCCIÓN.....	7
1.	PLANTEAMIENTO DEL PROBLEMA.....	9
2.	OBJETIVOS.....	11
2.1.	GENERAL.....	11
2.2.	ESPECÍFICOS.....	11
3.	MARCO TEÓRICO O MARCO CONCEPTUAL.....	12
3.1	LIMPIEZA DE PANELES SOLARES.....	12
3.1.1	Paneles solares en entornos urbanos.....	12
3.1.2	Incidencia de la deposición de polvo y ángulo inclinación en la frecuencia de limpieza de paneles solares.....	13
3.2	TÉCNOLOGIAS Y METODOS DE LIMPIEZA PARA PANELES SOLARES.....	13
3.3	SIMULACIÓN Y VALIDACIÓN VIRTUAL DE PROTOTIPOS ROBÓTICOS.....	16
3.3.2	Robotic Operating System (ROS).....	16
3.3.3	Webots.....	17
3.4	SENSORES Y MÉTODOS DE CONTROL PARA NAVEGACIÓN AUTÓNOMA.....	19
3.4.1	Sensores Ultrasónicos.....	19
3.4.2	Encoders.....	19
3.4.3	Unidad de Medición Inercial (IMU).....	20
3.5	ODOMETRÍA Y ESTIMACIÓN DE POSICIÓN.....	21
3.5.1	Fuentes de error en estimación de odometría.....	22
3.5.2	Odometría por ruedas.....	23
3.5.3	Odometría Inercial (Inertial Navigation System).....	24
3.6	TÉCNICAS DE CORRECCIÓN PARA ODOMETRÍA: EKF Y ZUPT.....	26
3.6.1	Extended Kalman Filter (EKF).....	26
3.6.2	Zero Velocity Update (ZUPT).....	27
3.7	ALGORITMOS DE NAVEGACIÓN Y CONTROL PARA ROBOTS.....	28
4.	DISEÑO METODOLÓGICO O METODOLOGÍA.....	31
4.1	ALCANCE DEL PROYECTO.....	31
4.2	PASOS METODOLÓGICOS.....	31

5.	MATERIALES Y MÉTODOS.....	34
5.1	ESTADO DEL ARTE Y ANÁLISIS COMPARATIVO DE PROTOTIPOS.....	34
5.2	SELECCIÓN DE CARACTERÍSTICAS Y DISEÑO DEL PROTOTIPO SIMPLIFICADO.....	38
5.2.1	Modelado del prototipo robótico.....	41
5.3	DISEÑO DEL ENTORNO VIRTUAL EN WEBOTS.....	44
5.4	IMPLEMENTACIÓN EN ROS.....	46
5.4.1	Arquitectura del paquete en ROS 2.....	46
5.4.2	Algoritmos de navegación.....	48
5.4.3	Estimación de odometría.....	50
5.4.4	Registro de datos.....	50
5.4.5	Protocolo para ejecución de las simulaciones.....	51
6	RESULTADOS.....	52
7	CONCLUSIONES Y RECOMENDACIONES.....	64
9	REFERENCIAS.....	66

LISTA DE TABLAS

pág.

Tabla 1. Métodos de limpieza para paneles solares.....	14
Tabla 2. Matriz comparativa de prototipos robóticos	35
Tabla 3. Matriz de decisión para tipo de tracción	40
Tabla 4. Matriz de decisión para plataforma de movimiento	41
Tabla 5. Peso estimado del prototipo	42
Tabla 6. Protocolo de simulaciones	51
Tabla 7. Resultados en error de trayectoria real del prototipo en 0° de inclinación	53
Tabla 8. Resultados en error de estimación de trayectoria por odometría del prototipo en 0° de inclinación.....	55
Tabla 9. Resultados en error de trayectoria real del prototipo en 10° de inclinación	58
Tabla 10. Resultados en error de estimación de trayectoria por odometría del prototipo en 10° de inclinación.....	58
Tabla 11. Resultados en error de trayectoria real del prototipo en 20° de inclinación	59
Tabla 12. Resultados en error de estimación de trayectoria por odometría del prototipo en 20° de inclinación.....	59
Tabla 13. Resultados en error de trayectoria real del prototipo en 30° de inclinación	59
Tabla 14. Resultados en error de estimación de trayectoria por odometría del prototipo en 30° de inclinación.....	60
Tabla 15. Análisis comparativo de RMSE entre escenarios e inclinaciones	62

LISTA DE FIGURAS

pág.

Figura 1. Arquitectura general de ROS.....	17
Figura 2. Etapas de desarrollo de simulaciones en Webots.	18
Figura 3. Sistema Coordinado en IMUs.....	20
Figura 4. Métodos para estimación de odometría.....	21
Figura 5. Ejemplo en incrementos de error en odometría con incremento de distancias.	22
Figura 6a y 6b. Modelo cinemático para odometría por ruedas.....	24
Figura 7. Etapas de navegación en robots.....	29
Figura 8. Pasos Metodológicos	31
Figura 9. Diseño simplificado del prototipo robótico.....	42
Figura 10. Estructura de prototipo simulado en Webots.....	43
Figura 11. Prototipo en Webots	44
Figura 12. Entorno de simulación diseñado en Webots.....	45
Figura 13. Arquitectura del paquete en ROS 2	47
Figura 14. Algoritmos de Navegación implementados.....	49
Figura 15. Cobertura de limpieza del prototipo en 0° de inclinación	53
Figura 16. Trayectorias Esperadas y Reales de las simulaciones en 0° de inclinación	54
Figura 17. Trayectorias Esperadas y por odometría de las simulaciones en 0° de inclinación	56
Figura 18. Cobertura de limpieza del prototipo en inclinaciones de 10°, 15°, 20° y 30°	57
Figura 19. Modificación a trayectorias esperadas en relación con inclinación de los paneles solares ...	58
Figura 20. Trayectorias reales y estimadas por odometría para Escenario 1 a diversas inclinaciones ...	60
Figura 21. Trayectorias reales y estimadas por odometría para Escenario 2 a diversas inclinaciones ...	61
Figura 22. Trayectorias reales y estimadas por odometría para Escenario 3 a diversas inclinaciones ...	61
Figura 23. Trayectorias reales y estimadas por odometría para Escenario 4 a diversas inclinaciones ...	62

Resumen

Este trabajo presenta el modelado y evaluación de un prototipo robótico de limpieza para paneles solares instalados en fachadas urbanas con características típicas de entornos urbanos como los de la ciudad de Medellín. El estudio integra el desarrollo mecánico del robot simplificado, la construcción del entorno virtual en Webots y la implementación de diferentes metodologías de estimación de posición y navegación autónoma en ROS 2 incluyendo odometría por encoders, navegación inercial (INS), INS con ZUPT y fusión sensorial mediante filtro de Kalman extendido (EKF).

Se definió una metodología de navegación orientada a cubrir completamente la superficie de los paneles mediante trayectorias repetitivas y controladas, evaluando el desempeño del prototipo a 0°, 10°, 20° y 30° de inclinación. Para cada caso se analizaron los errores entre la trayectoria real ejecutada por el robot y la trayectoria esperada, así como los errores de estimación de la odometría frente a la trayectoria real del prototipo.

Los resultados demuestran que el deslizamiento inducido por la inclinación afecta de manera significativa el rendimiento de los métodos basados únicamente en encoders y en navegación inercial, especialmente durante las rotaciones. La técnica ZUPT pierde efectividad al no existir verdaderos estados de reposo en pendientes. En contraste, el método de fusión sensorial mediante EKF mostró el mejor desempeño global, ofreciendo menores errores y mayor estabilidad en todas las inclinaciones evaluadas.

El trabajo concluye que la fusión sensorial es esencial para aplicaciones robóticas en superficies inclinadas y que la navegación debe adaptarse dinámicamente a los fenómenos de deslizamiento presentes en este tipo de entornos.

Palabras clave: Simulación, prototipo robótico, paneles solares, fachadas urbanas, algoritmo de navegación, odometría, ROS, Webots.

0. INTRODUCCIÓN

El uso de energías renovables es indispensable para el desarrollo energético sostenible, para la disminución del uso de combustibles fósiles y para la mitigación del cambio climático [1]. En este contexto, la energía solar destaca como una de las fuentes de energía renovable de más fácil adaptación a los entornos urbanos y por ello las ciudades juegan un papel fundamental en la implementación de sistemas de producción de energía solar a gran escala [2]. La Agencia Internacional de Energía (IEA) proyecta que para el año 2050 el 50% de la capacidad fotovoltaica estará instalada en construcciones residenciales y comerciales [3]. En el contexto colombiano, se ha observado un aumento considerable en la concentración de población en entornos urbanos. Según datos de las Naciones Unidas se espera que en 2025 el 82.9% de la población colombiana resida en zonas urbanas [4]. En la ciudad de Medellín se proyecta que para 2030 el 19% de los edificios residenciales y comerciales contarán con instalaciones de sistemas fotovoltaicos [5].

A lo largo de su vida útil, los paneles solares y su capacidad de producción están influenciados por diversos factores medioambientales tales como radiación solar disponible, magnitud y dirección de viento, temperatura ambiental, humedad y polvo atmosférico [6]. La acumulación de polvo es una problemática de gran importancia pues la suciedad reduce la transmitancia y potencia nominal generada [7].

Es por ello, que la limpieza de paneles solares es de gran relevancia en el contexto de esta industria para garantizar una buena eficiencia en la producción energética [8]. La selección de un método óptimo de limpieza debe tener en cuenta la disponibilidad de recursos, costos y capacidad de reutilización, tipo de partículas presentes en el aire, entre otros. La limpieza mecánica de paneles solares mediante plataformas robóticas busca garantizar un mantenimiento eficiente y disponible durante todo momento. A su vez, permite programar mantenimientos cíclicos que adapten las condiciones específicas de cada instalación con menor riesgo de accidentes y costos debido a su mínima intervención humana [8] [9].

El presente trabajo de grado buscará validar mediante simulaciones el prototipo de una plataforma robótica seleccionada. Se evaluará la capacidad de movimiento y navegación autónoma en paneles solares con diversas inclinaciones. Los entornos de simulación físicos son una herramienta de alto valor para la ingeniería dado que estos entornos controlados aprovechan los modelos computacionales y basados en física para simular los sensores y plataforma de movilidad de prototipos [10]. La simulación

de un prototipo robótico adaptado a las condiciones de instalación fotovoltaicas propias de los entornos urbanos y fachas de la ciudad de Medellín resulta valioso para predecir el comportamiento y capacidad de navegación autónoma. De esta forma podrán tomarse decisiones sobre el diseño propuesto en etapas tempranas del diseño y antes de la realización de pruebas físicas que incurran en tiempos y costos de fabricación mayores.

1. PLANTEAMIENTO DEL PROBLEMA

La operación eficiente de sistemas fotovoltaicos depende en gran medida de un mantenimiento y limpieza adecuados. Son diversos los métodos de limpieza que se han implementado para garantizar la potencia nominal de los sistemas. Estos se han clasificado generalmente en métodos de limpieza manual efectuada por mano de obra, métodos mecánicos con agua, métodos mecánicos en seco, limpieza química y limpieza electrostática [11] [12]. La selección de un método óptimo de limpieza debe tener en cuenta la disponibilidad de recursos, costos y capacidad de reutilización, tipo de partículas presentes en el aire, entre otros. En zonas con menor disponibilidad de agua, son más comunes las soluciones automáticas en seco y en general las tecnologías más implementadas son los métodos mecánicos dados sus bajos costos y mínima necesidad de supervisión durante el funcionamiento [13] [12].

El desarrollo prototipos robóticos en un proceso de ingeniería de alto nivel de detalle con diversas etapas y ciclos de desarrollo. Existen dos etapas durante el diseño que suelen consumir una gran cantidad de tiempo: El diseño mecánico y el diseño de control. En ambas etapas se requiere la puesta a prueba de múltiples prototipos consecutivos hasta alcanzar un prototipo aceptable [14]. La ejecución de las múltiples pruebas conlleva altos costos de fabricación, mayores riesgos y largos tiempos de desarrollo [15] [16].

En este escenario, las simulaciones en Software especializado con motores de física avanzado resultan una opción más viable y de menor costo durante el desarrollo de un prototipo pues aprovechan los modelos computacionales para simular en detalle los sensores y plataforma de movilidad de prototipos. Mediante las simulaciones es posible predecir los comportamientos autónomos en un robot antes de su fabricación [10]. En el caso específico de propuestas para la limpieza de paneles solares se han presentado publicaciones académicas presentando simulaciones para la planeación de navegación de vehículos de limpieza en granjas solares de gran escala [17] [18] y un robot de limpieza de fachas verticales que puede adaptarse a paneles solares [19]. Sin embargo, la simulación de un prototipo adaptado a las condiciones de instalación fotovoltaicas propias de los entornos urbanos y fachadas similares a las de la ciudad de Medellín aún no han sido abordadas. Las simulaciones resultan valiosas para predecir el comportamiento y capacidad de navegación autónoma. Los desafíos únicos de estas instalaciones tales como su accesibilidad limitada, áreas de trabajo reducidas y diversos grados de inclinación requieren procesos de validación exhaustivos y reiterativos que conllevan altos costos y tiempo durante la fase de pruebas y validación.

Teniendo en cuenta estas necesidades del desarrollo y diseño de un prototipo robótico, es necesario el planteamiento de un marco de simulaciones adaptado a las condiciones específicas de fachadas urbanas tales como las de Medellín. El entorno virtual y las simulaciones deben contar con las variables, parámetros y configuraciones que influyen directamente en el desempeño de navegación del prototipo robótico. Mediante los resultados, será posible replicar el comportamiento del prototipo antes de su construcción, facilitando la toma de decisiones en el diseño durante etapas tempranas del desarrollo tecnológico, a la vez que son optimizados los costos y tiempos de fabricación.

2. OBJETIVOS

2.1.GENERAL

Validar la estabilidad y desempeño de navegación de un prototipo robótico de limpieza de paneles solares bajo diversas condiciones de inclinación instalados en techos y fachadas urbanas de Medellín.

2.2.ESPECÍFICOS

1. Analizar sistemas de movilidad adecuados para robots de limpieza en paneles solares con superficies inclinadas, considerando variables propias a las instalaciones en fachadas urbanas de Medellín.
2. Diseñar un entorno virtual controlado que permita la recolección de datos de posición, movimiento y tiempo de limpieza, y que facilite el análisis del desempeño de movilidad del prototipo bajo diversas condiciones de inclinación de paneles solares.
3. Evaluar el seguimiento de algoritmos de navegación autónomos, tiempo de limpieza y precisión de odometría del prototipo robótico en el entorno virtual para verificar su desempeño en diferentes superficies inclinadas de paneles solares.

3. MARCO TEÓRICO O MARCO CONCEPTUAL

3.1 LIMPIEZA DE PANELES SOLARES

3.1.1 Paneles solares en entornos urbanos

La implementación de sistemas solares en entornos urbanos es de gran importancia pues permite descentralizar la producción de energía y evitar la dependencia de fuentes remotas y unificadas. A su vez, permite valorizar y aprovechar los techos comerciales y residenciales de construcciones urbanas para contribuir a la producción de la alta demanda energética de las ciudades. Durante los últimos años han sido significativos los avances tecnológicos y menores costos de inversión para la instalación de paneles solares. Es por ello, que la integración de paneles en construcciones urbanas resulta ser cada vez más atractiva para los sectores residenciales, comerciales e industriales [20] [21].

La integración de paneles solares en edificios urbanos conlleva consideraciones arquitectónicas y estructurales importantes. Los paneles deben instalarse de forma que su producción sea maximizada a la vez que su ubicación sea soportada de manera segura por la estructura [22]. La optimización en el diseño para techos está determinada por el ángulo de inclinación de la instalación, el espacio entre filas de paneles y área disponible [23][24]. Por ejemplo, en [23] se concluye que un ángulo de 13° es óptimo para instalaciones urbanas y puede aumentar la producción entre un 2-16% [23]. Otros estudios han demostrado que, en regiones con mayor radiación solar y latitudes al norte, se deben optimizar los ángulos de instalación en rangos desde 18° hasta 22° [24] [25]. Muchas veces las condiciones de instalación no están determinadas únicamente por el ángulo óptimo sino por las condiciones reales del techo donde será instalado el sistema: El área disponible, la geometría e inclinación y las condiciones de sombra pueden condicionar el área disponible e inclinación final del sistema [26]. En el caso específico de Colombia y la ciudad de Medellín, al ser una región cercana a la línea ecuatorial, el ángulo de inclinación óptimo es de aproximadamente $\pm 10^\circ$ [27] [28]. Sin embargo, este valor puede variar debido a las mismas razones de limitación por geometría, área disponible y sombreado. De la misma forma, el espacio entre filas de paneles solares en techos planos o de bajas inclinaciones es optimizado y los valores promedio en instalación se encuentran entre 1 y 2 metros. Dado que el objetivo de una instalación en techos urbanos es maximizar el área de producción, se dejan espacios mínimos para el mantenimiento y evitar el sombreado generado por la inclinación entre filas de paneles [25] [29].

3.1.2 Incidencia de la deposición de polvo y ángulo inclinación en la frecuencia de limpieza de paneles solares

Han sido diversos los autores que han estudiado la relación entre la acumulación de polvo y reducción de eficiencia en potencia de paneles solares. Las partículas de polvo y suciedad se acumulan sobre la superficie de los paneles, alterando las propiedades ópticas de las celdas, aumentando las temperaturas de operación y consecuentemente reduciendo la producción energética [13] [30] [31] [32]. En [6], se realizaron pruebas controladas en Arabia Saudita que estudiaron el efecto del polvo sobre la transmitancia del vidrio, con el objetivo de interpolar estos resultados y predecir la pérdida de potencia. Allí se demostró que un periodo prolongado de exposición mayor a 6 meses puede provocar descensos de hasta el 50% de producción. Experimentos realizados en [33] demostraron que el polvo adherido a paneles solares aumenta significativamente la temperatura de operación de las celdas y resulta en menores producciones de energía. De igual forma se ha concluido, que las condiciones climáticas de la región en que son instalados los paneles tienen una alta influencia en la acumulación de polvo sobre estas superficies [8].

El ángulo de inclinación de los paneles solares también tiene un efecto importante en las tasas de deposición de suciedad y polvo sobre la superficie de paneles. En inclinaciones entre 0° y 15° la reducción de potencia debido a la acumulación de polvo puede alcanzar valores del 45%, mientras que en inclinaciones cercanas a 45° representan un 25% de pérdidas de potencia nominal [34] [35]. Es por ello que la acumulación de suciedad es una problemática de mayor importancia en paneles solares planos [32] [36]. Aunque la lluvia ligera puede ayudar en la limpieza de las superficies solares sin intervención humana, en condiciones de inclinación extremas como 0° o 90° su efecto no es tan significativo, por lo cual es aún necesario contar métodos de limpieza periódicos [35].

Contar con métodos de limpieza eficientes es indispensable en las instalaciones de paneles solares para asegurar la vida útil de los sistemas y obtener mayores beneficios económicos y ambientales. Son diversos los estudios que han resaltado la importancia de un esquema de mantenimiento periódico que disminuya los efectos producidos por las condiciones ambientales de la instalación.

3.2 TÉCNOLOGÍAS Y MÉTODOS DE LIMPIEZA PARA PANELES SOLARES

Existen diversos métodos de limpieza propuestos para la eliminación periódica de suciedad en la superficie de paneles solares. El mantenimiento periódico puede considerarse como la estrategia más efectiva para la reducción en eficiencia y producción en todos los sistemas fotovoltaicos. A continuación, se presenta una comparación entre los diferentes métodos.

Tabla 1. Métodos de limpieza para paneles solares

Método	Descripción	Ventajas	Desventajas
Natural [8] [12] [36]	Interacción del viento y agua de lluvia para limpiar sin ninguna intervención humana la superficie de paneles	Sin ningún costo asociado, no requiere tampoco ninguna intervención ni monitoreo	Depende de las condiciones climáticas. No asegura una limpieza adecuada de la superficie a largo plazo
Manual [12] [36]	Uso de herramientas como cepillos, paños y escobas para eliminar la suciedad de superficie, pudiendo usar agua o realizarse en seco. En general el uso de agua ha sido considerado mucho más efectivo que los métodos en seco	Asegura un buen nivel de control y monitoreo del proceso de limpieza. Fácil ejecución y costos de inversión en herramienta	Requiere mano de obra calificada. El uso de cepillos y paños de buena calidad es indispensable para prevenir daños en la superficie. Usa grandes cantidades de agua para asegurar una buena limpieza de superficie
Mecánico [7] [8] [12] [36]	Sistemas de limpieza automáticos que utilizan dispositivos, robots y drones para ejecutar ciclos de limpieza periódicos y programados. Puede realizarse mediante diversas técnicas como limpieza con agua, flujo de aire comprimido, vibración o limpieza ultrasónica	Reduce los costos asociados a mano de obra y asegura una ejecución precisa en la periodicidad del mantenimiento	Costos iniciales más elevados. Requieren mantenimiento propio del sistema de limpieza. El contacto de cepillos y elementos móviles requiere un control preciso para evitar daño superficial en los paneles solares
Eléctrico [8] [12][13] [36]	Escudos electrodinámicos para el polvo (EDS por sus siglas en ingles). Aprovecha los principios de fuerzas electrostáticas para cargar eléctricamente las partículas de polvo y que son repelidas por un campo eléctrico	No requiere ningún tipo de herramienta, suministro de agua o intervención por lo cual es muy efectivo en zonas áridas o desérticas. La limpieza es constante al	Puede reducir en hasta un 15% la eficiencia de los paneles solares debido a la alta demanda de voltaje. No es efectivo en zonas con alta humedad y con partículas de mayor diámetro. Requiere costos de inversión e

	generado sobre la superficie del panel mediante diodos	mantener activamente el campo eléctrico	instalación más altos comparados con otros métodos
Químico [8][13] [36]	<p>Recubrimientos aplicados sobre la superficie de los paneles para proteger y realizar una limpieza pasiva constante. Al modificar las propiedades de la superficie tales como su adhesión, humectabilidad, resistencia a la corrosión y repelencia, se evita la deposición de partículas de polvo. Existen dos tipos de recubrimientos: hidrofóbicos en donde el agua actúa como una barrera protectora para encapsular las partículas y posteriormente ser removidas al rodar sobre la superficie. También pueden ser hidrofílicos, en donde se incrementa la afinidad de la superficie con el agua, favoreciendo un proceso de fotocatalisis. En este proceso, la radiación solar desintegra las partículas de polvo, las cuales son posteriormente removidas por la lluvia o la humedad ambiental</p>	<p>Bajo mantenimiento, métodos amigables con el ambiente. Además, ayudan a mejorar la resistencia del panel a condiciones ambientales y por ende aumentan su eficiencia y vida útil.</p>	<p>Son frecuentemente usados como métodos complementarios de otro mecanismo de limpieza al depender de las condiciones ambientales de los paneles solares y tener una efectividad reducida frente a partículas o manchas de gran tamaño. Su implementación requiere aplicación especializada y de alto costo. Además, son sensibles a la radiación ultravioleta por lo cual tienden a degradarse y reducir su efectividad.</p>

Es importante conocer las características y propiedades del polvo en la ubicación, las condiciones climáticas, las imitaciones geométricas y físicas de la instalación y los costos asociados a cada método [12] [36]. En zonas con menor disponibilidad de agua, son más comunes las soluciones automáticas en seco y en general las tecnologías más implementadas son los métodos mecánicos dados sus bajos costos y mínima necesidad de supervisión durante el funcionamiento [10] [17]. En general, las revisiones sistémicas de soluciones para limpieza han demostrado que los sistemas automáticos y especialmente las soluciones robóticas son soluciones ideales y mayormente aceptadas en la industria dados sus costos menores, capacidad de adaptación y crecimiento e innovación a largo plazo [7] [9]. Es por ello, que el presente trabajo se ha enfocado en el desarrollo de un prototipo robótico para limpieza de paneles solares teniendo en cuenta las características específicas de instalaciones urbanas en Medellín.

3.3 SIMULACIÓN Y VALIDACIÓN VIRTUAL DE PROTOTIPOS ROBÓTICOS

Dentro del campo de la robótica se han desarrollado diversas herramientas de simulación y software cuyo objetivo ha sido modelar a nivel computacional y mediante motores de física avanzado los sensores y plataformas de movimiento de prototipos y vehículos no tripulados [10]. Cuando las simulaciones son basadas en leyes de física se ven involucrados los factores de peso, centro de gravedad y fricción principalmente. Las leyes ópticas también podrán verse involucradas al simular sensores y su capacidad de respuesta ante factores del entorno. El código de software que almacena y resuelve las ecuaciones físicas y ópticas simuladas es denominado motor dinámico [37]. Cada plataforma de simulación cuenta con un motor dinámico integrado que se encarga de resolver las ecuaciones de las simulaciones mediante aproximaciones numéricas para cada punto del entorno. En la práctica, una simulación robótica será más exitosa siempre y cuando las aproximaciones y métodos numéricos utilizados provean predicciones aceptables en el comportamiento dinámico del sistema [37]. A continuación, se presenta una descripción de entorno de desarrollo y plataforma de simulación utilizada en este trabajo.

3.3.2 Robotic Operating System (ROS)

ROS es un paquete de Software compuesto por diversas herramientas, librerías y convenciones de diseño que buscan simplificar la creación de aplicaciones robóticas. Este entorno de desarrollo surgió como una alternativa de libre uso cuyo objetivo es proveer servicios de operación estándar tales como abstracción de hardware y control de dispositivos a bajo nivel para todo tipo de implementaciones robóticas [16] [38]. Los principales objetivos filosóficos de ROS son proveer su entorno de trabajo garantizando una comunicación directa entre sus componentes sin la necesidad de un servidor central, multilenguaje, software ligero, libre y de código abierto y un diseño modular que permite la adaptación de todo tipo de diseños especializadas que interactúen entre si [39]. Los procesos de trabajo de ROS son representados por una arquitectura de grafos donde los procesos ocurren dentro de nodos que pueden recibir y enviar mensajes mediante servicios y tópicos. A pesar de su baja latencia y reactividad, ROS no es considerado un sistema de operación en tiempo real.

Los conceptos fundamentales dentro del entorno de ROS son:

Nodos: Son procesos que ejecutan la computación. Constituyen los bloques básicos para la construcción de cualquier sistema. Los nodos de ROS son procesos completamente independientes que interactúan

entre si recibiendo y enviando mensajes. De la misma forma los nodos pueden suscribirse y publicar en tópicos. Los nodos tienen diversos parámetros asociados [38] [39][40][41].

Mensajes: Son unidades de datos que son intercambiadas entre nodos. Son descritos mediante archivos “.msg” y son publicados mediante tópicos y recibidos cuando otro nodo se suscribe al mismo. Pueden contener cualquier tipo de datos tales como lecturas de sensores, comandos, imágenes, etc. [41]

Tópicos y servicios: El sistema de envío de mensajes utiliza un esquema de publicador y suscriptor cuyo identificador es un string denominado como tópico. Cuando un nodo está interesado en un tipo específico de datos, se suscribe al tópico en cuestión. Puede haber múltiples publicadores y suscriptores concurrentes para un tópico. Aunque el esquema de publicador y suscriptor es muy flexible, no es adecuado para necesidades de interacción en la que se requieran solicitudes y consecuentes respuesta o acciones. Esta interacción es realizada mediante servicios, que son definidos por dos estructuras de mensajes: Una para la solicitud y otra para la respuesta. A diferencia de los tópicos, solo un nodo puede publicar un servicio para múltiples nodos solicitantes [39] [40].

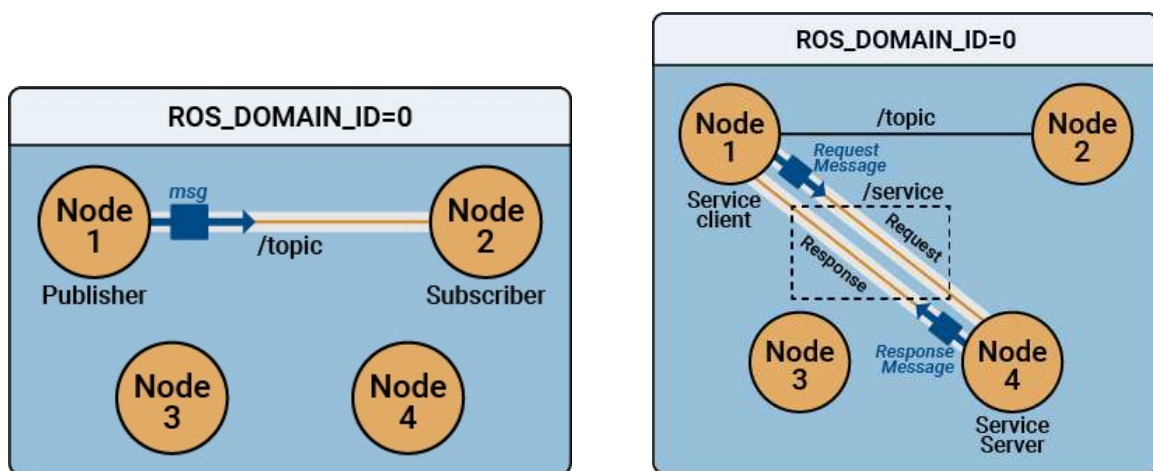


Figura 1. Arquitectura general de ROS. Fuente: The Mathworks, Inc

3.3.3 Webots

Webots es una plataforma de software libre y código abierta producida por Cyberbotics desde 1998 para la simulación de robots. Provee un ambiente de desarrollo completo que permite diseñar modelos de todo tipo de robots y sistemas de locomoción. A su vez incluye una librería completa de sensores y actuadores que pueden ser programados en múltiples lenguajes de programación y otras plataformas de desarrollo

robótico. A su vez, utiliza un motor de físicas ODE (Open Dynamics Engine) para la simulación realista de la interacción del robot con su ambiente. Webots provee un ambiente óptimo para la puesta a prueba de prototipos robóticos en ambientes tridimensionales realistas. Los parámetros físicos del robot y del mundo pueden ser ajustados de manera detallada para las simulaciones, incluyendo su masa y centro de gravedad, coeficientes de fricción, geometrías, velocidad, posición, etc. A su vez, todos los sensores y actuadores pueden ser programados para replicar de manera realista el funcionamiento de los prototipos [16] [42] [43].

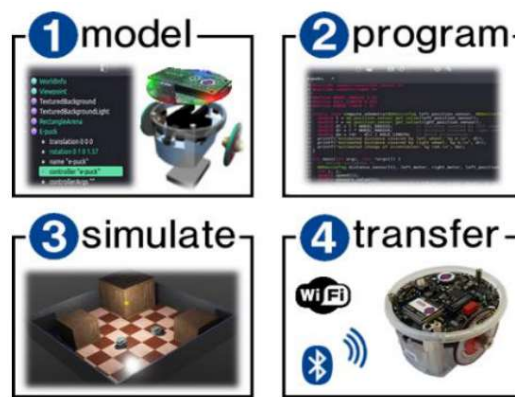


Figura 2. Etapas de desarrollo de simulaciones en Webots. Fuentes: [42] [44]

Para la interacción entre ROS y Webots, ambas compañías han desarrollado soporte y documentación que permite la integración de sistemas diseñados y programados dentro de la arquitectura de ROS que puedan ser simulados dentro del ambiente de Webots. “webots_ros2” es un paquete desarrollado por Cyberbotics que provee todas las interfaces entre las dos plataformas. Este paquete y sus diversos subcomponentes permiten controlar las simulaciones desarrolladas en Webots mediante nodos diseñados en ROS. De esta forma es posible ejecutar pruebas del prototipo en un ambiente físico simulado y detallado, manteniendo una estructura de software que sea fácilmente exportable y adaptable a un posterior prototipo físico [45] [46].

La integración de estas dos herramientas resulta especialmente útil para evitar los altos costos de fabricación y validación de un prototipo físico, permitiendo obtener datos precisos acerca del desempeño del robot durante etapas tempranas del desarrollo.

3.4 SENSORES Y MÉTODOS DE CONTROL PARA NAVEGACIÓN AUTÓNOMA

3.4.1 Sensores Ultrasónicos

Los sensores ultrasónicos son comúnmente utilizados en la navegación y detección de obstáculos para vehículos y robots. Estos sensores utilizan el principio de pulso-eco para calcular la distancia entre el sensor y el objeto reflejado al multiplicar la velocidad del sonido en el aire por el tiempo de vuelo medido [47] [48]

$$d = \frac{1}{2} c_{sonido} \Delta t \quad (1)$$

Donde

d = distancia calculada por el sensor

c_{sonido} = velocidad del sonido en el aire

Δt = tiempo medido entre la emisión y la recepción

Los sensores usan discos de cerámica piezoeléctricas para transmitir los pulsos y recibir ecos. La distancia mínima detectable de la materia de referencias comerciales para este tipo de sensores es de 0.2m. La presencia de objetos más cercanos a este mínimo puede realizarse mediante el sensor transmisor, pero no se obtendrán valores absolutos de distancia [47]. Diversos sensores cuentan con diversos rangos de alcance y diversos anchos del haz que determinan la capacidad de detección de obstáculos [48]. Un único sensor solo permite detectar la distancia hacia el objeto. Para localizar un objeto se requieren al menos dos sensores con rangos parcialmente superpuestos [47].

3.4.2 Encoders

Los encoders son sensores que generan señales digitales en respuesta al movimiento al utilizar tecnologías ópticas o magnéticas para medir las revoluciones o fracciones de revoluciones en un motor rotativo. También están disponibles para detectar movimiento lineal son utilizados como transductores de retroalimentación para el control de velocidad de motores, como sensores para medición de posición y para entrada de velocidad y controles de rango [49].

Mediante la medición de la rotación en una rueda o disco adjunta a un motor rotativo, es posible calcular la odometría de un robot. La circunferencia de un disco con un radio r está determinada por $2\pi r$, por lo cual la distancia recorrida por esta rueda está dada por la ecuación [50]:

$$d = 2\pi nr \quad (2)$$

Donde d = distancia recorrida

n = numero de revoluciones medidas por el encoder

r = radio del disco o rueda medido

3.4.3 Unidad de Medición Inercial (IMU)

Las unidades de medición inercial (IMU por sus siglas en inglés) son dispositivos electrónicos compuestos por un grupo de sensores que recolecta e integra la información de cada uno de ellos. Los principales componentes de una unidad inercial son acelerómetros, giroscopios y magnetómetros. La información recolectada por la IMU incluye datos de aceleración, velocidad angular y orientación en 3 direcciones. En unidades más complejas y avanzadas, se pueden obtener hasta 9 parámetros distintos (9 DOF). Los sensores son referenciados en un sistema ortogonal en donde cada sensor toma un eje coordinado para medir movimiento lateral, rotación y orientación magnética correspondientes a los ejes lateral, longitudinal y vertical de la IMU [51] [52] [53]

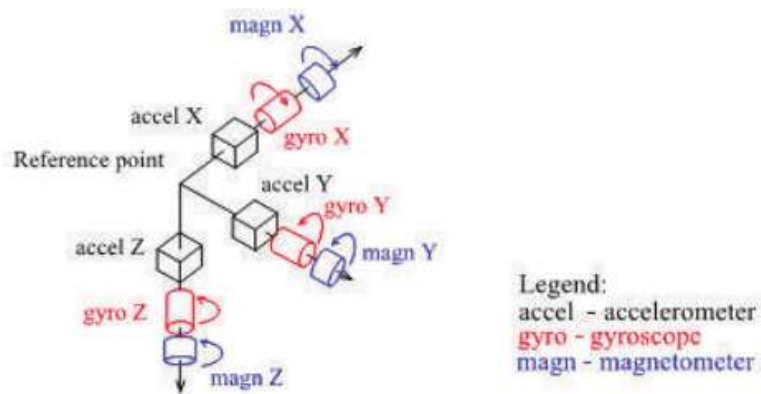


Figura 3. Sistema Coordinado en IMUs. Fuente [53]

La aplicación de estos sensores en la estimación de posición y control de navegación de robots está altamente influenciada por la exactitud de la IMU empleada. Las IMUs de bajo costo se fabrican como Sistemas Micro Electromecánicos (MEMS) cuyas precisiones y exactitud pueden ser menores respecto

a referencias comerciales diseñadas para aplicaciones tácticas. Una de las aplicaciones más utilizadas para las IMU es como un método complementario al sistema de posicionamiento global (GNSS), donde las mediciones realizadas por el sistema de navegación inercial (INS) se emplean para interpolar la trayectoria determinada mediante GNSS y así obtener mediciones mucho más precisas. De igual forma, las IMU se utilizan para la navegación de vehículos o peatones en entornos interiores y exteriores [53].

3.5 ODOMETRÍA Y ESTIMACIÓN DE POSICIÓN

La odometría se refiere a la medición de distancia y es un método usado por robots para la navegación y estimación de posición. La odometría es una forma de localización, midiendo de forma indirecta el tiempo transcurrido y la velocidad de movimiento, es posible estimar la posición del robot en su entorno relativa a su posición inicial. Una desventaja de este método está dada por la naturaleza indirecta de su medición, la cual puede incurrir en errores en los datos puesto que la relación entre la velocidad de los motores y la rotación de las ruedas pueden variar y no ser lineales en el tiempo [50] [54] [55].

Se han propuesto diversos métodos para la odometría. En [55] se realiza un estudio que busca categorizar y describir las diferentes técnicas de cada método propuesto. Basado en los resultados, los métodos de odometría se clasifican en:

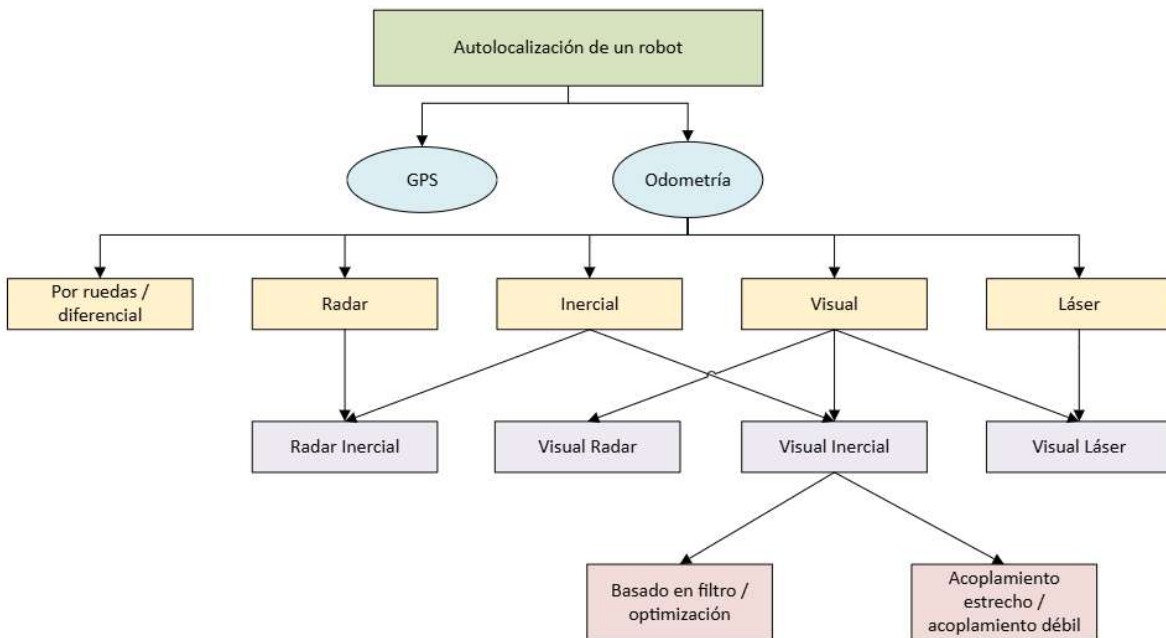


Figura 4. Métodos para estimación de odometría. Fuente: [55]

Teniendo en cuenta este estudio, las técnicas de odometría pueden clasificarse de acuerdo con los sensores empleados y la forma en que se realiza la estimación de posición. Entre las más utilizadas se encuentra la odometría por ruedas o odometría diferencial, la odometría inercial, y los métodos combinados o de fusión sensorial, en los cuales se integran diferentes fuentes de datos mediante algoritmos como el Filtro de Kalman Extendido (EKF) para mejorar la precisión y reducir los errores en la medición. De igual forma, en situaciones donde el robot se encuentra detenido o con movimientos muy lentos, también pueden aplicarse técnicas complementarias como ZUPT (Zero Velocity Update), que permiten corregir la deriva en la estimación [54] [55]. Para la realización de este trabajo se emplearán algunas de los métodos existentes para analizar la precisión y estabilidad de la estimación del robot bajo diversas condiciones de inclinación en los paneles solares.

3.5.1 Fuentes de error en estimación de odometría

Como se ha mencionado anteriormente, la condición indirecta de medición y las inconsistencias en la medición de datos por odometría puede causar errores significativos en la estimación de posición de un robot [50]. En general, las fuentes de error pueden clasificarse entre errores sistemáticos y no sistemáticos. Los sistemáticos se refieren a todos aquellos que son resultado de los defectos cinemáticos del robot tales como asimetría en diámetro de ruedas o la incertidumbre e inexactitud en la dimensión de estas. Los errores no sistemáticos se refieren a todos aquellos que son resultado de una interacción entre el entorno con el robot tales como deslizamiento de ruedas o ruido en sensores [54].

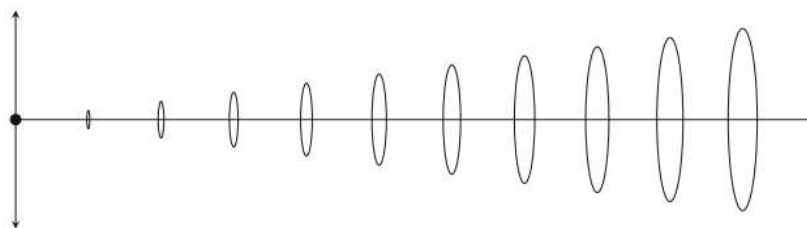


Figura 5. Ejemplo en incrementos de error en odometría con incremento de distancias. Fuente:[50]

La acumulación del error en odometría es significativa cuanto mayor sea la distancia recorrida. Los errores angulares suelen ser mucho más significativos que los errores lineales. Puesto que el error en la estimación es inevitable, la confirmación de la posición absoluta del robot mediante puntos de referencia o fusión de sensores es la técnica más ampliamente utilizada para asegurar mejor precisión y exactitud en la medición y estimación.

3.5.2 Odometría por ruedas

La odometría por ruedas está basada en la aplicación de ecuaciones simples a un robot diferencial con tracción por deslizamiento (skid-steering) tales como robots de dos y cuatro ruedas y robots de tipo oruga [50] [55]. En estos robots, las ruedas del lado derecho e izquierdo pueden operarse de forma independiente con diferentes velocidades y direcciones. Esta odometría está basada en la recolección de información proveniente de encoders ubicado en cada grupo de ruedas. El número de revoluciones de cada rueda es integrado en un modelo dinámico para determinar la posición relativa del robot respecto a su punto inicial [55].

El modelo cinemático de un robot diferencial permite describir la relación matemática entre la velocidad de rotación de las ruedas y el movimiento del robot sobre el plano. Como modelo general, cuando el robot cuenta con dos ruedas separadas por una distancia de $2b$, la posición del robot es representada mediante un vector de posición o estado $q = [x_Q, Y_Q, \varphi]^T$ donde x_Q y Y_Q representan a la ubicación del punto central del eje del robot y φ indica su orientación respecto al sistema de coordenadas global [56]. A partir de esto, la relación entre la velocidad lineal v y la velocidad angular w con el cambio de posición de robot puede representarse como: [56]

$$\begin{bmatrix} \dot{x}_Q \\ \dot{y}_Q \\ \dot{\varphi} \end{bmatrix} = \begin{bmatrix} \cos \varphi & 0 \\ \sin \varphi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ w \end{bmatrix} \quad (3)$$

Así mismo a partir de las velocidades angulares de las ruedas derecha ($\dot{\theta}_R$) e izquierda ($\dot{\theta}_L$), se puede determinar la velocidad lineal y angular del robot tal que: [56]

$$v = \frac{r}{2}(\dot{\theta}_R + \dot{\theta}_L) \quad y \quad w = \frac{r}{2b}(\dot{\theta}_R - \dot{\theta}_L) \quad (4) (5)$$

Donde: r = radio de las ruedas

$2b$ = distancia entre el centro de ambas ruedas

Al integrar estas ecuaciones en el tiempo se puede calcular entonces el movimiento del robot en el plano respecto a su punto inicial: [56]

$$\dot{x}_Q = \frac{r}{2}(\dot{\theta}_R + \dot{\theta}_L) \cos \varphi \quad (6)$$

$$\dot{y}_Q = \frac{r}{2}(\dot{\theta}_R + \dot{\theta}_L) \sin \varphi \quad (7)$$

$$\dot{\varphi} = \frac{r}{2b}(\dot{\theta}_R - \dot{\theta}_L) \quad (8)$$

Estas ecuaciones resultan la base para la estimación de posición y orientación del robot.

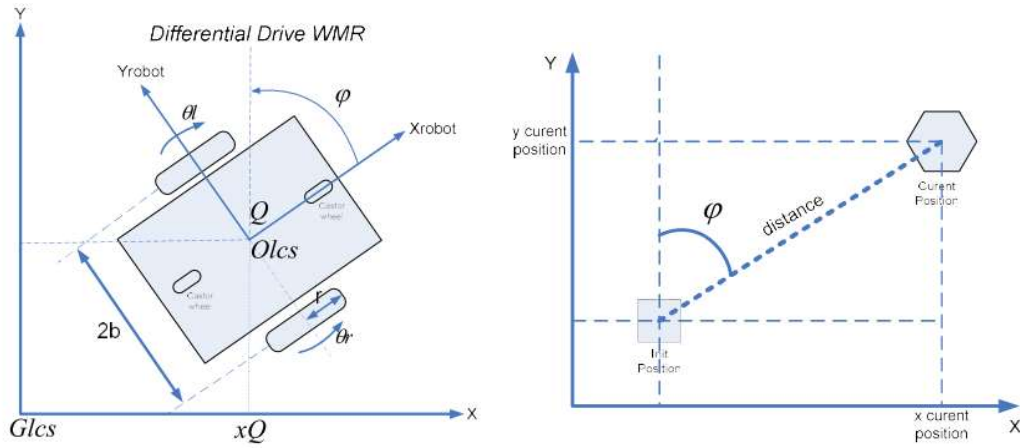


Figura 6a y 6b. Modelo cinemático para odometría por ruedas. Fuente: [56]

La odometría por ruedas resulta válida cuando el número de revoluciones de una rueda puede ser trasladado de manera precisa a valores de desplazamiento lineal respecto al suelo. Sin embargo, cuando el fenómeno de deslizamiento se presenta en las ruedas, la transformación de rotación en movimiento lineal puede no ser proporcional y esto puede llevar a acumulaciones de error considerables en la estimación [54].

3.5.3 Odometría Inercial (Inertial Navigation System)

Los sistemas de navegación inercial (INS por sus siglas en inglés), es otro método de odometría en el cual son usadas las lecturas de acelerómetros y giróscopos para estimar la posición y orientación de un robot respecto a su posición inicial. Para ello, se utilizan los 3 ejes y grupos de datos extraídos de una IMU. El concepto básico de un sistema INS es la medición de cambios en movimiento relativo. Dado que cada sensor mide en su propio sistema de coordenadas, las lecturas deben transformarse al marco global para reconstruir la posición y trayectoria del robot [50] [54] [55] [57].

El acelerómetro de una IMU entrega mediciones y datos que pueden representarse como: [58]

$$a_m = R_b^n(a + g + \omega \times v) + a_{bias} + a_{noise} \quad (8)$$

Donde a_m representa la aceleración medida en el marco del sensor, R_b^n es la matriz de rotación del robot respecto al marco global de referencia, a es la aceleración real del robot, g la gravedad y a_{bias} , a_{noise} representan el sesgo (bias) y el ruido del acelerómetro.

De igual forma, el giróscopo puede medir la velocidad angular media tal que: [58]

$$\omega_m = \omega + \omega_{bias} + \omega_{noise} \quad (9)$$

Donde ω_m representa la aceleración angular medida en el mismo marco del sensor, ω es la velocidad angular real del robot y $\omega_{bias}, \omega_{noise}$ representan nuevamente el sesgo y ruido del dispositivo.

La orientación relativa del robot respecto a la orientación inicial es corregida ($\omega = \omega_m - \omega_{bias}$) y puede obtenerse la dinámica rotacional mediante la cinemática de los cuaterniones que describen la orientación del robot: [58]

$$\dot{q}(t) = \frac{1}{2} q(t) \otimes \Omega(t) \quad (10)$$

Donde $q(t)$ es el cuaternión que describe la rotación en el marco global, $\Omega(t)$ es un cuaternión construido a partir de la velocidad angular corregida y \otimes es el producto cuaternión – cuaternión.

Asi mismo, la aceleración medida se transforma al marco global de referencia para obtener la aceleración traslacional del robot: [58]

$$a = R_b^n(a_m - a_{bias}) - g$$

Con esta aceleración, se integra entonces el resultado para obtener la velocidad y posición del robot: [58] [50]

$$v(t) = \int_0^t a(t) dt + v(0) \quad (11)$$

$$x(t) = \int_0^t v(t) dt + x(0) \quad (11)$$

En la práctica y al tener sistemas discretos las integraciones son reemplazadas por sumatorias, tomando las medidas tomadas en el tiempo. A mayor frecuencia de tiempo y mediciones, mejor es la estimación de posición obtenida: [50]

$$v_n = \sum_{i=0}^n a_n \Delta t \quad \theta_n = \sum_{i=0}^n \omega_i \Delta t \quad (12), (13)$$

Los sistemas INS están sujetos a errores causados por las inexactitudes y ruidos en las mediciones tomadas por la IMU. Factores ambientales como la temperatura, vibraciones o el desgaste del sensor introducen pequeñas desviaciones al calculo que, al integrarse repetidamente en el tiempo, generan errores acumulativos significativos en la estimación de posición y orientación del robot. Además, los sensores MEMS utilizados en muchas IMUs presentan ruido de parpadeo, un tipo de ruido de baja frecuencia que produce fluctuaciones en el sesgo y que suele modelarse de forma aleatoria. Este fenómeno provoca que la incertidumbre en la velocidad crezca proporcionalmente a $t^{\frac{3}{2}}$ y que el error en la posición aumente aún más rápidamente, con una dependencia proporcional a $t^{\frac{5}{2}}$ [55] [58] [59]

A los anteriores errores se suma el hecho de que la estimación de la aceleración real del robot requiere restar la contribución de la gravedad a partir de la orientación calculada. Esto implica que cualquier

pequeño error en la orientación se traduce directamente en una cancelación incorrecta de la gravedad, generando aceleraciones ficticias que también se integran a lo largo del movimiento. Este efecto es especialmente crítico porque las aceleraciones del robot suelen ser mucho menores que la gravedad, por lo que incluso pequeñas desviaciones angulares introducen derivas perceptibles en la trayectoria estimada. Por todo lo anterior, no es recomendable utilizar la odometría como fuente primaria de información del robot durante la planeación de navegación del robot y por ello suelen usarse técnicas como la integración de fuentes externas de información mediante el filtro extendido de Kalman o el apoyo con técnicas como Zero Velocity Update (ZUPT) para corregir periódicamente los resultados obtenidos mediante INS [57] [58] [59].

3.6 TÉCNICAS DE CORRECCIÓN PARA ODOMETRÍA: EKF Y ZUPT

3.6.1 Extended Kalman Filter (EKF)

El filtro extendido de Kalman es un método matemático que fue desarrollado para integrar información proveniente de sistemas no lineales en donde se realizan aproximaciones mediante sistemas lineales locales. Cuando se combinan fuentes de información que por si solas son ruidosas o inexactas (tales como las provenientes de IMU o Encoders), el método matemático predice y corrige los estados del robot para disminuir los sesgos y reducir las inexactitudes en la predicción de posición y orientación de un robot [58] [59].

La exactitud en la aproximación linear de este método depende de la varianza del ruido de las fuentes. Esto implica que incertidumbres grandes o amplitudes en el ruido considerables conllevan a aproximaciones lineales pobres. En general, un sistema no lineal puede ser descrito tal que: [58]

$$x_k = f(x_{k-1}, U_{k-1}, w_{k-1}) \quad z_k = h(x_k) + v_k \quad (14)$$

Donde el ruido w_k puede ocurrir en las fuentes de información de entrada o influenciar los estados del sistema directamente. El filtro EKF consta de una parte de predicción cuyas ecuaciones son: [58]

$$\hat{x}_{k|k-1} = f(\hat{x}_{k-1|k-1}, u_{k-1}) \quad P_{k|k-1} = A P_{k-1|k-1} A^T + F Q_{k-1} F^T \quad (15)$$

Y una parte de corrección dada por las ecuaciones: [58]

$$K_k = P_{k|k-1} C^T (C P_{k|k-1} C^T + R_k)^{-1} \quad (16)$$

$$\hat{x}_{k|k} = \hat{x}_{k|k-1} + K_k (\hat{z}_k - z_k) \quad (16)$$

$$P_{k|k} = P_{k|k-1} - K_k C P_{k|k-1} \quad (16)$$

En la parte de predicción, el sistema no lineal se utiliza para calcular la estimación de la predicción de estado del sistema. Para calcular la matriz de covarianza del ruido de las entradas, se debe calcular la matriz Jacobiana A que describe como se propaga el ruido desde los estados previos hasta los estados actuales. Se calcula también la matriz F que describe la propagación del ruido desde las entradas hacia los estados. En la parte de corrección, la estimación de la medición es calculada en función del estado predicho. La matriz Jacobiana C se determina para describir como se propaga el ruido desde los estados hacia las mediciones [58].

$$A = \frac{\partial f}{\partial x} \Big|_{(\hat{x}_{(k-1)|k}^{-1}), u_{(k-1)}^{(k-1)}} \quad (17)$$

$$F = \frac{\partial f}{\partial w} \Big|_{(\hat{x}_{(k-1)|k}^{-1}), u_{(k-1)}^{(k-1)}} \quad (18)$$

$$C = \frac{\partial h}{\partial x} \Big|_{(\hat{x}_{(k)|k}^{-1})} \quad (19)$$

Finalmente, las matrices de covarianza del ruido representan la incertidumbre asociada a los errores que provienen del modelo y de los sensores como fuentes de entrada. Estas matrices son definidas tal que [58]:

$$Q_k = E\{w(k)w(k)^T\} \quad R_k = E\{v(k)v(k)^T\} \quad (20), (21)$$

Estas ecuaciones representan cuanta confianza tiene el filtro tanto en las predicciones de su modelo y en las mediciones de entrada. Si Q_k es grande, el modelo interpreta que la predicción del modelo es poco confiable y da mayor peso a las mediciones de los sensores. Si R_k es grande significa que los sensores de entrada son ruidosos y poco confiables, y por ello el filtro preferirá las predicciones del modelo [58].

Los filtros EKF son ampliamente utilizados y son una herramienta ampliamente validada en la robótica móvil para combinar fuentes como odometría por ruedas, IMU, ultrasonidos, sensores GPS entre muchos otros. Así mismo son muy útiles para realizar el mapeo o SLAM puesto que permite estimar la posición del robot y al mismo tiempo caracterizar el entorno en el que se mueve. Sin embargo, el filtro EKF solo es confiable en sistemas con modelos Gaussianos y con no linealidades limitadas [55] [58][59].

3.6.2 Zero Velocity Update (ZUPT)

La técnica comúnmente conocida como ZUPT es un método de supresión de error y compensación de posicionamiento ampliamente utilizada en los sistemas por navegación inercial INS. Este aprovecha los intervalos de parada de un robot para asumir que el vector de velocidad del robot v_n expresado en el marco de navegación será igual a cero. La diferencia entre la velocidad estimada mediante la odometría inercial o INS y este vector cero artificial se utiliza como un residual de medición. Este residual corrige

directamente la velocidad y de manera indirecta reduce también la deriva en la orientación y posición del robot [59] [60] [61].

Esta medición residual puede expresarse como: [59]

$$\delta z_{ZVN} = v_{INS^n} - 0^{(3 \times 1)} = H_{ZVN}\delta x + v_{ZVN} \quad (22)$$

Donde v_{INS^n} es el vector de velocidad calculada por el sistema INS y v_{ZVN} es el ruido blanco Gaussiano asociado a esta medición artificial. Este residual se relaciona con el error del estado a través de la matriz de medición H_{ZVN} , que selecciona únicamente los componentes de velocidad del vector de estado. La matriz suele contener una identidad en la parte correspondiente a la velocidad y ceros en el resto, pues esta técnica solo corrige de manera directa la velocidad del robot [59]

La detección correcta de los intervalos de parada es crucial para asegurar que los resultados de esta técnica son confiables. Si el modelo detecta incorrectamente momentos de parada cuando en realidad el robot se está moviendo, las predicciones de posición del robot pueden ser aún peores a los resultados previos a el uso de esta técnica. Es por ello por lo que es fundamental que el ruido de los sensores de la IMU utilizada en el sistema INS sean moderados y que el modelo cuente con un buen sistema de detección de paradas para asegurar buenos resultados en el posicionamiento del robot [59] [61].

3.7 ALGORITMOS DE NAVEGACIÓN Y CONTROL PARA ROBOTS

La navegación autónoma en robots es uno de los procesos más importantes para asegurar un desplazamiento y seguro dentro del entorno. Para ello, los robots integran la información proveniente de sus diversos sensores, modelos de movimiento y reglas de control con el objetivo de determinar la trayectoria de desplazamiento más adecuada de acuerdo con su objetivo [44] [62] [63].

Los algoritmos de navegación pueden agruparse en tres etapas fundamentales: La percepción del entorno, planificación de trayectoria y finalmente el control de movimiento. La percepción del entorno es el proceso mediante el cual el robot utiliza los sensores para adquirir información acerca de los obstáculos, límites y características de su entorno. Para ello, utiliza sensores como ultrasonidos, infrarrojos, IMU, cámaras o LiDAR que pueden complementarse y fusionarse mediante modelos matemáticos que den mayor precisión a la percepción obtenida [62] [64] [65]. Muchos trabajos han demostrado la eficacia de diversos métodos para la detección de obstáculos. Por ejemplo, en [65] se diseña un algoritmo de detección y evasión de obstáculos basado en un LIDAR 2D, empleando segmentación y análisis geométrico para generar rutas libres de colisiones en tiempo real. De igual manera, en [63] se utilizan sensores infrarrojos para realizar el seguimiento de trayectorias y evasión de obstáculos en un robot de

prueba dentro de Webots. En [44] se propone un algoritmo de navegación autónoma que integra sensores ultrasónicos orientados a la detección de bordes de paneles solares y una IMU para mantener la estabilidad del robot de prueba. Este último trabajo destaca la importancia de validar los algoritmos en entornos simulados antes de ejecutar pruebas físicas sobre superficies solares reales.

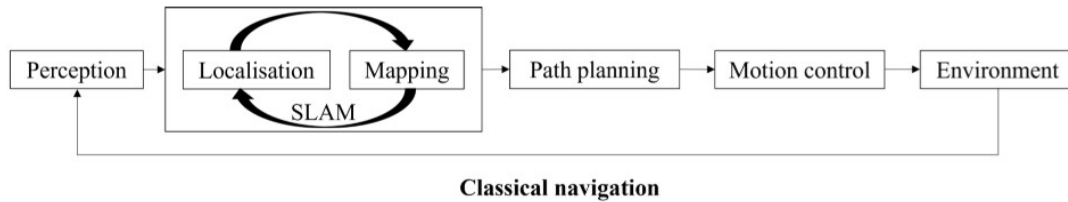


Figura 7. Etapas de navegación en robots. Fuente: [64]

Durante la fase de planificación de trayectorias se utiliza la información recolectada acerca del entorno para estimar la ruta y trayectoria más adecuada de acuerdo con su objetivo de navegación. Comúnmente el objetivo de navegación consiste en la llegada a una meta u objetivo [63]. Las técnicas de planificación de trayectoria varían de acuerdo con la complejidad del entorno en el que se encuentra el robot. Las diversas técnicas que pueden ser empleadas para la planificación de trayectoria pueden dividirse en: Navegación reactiva, planificación de trayectoria simple y algoritmos de navegación basados en percepción avanzada.

La navegación reactiva se basa en principios simples que dependen directamente de la información recolectada durante la percepción del entorno. Entre ellos podemos encontrar métodos basados en la detección de obstáculos y orientación óptima tal como se expone en [44] y [65].

Por otro lado, se encuentran técnicas de planificación más avanzadas que buscan generar caminos de navegación óptimos hacia el objetivo, considerando las restricciones de geometría del entorno que han sido previamente identificadas. Entre estas técnicas podemos encontrar el seguimiento de líneas tal como se expone en [63], o modelos de navegación adaptativos tales como el que se expone en [66].

Finalmente se encuentran algoritmos de navegación que incluyen mapeos y percepciones avanzadas del entorno. Estos métodos incluyen visión por computador, SLAM y sensores de mayor complejidad para construir representaciones más reales del entorno. Entre estos métodos podemos encontrar el propuesto en [62], que utiliza visión por computador para detectar líneas en el entorno además de utilizar odometría magnética para la mejora de navegación en espacios interiores.

El control de movimiento es realizado durante la ejecución del algoritmo de navegación utilizado. Estos métodos de control permiten ajustar velocidades, giros y realizar correcciones dinámicas de la orientación y posición del robot para asegurar el cumplimiento del objetivo. En robots diferenciales, este

control usualmente es implementado mediante controladores PID o derivados que compensan variaciones en el entorno o errores en mediciones. En el trabajo desarrollado en [62] y [56] se utilizan controles de este tipo para ajustar las desviaciones en la trayectoria de navegación de prototipos robóticos.

4. DISEÑO METODOLÓGICO O METODOLOGÍA

Para el alcance de los objetivos propuestos en este trabajo de grado, se utilizará una evaluación con enfoque adaptado a las etapas de proyecto, de tipo aplicado y de carácter experimental simulado. Se utilizarán simulaciones computacionales, recolección de datos y técnicas estadísticas de análisis para obtener conclusiones y recomendaciones acerca del comportamiento del prototipo.

4.1 ALCANCE DEL PROYECTO

Este trabajo de grado se enfocará en la ejecución experimental de simulaciones para un prototipo robótico de limpieza para paneles solares adaptado a las condiciones de instalación propias de entornos urbanos. El alcance está enfocado en la selección y diseño simplificado del prototipo, así como el modelado y ejecución de pruebas en un entorno simulado.

El análisis de resultados se centra en el desempeño de navegación y seguimiento de patrones y algoritmos de navegación bajo diversas condiciones de inclinación. Se busca obtener resultados que permitan identificar limitaciones, precisión en seguimiento de algoritmos de navegación, resultados de odometría y oportunidades de mejora del para el prototipo.

El presente trabajo no incluye la integración de hardware, diseño eléctrico, diseño mecánico de detalle ni validación real en campo. Sin embargo, establece un marco metodológico de simulación y análisis que puede ser usado para el futuro desarrollo del prototipo físico.

4.2 PASOS METODOLÓGICOS

Las etapas definidas de acuerdo con el alcance del proyecto son:

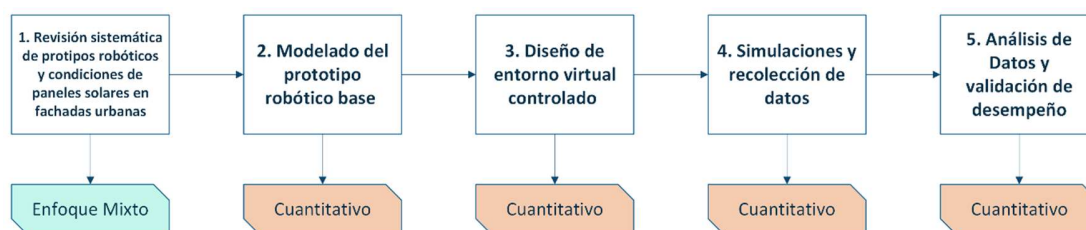


Figura 8. Pasos Metodológicos

Etapa 1: Revisión Sistemática de prototipos robóticos y condiciones de paneles solares en fachadas urbanas

Durante esta etapa se realizará una revisión sistemática de la literatura académica con el objetivo de recolectar información de robots previamente propuestos para la limpieza de paneles solares. Adicionalmente, se realizará una clasificación técnica de los prototipos analizados mediante matrices comparativas que permitan identificar variables clave para tener en cuenta durante la modelación y ejecución de simulaciones.

Etapa 2: Modelado del prototipo robótico base

Teniendo en cuenta la información recopilada en la primera etapa, se procederá a modelar el prototipo robótico a utilizar durante las simulaciones. Teniendo en cuenta el alcance del proyecto, este modelo simplificado se enfocará en sistemas tales como la estructura, ruedas, sistema de tracción y sensores que permitan recolectar información relevante a la estabilidad y desempeño de navegación. De igual forma se establecerá un controlador de navegación que será puesto a prueba durante la ejecución en el entorno de prueba.

Mediante el modelado del prototipo robótico se buscará representar de forma funcional y simplificada los principales sistemas de navegación. De esta forma el modelo podrá responder a los parámetros físicos del entorno de simulación a definir.

Etapa 3: Diseño de entorno virtual controlado

Se desarrollará un entorno virtual controlado que represente las condiciones físicas clave para representar paneles solares en condiciones típicas de instalaciones urbanas. De acuerdo con las variables establecidas se definirán los escenarios de simulación y se diseñarán de forma que sea posible recolectar la información relevante de manera estructurada.

Etapa 4: Simulaciones y recolección de datos

Teniendo en cuenta los escenarios previamente definidos, en esta etapa se ejecutarán las simulaciones de forma sistemática. Se registrarán los datos relacionados con el desempeño de navegación del prototipo bajo las variables de entorno establecidas. Esta etapa corresponde a la ejecución experimental de la investigación y la recolección de datos será realizada mediante la observación de las simulaciones.

Etapa 5: Análisis de Datos y validación de desempeño

Se realizará un análisis cuantitativo de los datos registrados durante la fase experimental de simulaciones, con el fin de evaluar el desempeño de navegación del prototipo. La utilización de técnicas descriptivas permitirá caracterizar y sustentar las conclusiones de la investigación.

5. MATERIALES Y MÉTODOS

5.1 ESTADO DEL ARTE Y ANÁLISIS COMPARATIVO DE PROTOTIPOS

Previo al diseño conceptual y selección de plataforma móvil y sensores a utilizar en las simulaciones, es necesario realizar una revisión de la bibliografía actual y comparar los diseños de prototipos robóticos propuestos para la limpieza de paneles solares. Para ello, se realizó la búsqueda bibliográfica en plataformas como Scopus, Science Direct y IEEE Xplore. Las búsquedas bibliográficas fueron realizadas mediante ecuaciones de búsqueda y palabras claves que permitieran extraer artículos y trabajos específicamente relacionados con la limpieza de paneles solares y soluciones robóticas. Las ecuaciones de búsqueda utilizadas para la extracción de información son:

- robot AND clean* AND "solar panel"
- robot AND clean* AND ("solar panel" OR pv OR solar)
- robot AND automat* AND "solar panel"

De igual forma se establecieron algunos criterios de selección para el filtro de prototipos a analizar:

- Los artículos deben estar escritos en ingles
- Los artículos deben provenir de revistas académicas indexadas, conferencias IEEE o repositorios académicos reconocidos
- Los artículos deben describir un prototipo robótico real o un diseño propuesto ya sea a nivel experimental o en simulaciones
- El robot debe estar diseñado para limpieza de paneles solares o superficies similares tales como fachadas

Finalmente se seleccionaron los criterios y características de comparación entre los prototipos. A partir de la búsqueda bibliográfica, se extrajeron 15 artículos de revista o conferencia con los cuales se construyó una matriz comparativa para el análisis de la información:

Tabla 2. Matriz comparativa de prototipos robóticos

Estudio	Método de Validación	Tipo de Tracción	Tipo de Plataforma Móvil	Condiciones de Climáticas simuladas	Sensores	Patrón / Algoritmo de limpieza	Indicador de Desempeño medido	Tipo de Limpieza
[44]	Simulación (Webots)	2 ruedas	Robot de movimiento libre	Paneles solares en inclinación (no se describen en detalle)	2 acelerómetros y 1 sensor ultrasónico	3 algoritmos creados	Error de orientación del robot Error de detección de bordes	En seco + aspiración
[9]	Validación experimental con prototipo	2 orugas/Correas	Robot de movimiento libre	Condiciones de polvo en desierto. Paneles en inclinación de hasta 30°	No se mencionan	1 algoritmo creado	Tiempo de recuperación de trayectoria	En seco
[67]	Validación experimental con prototipo	Vehículo de 4 ruedas	Vehículo para granjas solares	Granjas solares de gran escala	cámara, LiDAR, infrarrojo. Mapeo del entorno mediante RTK y LiDAR.	1 algoritmo creado	Tasa de limpieza del panel	Con agua
[68]	Validación experimental con prototipo	2 ruedas	Robot de movimiento libre	Paneles solares (No se especifican las condiciones)	Acelerómetros + sensor ultrasónico	1 algoritmo creado	Velocidad de navegación	En seco + aspiración
[69]	Simulación cinemática (Matlab / Simulink)	Vehículo de 4 ruedas	Vehículo para granjas solares	Granjas solares de gran escala	No se mencionan	Modelo de control mediante Machine Learning: Brain	Mean Squared Error (MSE), Root Mean Squared Error (RMSE), Mean Absolute	No mencionado

						Emotial Learning (BEL)	Percentage Error (MAPE) respecto a un controlador PID	
[70]	Validación experimental con prototipo	2 ruedas	Robot de movimiento libre (Robot comercial adaptado)	No se mencionan	No se mencionan	1 algoritmo creado	No se mencionan	En Seco + Aspiración (Modelo comercial)
[71]	Validación experimental con prototipo	2 orugas / correas	Robot de movimiento libre	Paneles en inclinación de hasta 20°	6 sensores ultrasónicos + IMU Cámara	1 algoritmo creado	Precisión en procesamiento de imagen para detección de bus bar del panel solar	En Seco / Con Agua
[72]	Validación experimental con prototipo	4 ruedas Omnidireccionales	Robot de movimiento libre	Paneles en inclinación de hasta 10°	4 encoders (uno para cada rueda) + fotosensores para detección de bordes	No se mencionan	Velocidad de navegación. Error en posicionamiento	No se mencionan
[73]	Validación experimental con prototipo	4 ruedas	Robot de movimiento libre	Paneles en inclinación de hasta 20°	4 sensores ultrasónicos	1 algoritmo creado	Velocidad de navegación. Capacidad de arrastre. Error de sensores en condiciones diversas	No se mencionan
[17]	Simulación (V-REP - ROS)	Vehículo de 4 ruedas	Vehículo para granjas solares	Granjas solares de gran escala	2 sensores laser y SLAM (Simultaneous Localization and Mappin)	Planeación Global: Algoritmo Dijkstra Planeación local: Dynamic Window Approach (DWA)	No se mencionan	No se mencionan

[74]	Simulación cinemática (Matlab / Simulink)	4 ruedas	Robot de movimiento libre	No mencionado	4 sensores infrarrojos	1 algoritmo creado	Tiempo de navegación	En seco
[75]	Validación experimental con prototipo	Rieles fijos sobre bordes del panel	Robot fijo con movimiento en ejes x,y	Plantas solares industriales de gran escala	Sensores de proximidad para movimiento en eje x , y. Sensor infrarrojo para distancia de superficie con cepillos	1 algoritmo creado	No mencionado, el estudio se centra más en la conexión a red del robot	En seco
[76]	Validación experimental con prototipo	Rieles fijos sobre bordes del panel	Robot fijo con movimiento en ejes x,y	Plantas solares industriales de gran escala	Switches Limitadores	1 algoritmo creado	Producción de potencia del panel antes y después de la limpieza	Con agua
[77]	Validación experimental con prototipo	4 ruedas + Sistema de sellado por presión negativa	Robot de movimiento libre	Paneles en inclinación de hasta 30°	No se mencionan	No se menciona. El estudio se centra en el rendimiento del robot y su presión negativa en diversos ángulos	Transmitancia del panel, eficiencia del panel	En seco + Aspiración
[78]	Validación experimental con prototipo	2 orugas/Correas	Robot de movimiento libre	Paneles en inclinación de hasta 30°	Sensores Inductivos + Sensores ultrasónicos	1 algoritmo creado	Tiempo de limpieza, velocidad cruceo promedio del robot	En seco + Aspiración

A partir de la información y observaciones recolectadas de estos estudios se diseñó el prototipo simplificado que fue utilizado durante las simulaciones. Puesto que el objetivo del trabajo es validar

únicamente el desempeño de navegación y no el desempeño de limpieza, únicamente se tuvieron en cuenta algunos aspectos de los prototipos analizados para la consideración y diseño del prototipo a simular.

5.2 SELECCIÓN DE CARACTERÍSTICAS Y DISEÑO DEL PROTOTIPO SIMPLIFICADO.

El diseño simplificado del prototipo se realizó teniendo en cuenta las capacidades de simulación de ROS2 y Webots. Para asegurar que el desempeño de navegación fuera evaluable, se seleccionaron algunas de las características importantes de los diferentes estudios obtenidos durante la revisión bibliográfica. A partir de la información recolectada, se determinó que las características más relevantes en el diseño del prototipo son:

- Tipo de tracción
- Plataforma de movimiento
- Sensores empleados
- Algoritmo de navegación utilizado

El análisis de estas características debe ser realizado a partir de criterios que reflejen las condiciones específicas de paneles solares en fachadas urbanas como las de Medellín. Es por ello que fue necesario también realizar un estudio para determinar los criterios de selección que fueron utilizados para la evaluación:

Capacidad de Navegación en superficies inclinadas

En ciudades como Medellín el ángulo óptimo de instalación, desde un punto de vista de generación eléctrica, suele estar entre 10° y 15° [27] [28]. Sin embargo, en muchos casos la inclinación de la instalación está determinada por la geometría de los techos y fachadas [26]. De acuerdo con los rangos de inclinaciones más comunes que fueron encontrados en los estudios y teniendo en cuenta que en cubiertas o techos inclinados suelen observarse hasta 7/12 de inclinación, se determina 30° de inclinación respecto a la superficie horizontal es el límite de validación para las simulaciones. Este valor garantiza una buena representación de las condiciones más comunes en techos urbanos como los de Medellín.

Peso del prototipo

El peso es un parámetro crítico ya que debe ser lo suficientemente bajo para no comprometer la integridad estructural de los paneles y a su vez debe garantizar la estabilidad y tracción del prototipo durante la navegación de las superficies inclinadas. Son varios los estudios que destacan los fenómenos a los cuales se ven expuestos robots que se desplazan sobre superficies inclinadas. En general, se considera que estos robots experimentan fenómenos de deslizamiento (movimiento no controlado producto de la baja gravedad y fricción) y vuelco (fenómeno de rotación no controlada cuando el centro de gravedad se desplaza por fuera de su área de apoyo) [79] [80] [81]. El deslizamiento de un robot puede modelarse geoméricamente a partir de la relación entre la pendiente de una superficie y coeficiente de fricción de Coulomb entre los materiales [82]. Mayores áreas de contacto incrementan la fracción y por ende mejorar la estabilidad y resistencia al deslizamiento en superficies inclinadas [83] [84]. Por ello, el peso y tipo de tracción deben evaluarse de forma conjunta considerando la influencia de estos factores sobre la fuerza normal, torque requerido para motores y la resistencia al deslizamiento.

Flexibilidad

El prototipo diseñado deber ser capaz de adaptarse a las diversas condiciones de instalación presentes en entornos urbanos. Entre diferentes paneles solares pueden encontrarse condiciones de inclinación, orientación, presencia de obstáculos o áreas de maniobra disponibles. Esta flexibilidad está relacionada con la capacidad del robot de ejecutar su navegación autónoma de forma exitosa en diversas configuraciones. Esta necesidad implica también una adecuada planeación de navegación y un control robusto o ante perturbaciones del medio ambiente.

Simplicidad de Control

Teniendo en cuenta las características y limitaciones de la ejecución de pruebas en un entorno virtual como el de Webots, es necesario encontrar un equilibrio entre la precisión y simplicidad computacional. Es importante que el comportamiento del prototipo pueda validarse de forma reproducible en el entorno virtual sin dificultades mayores. En este contexto se priorizan esquemas de control y algoritmos de navegación que no sean excesivamente complejos o dependientes de datos visuales.

Con base en estos criterios de evaluación, se elaboró una matriz de decisión ponderada que incluye los atributos que influyen directamente en el comportamiento dinámico del robot sobre superficies inclinadas. Estos aspectos se seleccionaron de acuerdo con la revisión bibliográfica realizada y al tener una relación directa con la movilidad y estabilidad del robot. De las características determinadas para el diseño del prototipo, solo el Tipo de tracción y Plataforma de movimiento fueron sometidas a evaluación mediante las matrices. Este debido a que corresponden a características estructurales que deben ser definidas previo al diseño de las simulaciones. Por otro lado, los sensores y algoritmos de navegación empleados serán evaluados durante la ejecución de las simulaciones para comparar el desempeño de múltiples combinaciones bajo diversas condiciones de inclinación.

Tabla 3. Matriz de decisión para tipo de tracción

	Capacidad de movimiento en	Resistencia al vuelco	Radio de giro / Maniobrabilidad	Complejidad de control y planeación de trayectoria	Área de contacto	Capacidad de superar obstáculos	Slip en giro	Flexibilidad	Cambio en coeficiente de fricción (húmedo, seco, etc.)	
Peso Ponderado	20%	15%	12%	10%	10%	5%	10%	10%	8%	Resultado
Ruedas Convencionales	3	3	3	5	2	2	3	3	2	2,97
Orugas	5	4	2	4	5	5	2	4	4	3,91
Ruedas Omnidireccionales	2	3	5	2	2	2	3	5	2	2,91
Mixto	4	4	4	2	4	4	3	5	3	3,72

Tabla 4. Matriz de decisión para plataforma de movimiento

	Capacidad de movimiento en pendiente	Resistencia al vuelco	Adaptabilidad	Cobertura Área de limpieza	Precisión de Posicionamiento	
Peso Ponderado	20%	15%	25%	25%	15%	Resultado
Movimiento libre	3	2	5	4	3	3,6
Estructura fija (Rieles)	4	5	2	3	4	3,4

Los resultados de la ponderación brindan resultados muy cercanos entre un sistema de tracción tipo oruga y un sistema de tracción mixto o híbrido. Dado que el resultado de tipo oruga obtiene resultados ligeramente mejores y su implementación en el entorno de Webots cuenta con facilidades de diseño, se opta por esta opción para el prototipo. De manera similar, ambas opciones entre las plataformas de movimiento obtienen resultados muy cercanos. Priorizando el requerimiento de flexibilidad que pueden brindar potencialmente las plataformas de movimiento libre, se selecciona esta opción para el diseño. Es por ello por lo que el prototipo robótico que será diseñado constará de un sistema de tracción de tipo oruga y una plataforma de movimiento libre sobre la superficie de paneles.

5.2.1 Modelado del prototipo robótico

Teniendo en cuenta las características seleccionadas, se modeló el CAD del prototipo en el programa Autodesk Fusion. La modelación fue realizada teniendo en cuenta dimensiones y características de diseño que fueran acordes a los requerimientos de maniobrabilidad y trabajos previos relacionados. El sistema de orugas fue modelado de forma simplificada, incluyendo únicamente las ruedas principales, ruedas secundarias y banda puesto que estos detalles serán los únicos requeridos para la ejecución de las simulaciones. Como característica importante, se estimó un peso fijo para la utilización en simulaciones de acuerdo con los componentes típicos de un robot de limpieza de paneles solares. En el presente trabajo no se consideró un cambio dinámico de peso durante la limpieza y navegación por el vaciamiento de los tanques de agua o líquido de limpieza. Esta consideración se dejará para futuros desarrollos o simulaciones.

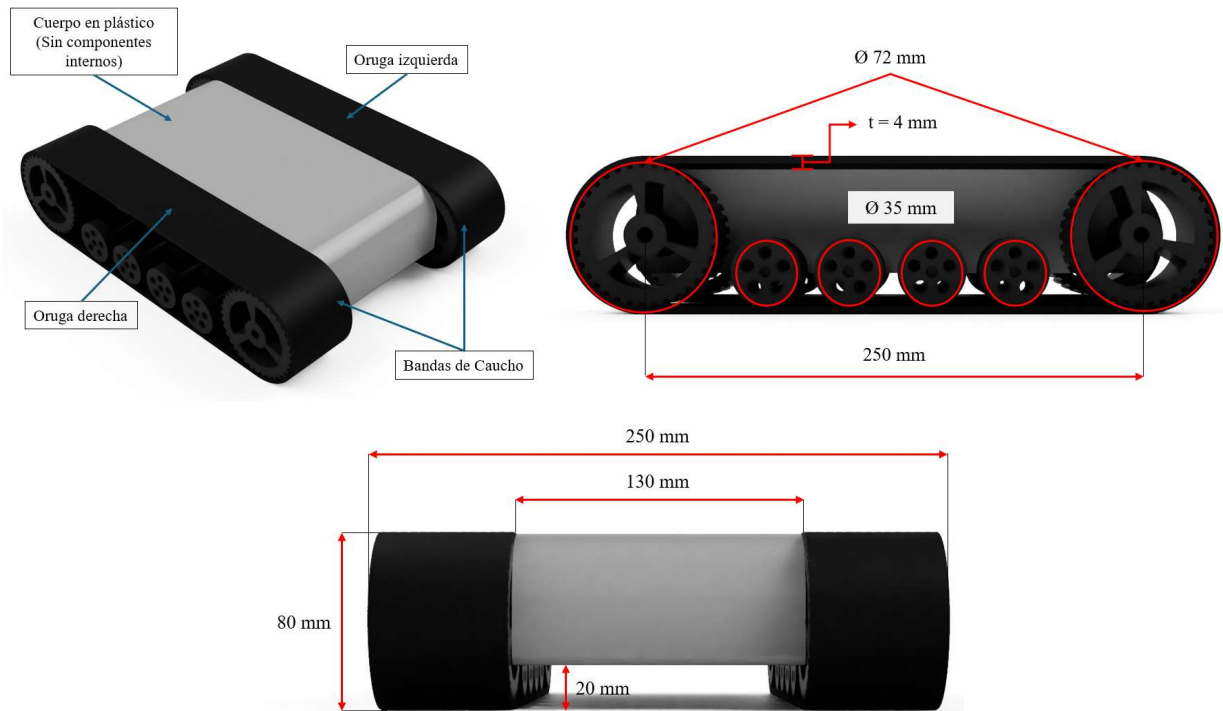


Figura 9. Diseño simplificado del prototipo robótico

Los materiales utilizados en la modelación fueron seleccionados de manera genérica como caucho para las bandas, Nylon para ruedas principales y secundarias y plástico para el cuerpo del prototipo. No se modelaron los componentes internos del robot, pero estos serán tenidos en cuenta para el cálculo estimado de peso y simulación de centro de gravedad en Webots.

Tabla 5. Peso estimado del prototipo

Componente	Peso Estimado
Estructura	1.9 kg
Baterías	1.6 kg
Componentes electrónicos y cableado	0.6 kg
Sensores	0.4 kg
Motores y sistema de tracción	1.5 kg
Sistema de Limpieza (Lleno)	2 kg
Masa total estimada del prototipo	8 kg

Posteriormente, el modelo del prototipo fue importado al entorno de Webots. Este entorno permite simular en detalle el funcionamiento dinámico de las orugas y sensores a implementar en el prototipo. A continuación, se detallan los componentes utilizados para la configuración del robot dentro del entorno que será simulado:

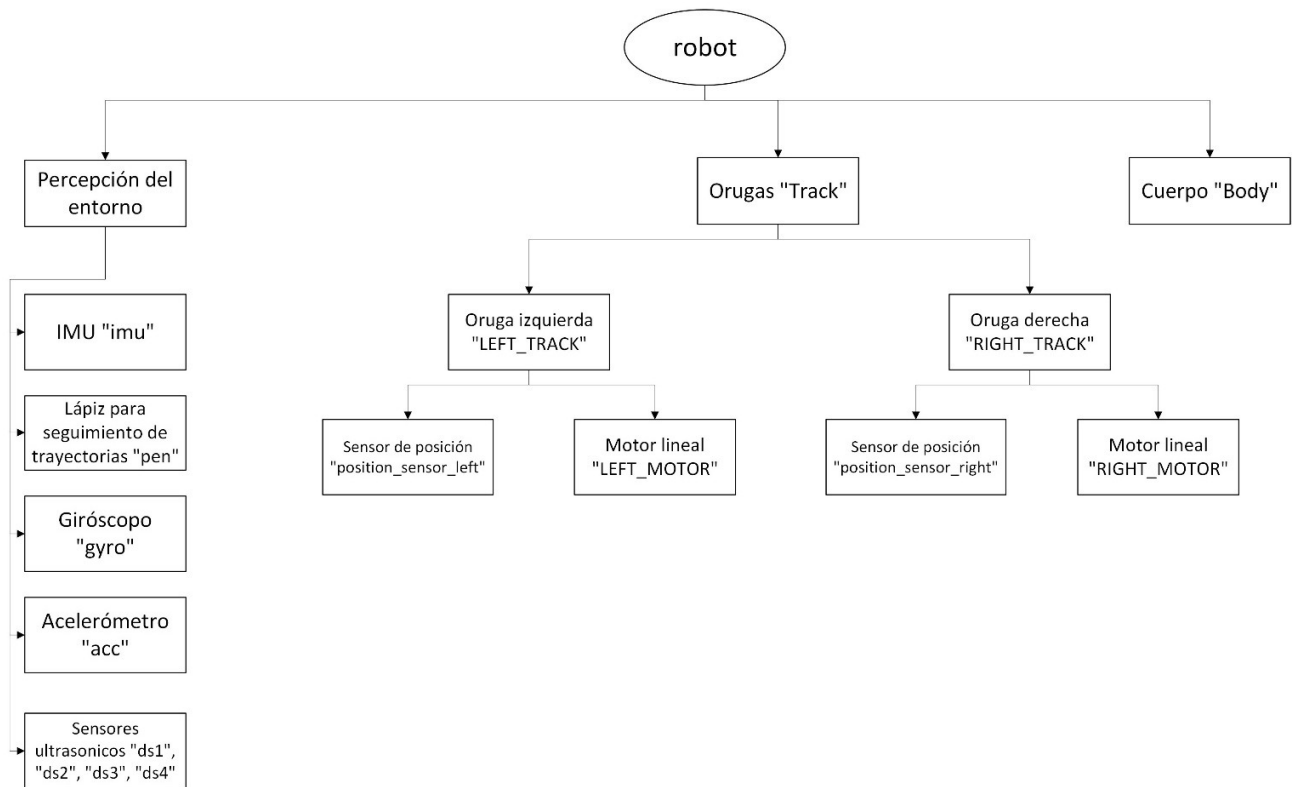


Figura 10. Estructura de prototipo simulado en Webots

Se configuraron los parámetros físicos del robot incluyendo su masa, límites de colisión de acuerdo con la geometría del prototipo, centro de gravedad y parámetros de funcionamiento de cada uno de los sensores a utilizar. La IMU es simulada con un ruido Gaussiano de 0.003 radianes para simular las condiciones reales de perturbación e imperfección que pueden tener los componentes de este dispositivo en referencias comercial de calidad media. Los sensores ultrasónicos son modelados con un rango máximo de medición de 0.4 m.

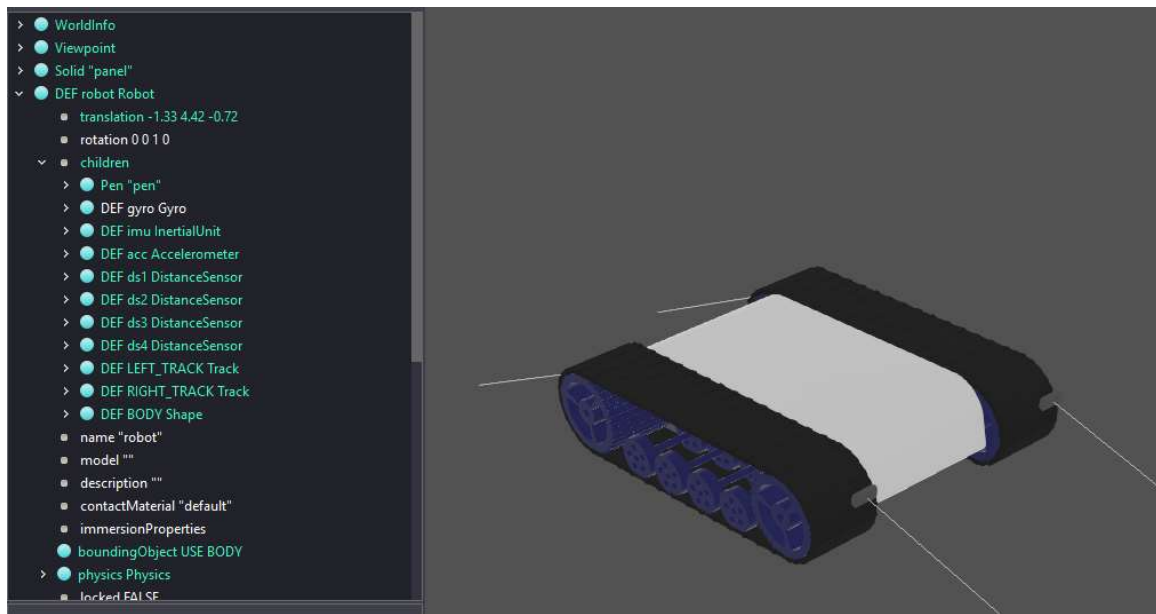
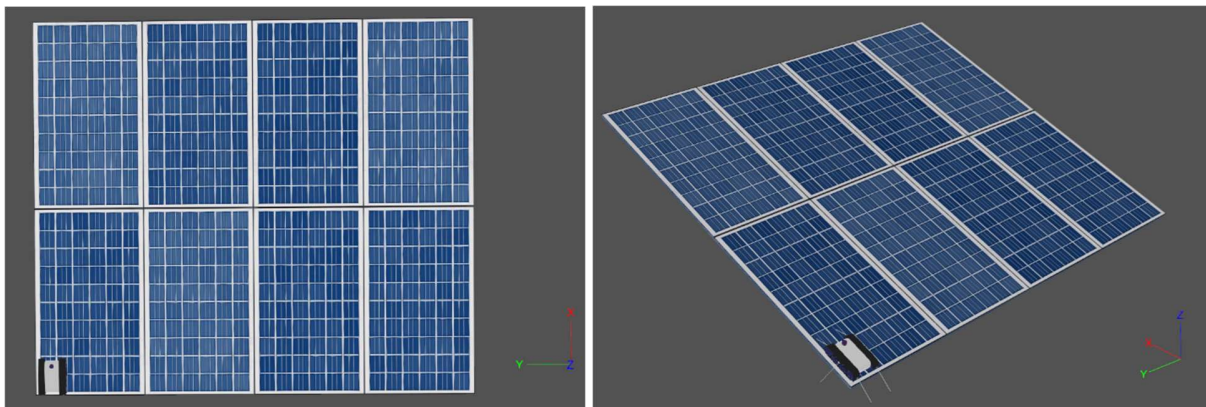


Figura 11. Prototipo en Webots

5.3 DISEÑO DEL ENTORNO VIRTUAL EN WEBOTS

El entorno virtual o mundo en Webots fue diseñado para simular las condiciones de inclinación de un arreglo de paneles solares típico de instalaciones en fachadas urbanas. El arreglo de paneles solares de prueba tiene unas dimensiones de 3.4 m x 4 m que deberán ser recorridos por el prototipo en su totalidad para simular la trayectoria de limpieza. Se establecieron múltiples entornos de prueba para diversas inclinaciones en el arreglo de paneles solares: 0°, 5°, 10°, 15°, 20°, 25° y 30° como límite de operación del prototipo. No se consideraron para este trabajo las separaciones propias entre paneles solares ni las irregularidades sobre su superficie pues se reservan estas consideraciones para trabajos posteriores.



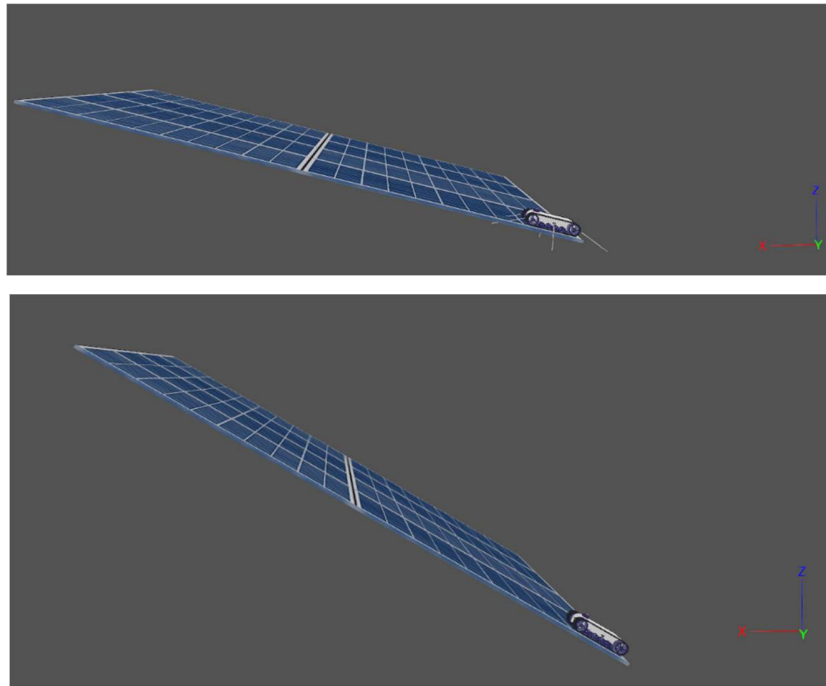


Figura 12. Entorno de simulación diseñado en Webots

Otro aspecto importante en la definición del mundo es la configuración de parámetros de fricción entre el material de las orugas del prototipo (caucho) y la superficie del panel solar (vidrio). El nodo “ContactProperties” de Webots permite establecer el comportamiento durante el contacto de dos materiales establecidos en el mundo [85]. Para esta simulación se establecieron los materiales “panel” y “track material” asignados respectivamente al arreglo y a las bandas de las orugas del prototipo. El parámetro “CoulumbFriction” permite configurar dos sub-parámetros μ_1 y μ_2 que actúan en direcciones ortogonales a la superficie de contacto. Estudios experimentales han demostrado que el caucho puede alcanzar coeficientes de fricción elevados cuando desliza sobre materiales como el vidrio bajo condiciones secas [86]. Estudios clásicos tales como [87] demuestran que, en superficies lisas como el vidrio, la fricción del caucho depende de la adherencia propia del material y la geometría del contacto. En [88] se reportaron valores de fricción cinética entre 0.74 y 1.34 para caucho en contacto con vidrio dependiente la geometría de los elementos y la velocidad de deslizamiento. Teniendo en cuenta las condiciones de inclinación requeridas para el funcionamiento del robot de 30° , se estableció un coeficiente de fricción mínimo de 0.8 que garantiza la adherencia necesaria en la superficie y un segundo coeficiente de 1.5 ortogonal, para representar una mayor tracción ortogonal típica de las orugas [89]. Respecto a los demás parámetros de contacto, se asignó un valor de restitución muy bajo “bounce” de 0,02 para evitar rebotes o vibraciones numéricas no deseadas. Asimismo, se incluyó un valor pequeño

en el parámetro “forceDependentSlip” de 0,01 que permite que el modelo reproduzca una transición más suave entre adherencia y deslizamiento. Esto resulta coherente con el comportamiento viscoelástico del caucho, donde el movimiento no comienza de forma brusca sino mediante pequeños micro deslizamientos controlados que ayudan a mantener la tracción sin perder estabilidad [89].

Este trabajo no se incorporaron condiciones de humedad ni suciedad significativas sobre la superficie del panel, ya que el objetivo principal de la simulación es aislar el comportamiento básico de interacción caucho-vidrio bajo condiciones controladas. La literatura muestra que tanto la presencia de agua como de partículas contaminantes pueden reducir de manera significativa la fricción efectiva, modificando la transición entre fricción estática y cinética [86] [87]. Sin embargo, estas condiciones requieren modelaciones más complejas y con mayor demanda computacional que podrán ser estudiadas en trabajos posteriores o en desarrollos experimentales reales.

5.4 IMPLEMENTACIÓN EN ROS

5.4.1 Arquitectura del paquete en ROS 2

El control del prototipo y la adquisición de datos de las simulaciones fue realizado dentro del entorno de ROS 2 y para ello fue desarrollado un paquete que integra diversos nodos encargados de ejecutar distintos esquemas de navegación y estimación del prototipo. Las simulaciones realizadas tienen como objetivo comparar como estas diversas combinaciones de sensores y algoritmos de navegación reflejan la posición y trayectoria del prototipo bajo las diversas condiciones de inclinación. La arquitectura del paquete fue desarrollada de manera modular, permitiendo que nodos específicos del paquete fueran utilizados según la combinación de sensores y algoritmos de cada simulación.

El paquete está compuesto por tres grupos principales de nodos:

- **Nodo de percepción:** Encargado de extraer los datos medidos por cada uno de los sensores del prototipo (sensores ultrasónicos, IMU y encoders) y publicar estos datos en tópicos de ROS independientes que puedan ser usados posteriormente por otros nodos. La extracción de los datos provenientes de Webots es realizada mediante la definición de un archivo URDF que describe la estructura funcional del prototipo, sus marcos internos de referencia y ubicación física de cada sensor. A partir de este archivo, el nodo `ros2_supervisor` exporta estos datos simulados y los publica en tópicos independientes asociados a cada sensor.

- Nodos de navegación: Encargados de controlar el movimiento autónomo del prototipo y modificar su comportamiento a partir de las mediciones obtenidas por los sensores. Teniendo en cuenta los sensores que han sido implementados para el prototipo, se optó por utilizar algoritmos de navegación reactiva que detectan bordes del arreglo de paneles para ejecutar trayectorias definitivas que cubren la superficie completa a ser limpiada
- Nodos de estimación de odometría: Encargados de procesar los datos obtenidos por los sensores utilizados en cada simulación para estimar la posición y trayectoria recorrida por el prototipo.

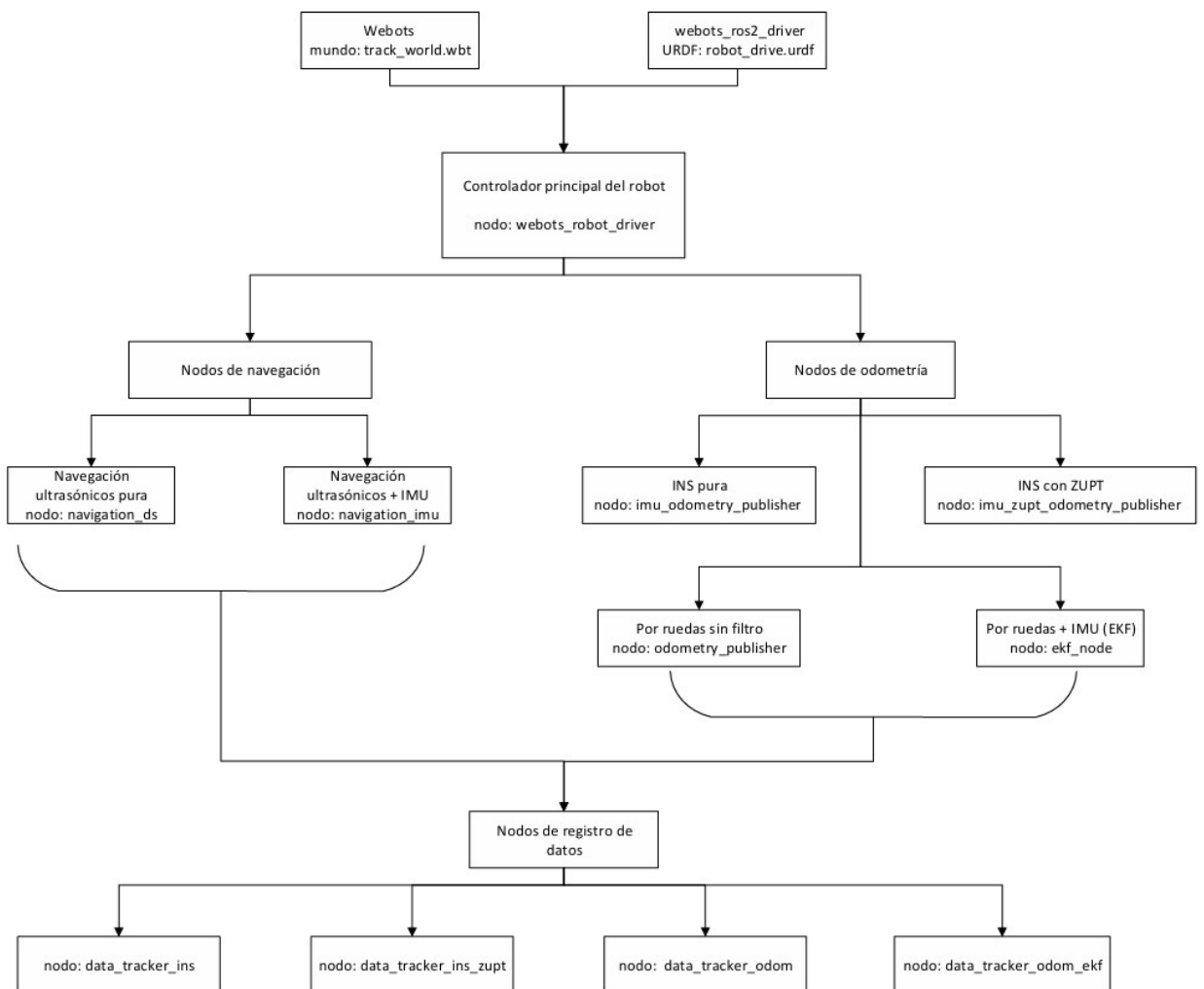


Figura 13. Arquitectura del paquete en ROS 2

5.4.2 Algoritmos de navegación

Se implementaron dos algoritmos de navegación reactiva que permiten al robot recorrer el arreglo de paneles solares mediante estrategias de detección de bordes, giros de 90° y avances. El patrón de recorrido del panel fue implementado basándose en estrategias de control de prototipos robóticos similares encontrados en la literatura [44] [68]. Este patrón permite limpiar completamente la superficie de los arreglos de manera eficiente y mediante la ejecución de un ciclo predefinido. En ambos algoritmos el prototipo inicia en la esquina inferior derecha del arreglo de paneles y ejecuta el ciclo hasta detectar el borde superior o inferior del lado derecho, momento en que el prototipo es detenido. De igual forma, se implementaron dos variaciones del algoritmo: un algoritmo que utiliza únicamente las lecturas de sensores ultrasónicos para la detección de bordes y otro algoritmo que adicionalmente se apoya en las lecturas de la IMU para el control de giros. Ambos algoritmos fueron utilizados en la combinación de simulaciones correspondientes.

Navegación por ultrasonido: Este constituye el control reactivo más sencillo. Se controla el avance y las transiciones entre franjas de limpieza del panel. Se ejecuta una máquina de estados finitos y se realiza control de velocidad de ambos motores (ubicados en cada una de las orugas del prototipo) para realizar los avances y giros requeridos. La ejecución de los giros se realiza únicamente mediante estimación del tiempo requerido para alcanzar los 90° de giro.

Navegación por ultrasonido + IMU: Utiliza una lógica de estados similar al algoritmo anterior pero que implementa un control de giros usando las lecturas de “yaw” provenientes de la IMU. Los giros propios del patrón son ejecutados en dos fases. Una fase inicial de mayor velocidad angular y menor precisión que posteriormente hace una transición hacia un giro con menor velocidad y mayor control que asegura una precisión mayor en el objetivo de 90° . El control de giro es realizado mediante un controlador PID. Finalmente implementa también otro controlador PID de rumbo para compensar cualquier error o ruido presente en las trayectorias de avance recto.

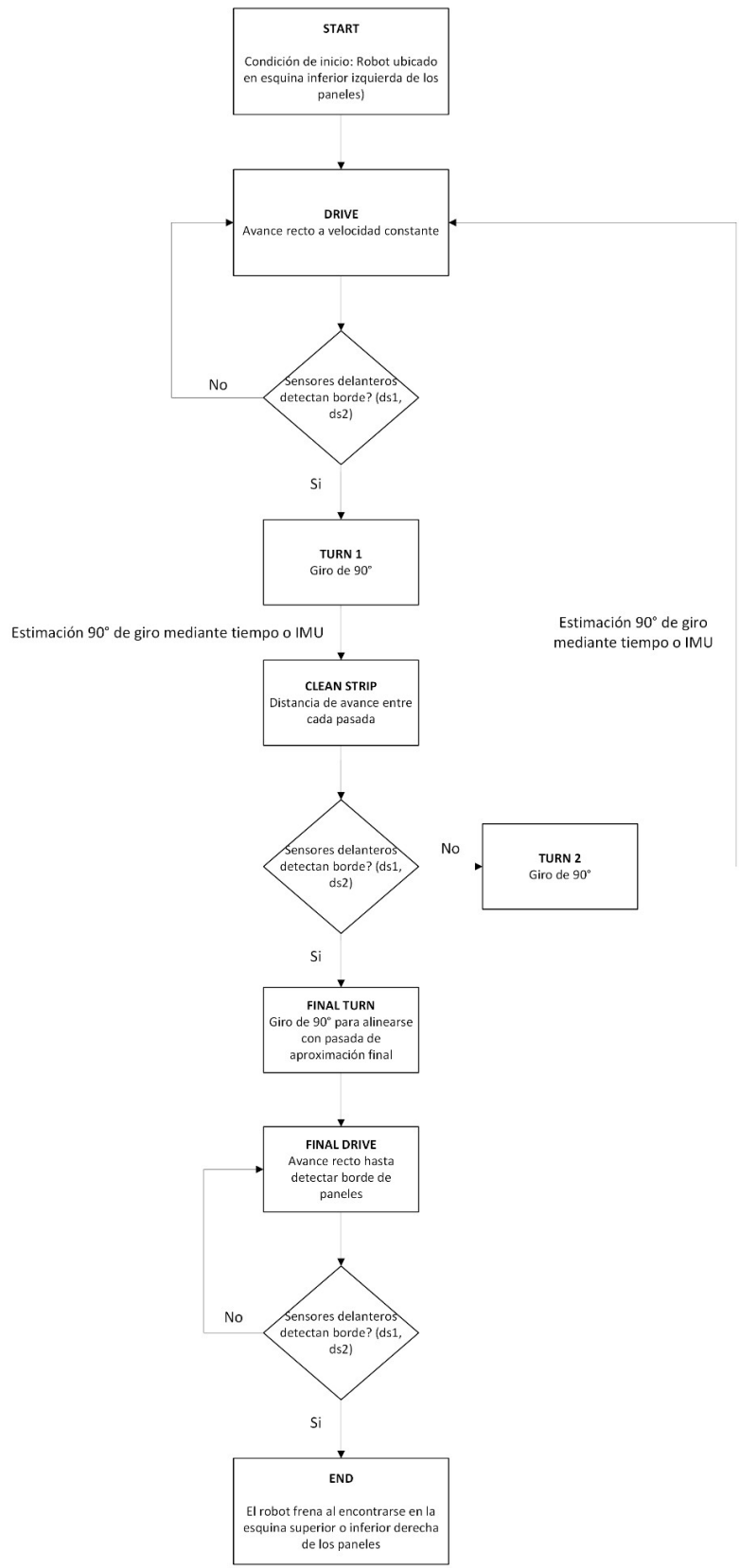


Figura 14. Algoritmos de Navegación implementados

5.4.3 Estimación de odometría

Se implementaron cuatro técnicas diferentes para obtener las estimaciones de posición y trayectorias recorridas por el robot al seguir el algoritmo de navegación correspondiente a cada caso. Los métodos de estimación de odometría utilizados para estas simulaciones son:

- Odometría Inercial Pura (INS): Integra las mediciones del giroscopio y acelerómetro provenientes de la IMU para integrar temporalmente la aceleración lineal y angular y con ello estimar la posición y rotación del robot respecto a su localización inicial.
- INS corregida mediante ZUPT (Zero Velocity Update): Se incorpora esta técnica al método de odometría inercial para aprovechar los instantes en que el robot se detiene antes de la realización de un giro y aplicar una corrección al error acumulado.
- Odometría por ruedas sin filtro: Integra las mediciones provenientes de los sensores lineales y la geometría de cada oruga. En Webots, las orugas son implementadas mediante motores lineales y sensores de posición lineales. Por ello, la estimación de odometría por ruedas es modificada respecto a las aproximaciones clásica de robots diferenciales tal que:

$$v = \frac{(v_R + v_L)}{2} \quad \omega = \frac{(v_R - v_L)}{2b} \quad (23)$$

$$\dot{x} = v \cos(\theta) \quad \dot{y} = v \sin(\theta) \quad \dot{\theta} = \omega \quad (24)$$

Donde b es igual a la mitad de la distancia entre las orugas y v_R, v_L corresponden a las velocidades lineales otorgadas por cada sensor lineal provenientes de Webots.

- Odometría de ruedas fusionada con IMU mediante EKF: Combina las estimaciones de posición provenientes de la odometría por ruedas con las lecturas provenientes de la IMU. El filtro extendido de Kalman (EKF) utiliza las lecturas de la odometría por ruedas para realizar la predicción del modelo y las mediciones de “yaw” y velocidad lineal para realizar la corrección y reducir los errores propios de ambas técnicas individuales.

5.4.4 Registro de datos

El paquete de ROS implementa también nodos de registro de datos que permitió almacenar la información de cada una de las simulaciones ejecutadas. Estos nodos crean archivos .csv con sello de tiempo, que registran los datos de posición en x, y (respecto a los marcos de referencia correspondientes) y orientación con una frecuencia de toma de datos de 10 Hz (cada 0.1 segundos). En cada archivo se

registran los valores reales de posición del robot en el mundo de Webots (“ground_truth_pose”) y los valores de posición estimados por cada método de odometría correspondiente (“odom_pose”). Este sistema de registro garantiza la recolección sincronizada de datos que permitirá realizar comparaciones cuantitativas entre las diferentes técnicas de odometría y algoritmos de navegación implementados.

5.4.5 Protocolo para ejecución de las simulaciones

Con el objetivo de estructurar y asegurar la reproducibilidad de las simulaciones, se definió un protocolo experimental para las combinaciones de algoritmos de navegación y estimación de odometría realizadas. Se implementaron cuatro escenarios principales, cada uno evaluado en configuraciones de inclinación de 0°, 5°, 10°, 15°, 20°, 25° y 30°. Con el objetivo de minimizar los errores en ejecución y asegurar una aleatoriedad en los ruidos y errores presentes en cada iteración, se ejecutaron 3 simulaciones por cada escenario de configuración e inclinación. En total se ejecutaron 84 iteraciones diferentes, asegurando en cada una de ellas el registro adecuado de datos de posición real y de posición por estimación de odometría.

Tabla 6. Protocolo de simulaciones

Sensores para navegación	Método de Estimación de Odometría	Configuración ejecutada en ROS	Número de escenario
Ultrasónico	Encoders (Sin Filtro)	robot_odom_launch	1
Ultrasónico + IMU	IMU (INS Pura)	robot_ins_launch	2
	IMU (INS con ZUPT)	robot_ins_zupt_launch	3
	Encoders + IMU (EFK)	robot_odom_ekf_launch	4

6 RESULTADOS

Los resultados de cada una de las configuraciones de sensores y odometría obtenidos fueron condensados en inclinaciones representativas de 0° , 10° , 20° y 30° . En cada simulación, las posiciones estimadas en los ejes x e y se compararon con las posiciones reales registradas dentro del entorno de simulación en Webots, así como con la trayectoria estimada para cada iteración. De igual forma el dispositivo “pen” configurado para el prototipo dentro de Webots, permitió representar la acción de limpieza del robot al pasar sobre la superficie del panel. Para ello, el lápiz cuenta con un ancho de 0.35 m que representa el área efectiva de limpieza que tendría el sistema de cepillos y/o aspiración del prototipo. Durante la simulación, el lápiz colorea la superficie de amarillo con el ancho especificado. El resultado esperado al finalizar cada simulación es que toda la superficie del arreglo de paneles sea coloreada de amarillo. Así se demostrará la efectividad en la capacidad de limpieza del prototipo cuando son empleados los algoritmos de navegación propuestos. Para el análisis de resultados se realizó un promedio de las tres iteraciones correspondientes a cada escenario e inclinación simulados.

Inclinación de 0° : El escenario sin inclinación en el arreglo de paneles constituye el escenario ideal para el funcionamiento del robot puesto que los deslizamientos son mínimos y la efectividad en los métodos de estimación de odometría fueron los mejores. La efectividad de cobertura de limpieza en los cuatro escenarios de configuración: odometría de ruedas pura, INS pura, INS con ZUPT y odometría de ruedas con filtro EKF resultó similar y en todos los casos se aseguró una cobertura casi completa de toda la superficie de los paneles. Las desviaciones en la alineación del robot durante etapas de avance recto pueden deberse a los ruidos Gaussianos simulados para la IMU.

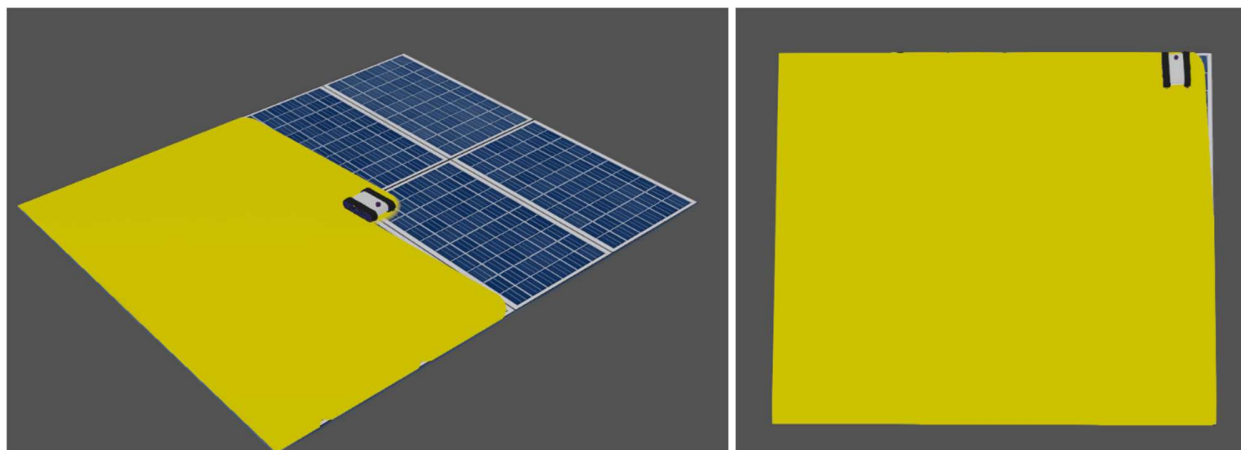


Figura 15. Cobertura de limpieza del prototipo en 0° de inclinación

Comparando la trayectoria esperada y las trayectorias reales ejecutadas para cada uno de los escenarios, se observó que el error cuadrático medio RMSE, la desviación estándar y el error promedio del escenario número 1 fue mayor respecto a los otros tres escenarios. Esto debido a que este escenario particular no cuenta con el apoyo de las lecturas de la IMU en su algoritmo de navegación y por ello el error acumulado en la estimación de los giros de 90°, al detectar los bordes, implica una desviación mucho más considerable respecto a los otros escenarios. El error fue calculado respecto a los puntos de la trayectoria ideal más cercanos a cada punto de la trayectoria real del robot. Así mismo, este escenario requirió de más pasadas para completar la limpieza del panel y por ende el tiempo de limpieza es considerablemente más alto. A pesar de ello, todas las configuraciones fueron lo suficientemente robustas para asegurar que la totalidad del panel fuera limpiada durante la navegación del prototipo.

Tabla 7. Resultados en error de trayectoria real del prototipo en 0° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	387	0,19	0,12	0,14
2. INS	261	0,04	0,02	0,04
3. INS con ZUPT	260	0,07	0,04	0,06
4. Encoders con EKF	270	0,09	0,05	0,08

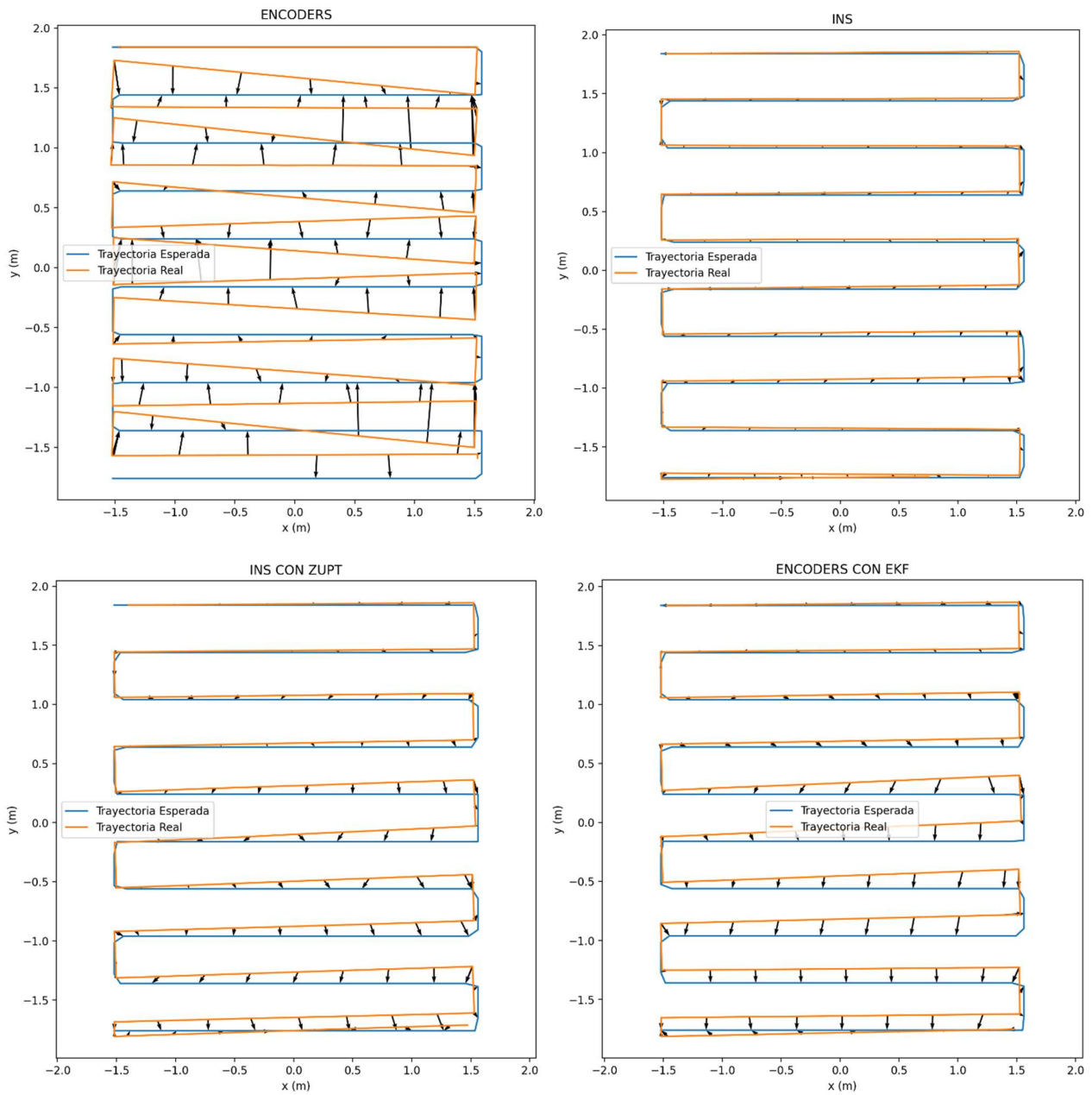
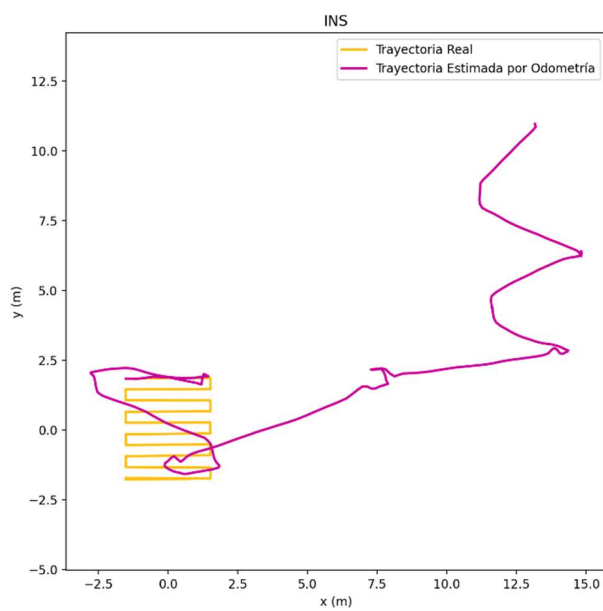
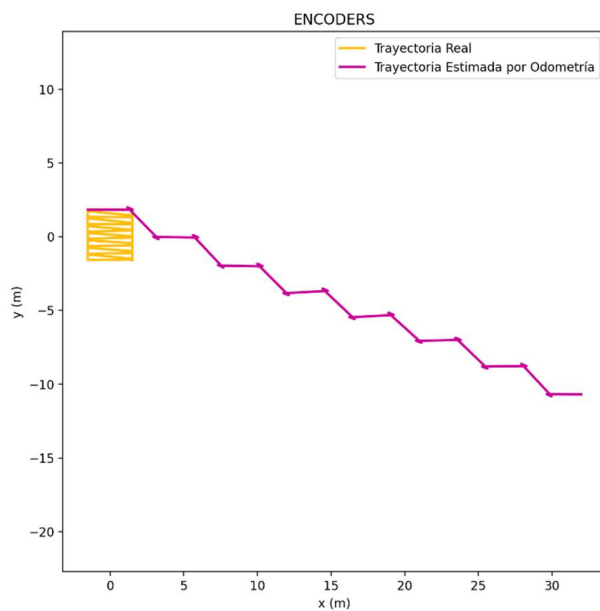


Figura 16. Trayectorias Esperadas y Reales de las simulaciones en 0° de inclinación

En cuanto a las estimaciones de posición y trayectorias calculadas por cada uno de los métodos de odometría, los resultados fueron contundentes al demostrar que el único método que refleja de manera aproximada la trayectoria real que realizó el robot durante su limpieza fue la fusión de sensores mediante el filtro EKF. Los resultados en los otros escenarios demostraron que la acumulación de error y deriva es considerable durante la ejecución de giros y por ende resulta complejo utilizar la información proveniente de estos métodos para estimar la posición del robot en el tiempo.

Tabla 8. Resultados en error de estimación de trayectoria por odometría del prototipo en 0° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	386	17,95	9,99	14,91
2. INS	261	9,97	6,51	7,55
3. INS con ZUPT	260	16,55	10,62	12,70
4. Encoders con EKF	270	0,05	0,03	0,05



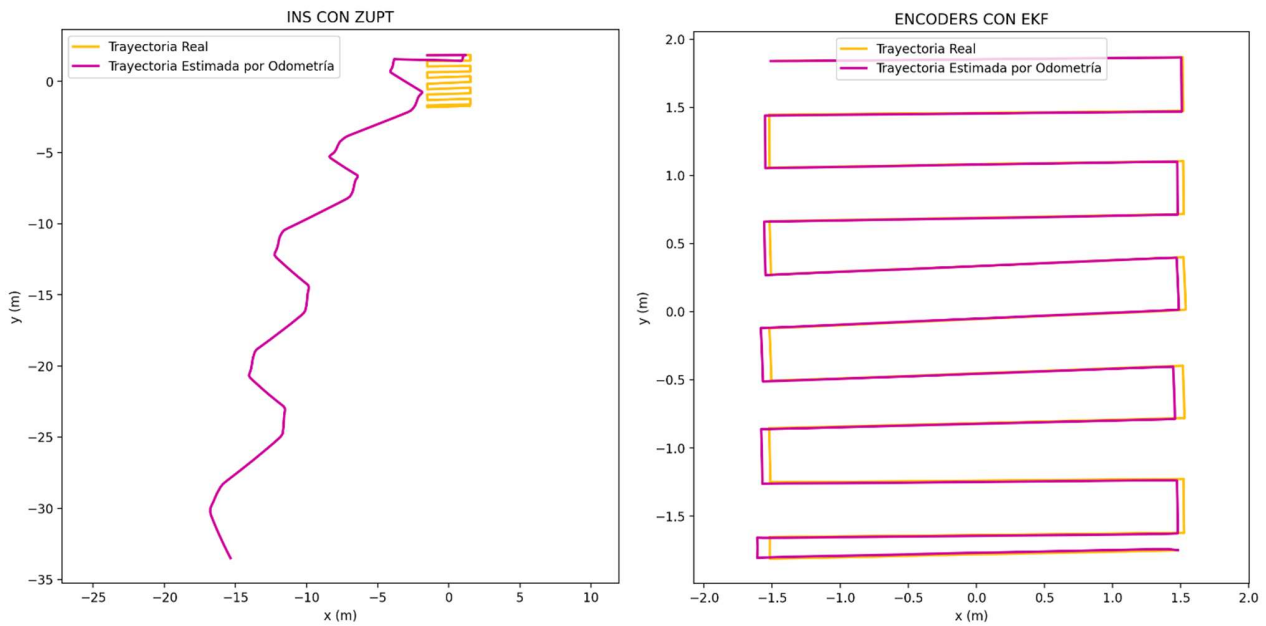


Figura 17. Trayectorias Esperadas y por odometría de las simulaciones en 0° de inclinación

Inclinaciones de 10° , 20° y 30° : En condiciones de fachadas urbanas típicas de Medellín será poco probable encontrar instalaciones de paneles solares ajustadas exactamente a 0° de inclinación. Esto teniendo en cuenta las consideraciones de instalación previamente expuestas en este trabajo.

Con el incremento en la inclinación de los paneles, se evidenció un fenómeno de deslizamiento considerable del prototipo durante la ejecución de los giros. Este comportamiento observado es consistente con las conclusiones expuestas en [89], en donde se concluye que los robots tipo oruga pueden desplazarse con buena estabilidad en trayectorias rectas, pero sufren deslizamientos notorios cuando realizan rotaciones puras sobre superficies inclinadas y por ello, estas maniobras deberían evitarse en dichas condiciones. Para contrarrestar este efecto de deslizamiento, en este trabajo se optó por aumentar la distancia de detección de bordes en los algoritmos de navegación del prototipo a medida que es aumentado el ángulo de inclinación. De esta forma, en inclinaciones mayores el prototipo se detiene mucho más lejos del borde, permitiendo un margen suficiente para realizar el giro aun considerando el deslizamiento que se produce durante la maniobra. Esta solución permite superar el problema de deslizamiento de forma práctica, pero es sacrificada la cobertura de limpieza del prototipo. El área efectiva de limpieza será reducida cerca de los bordes de los paneles solares. Esto es observable en los resultados de coloreado de lápiz a las diversas inclinaciones.

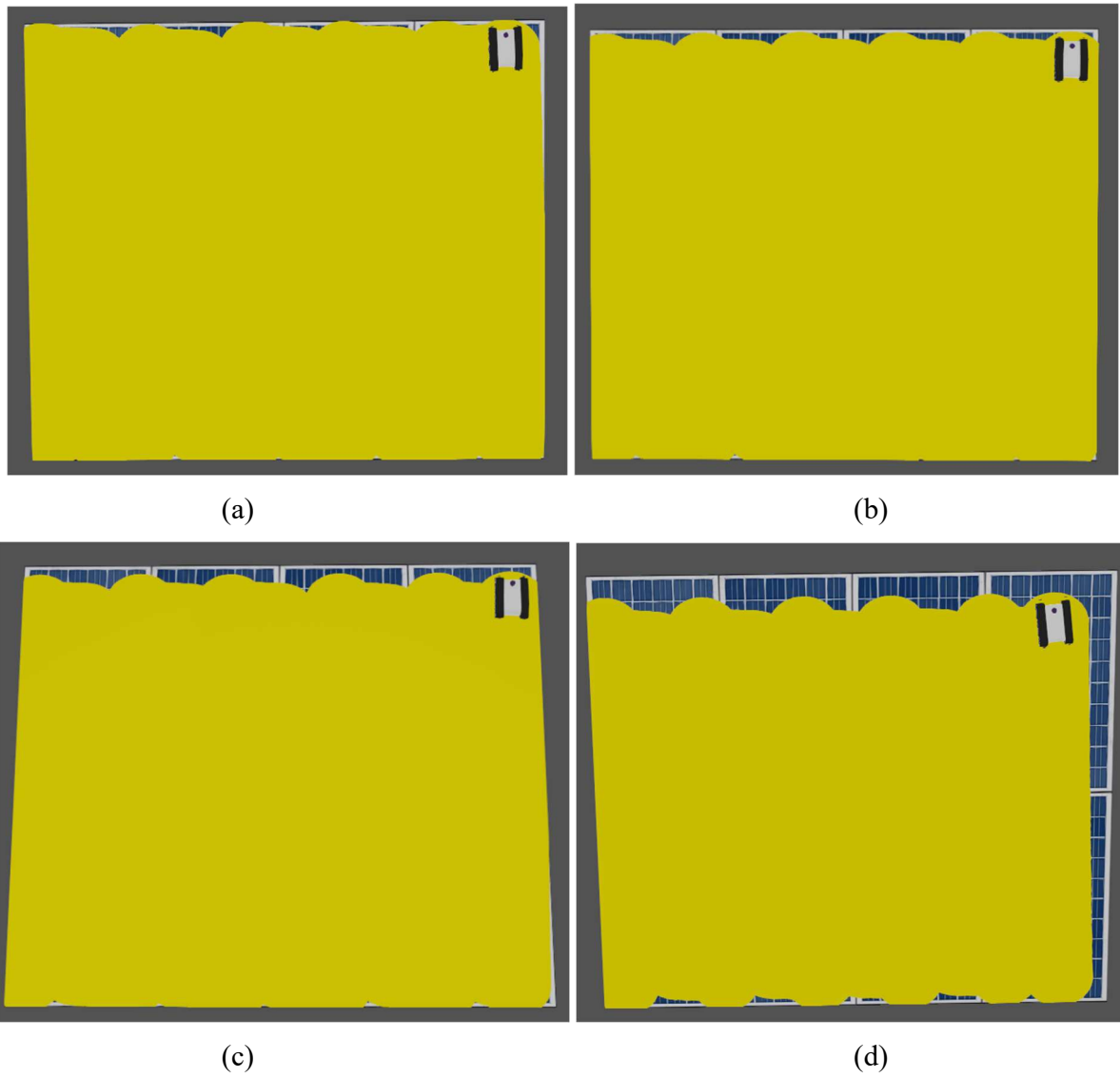


Figura 18. Cobertura de limpieza del prototipo en inclinaciones de 10° (a), 15° (b), 20° (c) y 30° (d)

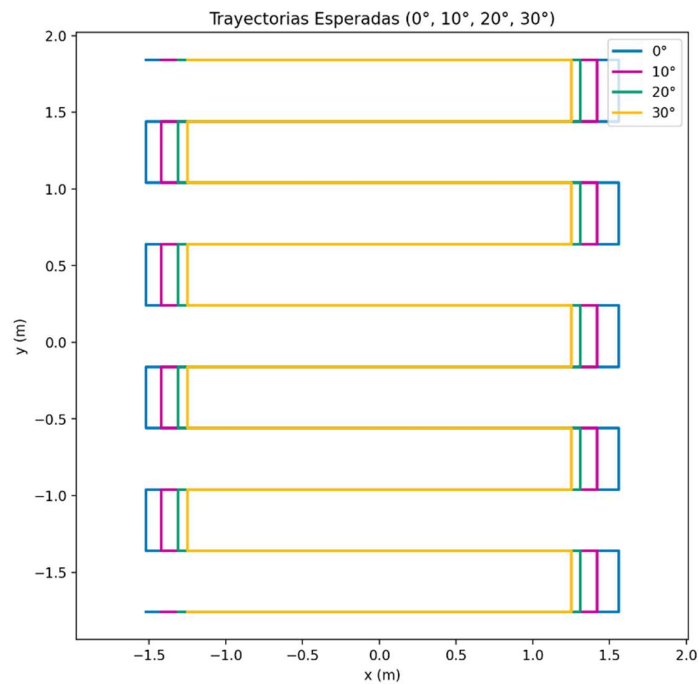


Figura 19. Modificación a trayectorias esperadas en relación con inclinación de los paneles solares

A continuación, se presentan los resultados condensados de cada escenario e inclinación simulados:

Tabla 9. Resultados en error de trayectoria real del prototipo en 10° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	299	0,17	0,11	0,13
2. INS	299	0,11	0,06	0,10
3. INS con ZUPT	300	0,11	0,05	0,10
4. Encoders con EKF	300	0,18	0,11	0,14

Tabla 10. Resultados en error de estimación de trayectoria por odometría del prototipo en 10° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	299	14,20	8,02	11,72

2. INS	299	12,32	7,96	9,40
3. INS con ZUPT	300	33115,77	22082,99	24677,84
4. Encoders con EKF	300	0,56	0,45	0,33

Tabla 11. Resultados en error de trayectoria real del prototipo en 20° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	272	0,15	0,10	0,11
2. INS	294	0,11	0,05	0,10
3. INS con ZUPT	291	0,12	0,05	0,10
4. Encoders con EKF	275	0,09	0,04	0,09

Tabla 12. Resultados en error de estimación de trayectoria por odometría del prototipo en 20° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	272	12,22	6,90	10,08
2. INS	294	38,59	24,45	29,85
3. INS con ZUPT	291	61221,47	40814,34	45631,76
4. Encoders con EKF	275	1,33	0,99	0,89

Tabla 13. Resultados en error de trayectoria real del prototipo en 30° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	206	0,11	0,05	0,10
2. INS	259	0,07	0,04	0,06

3. INS con ZUPT	282	0,13	0,06	0,11
4. Encoders con EKF	283	0,16	0,09	0,13

Tabla 14. Resultados en error de estimación de trayectoria por odometría del prototipo en 30° de inclinación

Escenario	Número de Muestras	RMSE [m]	Desviación Estándar SD	Error Absoluto Medio MAE
1. Encoders	206	8,88	5,08	7,28
2. INS	259	13,32	7,83	10,77
3. INS con ZUPT	282	83421,99	55575,17	62214,37
4. Encoders con EKF	283	2,41	1,54	1,86

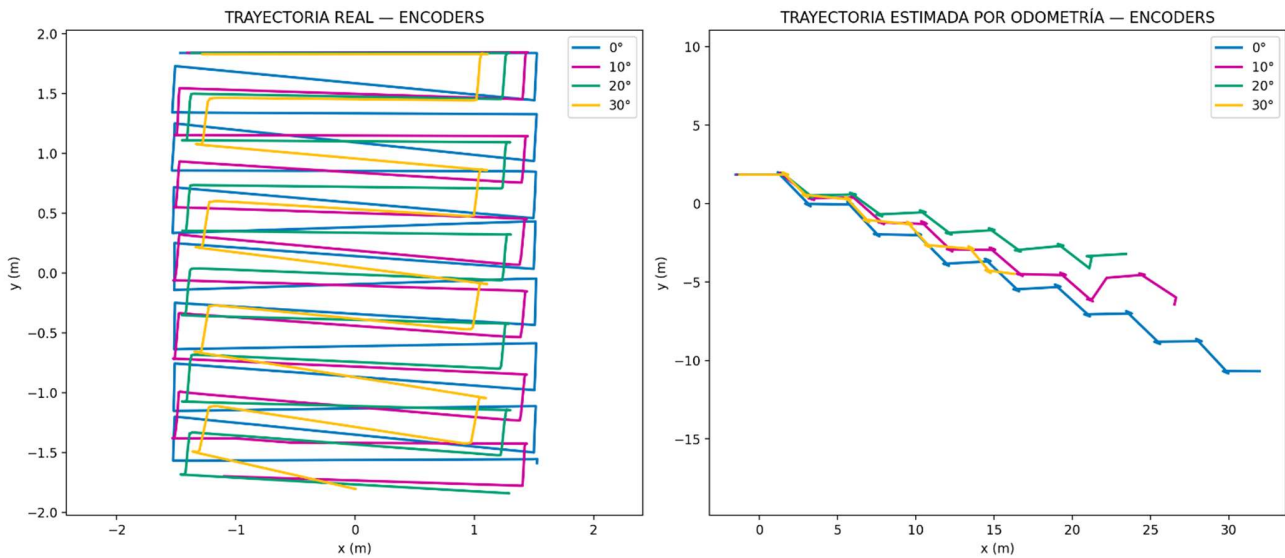


Figura 20. Trayectorias reales y estimadas por odometría para Escenario 1 a diversas inclinaciones

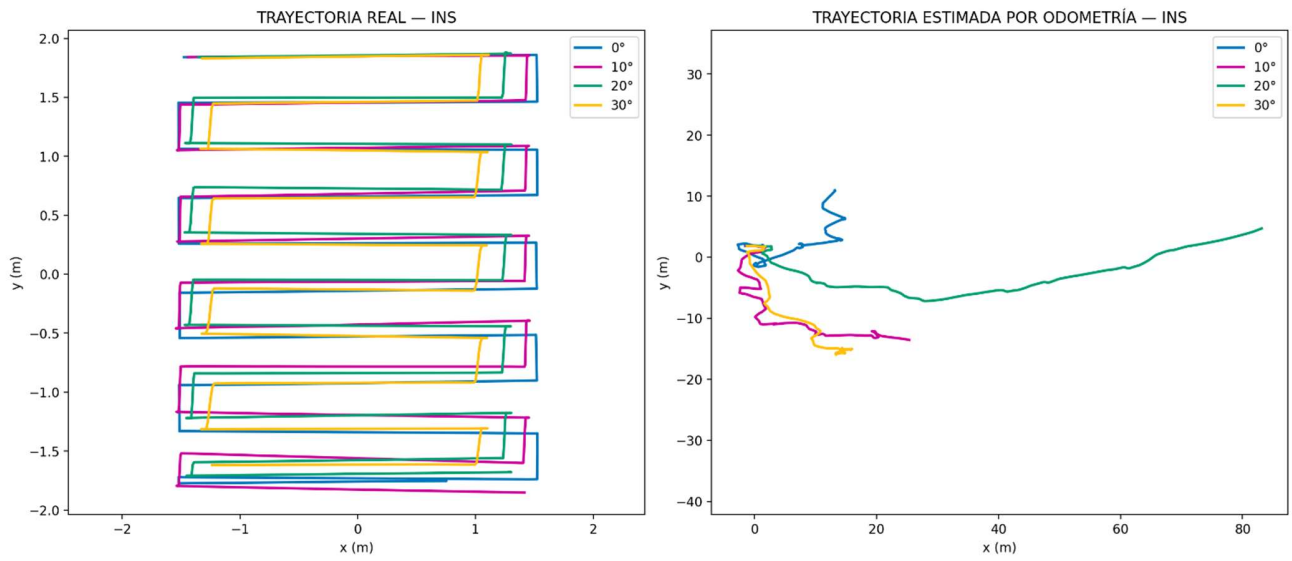


Figura 21. Trayectorias reales y estimadas por odometría para Escenario 2 a diversas inclinaciones

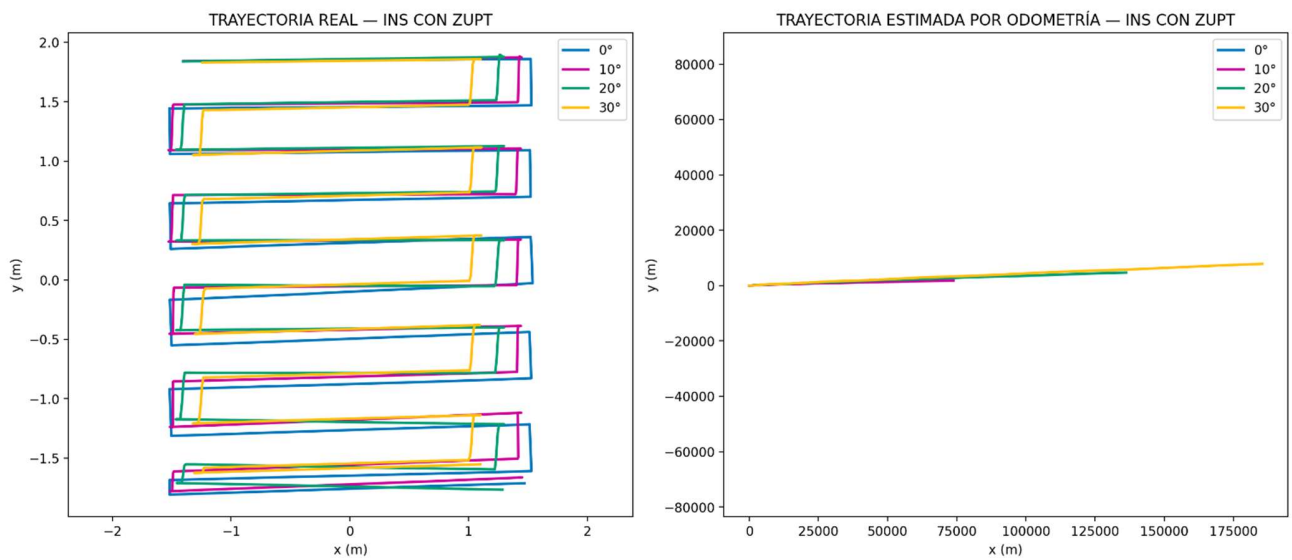


Figura 22. Trayectorias reales y estimadas por odometría para Escenario 3 a diversas inclinaciones

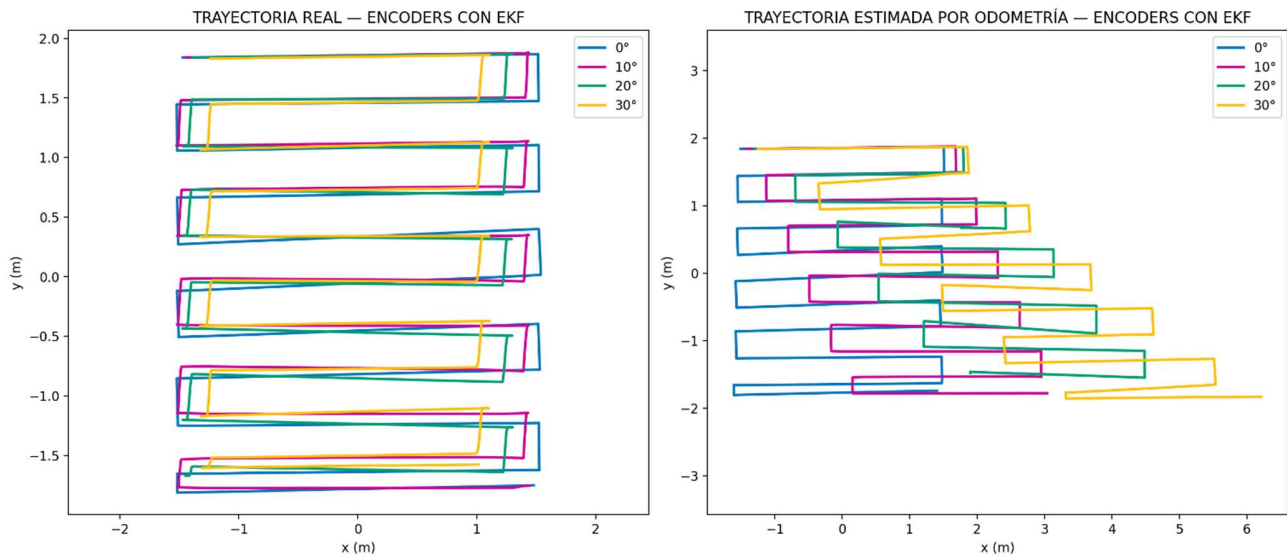


Figura 23. Trayectorias reales y estimadas por odometría para Escenario 4 a diversas inclinaciones

Tabla 15. Análisis comparativo de RMSE entre escenarios e inclinaciones

Inclinación [0°]	Escenario	RMSE en trayectoria real vs esperada	Cambio de RMSE en trayectoria real vs 0°	RMSE en odometría vs trayectoria real	Cambio de RMSE en odometría vs 0°	Mejor método (Menor RMSE en ambas mediciones)
0	Encoders	0,19	0	17,95	0	Encoders con EKF
	INS	0,04	0	9,97	0	
	INS con ZUPT	0,07	0	16,55	0	
	Encoders con EKF	0,09	0	0,05	0	
10	Encoders	0,17	-0,01	14,20	-3,76	Encoders con EKF
	INS	0,11	0,07	12,32	2,35	
	INS con ZUPT	0,11	0,04	33115,77	33099,22	
	Encoders con EKF	0,18	0,08	0,56	0,51	
20	Encoders	0,15	-0,03	12,22	-5,74	Encoders con EKF
	INS	0,11	0,06	38,59	28,62	
	INS con ZUPT	0,12	0,04	61221,47	61204,91	
	Encoders con EKF	0,09	0,002	1,33	1,28	
30	Encoders	0,11	-0,08	8,88	-9,08	Encoders con EKF
	INS	0,07	0,03	13,32	3,35	
	INS con ZUPT	0,13	0,05	83421,99	83405,43	
	Encoders con EKF	0,16	0,07	2,41	2,36	

Los resultados obtenidos mantienen la tendencia esperada: el escenario basado únicamente en encoders presenta los errores más altos en el seguimiento de la trayectoria real vs. la trayectoria esperada. Sin embargo, aparece un fenómeno que no había sido evidente en el análisis a 0° de inclinación: debido a que la inclinación del panel introduce deslizamientos durante los momentos de giro, el método INS con ZUPT se ve particularmente afectado. Bajo estas condiciones de inclinación, el robot nunca alcanza un estado de reposo verdadero y por tanto las actualizaciones ZUPT no pueden aplicarse correctamente por lo cual el sistema acumula errores cada vez mayores. Por otra parte, los errores de estimación de trayectoria por odometría también se incrementan de forma significativa con la inclinación. Los métodos más afectados son INS y ZUPT, donde el deslizamiento altera las mediciones inerciales y deteriora la integración.

Se observa también un patrón consistente: la fusión de sensores mediante EKF sigue ofreciendo el mejor comportamiento global al obtener los menores resultados de error tanto en el seguimiento de trayectoria esperada como en la estimación de odometría y menor variabilidad en todos los escenarios y grados de inclinación simulados. Aunque también se ve afectado por los deslizamientos ya que el modelo no logra capturar completamente los cambios de posición causados durante los deslizamientos en giros, se consolida como la opción más robusta frente a condiciones de pendiente

7 CONCLUSIONES Y RECOMENDACIONES

- El diseño de un robot de limpieza para paneles solares debe tener en cuenta las condiciones específicas de una instalación en fachadas urbanas. La simulación del desempeño físico y del comportamiento de navegación permitió anticipar problemas y oportunidades de mejora antes de pasar a etapas más costosas de fabricación o pruebas experimentales. Esto se traduce en un desarrollo más eficiente y en una optimización de recursos desde fases tempranas del desarrollo.
 - Las simulaciones realizadas en este estudio se llevaron a cabo bajo condiciones de fricción favorables para el desplazamiento del robot. Los coeficientes utilizados se basaron en valores reportados en la literatura para superficies limpias y secas, por lo que representan escenarios ideales. Será necesario profundizar este análisis en el futuro utilizando rangos más amplios de fricción dentro de Webots, de modo que el diseño final pueda responder adecuadamente a superficies con polvo, humedad o suciedad. Este trabajo se planteó como un punto de partida metodológico para permitir simulaciones más exigentes y realistas a futuro.
 - también será importante considerar a futuro cómo afectan al rendimiento del robot ciertos factores dinámicos que pueden presentarse durante la limpieza tales como: ráfagas de viento, objetos inesperados sobre los paneles, presencia de animales, entre otros. De igual forma, deberá evaluarse el efecto de variaciones en la masa y en el centro de gravedad del robot propios de un robot de limpieza, ya que estos parámetros pueden cambiar durante la operación e influir directamente en su velocidad, estabilidad, deslizamiento y capacidad de adherencia.
 - Los resultados de las simulaciones evidenciaron con claridad las limitaciones asociadas al sistema de tracción por orugas seleccionado. Aunque el robot logró un buen desempeño en desplazamientos rectilíneos, las maniobras de giro produjeron deslizamientos importantes, especialmente en superficies inclinadas. Ante este comportamiento, podría considerarse la adopción de configuraciones híbridas que combinen las ventajas de las orugas con elementos omnidireccionales o incluso mecanismos tipo pata que permitan reducir el deslizamiento durante los cambios de dirección y, con ello, asegurar la cobertura completa del área de limpieza.
- 8 Finalmente, los experimentos confirmaron que la estrategia de navegación más consistente fue la basada en la fusión de múltiples sensores mediante un filtro extendido de Kalman (EKF). Aunque los distintos escenarios simulados lograron completar la trayectoria de limpieza prevista, la combinación de encoders, IMU y filtro extendido de Kalman fue la que presentó una estimación de odometría más cercana al comportamiento real del prototipo. Para mejorar el desempeño en

inclinaciones mayores, sería conveniente incorporar nuevas fuentes de información al EKF, como detección de bordes mediante sensores ultrasónicos, mediciones LiDAR para mapear el entorno o sistemas de visión que fortalezcan la estimación de posición bajo distintas condiciones de operación.

9 REFERENCIAS

- [1] V. Masson, M. Bonhomme, J. L. Salagnac, X. Briottet, y A. Lemonsu, “Solar panels reduce both global warming and urban heat island”, *Front Environ Sci*, vol. 2, núm. JUN, jun. 2014, doi: 10.3389/fenvs.2014.00014.
- [2] A. Boccalatte, M. Thebault, C. Ménézo, J. Ramousse, y M. Fossa, “Evaluating the impact of urban morphology on rooftop solar radiation: A new city-scale approach based on Geneva GIS data”, *Energy Build*, vol. 260, abr. 2022, doi: 10.1016/j.enbuild.2022.111919.
- [3] “Technology Roadmap Solar Photovoltaic Energy - 2014 edition”, 2014. [En línea]. Disponible en: www.iea.org
- [4] U.N. Desa, “Percentage of Population at Mid-Year Residing in Urban Areas by Region, Subregion, Country and Area, 1950-2050”, World Urbanisation Prospects 2018. Consultado: el 27 de marzo de 2025. [En línea]. Disponible en: <https://population.un.org/wup/downloads>
- [5] N. Arcila Marín *et al.*, *Plan de Acción Climática Medellín 2020 - 2050*, 1a ed. Medellín, Colombia: Municipio de Medellín, 2021. Consultado: el 2 de marzo de 2025. [En línea]. Disponible en: https://www.medellin.gov.co/es/wp-content/uploads/2024/03/PAC_Medellin_Libro_Digital.pdf
- [6] M. J. Adinoyi y S. A. M. Said, “Effect of dust accumulation on the power outputs of solar photovoltaic modules”, *Renew Energy*, vol. 60, pp. 633–636, dic. 2013, doi: 10.1016/j.renene.2013.06.014.
- [7] N. Sarode, P. Ghugal, S. Yadav, S. Dantule, y P. Nandankar, “A comprehensive review on solar panel cleaning robot technologies”, en *AIP Conference Proceedings*, American Institute of Physics Inc., abr. 2023. doi: 10.1063/5.0127800.
- [8] M. Rudnicka y E. Klugmann-Radziemska, “CLEANING METHODS FOR DUST DEPOSITED ON THE FRONT COVER OF PHOTOVOLTAIC MODULE”, *Ecological Chemistry and Engineering S*, vol. 30, núm. 4, pp. 505–516, dic. 2023, doi: 10.2478/ECES-2023-0045.
- [9] M. G. Antonelli, P. Beomonte Zobel, A. De Marcellis, y E. Palange, “Autonomous robot for cleaning photovoltaic panels in desert zones”, *Mechatronics*, vol. 68, jun. 2020, doi: 10.1016/j.mechatronics.2020.102372.
- [10] P. J. Durst, D. McInnis, J. Davis, y C. T. Goodin, “A novel framework for verification and validation of simulations of autonomous robots”, *Simul Model Pract Theory*, vol. 117, may 2022, doi: 10.1016/j.simpat.2022.102515.
- [11] D. Deb y N. L. Brahmhatt, “Review of yield increase of solar panels through soiling prevention, and a proposed water-free automated cleaning solution”, *Renewable and Sustainable Energy Reviews*, vol. 82, Elsevier Ltd, pp. 3306–3313, 2018. doi: 10.1016/j.rser.2017.10.014.
- [12] H. A. Kazem, M. T. Chaichan, A. H. A. Al-Waeli, y K. Sopian, “A review of dust accumulation and cleaning methods for solar photovoltaic systems”, el 10 de diciembre de 2020, *Elsevier Ltd*. doi: 10.1016/j.jclepro.2020.123187.
- [13] A. Syafiq, A. K. Pandey, N. N. Adzman, y N. A. Rahim, “Advances in approaches and methods for self-cleaning of solar photovoltaic panels”, *Solar Energy*, vol. 162, Elsevier Ltd, pp. 597–619, 2018. doi: 10.1016/j.solener.2017.12.023.
- [14] H. Choi *et al.*, “On the use of simulation in robotics: Opportunities, challenges, and suggestions for moving forward”, vol. 118, núm. 1, 2020, doi: 10.1073/pnas.1907856118/-/DCSupplemental.
- [15] A. Afzal, D. S. Katz, C. Le Goues, y C. S. Timperley, “A Study on the Challenges of Using Robotics Simulators for Testing”, abr. 2020, [En línea]. Disponible en: <http://arxiv.org/abs/2004.07368>

- [16] C. Camargo, J. Gonçalves, M. Conde, F. J. Rodríguez-Sedano, P. Costa, y F. J. García-Peñalvo, “Systematic literature review of realistic simulators applied in educational robotics context”, el 2 de junio de 2021, *MDPI AG*. doi: 10.3390/s21124031.
- [17] N. K. Memon, “Autonomous vehicles for cleaning solar panels”, en *Proceedings of 2016 International Renewable and Sustainable Energy Conference, IRSEC 2016*, 2017, pp. 633–637. doi: 10.1109/IRSEC.2016.7983975.
- [18] F. Hajiahmadi, P. Zarafshan, M. Dehghani, S. A. A. Moosavian, y S. R. Hassan-Beygi, “Dynamics Modeling and Position Control of a Robotic Carrier for Solar Panel Cleaning System”, en *2019 7th International Conference on Robotics and Mechatronics (ICRoM)*, IEEE, 2019, pp. 613–618. doi: 10.1109/ICRoM48714.2019.9071821.
- [19] R. S. Bisht, P. M. Pathak, y S. K. Panigrahi, “Design and development of a glass façade cleaning robot”, *Mech Mach Theory*, vol. 168, feb. 2022, doi: 10.1016/j.mechmachtheory.2021.104585.
- [20] M. Bagheri, S. H. Delbari, M. Pakzadmanesh, y C. A. Kennedy, “City-integrated renewable energy design for low-carbon and climate-resilient communities”, *Appl Energy*, vol. 239, pp. 1212–1225, abr. 2019, doi: 10.1016/j.apenergy.2019.02.031.
- [21] J. Bazán, J. Rieradevall, X. Gabarrell, y I. Vázquez-Rowe, “Low-carbon electricity production through the implementation of photovoltaic panels in rooftops in urban environments: A case study for three cities in Peru”, *Science of the Total Environment*, vol. 622–623, pp. 1448–1462, may 2018, doi: 10.1016/j.scitotenv.2017.12.003.
- [22] D. S. Vijayan *et al.*, “Advancements in Solar Panel Technology in Civil Engineering for Revolutionizing Renewable Energy Solutions—A Review”, el 1 de septiembre de 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/en16186579.
- [23] J. Y. Han, Y. C. Chen, y S. Y. Li, “Utilising high-fidelity 3D building model for analysing the rooftop solar photovoltaic potential in urban areas”, *Solar Energy*, vol. 235, pp. 187–199, mar. 2022, doi: 10.1016/j.solener.2022.02.041.
- [24] P. Y. Najeeb, A. Aboshosha, y A. Haggag, “Intelligent Optimization of Photovoltaic Stations Performance Relying on Integrated Solar Techniques”, en *Proceedings of the International Middle East Power System Conference, MEPCON*, Institute of Electrical and Electronics Engineers Inc., 2024. doi: 10.1109/MEPCON63025.2024.10850229.
- [25] A. Al-Quraan *et al.*, “A New Configuration of Roof Photovoltaic System for Limited Area Applications—A Case Study in KSA”, *Buildings*, vol. 12, núm. 2, feb. 2022, doi: 10.3390/buildings12020092.
- [26] A. Barbón, M. Ghodbane, L. Bayón, y Z. Said, “A general algorithm for the optimization of photovoltaic modules layout on irregular rooftop shapes”, *J Clean Prod*, vol. 365, sep. 2022, doi: 10.1016/j.jclepro.2022.132774.
- [27] M. Z. Jacobson y V. Jadhav, “World estimates of PV optimal tilt angles and ratios of sunlight incident upon tilted and tracked PV panels relative to horizontal panels”, *Solar Energy*, vol. 169, pp. 55–66, jul. 2018, doi: 10.1016/j.solener.2018.04.030.
- [28] T. M. Yunus Khan *et al.*, “Optimum location and influence of tilt angle on performance of solar PV panels”, *J Therm Anal Calorim*, vol. 141, núm. 1, pp. 511–532, jul. 2020, doi: 10.1007/s10973-019-09089-5.
- [29] A. Barbón, C. Bayón-Cueli, L. Bayón, y C. Rodríguez-Suanzes, “Analysis of the tilt and azimuth angles of photovoltaic systems in non-ideal positions for urban applications”, *Appl Energy*, vol. 305, ene. 2022, doi: 10.1016/j.apenergy.2021.117802.
- [30] S. A. M. Said, G. Hassan, H. M. Walwil, y N. Al-Aqeeli, “The effect of environmental factors and dust accumulation on photovoltaic modules and dust-accumulation mitigation strategies”, 2018, *Elsevier Ltd*. doi: 10.1016/j.rser.2017.09.042.

- [31] M. R. Maghami, H. Hizam, C. Gomes, M. A. Radzi, M. I. Rezadad, y S. Hajighorbani, “Power loss due to soiling on solar panel: A review”, el 1 de junio de 2016, *Elsevier Ltd.* doi: 10.1016/j.rser.2016.01.044.
- [32] K. Ilse *et al.*, “Techno-Economic Assessment of Soiling Losses and Mitigation Strategies for Solar Power Generation”, el 16 de octubre de 2019, *Cell Press.* doi: 10.1016/j.joule.2019.08.019.
- [33] L. T.A. y T. I., “Environmental Factors and the Performance of PV Panels: An Experimental Investigation”, *African Journal of Environment and Natural Science Research*, vol. 6, núm. 3, pp. 231–247, dic. 2023, doi: 10.52589/ajensr-ga3smdhp.
- [34] E. Abdeen, M. Orabi, y E. S. Hasaneen, “Optimum tilt angle for photovoltaic system in desert environment”, *Solar Energy*, vol. 155, pp. 267–280, 2017, doi: 10.1016/j.solener.2017.06.031.
- [35] A. Ullah, A. Amin, T. Haider, M. Saleem, y N. Z. Butt, “Investigation of soiling effects, dust chemistry and optimum cleaning schedule for PV modules in Lahore, Pakistan”, *Renew Energy*, vol. 150, pp. 456–468, may 2020, doi: 10.1016/j.renene.2019.12.090.
- [36] L. Wan, L. Zhao, W. Xu, F. Guo, y X. Jiang, “Dust deposition on the photovoltaic panel: A comprehensive survey on mechanisms, effects, mathematical modeling, cleaning methods, and monitoring systems”, el 15 de enero de 2024, *Elsevier Ltd.* doi: 10.1016/j.solener.2023.112300.
- [37] C. K. Liu y D. Negrut, “The Role of Physics-Based Simulators in Robotics”, *Robotics, and Autonomous Systems Annu. Rev. Control Robot. Auton. Syst.*, vol. 22, pp. 35–58, 2025, doi: 10.1146/annurev-control-072220.
- [38] Open Robotics, “ROS - Robot Operating System”. Consultado: el 2 de noviembre de 2025. [En línea]. Disponible en: <https://www.ros.org/>
- [39] M. Quigley *et al.*, “ROS: an open-source Robot Operating System”, en *ICRA Workshop on Open Source Software*, nov. 2009.
- [40] Open Robotics, “ROS / Concepts”. Consultado: el 2 de noviembre de 2025. [En línea]. Disponible en: <https://wiki.ros.org/ROS/Concepts>
- [41] I. The MathWorks, “Introduction to Robot Operating System 2 (ROS 2)”. Consultado: el 2 de noviembre de 2025. [En línea]. Disponible en: <https://www.mathworks.com/help/ros/gs/robot-operating-system-ros2-basic-concepts.html>
- [42] O. Michel, “Cyberbotics Ltd. Webots TM : Professional Mobile Robot Simulation”, *International Journal of Advanced Robotic Systems*, vol. 1, pp. 39–42, 2004. [En línea]. Disponible en: <http://www.cyberbotics.com>
- [43] Cyberbotics, “Introduction to Webots”.
- [44] M. Farah, M. Alshehab, y K. Alshehri, “Design and Implementation of an Autonomous Mobility Algorithm for Photovoltaic Panel Surface Cleaning Robots Based on Webots”, en *2022 6th International Conference on Robotics and Automation Sciences, ICRAS 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 138–143. doi: 10.1109/ICRAS55217.2022.9842259.
- [45] O. Michel, D. Mansolino, y S. Pedrazzi, “Webots ROS2 Interface”. Consultado: el 2 de noviembre de 2025. [En línea]. Disponible en: https://github.com/cyberbotics/webots_ros2?tab=readme-ov-file
- [46] Open Robotics, “ROS 2 Documentation Jazzy: Webots”. Consultado: el 2 de noviembre de 2025. [En línea]. Disponible en: <https://docs.ros.org/en/jazzy/Tutorials/Advanced/Simulators/Webots/Setting-Up-Simulation-Webots-Basic.html>
- [47] D. Bank, “P2-37: A Novel Ultrasonic Sensing System for Autonomous Mobile Systems”, en *Proceedings of IEEE Sensors*, 2002.

- [48] S. Kim y H. Bin Kim, “High Resolution Mobile Robot Obstacle Detection Using Low Directivity Ultrasonic Sensor Ring”, en *Advanced Intelligent Computing Theories and Applications. With Aspects of Artificial Intelligence*, Berlin: Springer, 2010.
- [49] West Instruments de Mexico S.A, “MANUAL DE APLICACIÓN DE ENCODERS Una guía de referencia y tutoría sobre encoders para control de movimiento: Tipos, tecnologías, aplicaciones, e instalaciones”. Consultado: el 3 de noviembre de 2025. [En línea]. Disponible en: <https://www.lagos.udg.mx/sites/default/files/adjuntos/encoders.pdf>
- [50] M. Ben-Ari y F. Mondada, “Robotic Motion and Odometry”, en *Elements of Robotics*, Springer International Publishing, 2018, pp. 63–93. doi: 10.1007/978-3-319-62533-1_5.
- [51] *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE) : 30 April - 3 May 2017 in Windsor, ON, Canada*. IEEE, 2017.
- [52] G. G. Samatas y T. P. Pachidis, “Inertial Measurement Units (IMUs) in Mobile Robots over the Last Five Years: A Review”, el 1 de febrero de 2022, *MDPI*. doi: 10.3390/designs6010017.
- [53] P. Kajánek, “Testing of the Possibilities of Using IMUs with Different Types of Movements”, 2014.
- [54] J. Borenstein, H. R. Everett, L. Feng, y D. Wehe, “Mobile Robot Positioning: Sensors and Techniques”, 1997.
- [55] S. A. S. Mohamed, M. H. Haghbayan, T. Westerlund, J. Heikkonen, H. Tenhunen, y J. Plosila, “A Survey on Odometry for Autonomous Navigation Systems”, *IEEE Access*, vol. 7, pp. 97466–97486, 2019, doi: 10.1109/ACCESS.2019.2929133.
- [56] Fahmizal, A. Priyatmoko, E. Apriaskar, y A. Mayub, “Heading Control on Differential Drive Wheeled Mobile Robot with Odometry for Tracking Problem”, en *2019 International Conference on Advanced Mechatronics, Intelligent Manufacture and Industrial Automation (ICAMIMIA)*, IEEE, 2019. doi: 10.1109/ICAMIMIA47173.2019.9223412.
- [57] M. Park y Y. Gao, “Error and Performance Analysis of MEMS-based Inertial Sensors with a Low-cost GPS Receiver”, *Sensors*, vol. 8, pp. 2240–2261, 2008, [En línea]. Disponible en: www.mdpi.org/sensors
- [58] G. Klančar, A. Zdešar, S. Blažič, y I. Škrjanc, “Chapter 5 - Sensors Used in Mobile Systems”, en *Wheeled Mobile Robotics*, G. Klančar, A. Zdešar, S. Blažič, y I. Škrjanc, Eds., Butterworth-Heinemann, 2017, pp. 207–288. doi: <https://doi.org/10.1016/B978-0-12-804204-5.00005-6>.
- [59] D. Engelsman y I. Klein, “Information-Aided Inertial Navigation: A Review”, *IEEE Trans Instrum Meas*, vol. 72, pp. 1–18, 2023, doi: 10.1109/TIM.2023.3303496.
- [60] C.-Y. Liu, C.-A. Lin, K.-W. Chiang, S.-C. Huang, C.-C. Chang, y J.-M. Cai, “Performance Evaluation of Real-time MEMS INS/GPS Integration with ZUPT/ZIHR/NHC for Land Navigation”, en *2012 12th International Conference on ITS Telecommunications*, IEEE, 2012.
- [61] L. Xiaofang, M. Yuliang, X. Ling, C. Jiabin, y S. Chunlei, “Applications of Zero-velocity Detector and Kalman Filter in Zero Velocity Update for Inertial Navigation System”, en *2014 IEEE Chinese Guidance, Navigation and Control Conference*, IEEE, 2014. doi: 978-1-4799-4699-0/14/\$31.00©2014IEEE.
- [62] B. Sudantha, K. A. Sumathipala, C. Premachandra, K. Warnakulasooriya, C. Elvitigala, y Y. Jayasuriya, *Adaptive Navigation and Motion Planning for a Mobile Track Robot*. IEEE, 2017.
- [63] M. M. Almasri, A. M. Alajlan, y K. M. Elleithy, “Trajectory Planning and Collision Avoidance Algorithm for Mobile Robotics System”, *IEEE Sens J*, vol. 16, núm. 12, pp. 5021–5028, jun. 2016, doi: 10.1109/JSEN.2016.2553126.
- [64] L. Wijayathunga, A. Rassau, y D. Chai, “Challenges and Solutions for Autonomous Ground Robot Scene Understanding and Navigation in Unstructured Outdoor Environments: A Review”, el 1 de septiembre de 2023, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/app13179877.

- [65] D. Ghorpade, A. D. Thakare, y S. Doiphode, “Obstacle Detection and Avoidance Algorithm for Autonomous Mobile Robot using 2D LiDAR”, en *2017 International Conference on Computing, Communication, Control and Automation (ICCUBEA)*, IEEE, 2017. doi: 978-1-5386-4008-1/17/\$31.00.
- [66] Y. Abessolo Mindzie *et al.*, “Enhancing Solar Panels Efficiency: The Impact of Robotic Cleaning and Optimal Trajectory Tracking in the Presence of Disturbances Using Model Reference Adaptive Control”, *Jurnal Elektronika dan Telekomunikasi*, vol. 24, núm. 2, p. 137, dic. 2024, doi: 10.55981/jet.645.
- [67] T. Li *et al.*, “A Mobile Robot Design for Efficient and Large-Scale Solar Panel Cleaning”, en *2022 IEEE International Conference on Robotics and Biomimetics, ROBIO 2022*, Institute of Electrical and Electronics Engineers Inc., 2022, pp. 70–75. doi: 10.1109/ROBIO55434.2022.10011850.
- [68] Aravind G, Gautham Vasam, Gowtham Kumar T.S.B, y Naresh Balaji R, “A Control Strategy for an Autonomous Robotic Vacuum Cleaner for Solar Panels”, en *2014 Texas Instruments India Educators’ Conference*, IEEE, 2014.
- [69] F. Hajiahmadi, M. Jafari, y M. Reyhanoglu, “Machine Learning-Based Control of Autonomous Vehicles for Solar Panel Cleaning Systems in Agricultural Solar Farms”, *AgriEngineering*, vol. 6, núm. 2, pp. 1417–1435, jun. 2024, doi: 10.3390/agriengineering6020081.
- [70] A. Gheitasi, A. Almaliki, y N. Albaqawi, “Development of an automatic cleaning system for photovoltaic plants”, en *2015 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*, 2015. doi: 10.1109/APPEEC.2015.7380938.
- [71] I. Munasinghe, V. Vijenayake, S. Viduranga, Y. Lokugama, y P. Jayasekara, “Design and Implementation of a Semi-Autonomous Robotic System for Systematic Solar Panel Cleaning”, en *2024 9th International Conference on Control and Robotics Engineering, ICCRE 2024*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 61–67. doi: 10.1109/ICCRE61448.2024.10589856.
- [72] T. Sorndach, N. Pudchuen, y P. Srisungsitthisunti, “Rooftop Solar Panel Cleaning Robot Using Omni Wheels”, en *2018 2nd International Conference on Engineering Innovation (ICEI)*, 2018, pp. 7–12. doi: 10.1109/ICEI18.2018.8448530.
- [73] N. Ronnaronglit y N. Maneerat, “A Cleaning Robot for Solar Panels”, en *2019 5th International Conference on Engineering, Applied Sciences and Technology (ICEAST)*, IEEE, jul. 2019. doi: 10.1109/ICEAST.2019.8802521.
- [74] B. Sutam, D. Maneetham, P. N. Crisnapati, y W. Srichaipanya, “The Solar Panels Cleaning Robot Control via IoT”, en *2023 11th International Conference on Cyber and IT Service Management, CITSM 2023*, Institute of Electrical and Electronics Engineers Inc., 2023. doi: 10.1109/CITSM60085.2023.10455327.
- [75] G. Anilkumar *et al.*, “Design and development of wireless networking for solar PV panel cleaning robots”, en *IOP Conference Series: Materials Science and Engineering*, IOP Publishing Ltd, oct. 2020. doi: 10.1088/1757-899X/937/1/012024.
- [76] D. Fitriyanah, R. D. P. Saputra, I. Abadi, y A. Musyafa, “Optimal cleaning robot on solar panels with time-sequence input based on internet of things”, *International Journal of Electrical and Computer Engineering*, vol. 15, núm. 1, pp. 280–291, feb. 2025, doi: 10.11591/ijece.v15i1.pp280-291.
- [77] S. Fan, W. Liang, G. Wang, Y. Zhang, y S. Cao, “A novel water-free cleaning robot for dust removal from distributed photovoltaic (PV) in water-scarce areas”, *Solar Energy*, vol. 241, pp. 553–563, 2022, doi: 10.1016/j.solener.2022.06.024.
- [78] S. Mousavi y G. Farahani, “Introducing a new method of automatic cleaning of the PV array surface using a suction robot”, *Mechatronics*, vol. 85, ago. 2022, doi: 10.1016/j.mechatronics.2022.102845.

- [79] J. M. García, J. L. Martínez, A. Mandow, y A. García-Cerezo, “Slide-Down Prevention for Wheeled Mobile Robots on Slopes”, en *3rd International Conference on Mechatronics and Robotic Engineering*, Paris, 2017.
- [80] R. Bahri, R. Boucetta, y S. Bel Hadj Ali Naoui, “A novel frictional force control of a wheeled robot”, en *18th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, IEEE, dic. 2017.
- [81] Y. Nishimura y T. Yamaguchi, “Development of a steep slope mobile robot with propulsion adhesion”, en *IEEE International Conference on Intelligent Robots and Systems*, Institute of Electrical and Electronics Engineers Inc., oct. 2020, pp. 2592–2599. doi: 10.1109/IROS45743.2020.9341524.
- [82] D. R. Jones y K. A. Stol, “Modelling and Stability Control of Two-Wheeled Robots in Low-Traction Environments”.
- [83] D. Vu, M. Mikhail, y Q. Le Thu, “Study on the stability of a wheeled climbing robot”, en *Lecture Notes in Mechanical Engineering*, Springer Science and Business Media Deutschland GmbH, 2021, pp. 638–645. doi: 10.1007/978-3-030-69610-8_85.
- [84] M. T. Nguyen, C. T. Truong, V. T. Nguyen, V. T. Duong, H. H. Nguyen, y T. T. Nguyen, “Research on Adhesive Coefficient of Rubber Wheel Crawler on Wet Tilted Photovoltaic Panel”, *Applied Sciences (Switzerland)*, vol. 12, núm. 13, jul. 2022, doi: 10.3390/app12136605.
- [85] Cyberbotics, “Webots Reference Manual: ContactProperties”. Consultado: el 8 de noviembre de 2025. [En línea]. Disponible en: <https://www.cyberbotics.com/doc/reference/contactproperties?version=master>
- [86] F. L. Roth, R. L. Driscoll, y W. L. Holt, “FRICTIONAL PROPERTIES OF RUBBER”, *J Res Natl Bur Stand (1934)*, vol. 28, 1942.
- [87] M. Barquins y A. D. D. Roberts, “Rubber friction variation with rate and temperature: some new observations”, *J. Phys. D: Appl. Phys*, vol. 19, pp. 547–563, 1986.
- [88] A. Ishizako, M. Tomosada, K. Hokkirigawa, y T. Yamaguchi, “Impact of rubber block end-face corner radius on the friction coefficient between rubber blocks and a glass plate under dry and lubrication conditions”, *Tribol Int*, vol. 174, oct. 2022, doi: 10.1016/j.triboint.2022.107705.
- [89] N. Ezra, A. Cohen, y D. Zarrouk, “Modeling, simulation, and experiments of a flexible track robot over rigid horizontal and inclined surfaces”, *Mech Mach Theory*, vol. 199, sep. 2024, doi: 10.1016/j.mechmachtheory.2024.105689.