

Universidad EAFIT
Escuela de Ciencias Aplicadas e Ingeniería

Robots autónomos en el manejo de bodegas

Modelación y Simulación 3

Autores:
Juan Tomás Macías
Santiago Beltrán
Laura González

Mayo 2025

1. Introducción

El uso de las tecnología de automatización, especialmente de aquellas que manejan bodegas para la recepción y entrega de producto que tienen como destino su comercialización ha aumentado significativamente, esto producto del incremento en el comercio digital y las tendencias producidas por las redes sociales a partir del reciente golpe por el COVID-19 (Martinez, L.G, 2022).

Como en cualquier industria, los administradores de bodegas tienen interés en adoptar tecnologías para automatizar como lo son las tecnologías de identificación de objetos por ondas de radio (RFID). El crecimiento del mercado en tecnologías en el manejo de materiales, incluido el sistema de transportadora, ha llegado a 201 miles de millones de dólares en el último año, un incremento del 34 % desde el 2019 (Knowledge Sourcing Intelligence, 2025).

2. Descripción del problema

Este trabajo fue elaborado a partir del artículo *"Autonomous mobile robot travel under deadlock and collision prevention algorithms by agent-based modelling in warehouses"*, de Turhanlar et al. (2022). Este artículo habla acerca de la implementación de un algoritmo de movilidad flexible para el movimiento de un conjunto de robots en una bodega con corredores y transiciones. Este artículo contiene especificaciones acerca del algoritmo de decisión de tareas y de movimiento flexible para evitar bloqueos y choques al ejecutar una tarea.

Turhanlar et al. (2022) concluyen con diversas preguntas y oportunidades de optimización y variedad acerca de las dinámicas que puede tener el comportamiento de los robots en el modelo que ellos proponen. Una de las cuales causó interés consiste en la implementación de un sistema que permita intercambiar tareas durante la ejecución de ellas en el modelo. Este comportamiento puede tener impacto en algunas situaciones específicas en el ambiente de la bodega, pero esta será discutido más adelante en el informe.

El modelo inherente al artículo se realizó a través del uso de agentes, y para su implementación se propuso hacerlo también así. El motivo del uso de modelación basada en agentes radica en poder diseñar el comportamiento propio de cada robot, el mapa de la bodega y el agente informativo, el cual ayuda a los robots a informarse acerca de sus tareas y lo que estos deben hacer; se ve beneficiado principalmente la autonomía del robot al momento de decidir sus tareas, o la ruta que va a tomar (Turhanlar et al., 2022).

3. Literatura del problema

Turnhalar et al. (2022) proponen un sistema flexible de movimiento de robots autónomos a partir de puntos de decisión, espera y escape; esto con el fin de evitar choques o y optimizar el movimiento de estos. Este se compara con un sistema rígido, el cual ocupa máximo un robot por corredor, de tal manera que un robot nunca interactúe con el otro. Se demostró una mejora hasta del 39 % en métricas importantes para el modelo, como el tiempo promedio de espera por tarea, el tiempo máximo por tarea o la proporción entre el tiempo de espera y el tiempo de tarea. Estas métricas ayudaron a verificar que un sistema flexible, incluso bajo un cambio de parámetros como la cantidad de robots o de fajas transportadoras, resulta más eficiente que un sistema rígido para la ejecución de tareas de transporte y de extracción de cargas.

Russell y Hung (2024) contextualizan sobre el uso de los ARM en las bodegas, informando que los AGV (Automated Guided Vehicle), si bien son útiles y fueron bastante utilizados en el pasado como herramientas de transporte gracias a su capacidad de realizar tareas repetitivas, han venido siendo desplazados por los ARM; eso gracias a las características de los robots autónomos, los cuales tienen la capacidad de tomar decisiones por sí mismos, y navegar por la bodega sin necesidad de una infraestructura preinstalada, esto gracias a sus avances tanto en hardware como en software, mostrando así su gran flexibilidad operativa. Sin embargo, estas tecnologías también tienen algunas limitaciones, como su capacidad máxima de carga, la velocidad máxima y sus ciclos de recarga. En el presente modelo no se tendrá en cuenta los tiempos, ni lugares de carga de los robots, tampoco su desgaste a través del tiempo.

4. Modelo Conceptual

Objetivo: Replicar el trabajo realizado por Turhanlar et al, sobre el uso de algoritmos que eviten choques y bloqueos en robots autónomos en bodegas, optimizando el tiempo de transacciones en bodega. Con esto, se puede comparar el rendimiento del nuevo modelo, el cuál tiene adicionado el algoritmo A* para la asignación de rutas con el propósito de escoger el camino más corto para los AMR, con las métricas de desempeño del modelo original. Para hacer la comparación, se hace uso del tiempo máximo de una transacción, el tiempo promedio de una transacción y el ratio entre el tiempo de espera y el tiempo total; además, esto es comparado con distintas dimensiones de bodega, diferente número de AMR y diferentes velocidades.

Agentes: Al momento de implementar el modelo, se contó con un único agente, que son los AMR (Autonomous Mobile Robot), los cuales son agentes inteligentes que interactúan con el entorno en tiempo real, y toman decisiones de manera autónoma basada en sus metas.

Entre sus atributos está la posición, su transacción, su ruta a un objetivo, y la velocidad, que está medida en *metros/segundo*. Sin embargo, este cuenta con dos estados dependiendo de su circunstancia. Uno de sus estados se presenta cuando a este se le asigna una tarea, donde su atributo llamado "has-payload?" se torna verdadero. Un estado representando el movimiento de este es si está causando un bloqueo o no. En caso de estarlo causando, su estado se llama "DAMR" (Deadlock AMR); y en caso de que no, su estado es llamado "AAMR" (Active AMR).

El paper de Turhanlar et al., cuenta con un agente adicional llamado el agente informativo; sin embargo, este no fue implementado en nuestro modelo ya que al usar NetLogo como lenguaje, no se vió necesario.

Entorno: El entorno de modelo se basa en una bodega flexible, que contiene pasillos verticales y corredores de transición horizontales, en donde los robots pueden cambiar de pasillo. Cada pasillo cuenta con estanterías a la izquierda y a la derecha, donde se ponen o se retiran productos por los robots. Los objetos de esta bodega son las estanterías, y los

búferes donde se entregan y reciben productos. A continuación, un ejemplo del entorno en donde A es el número de pasillos; N representa el número de ARM; B es la cantidad de áreas de almacenamiento por sección; y C es el número total de la capacidad de la bodega. La dimensiones del espacio físico no están especificadas en el modelo.

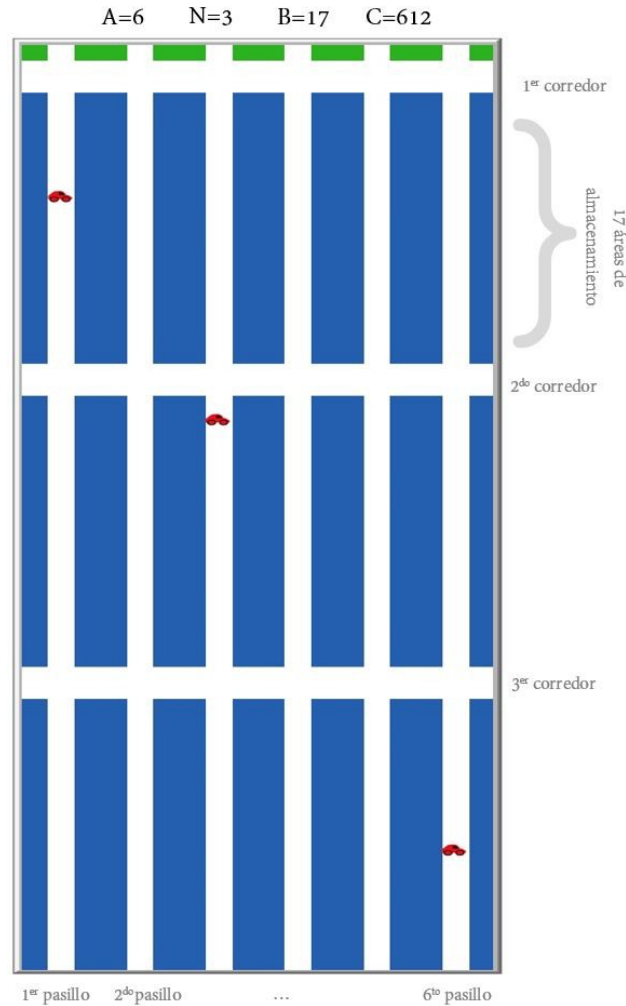


Figura 1: Bodega Flexible

Note además que en la parte superior del entorno, hay fajas transportadoras y zonas para recoger los productos, es decir, ese es el lugar donde se recogen y entregan pedidos.

Los agentes del modelo se comportan de la siguiente manera: Al tener solamente un tipo de agente, estos interactúan debido al algoritmo de asignación de tareas, que hace que los robots se muevan a través del mapa.

Además, al moverse desde los búferes hasta el punto de la tarea, los robots interactúan usando un algoritmo que les permite evitar un punto de bloqueo en la bodega. Este algoritmo está diseñado aprovechando las propiedades de las transiciones del mapa, para poder apartarse si pasa un robot en la dirección opuesta a este, o detenerse en caso de que un robot tenga que girar al frente. Para completar las tareas, es necesario que los robots pasen por la estantería en la cuál se debe dejar esta; o en su defecto, pasar por la ubicación del búfer establecido por la tarea. Luego de hacer esto, la tarea es eliminada y el robot se desocupa.

La emergencia de este modelo se da por el movimiento de estos robots, donde el cambio en el tiempo promedio de las tareas cambia de manera emergente; las posiciones en las que están distribuidas las tareas son asignadas de manera aleatoria; de igual forma que el tipo de tarea (llevar elementos al almacén, o recoger elementos del almacén).

5. Implementación

Este modelo fue implementado en Netlogo. Debido a que Turhanlar et al. (2022) no explica de manera explícita muchas de las decisiones del modelo para llevarlo a cabo, se tomaron algunas decisiones para lograr el mismo comportamiento al del paper. Al hacer el modelo, hubo varios cambios en comparación al modelo original dado que el lenguaje de Netlogo permitió descartar el uso del agente informativo originalmente hecho. Como aclaración importante, para nuestro modelo cada tick representa un segundo.

Para implementar el enrutamiento de los robots, se tomó la decisión de usar el algoritmo A*, un algoritmo que ayuda a buscar el camino más corto entre un punto de partida y llegada en un grafo ponderado. Este algoritmo combinado con la distancia Manhattan, que ayuda a calcular la distancia entre un parche A y B en un espacio limitado por estanterías (DataCamp, 2024). Con estas dos herramientas, se hizo un algoritmo para calcular la ruta que debe cumplir un robot para llegar hasta su objetivo.

Los tipos de tareas que pueden ser asignadas a los robots son de almacenar (storage), y de retirar de las estanterías (retrieve). Las tareas de almacenamiento tienen como parche inicial

un búfer aleatorio, y como parche final un punto aleatorio en las estanterías. Las tareas de retiro tienen como parche inicial una de las estanterías, y como parche final uno de los búferes aleatorios. Las tareas se asignan por prioridad, atendiendo primero las que tienen un tiempo de espera mayor al promedio, y luego las que se encuentran en pasillos menos congestionados; todas las asignaciones se hacen teniendo en cuenta el ARM más cercano a la tarea. Uno de los supuestos del modelo, es que cada transacción llega con distribución Poisson, el valor del promedio de llegadas puede ser modificado a la hora de implementar el modelo.

Para hacer la política de activación, usada para evitar bloqueos en el modelo, se usan los diagramas de flujo presentados por Turhanlar et al. (2022). Ellos presentan dos diagramas de flujo, uno de activación y otro de respuesta. El algoritmo de activación filtra los casos donde se debe hacer que el robot espere, evada o siga adelante. Para esto, hay que tener en cuenta que, en el modelo están establecidos algunos puntos por cada intersección entre pasillo y corredor, estos son punto de espera, decisión y escape. El funcionamiento de estos puntos es el siguiente, los robots se dirigen hacia el decision point más cercano, allí se le es asignado su tarea; cuando el robot calcula la ruta, y ve que hay otro AMR que está en su camino, y no hay otra ruta igual o más cercana para lograr su objetivo, entonces el AMR le pide al robot que le está bloqueando el camino que se mueva. Si el robot que está bloqueando no está ocupado, entonces este se moverá a un scape point o a un decision point, dependiendo de cuál esté más cercano, y el AMR ocupado se situará en el waiting point mientras le liberan el camino; si el robot sí está ocupado, se le da prioridad al AMR que esté saliendo de un pasillo. De esta manera, los agentes buscarán recorrer la menor distancia, sin chocar con otros robots; esto se hace gracias a que pueden interactuar con el entorno, y con otros agentes, recalculando en cada tick si la ruta más cercana está bloqueada por otro agente.

En la siguiente figura, se presenta una idea general de cómo funciona el modelo a grandes rasgos, sin hacer énfasis en la política para evitar bloqueos, ni el enrutamiento hacia el lugar de destino.

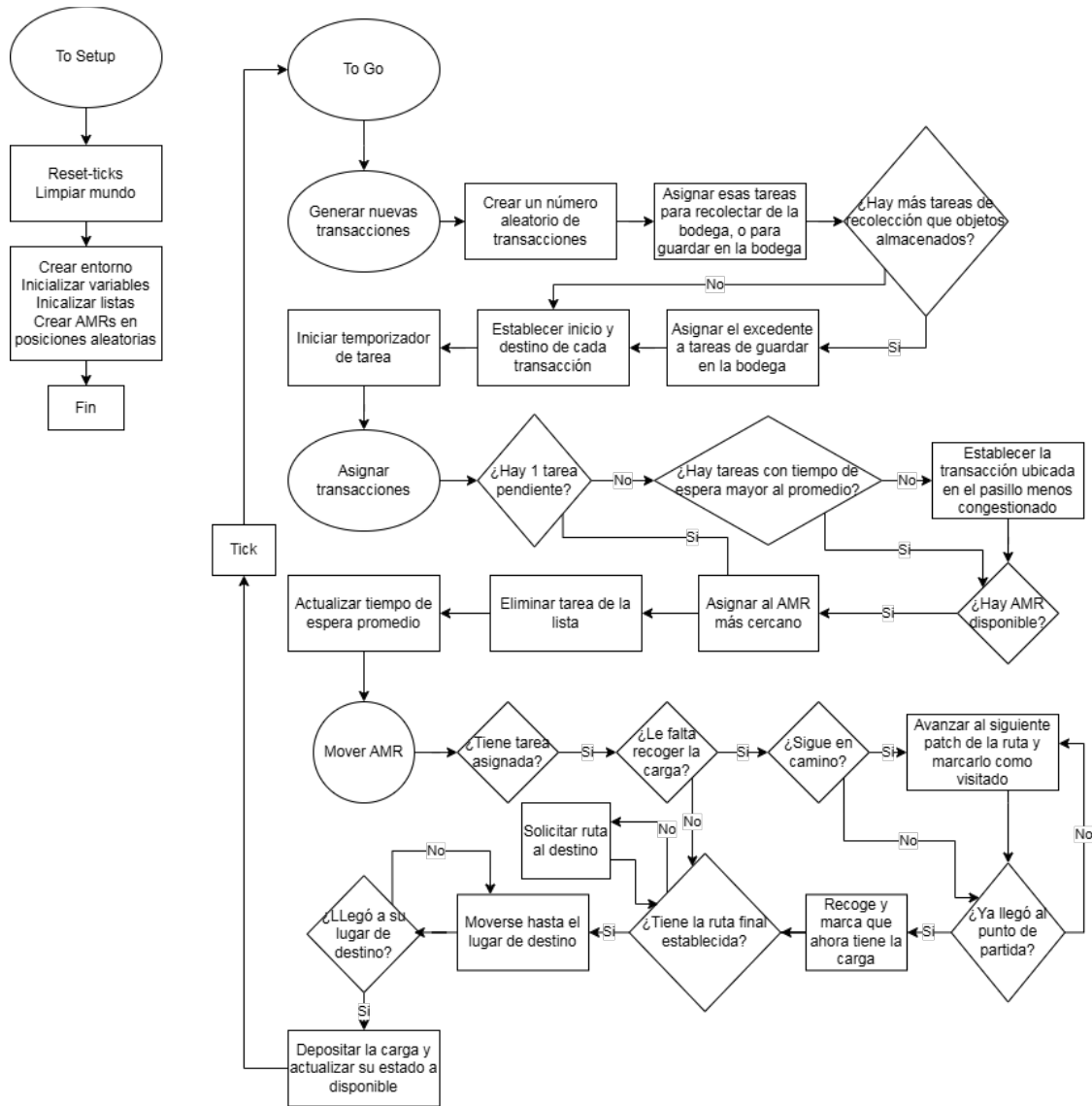


Figura 2: Diagrama de flujo

6. Análisis y Experimentación

Los datos de entrada usados para el modelo se basaron en el trabajo de Turhanlar et al. Datos como la cantidad de robots en bodega, el número de pasillos y corredores, y la capacidad del almacén. El dato de salida principal del modelo serían el tiempo promedio de transacción, el cual fue examinado según dos variables que se consideraron relevantes para el funcionamiento del modelo. Estas variables fueron Lambda (λ) como la taza promedio de

llegada de transacciones, y el numero de AMRs. Se ejerció un análisis de sensibilidad local de estas dos variables con respecto al tiempo promedio por ejecución de una transacción.

Análisis de Sensibilidad respecto a λ

Se ejecutaron 5 simulaciones por valor de λ con pasos de 0.1, desde 0.1 hasta 2. La respuesta del modelo respecto a estos valores fue muy variada. Especialmente incrementando el tiempo de transacciones para $\lambda = 1$. A continuación se muestra una gráfica de datos agrupados.

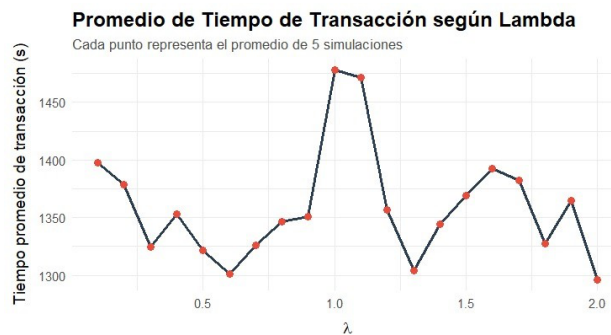


Figura 3: λ vs Tiempo

Análisis de Sensibilidad respecto al número de AMRs.

Se ejecutaron de nuevo 5 simulaciones por número de AMRs, variando desde 5 hasta 10. La respuesta del modelo fue decreciente. Esto es intuitivo, ya que al tener más AMRs, la asignación se puede hacer a los robots que estén más cerca de las tareas, por lo que se gasta menos tiempo en el desplazamiento inicial. A continuación se muestra una gráfica de datos agrupados.

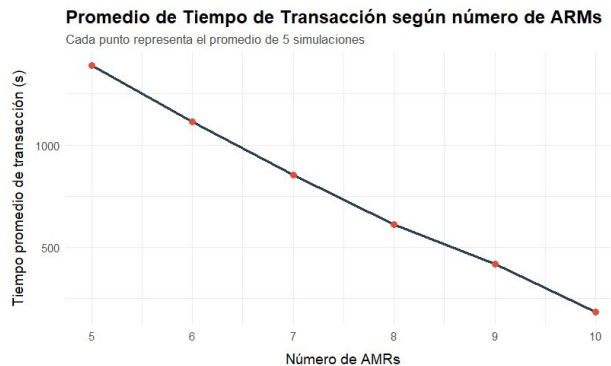


Figura 4: AMRs vs Tiempo

Análisis de Estocasticidad

Se realizó un análisis de estocasticidad del tiempo promedio de transacciones. Para esto, se ejecutó el modelo inicialmente 10 veces con los $\lambda = 5$, 5 AMRs, capacidad máxima de 1200 y 6 corredores. A partir de estas 10 ejecuciones, se halló que el número de ejecuciones óptimas para determinar las medidas de esta variable deb'ia ser m'ínimo 32.

Al ejecutar 32 veces el modelo, se halló que la media de tiempo de ejecuciones era de 1376.8 ticks, que equivale a 1376.8 segundos. Esta media está en un intervalo de incertidumbre de [1345,43 – 1408,17]. Además, la desviación estándar del tiempo es de 87 segundos. Aquí se muestra un histograma de la distribucion de probabilidad a partir de las ejecuciones.

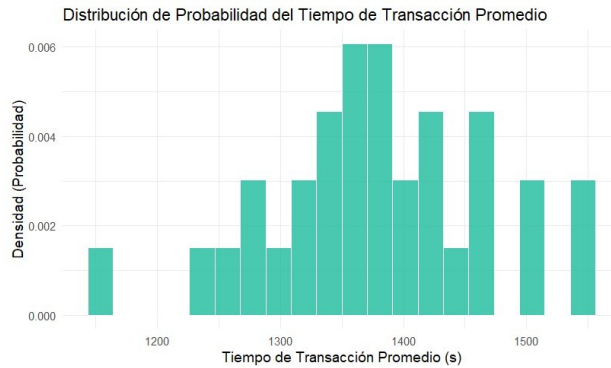


Figura 5: Enter Caption

Escenarios de comparación

Se desarrollaron dos escenarios de comparación del modelo, donde se ve si una mayor inversión en robots en el modelo minimiza el tiempo por transacción. Los parámetro usados en ambos escenarios son:

1. **Ejecución normal:** AMRs: 5, $\lambda = 2$, 6 corredores y 1200 de capacidad.
2. **Mayor inversión:** AMRs: 10, $\lambda = 2$, 6 corredores y 1200 de capacidad.

Se ejecutó el modelo 32 veces y se compararon resultados en el promedio de tiempos de transacción.

El promedio encontrado en el caso de ejecución normal es de unos 4270 segundos, con desviación estándar de 22 segundos y un intervalo de confianza de [4262,6 – 4278,5].

Mientras que en el caso de mayor inversión se redujo el tiempo promedio, ya que este es de 3870 segundos, con desviación estándar de 25.1 segundos y un intervalo de confianza de [3860,9 – 3879]. Por ende, se puede concluir que, efectivamente, bajo un escenario de mayor inversión, se reduce el tiempo promedio por transacción, específicamente en un 9.36 %.

7. Conclusiones

El modelo logra simular interacciones complejas entre robots, resultando en comportamientos emergentes con el tiempo promedio de tareas. La incompletitud de información en el modelo base dificulta encontrar similitudes en el enrutamiento, y resulta en el cumplimiento de transacciones; sin embargo, se modeló todo el entorno y algoritmo para evitar bloqueos y colisiones lo mejor posible bajo el lenguaje de NetLogo.

El modelo se comporta de manera esperada al variar parámetros visualmente tangibles como el número de ARMs, y de manera menos intuitiva bajo el cambio de tasa de llegada de transacciones.

El modelo tiene ciertas limitaciones, como la interacción entre robots y tareas, dejando como posibilidad de mejora o de exploración con dinámicas de intercambio de tareas o de toma de decisiones de último momento.

El modelo además tiene supuestos que lo hacen menos realista como:

- Los robots no son defectuosos, no se dañan ni se descargan.
- Las cargas toman el mismo tiempo en retirarse de los búferes y estanterías (3 segundos).
- Los carriles de las estanterías no son bidireccionales.

Puede que el modelo tenga mejores desempeños bajo distintos diseños de bodega; sin embargo, esto queda como un interrogante abierto a otra investigación.

Por último, este modelo está hecho para ilustrar y explorar el comportamiento flexible, no para tomar decisiones reales debido a las limitaciones del modelo, que harán que los resultados no sean iguales. Sin embargo, al ser este modelo escalable, tiene un valor para poder tener una visión aproximada de como se puede comportar una bodega bajo una variedad de parámetros.

Referencias

- [1] Martinez, L.G. (2022). Diseño de investigación para la implementación de un sistema de automatización para el manejo de almacenes en una empresa distribuidora de electrodomésticos, audio, video y tecnología. Universidad de San Carlos de Guatemala. V´inculo
- [2] Turhanlar, E. E., Ekren, B. Y., & Lerher, T. (2022). Autonomous mobile robot travel under deadlock and collision prevention algorithms by agent-based modelling in warehouses. *International Journal of Logistics Research and Applications*, 27(8), 1322–1341. V´inculo
- [3] Knowledge Sourcing Intelligence. (2025). Material Handling Market Size, Share, Opportunities, And Trends By Product (Automated Guided Vehicles, Automated Storage And Retrieval Systems, Trucks And Lifts, Cranes & Hoists, Conveyor System, Others), By Application (Warehousing And Distribution, Airport Baggage Handling, Bulk Handling, Postal And Parcel Delivery, Assembly, Packaging), By Industry Vertical (Manufacturing, Healthcare, Chemical, Paper, Food And Beverage, Warehousing, Others) And By Geography - Forecasts From 2025 To 2030. V´inculo
- [4] Hsueh. C. (2010). A simulation study of a bi-directional load-exchangeable automated guided vehicle system, *Computers & Industrial Engineering*. V´inculo
- [5] Kumar. R. (2024). El algoritmo A*: Guia completa. V´inculo
- [6] Russell, K., & Hung M. L. (2024). Review of Autonomous Mobile Robots for the Warehouse Environment. *Advanced Robotics and Automation (ARA) Lab*, University of Nevada, Reno.