Mariano Antonio Regalado Iglesias
Student ID: REG23556163

Smart Delivery Robot

Overview

The “Smart Delivery Robot” project, is an AI-driven autonomous system designed to navigate an urban grid, efficiently deliver packages, and optimize delivery routes using the Manhattan distance algorithm. The system also generates random obstacles, such as no-entry zones and one-way streets, which the robot must intelligently avoid.

You can find anything that you would like to now on my GitHub repository: https://github.com/marianor03/Smart-Delivery-Robot

Users can customize the environment by:

- Selecting the grid size.
- Activating or deactivating obstacles.
- Choosing a starting position for the robot.

The program dynamically displays a real-time grid update, showing the robot's movements and deliveries.

How the Program Works

1. User Inputs

The user is first prompted to enter:

1. Grid size (1 to 6) – defines the working area.
2. Obstacle activation (yes/no) – determines whether obstacles will be placed.
3. Starting position – sets the robot's initial location.

2. Grid Generation

- The grid is created based on the selected size.
- Delivery points are randomly placed within the grid.
- If obstacles are enabled, random obstacles are placed in different positions.
- The robot is placed at the user-defined starting position.

3. Autonomous Navigation & Delivery

- The robot identifies the nearest delivery point using the Manhattan distance formula.
- The robot moves step by step in the most efficient direction while avoiding obstacles.
- If an obstacle is encountered, the robot finds an alternative path.

- Upon reaching a delivery point, the robot marks the parcel as delivered and continues to the next delivery.
- The process repeats until all deliveries are completed.

4. Grid Visualization

- The grid updates dynamically to reflect the robot's movement.
- Special markers:
  - *R* – Robot's position
  - *D* – Delivery points
  - *X* – Obstacles
  - **.** – Empty space

5. Completion

- Once all deliveries are completed, a completion message is displayed.

Technical Features

1. Manhattan Distance Algorithm

The robot selects the nearest delivery point using the formula:

$$\mathbf{d = |x_2 + x_1| + |y_2 - y_1|}$$

where:

- is the robot's current position.
- is the delivery point.
- The absolute differences in the x and y coordinates give the shortest path in a grid-based system.

2. Dynamic Obstacle Avoidance

- If an obstacle is detected in the shortest path, the robot calculates an alternative path.
- The robot reroutes dynamically to ensure all deliveries are made efficiently.

3. Real-Time Display & Updates

- The system updates the grid in real-time, making movements visible step by step.
- The robot's actions are printed (e.g., "Moved right", "Delivered at (2,3)").

Reflections & Improvements

Challenges Faced

- Ensuring real-time movement updates in the grid while avoiding lag.
- Implementing dynamic rerouting when obstacles block the shortest path.

Potential Improvements

- Implementing user-defined obstacles instead of random generation.
- Allowing multiple delivery robots to work simultaneously.
- Introducing battery management (robot must return to a charging station when low on power).

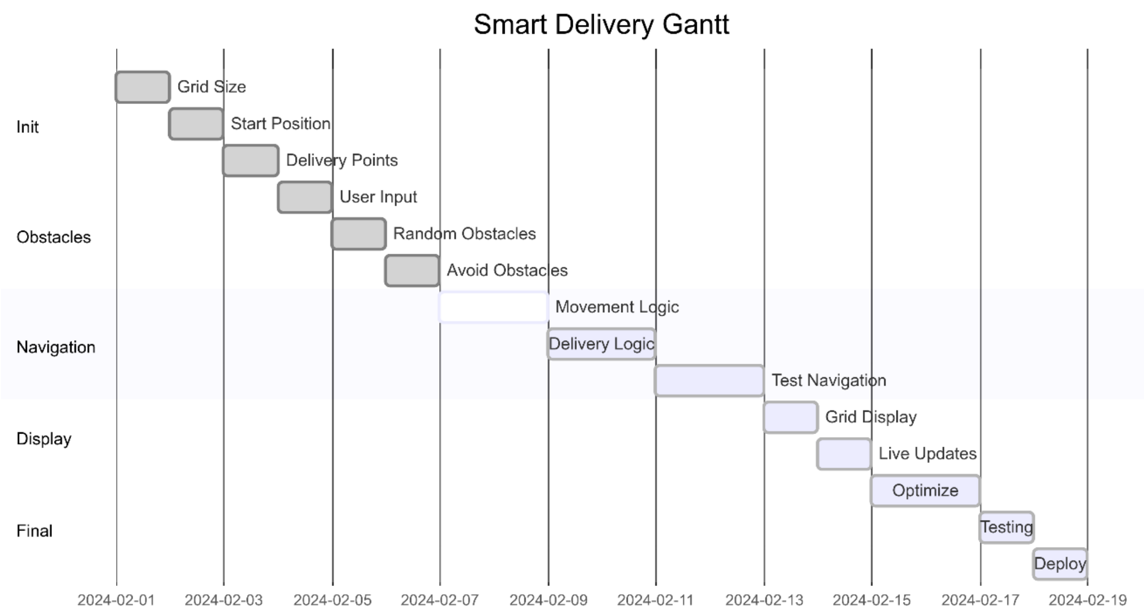
Conclusion

The “Smart Delivery Robot” efficiently navigates an urban environment, delivering parcels while dynamically avoiding obstacles. Using Manhattan distance for route optimization, the system ensures minimal travel time and efficient delivery operations. The interactive visualization enhances user understanding of how the AI makes decisions in real-time.

Mariano Antonio Regalado Iglesias
Student ID: REG23556163

Gallery

Gantt Chart:



Mariano Antonio Regalado Iglesias
Student ID: REG23556163

Flow Chart: