Mathematical Modeling and Simulation of an Autonomous Warehouse Robot System

Devi Krishna R, Gowri G, Arundhathy Devi L, Abhinand S, Adhi Lekshmi P Binu
Department of Robotic And Automation

Abstract

In this study, an Autonomous Warehouse Robot System is mathematically modeled and simulated to achieve efficient and collision-free navigation within a structured warehouse environment. The system integrates path planning, motion control, and kinematic modeling to enable autonomous movement between predefined pick-up and delivery points. The A* algorithm is employed for optimal path generation by minimizing travel distance, while a PID controller ensures smooth and stable trajectory tracking along the planned route. Transformation matrices are utilized to relate wheel angular velocities to the robot's linear and angular motion. Python-based simulations demonstrate accurate path generation, stable control response, and effective 3D visualization of robot navigation. The results validate the effectiveness of mathematical modeling and control strategies for developing reliable and intelligent robotic systems in warehouse automation.

1 Introduction

Automation has become an essential element in modern industrial systems, particularly in warehouse management, where efficiency, accuracy, and operational safety are critical for handling large-scale logistics operations. Traditional manual methods of material handling are often time-consuming, labor-intensive, and prone to human error. To overcome these challenges, the use of autonomous mobile robots (A.W.R.S) has emerged as a key solution for achieving intelligent and efficient warehouse automation. This study addresses the problem of autonomous navigation and path optimization for a warehouse robot operating in a structured environment with fixed obstacles. The primary objective is to design and simulate a robot system that can autonomously move between pick-up and delivery points while minimizing travel distance and avoiding collisions. The system integrates mathematical modeling, path planning, and control algorithms to ensure efficient and stable navigation. To accomplish this, the A* algorithm is utilized for optimal path planning, providing a computationally efficient method for determining the shortest collision-free route between predefined points in the warehouse layout. A Proportional-Integral-Derivative (PID) controller is implemented to regulate the robot's motion along the planned trajectory, ensuring smooth turns, precise velocity control, and minimal position error. Additionally, kinematic transformations are applied to relate wheel angular velocities to the robot's linear and angular motion, enabling accurate control of robot dynamics. The overall system is modeled and simulated in Python using computational tools such as NumPy, Matplotlib, and VPython for visualization. The simulation environment replicates a warehouse grid where the robot autonomously navigates between target points, validating the performance of both the path planning and control systems. Through this work, the study demonstrates how mathematical modeling, control theory, and simulation techniques can be integrated to design an efficient autonomous robot suitable for real-world warehouse automation applications.

Figure 1: Conceptual image of the A.W.R.S (created with Gemini)

2 Methodology

The project is divided into five major modules, each representing a mathematical concept that contributes to the development of an autonomous warehouse robot system. The implementation is done using Python and simulated in a virtual 3D environment.

2.1 Module 1: Environmental Setup And Mapping

The warehouse environment, including obstacles, shelves, and pathways, is represented as a 3D matrix. Each cell in the matrix corresponds to a specific position in the warehouse. Obstacles are marked with values of 1 and free space as 0 .

Mathematical Concept: Matrix representation, 3D coordinate system.

Warehouse Matrix W(x, y, z) is defined by **Piecewise Equation**

$$W(x, y, z) = \begin{cases} 1, & \text{if cell (x, y, z) contains an obstacle} \\ 0, & \text{if cell (x, y, z) is free space} \end{cases}$$

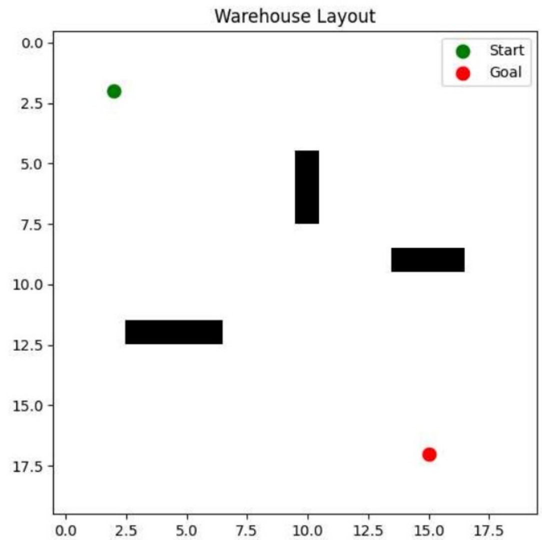

Figure 2: 2D visualization of the warehouse grid with start (green), goal (red), and obstacles (black).

2.2 Module 2: Path Planning using A^* Algorithm

Optimal path planning ensures that the robot moves from the starting position to the target with minimal travel distance while avoiding obstacles. Gradient descent and graph-based algorithms, A* are used to find the shortest path.

Mathematical Concept: Vector calculus, Optimization, Euclidean distance.

The Euclidean Distance heuristic is given by:

$$\mathrm{h}(n) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

where

- $x1, y1$: coordinates of the current node
- $x2, y2$: coordinates of the goal node
- h(n) : Straight line (Euclidean) distance from n to the goal

The cost function used by the algorithm is:

$$f(n) = g(n) + h(n)$$

where

- $g(n)$: actual cost from start node to the current node
- $h(n)$ is the heuristic estimate to the goal (using Euclidean or Manhattan distance).

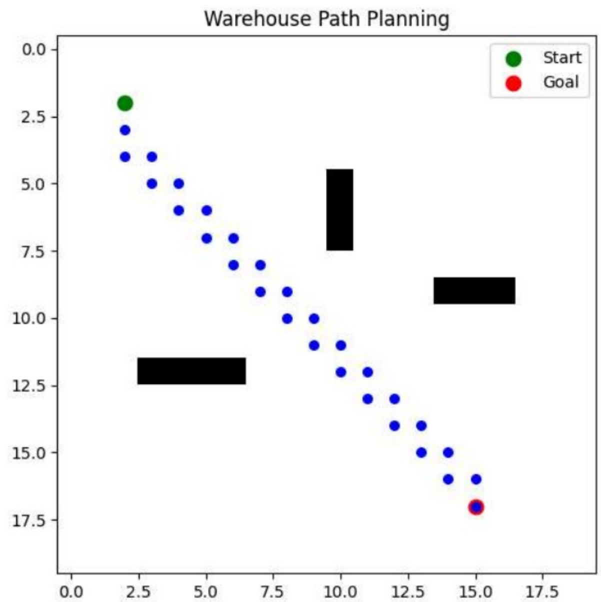

Figure 3: The optimal path (blue dots) computed by the A^* algorithm, navigating around all obstacles.

2.3 Module 3: Motion Control Using PID

This module focuses on developing a PID-based motion control system to achieve precise and stable movement of the autonomous warehouse robot. The PID (Proportional–Integral–Derivative) controller is implemented to regulate the wheel speed and heading of the robot according to the desired trajectory or target position.

Mathematical Concept: Calculus (differentiation and integration) is used in PID control to compute proportional, integral, and derivative terms for motion correction. .

Equation:

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt}$$

where

- $e(t)$: position error between target and actual coordinates.
- Kp : Proportional Grain (Response to present error)
- Ki : Integral Grain (Eliminates steady state error)
- Kd : Derivative gains (Anticipates future error)
- $u(t)$: The control output.

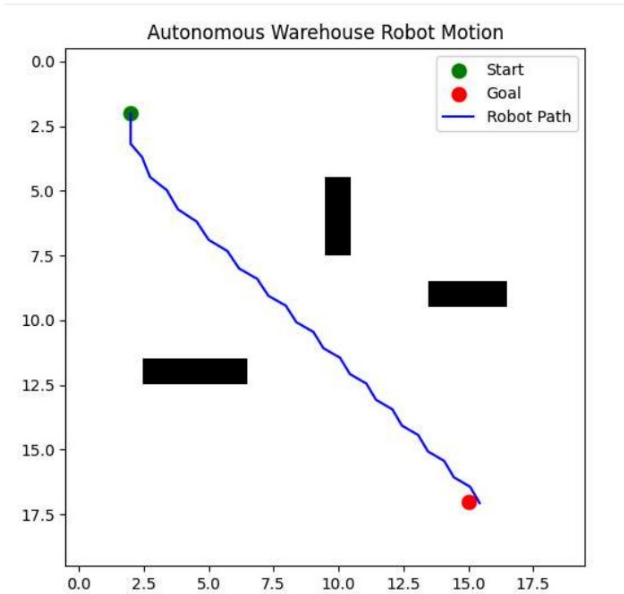

Figure 4: The smoothed trajectory of the robot (blue line) as it follows the path generated by the A^* algorithm, controlled by the PID system.

2.4 Module 4: Kinematic Transformation And Simulation

To represent and control the robot's position and orientation in space using mathematical transformations. Linear algebra, specifically 3x3 homogeneous transformation matrices, is used to calculate the robot's translation and rotation

The orientation angle θ is determined by the direction of movement along the path.

$$T = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & x \\ \sin(\theta) & \cos(\theta) & y \\ 0 & 0 & 1 \end{bmatrix}$$

The robot's pose was successfully tracked throughout its trajectory. At various points along the path, orientation vectors were calculated and visualized, correctly showing the robot's heading as it navigated towards the goal .

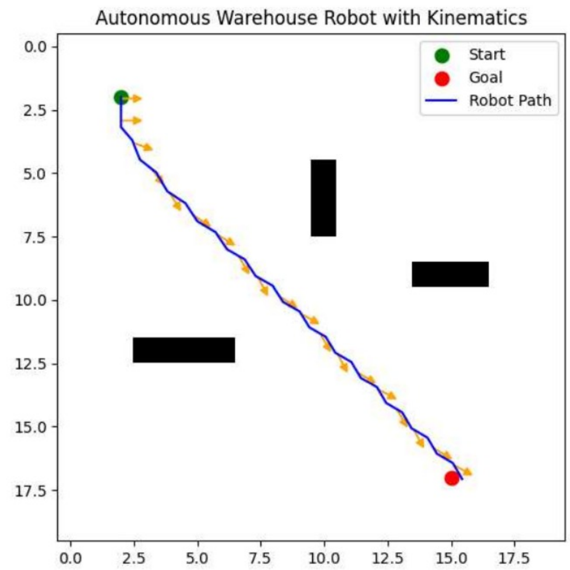

Figure 5: The robot's path with orientation vectors (orange arrows) shown at intervals, indicating its heading as it moves.

2.5 Module 5: Result Visualization And Analysis

To extend the simulation into a more realistic 3D environment, where the robot must navigate between different floors or levels of the warehouse. The concepts from the previous modules were extended to three dimensions. The environment became a 3D NumPy array, the A* algorithm searched a 3D grid, and robot kinematics were handled using 4x4 homogeneous transformation matrices .

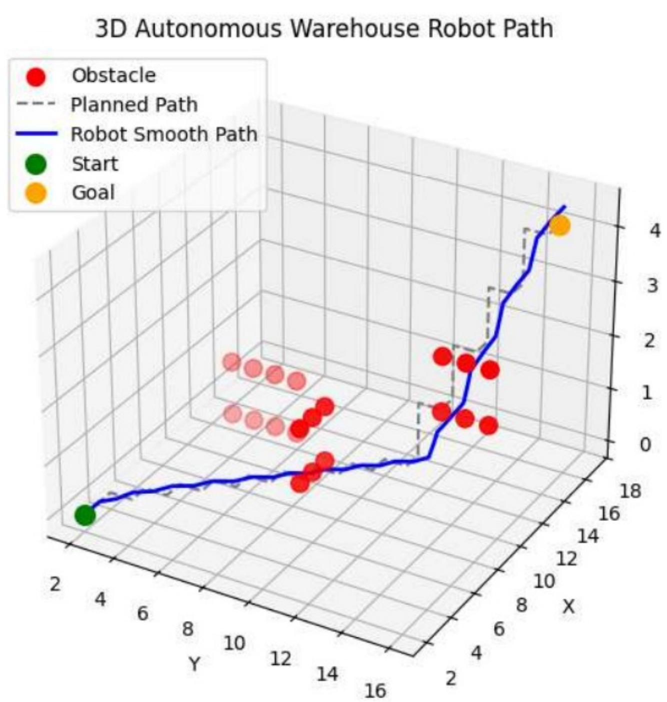

Figure 6: Final 3D simulation showing the robot's smooth path with orientation vectors (green arrows) indicating its heading in 3D space.

3 Conclusion

This project successfully demonstrates the integration of mathematical modeling and automation in robotics. The developed system efficiently combines theoretical concepts such as differential equations, optimization, and control algorithms with practical robotic applications. The study highlights how mathematical analysis supports decision-making, accuracy, and motion control in autonomous systems. Overall, this work strengthens the link between robotics and applied mathematics, providing a foundation for future research in intelligent and adaptive robotic control systems.

4 Limitations

- Real-time response may experience slight delays due to signal processing and computational limits.
- The project was tested only under controlled lab conditions, not in complex or outdoor environments.
- Environmental factors such as lighting, surface texture, and obstacles can affect system accuracy.
- Hardware limitations, including motor torque and battery capacity, may restrict long-duration operation.
- The mathematical model assumes ideal conditions and does not account for external disturbances or noise.
- Integration with advanced techniques like machine learning or adaptive control remains for future improvement.

5 Acknowledgment

I wish to express my deepest gratitude to my esteemed Mathematics Professor, Siju K. S, for his invaluable mentorship throughout this undertaking. His profound insights into Differential Equations, Optimization, Control System, Matrix Algebra, Applied Mathematics were foundational, particularly in the effective implementation of the path planning and the PID motion control systems. My sincere appreciation extends to my dedicated teammates. Their collective expertise in Linear Algebra and Kinematic Modeling, tireless work ethic, and commitment to collaboration were instrumental in navigating the complex integration of motion control and 3D environment mapping. Finally, I formally acknowledge the strategic use of Generative AI tools. These resources provided significant assistance in code generation for the 3D coordinate system and the debugging of complex Laplace Transform equations, thereby enhancing both the technical quality and the efficiency of the project execution.

6 Data and Code for the Project

All simulation codes and data files are available in the following GitHub repository: https://github.com/Minnusrd/Project

7 Reference

1. Saeed B Niku, **Introduction to Robotics Analysis, Systems, Applications**, 3rd Edition, 2005.
2. Richard C. Dorf and Robert H. Bishop, **Modern Control Systems**, 13th Edition, Pearson Education,2017.
3. Sebastian Thrun, Wolfram Burgard, and Dieter Fox, **Probabilistic Robotics**, MIT Press, 2005.
4. Jazar, R. N., **Theory of Applied Robotics: Kinematics, Dynamics, and Control**, Springer, 2010.
5. Erwin Kreyszig, **Advanced Engineering Mathematics**, Wiley , 10th Edition, (2016)