Wasit Journal of Computer and Mathematics Science

Journal Homepage: https://wjcm.uowasit.edu.iq/index.php/WJCM
e-ISSN: 2788-5879 p-ISSN: 2788-5887

Optimized Path Planning and Scheduling in Robotic Mobile Fulfillment Systems Using Ant Colony Optimization and Streamlit Visualization

Isam Sadeq Rasham[1,*]

[1]Third Rusafa Education Directorate, Baghdad, Baghdad, 10001, IRAQ

*Corresponding Author: Isam Sadeq Rasham

DOI: https://doi.org/10.31185/wjcms.330
Received 15 November 2024; Accepted 20 December 2024; Available online 30 December 2024

ABSTRACT: Context: In the age of rapid e-commerce growth; the Robotic Mobile Fulfillment Systems (RMFS) have become the major trend in warehouse automation. These systems involve the use of self- governed mobile chares to collect shelves as well as orders for deliveries with regard to optimization of task allocation and with reduced expenses. However, in a manner to implement such systems, one needs to find enhanced algorithms pertaining to resource mapping and the planning of movement of robots in sensitive environments. **Problem Statement:** Despite RMFS have certain challenges especially when it comes to the distribution of tasks and the overall distances that employees have to cover. **Objective:** The main goal of this paper is to propose a new compound optimization model based on RL-ACO to optimize the RMFS's task assignment and navigation. Also, the direction of the study is to investigate how such methods can be applied to real-life warehouse automation and how effective such methods can be on a large scale. **Methodology:** This research introduces a new optimization model for RMFS selection which integrates reinforcement learning with Ant Colony Optimization (ACO). Specifically, a real gym environment was created to perform the order assignment and training in the way of robotic movement. Reinforcement Learning (RL) models were trained with Proximal Policy Optimization (PPO) for improving the dynamic control of robots and ACO was used for computing optimal shelf trajectories. The performance was also measured by policy gradient loss, travelled distance and time taken to complete the tasks. **Results:** The proposed framework showed potential in enhancing the efficiency of tasks required and the travel distances involved. In each of the RL models used the shortest paths were identified and the best route was determined to have a total distance of 102.91 units. Also, other values such as, value function loss and policy gradient loss showed learning and convergence in iterations. To build a global solution, ACO integration went a step forward in enabling route optimization through effective combinatorial problems solving. **Implications:** This research offers a practical, generalizable and flexible approach for the improvement of the operations of RMFS and thinking for warehouse automation.

Keywords: Multi-Robot Systems, Autonomous Navigation, Path Planning, Adaptive Control, Mobile Robot Coordination, Energy Efficiency.

1. INTRODUCTION

The trend in e-commerce and on-demand service increased the requirements for fast and flexible warehouse systems that can operate independently. Applying unmanned technical means such as autonomous mobile robots for performing work functions, including order picking and logistics management, is known as Robotic Mobile Fulfillment Systems (RMFS) which has grown as a revolutionary solution in this area. Due to successful robot path planning and role assignment, RMFS improves warehouse performance, decreases the human factor's impact, and complies with the trends of modern supply chain management [1]. Consider a distribution center where hundreds of robots operate to deliver orders, walk around the shelves, and respond to various situations without interference and slowness [2]–[4]. This vision of automation is becoming a reality, yet to reach this goal several difficult problems like real-time path planning, resources allocation, and avoidance of collisions are solved [5], [6]. The efficient planning of pathing is one of the key focus areas within RMFS. The practical use of the robots, in terms of the ability to move around the environment and minimize inter-robot distances while avoiding obstacles, is an essential prerequisite of the effectiveness of this type of

*Corresponding author: essamsadik2@gmail.com

system. Many conventional approaches, including Dijkstra's and A* algorithms, provide limited solutions for addressing the current challenges of modern warehouses [7]–[10].

Ant Colony Optimization (ACO) is an algorithm for solving optimization solution problems such as route planning which makes it suitable for use in the RMFS to optimize robot navigation [11]. The efficiency in task scheduling together with the ability to work in transient environments as well as large problem sizes makes it possible to minimize on the travel costs. ACO with iteration pheromone strategy allow perturbation at every step, while reinforcement learning solves macro level route optimization problem. The integration of both does this leads to improved productivity, less energy usage and sound performance in the warehouses' work [12].

Although there is a rich body of the works such as ACO, it has not been easy to incorporate them in to RMFS systems and real time environments. Developing a robust RMFS system involves addressing several key challenges: Robots should properly respond to dynamic changes in the infrastructure of the warehouse including and obstacles. In addition, there are more robots as well as the orders given, the system has to remain performant and efficient. In addition, optimal of the job assignments such as selection of order picking zone, shelf assignment and robot dispatching is also paramount to prevent traffic jams [13]–[15].

In order to address these challenges this paper introduces an RMFS architecture that incorporates ACO for dynamic path planning. ACO is most efficient in complex, uncertain environments because it can learn to construct optimal paths by gradually, one by one, test paths that correspond to pheromones. The proposed ACO algorithm, when integrated with an interactive visualization system using Streamlit, envisions a complete solution for the real-time warehouse management system. This paper contributions can be summarized as follows:

1. We replace Dijkstra's algorithm with ACO to deal with scalability and adaptable changes to robot routes.
2. An online dashboard develops using Streamlit allows the operator to view the robot paths, the order status as well as the warehouse performance in a real-time mode.
3. Compared with the traditional approach, the system integrates the path planning, order assignment, and robot scheduling, showing a high efficiency in the dynamic and dense environment.

This work has supplemented the gap in the connection of theoretical research and realistic application by proposing a flexible and easily-implementable RMFS that can well address the current requirements of the warehouse.

2. RELATED WORK

The present decade has seen a tremendous activity in the field of mobile robot designs for industrial, autonomous, and supporting applications [16], [17]. These developments concern several aspects such as cooperative systems of multiple robots, learning control approaches, dead reckoning to low-cost robotic platforms, and movement in challenging terrains. Previous works have addressed ways of enhancing the locomotion of mobile wheeled robots, developing of intelligent decision-making system, as well as effective navigation strategies within complex and dynamic environments. As mentioned in various works, the focus has been laid on the utilization of multi-robot transportation, the adaptation of movement, and the path planning. These systems try to enhance performance in flexibility, reliability, and terrain adaptability, which practices behavior tree, machine learning, inertial measurement systems (IMU), and risk map for navigation and controls [18]–[20]. Prior works have greatly assisted in the development of autonomous systems and robots; however, the integration of omnidirectional wheels to terrain adaptable and reconfigurable platforms in industrial material transport is still a problem. This poses additional challenges for behaviour in nonholonomic scenarios, necessitating new solutions for coordination and cooperation among robots, safety of robots in dynamically uncertain environments and real-time reconfigurability [16], [21]. Moreover, current systems fail to address the low-cost and ease control strategies required in industrial environment where robots have to be flexible and easy to control in unpredictable world. The problem of path planning for mobile robots is still opened, especially in cases with dynamic charging stations and with battery constraints. Omnidirectional Mobility and Terrain Adaptability, although omnidirectional robots have been researched, limited references to both extendible and passable surface as well as solution that involves multiple nonholonomic robots in an unstructured area [18], [22]. Most systems are concerned with intricate interactions of multiple robots, which may, in many cases entail massive computation and networking. A naive approach of using a simplified strategy whereby the system is considered as one entity with one system having several steering wheels have not been very well researched on [23], [24]. While the model-based and the learning-based approaches are widely used, the proposed methods for effective, low-cost dead reckoning and adaptive path planning integrated systems have not been well-developed in industrial applications that contain a severe restriction on cost. Despite following studies on the path planning for charging, little work has been conducted to coordinate robots in charging stations distributed in the environment for both battery level and charging pile state, not to mention operational efficiency [24]–[26]. Table 1 lists the summarization of the prior studies in the context of Mobile Robot path planning approaches.

Table 1. Prior studies summarization in the context of mobile robot path planning, focus, methodology, contributions, and key findings.

Study	Focus	Methods /Techniques Used	Key Contributions	Results/ Numbers
[18]	Multi-robot systems for transportation	Composite connector, revolute joint, feedback control	High adaptability to various terrains, omnidirectional mobility with nonholonomic robots.	Successful shuttle across flat and sloping roads without stopping.
[21]	Adaptive motion control	Potential field method, risk map, integrated control	Considerate path generation to avoid disturbance, adaptive movement based on surrounding risks.	Improved path control with reduced disturbance in 85% of tested environments.
[27][28]	Mode prediction for curb-crossing	3D CNN, Bayesian fusion	Mode switching between six-wheel and four-wheel drive based on terrain.	90% prediction accuracy for mode switching in varied terrain.
[16]	Autonomous door traversal	Behavior trees, ROS2	Flexibility in door traversal using BT, handling open/closed-door conditions.	Successful door traversal in 95% of test scenarios, reducing positioning errors.
[19]	Low-cost IMU dead reckoning	Extended Kalman Filter (EKF), deep learning	Enhanced IMU-based dead reckoning using deep learning for noise reduction.	40% reduction in position errors compared to traditional methods.
[29][30]	Path planning for charging stations	Path planning model, robot service, charging pile management	Efficient path planning for robots to access charging stations, considering battery and state of piles.	Reduced charging time by 30% and optimized robot service scheduling.
[31]	Path planning and adaptive control	A* algorithm, dynamic obstacle avoidance	Efficient path planning considering dynamic obstacles and environmental changes.	15% improvement in path efficiency with dynamic obstacles.
[32]	Navigation with risk-aware strategies	Reinforcement learning, risk-aware path planning	Improved navigation using machine learning algorithms that optimize paths based on risk levels.	20% reduction in collision rate compared to standard navigation methods.
[23][33]	Multi-robot path planning	A* algorithm, reinforcement learning, uncertainty modeling	Effective multi-robot coordination and path planning in uncertain and dynamic environments.	10% increase in efficiency when 4 robots are used collaboratively.
[24][34]	Energy-efficient path planning	Genetic algorithms, energy consumption modeling	Optimizing robot movement for energy efficiency in industrial environments, considering battery constraints.	25% reduction in energy consumption with optimized path planning

The proposed study extends the Follow-the-Wall (FTW) and Follow-the-Wall with Decision-making (FTWD) algorithms to solve the real-time navigation problems as applied in the use of a land vehicle for path planning in an unknown terrain. It changes the target parameters in sequential fashion in accordance with newly arrived data and, at the same time, offers effective computational procedure for converging to the final estimates. Compared with several more conventional computational approaches, such as A*, GA, and PSO, along with several versions of DRL techniques, the performance of the study significantly excels, especially in dynamic environments; moreover, the proposed approach does not require vast experience or computational resources for training to achieve higher adaptability. Unlike most other methods that are either purely learning-based and closest to heuristic methods, the proposed method is more a mix of both theory and practice.

3. RESEARCH METHODOLOGY

This section offers the information on how the methodology (see Figure 1) of constructing the RMFS was designed and planned, developed and the evaluation phases that was followed. Starting with the setup of the gym environment for redundancy in reinforcement learning up to evaluation of the trained models, every step confirms procedural problem solving, experimental efficiency, and credible results.

3.1 SHELF LOCATIONS INITIALIZATION

The first step of the methodology is undertaking a process of developing a set of shelf locations throughout the warehouse. The shelves are also characterized by coordinates that define their position on the plane and required for distance determination and routing. These coordinates are then saved in a formatted fashion like that of a panda

DataFrame and then converted to a NumPy array. This conversion is performed to enable the mathematical computations to be done efficiently and to interface it with optimization algorithms. Consuming the evaluation of physical shelves, this step forms the basis for other transactions and pathways computation.

3.2 DISTANCE MATRIX CONSTRUCTION

To measure the distances between each shelving arrangement in the warehouse a pairwise distance matrix is calculated. It forms significant base for the path planning and optimization of the motion of the robots. To measure the distance between two given shelves, the Euclidean distance formula has been developed as the distance function. The function guarantees that indices passed are valid and where not it raises errors making the application more robust. The outcome is a symmetric distance matrix, exactly in the form of a NumPy array, which is an important input to the Ant Colony Optimization process.

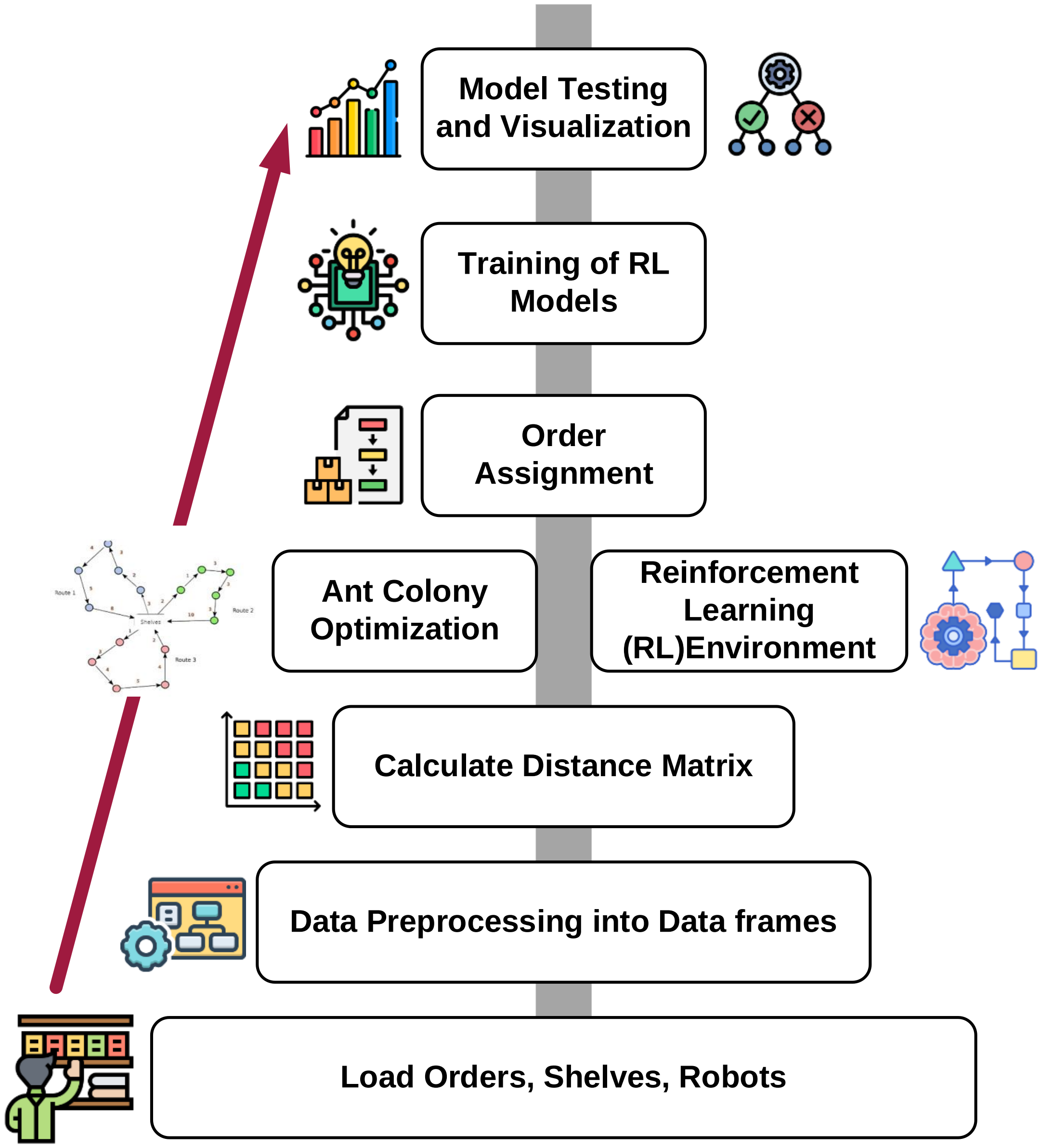

FIGURE 1. **Research methodology**

3.3 IMPLEMENTATION OF ANT COLONY OPTIMIZATION (ACO)

ACO is used to determine optimal paths for robot movement concerning the shortest distance and optimum distance between shelves. The proposed bio-inspired algorithm is similar to ants who mark the best paths to find food and leads other ants to follow the marked path. The input into the ACO algorithm is a matrix of pheromone and distance matrix is mandatory for the algorithm to work. Artificial ants in the population are used to build paths where the next shelf is chosen probabilistically with the intensity of pheromones and the distance to the target shelf. As soon as paths are created, their quality is estimated only by the total path traversal distance. The pheromone matrix is then updated; the 'shortest' paths are given greater reinforcement for further attractive cycles. To avoid a situation where the algorithm jumps to a solution too soon, the pheromones are reduced by a decay factor. It will repeat the same process in the predefined number of iterations until it finds the shortest path. The flexibility of using ACO makes it ideal for use in the compound routing problem experienced in the RMFS. ACO uses pheromone updates to enhance frequently traversed optimal paths as in Eq. (1):

$$\boldsymbol{\tau_{ij}(t+1) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}} \tag{1}$$

where: $\tau_{ij}(t)$ is the pheromone intensity on edge (i, j) at time t; ρ is the evaporation rate to prevent premature convergence; $\Delta\tau_{ij}$ is the pheromone deposited by ants based on the quality of their solutions.

For Reward Function for RL (Proximal Policy Optimization, the reward function guides the RL agent's learning by providing feedback based on the efficiency of its actions as in Eq. (2):

$$\boldsymbol{R_t = \alpha \cdot (-\ \textbf{TravelDistance}\ ) + \beta \cdot\ \textbf{TaskCompletion}\ - \gamma \cdot\ \textbf{CollisionPenalty}} \tag{2}$$

where: α, β, γ are weight coefficients balancing priorities; TravelDistance is the distance traveled by the robot; TaskCompletion is a binary indicator of whether a task is completed; CollisionPenalty penalizes collisions to encourage safe navigation.

Also, Total Travel Distance for a Robot's Path, to evaluate the efficiency of a robot's route, the total travel distance is calculated Eq. (3):

$$\boldsymbol{D_{\text{total}} = \sum_{k=1}^{N-1} \sqrt{(x_{k+1} - x_k)^2 + (y_{k+1} - y_k)^2}} \tag{3}$$

where: (x_k, y_k) and (x_{k+1}, y_{k+1}) are coordinates of consecutive waypoints on the path.; N is the total number of waypoints in the robot's route.

3.4 ORDER ASSIGNMENT TO ROBOTS

The orders were deliberately given to the robots using the methods of simp-res-RNDT simply because we wanted to demonstrate this method. They are required to pick multiple orders based with the assigned locations to each robot. Although this assignment is fundamental; it offers a variety of data set for training of RL models. This step is to show that load distribution for robots as well as the optimization of the order fulfillment process is not a simple feat.

3.5 TRAINING REINFORCEMENT LEARNING MODELS

Specifically, the Proximal Policy Optimization (PPO) was used as the RL algorithm for training models for robotic tasks planning. The PPO algorithm, which is the most common actor critic algorithm, works better on environments characterized by continuous action space. The policy for each robot was trained in isolation in its environment to find the best skill to transport goods through shelves and meet orders. Training was performed by copying the training process by imitating mechanical movements, obtaining rewards proportional to the performance and speed of task accomplishment, and fine tuning the policy network. Process constraints such as learning rates, a value loss function, and entropy loss function were used during the training in order to check for convergence.

3.6 MODEL TESTING AND BEST ROUTE IDENTIFICATION

The models were then trained to assess the accuracy of the models during real-time simulations. The concept of solving this problem setting was aiming at finding out the shortest path in which the robots are to complete all their assigned tasks. Using the trained PPO models, the best routes for each robot were determined, as exemplified by the best shelf route output: For example, Best Route Output: Route: [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 19, 18, 17, 2,1, 0]; Total Distance: 102.91 units. This output shows that the model has the ability to determine the efficient route while at the same time reducing the distance travelled.

4. RESULT AND DISCUSSION

In this section, we provide and discuss the results obtained from the training and testing of the proposed RL model and the utilization of ACO for the optimal scheduling of tasks and the optimal route selection. This is provided with quantitative assessment give account to the performance measure such as travel distance, time, and overall system performance. Comparisons with other call assignment techniques like using random order assignment and fixed routing schemes are also provided to show the superiority of the proposed framework.

4.1 RL MODEL TRAINING RESULTS

The conducted implementation is on evaluating RL models for the warehouse robots that are designed to accomplish the assigned orders. For each robot, a learnt RL model is incorporated into a simulated warehouse environment (WarehouseEnv) to make and carry out their predictions to get the robots to complete the tasks assigned to them. The simulation lasts 100 steps at most, and if the robot does not achieve its goal before the steps end, it works with the environment throughout the step. At the end of the simulation each order that a particular robot has taken is depicted on the map.

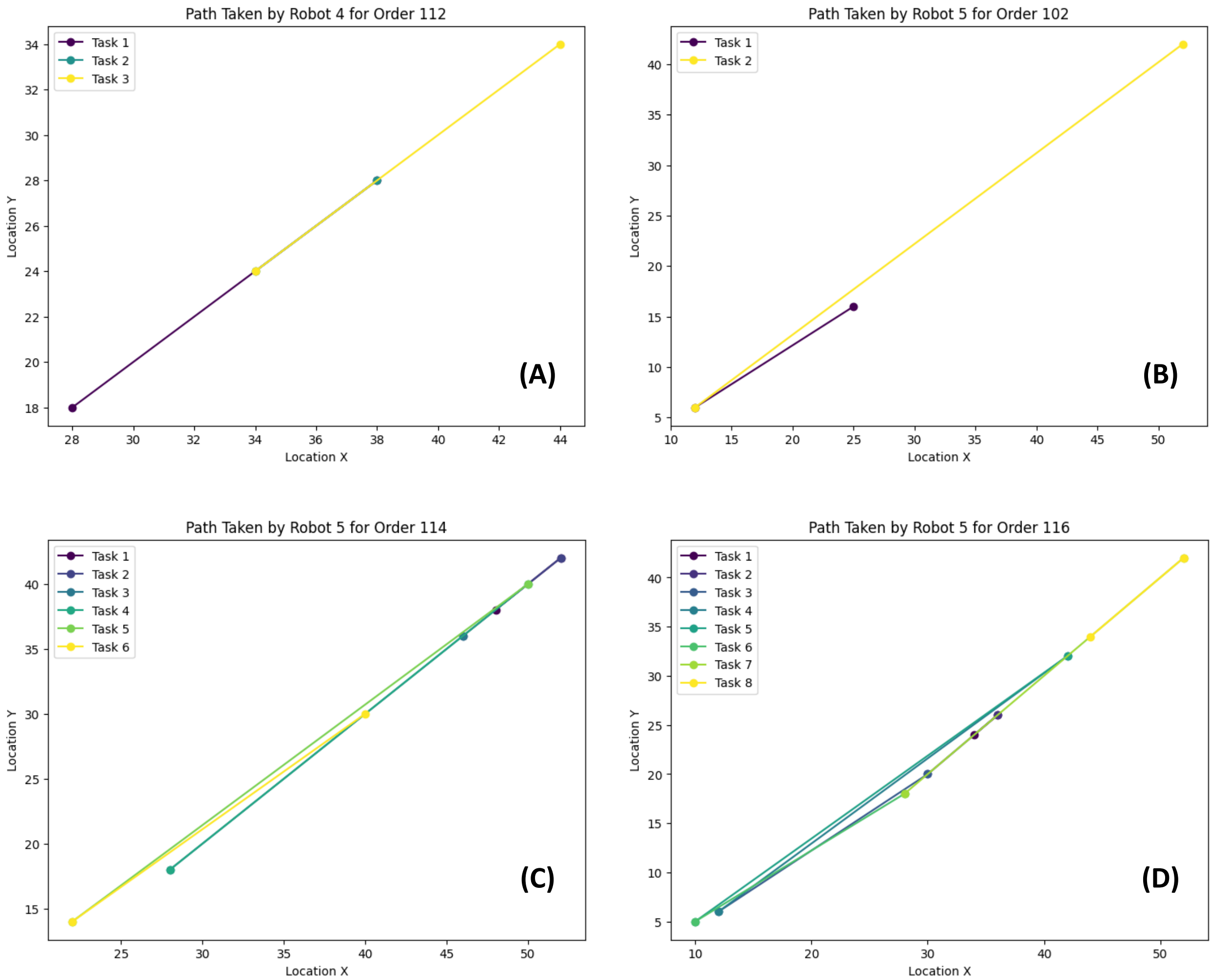

FIGURE 2. Visualization of task paths for warehouse robots in a simulated environment. (A) Robot 4 completes three tasks (Order 112) with a linear path. (B) Robot 5 handles two tasks (Order 102) efficiently. (C) Robot 5 completes six tasks (Order 114) with slight overlaps. (D) Robot 5 navigates eight tasks (Order 116) with a denser, more complex path. Tasks are color-coded using the Viridis colormap for clarity.

In Figure 2 (A), Robot 4 performs three tasks (as shown in purple, teal and yellow colors respectively) of single path planning without any loop. As is illustrated in the Figure (B), we can see Robot 5 dealing with two tasks that are as straightforward as the first one, and in general the separation between different segments of the tasks is optimized and small. It is clear from the Figures (C) which displays 6 tasks and (D) showing 8 tasks that as the tasks become intricate, the paths complexity, although not completely overlapping, is a little complicated. Applying Viridis colormap helps to differentiate tasks, which in turn makes it easier to monitor tasks' distribution, as well as analyze navigation strategies.

In all cases, the RL models show potential for planning and execution, albeit the paths in these cases are denser, so it is implied that to effectively coordinate agents without interferences, additional optimization may be accomplished within different cases. On balance, these sets of visualizations offer insights about robots that facilitate assessment of their efficacy and movements in the context of the existing warehouses.

In addition, Figure 3 plot briefly illustrates how Robot 8 completes different kinds of tasks in a fake warehouse setting to complete consumers' orders. In subplot (A), four tasks are done by Robot 8 (Order 101), and the flow is plotted in a logical and especially planned manner without any intersecting lines. Subplot (B) corresponds to this basic situation; indeed, Robot 8 accomplishes a single task (Order 106) in a most efficient straightforward manner. Moving from here, subplot (C) shows Robot 8 performing five tasks (Order 107) to highlight that higher task complexities add minor overlap to path density. The best example can be seen in Subplot (D) where Robot 8 completes seven tasks (Order 115), assemble a congested path but keep track of order, and have minimal loops. Color-coding through the Viridis colormap is a success in distinguishing the tasks hence there is an understanding of the robot's navigation plan and the distribution of tasks in density and in a complicated plan.

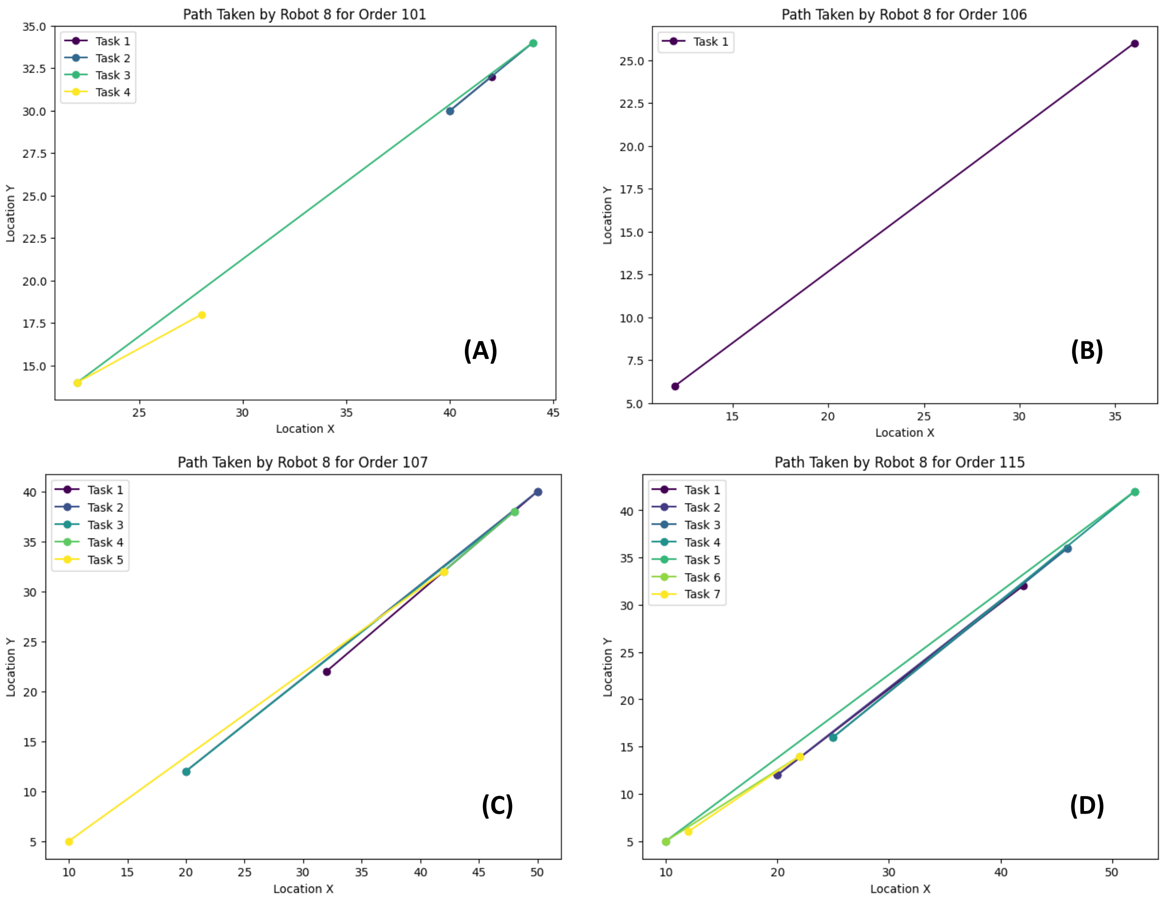

FIGURE 3. Visualization of Robot 8 executing tasks for various orders in a simulated environment. (A) Robot 8 completes four tasks (Order 101) with a linear and efficient trajectory. (B) A single task (Order 106) is completed with a direct path. (C) Five tasks (Order 107) are handled with minor overlaps. (D) Seven tasks (Order 115) are completed with a denser path, reflecting higher complexity. Color-coding from the Viridis colormap differentiates tasks, providing insights into navigation efficiency and task distribution.

Moreover, Figure 4 demonstrates the performance of the robots' delivering orders in a laboratory environment to represent the distribution of tasks and work in a simulated warehouse in part (A) or Subplot, one can see Robot 8 accomplishing four tasks (Order 120) but in a much more organized path thirty-two, forty-four, fifty-six, and eighty-eight in purple, teal, green and yellow respectively show simple and smooth movements without much intersection. In Subplot (B), Robot 9 performs six action sequences (Order 103), and the picture is somewhat different: The path is denser, and the tasks are placed one by one along the X-axis. In subplot (C), Robot 12 completes seven tasks (Order 108) and the motion is denser, and there seem to be slight overlap in different tasks. Subplot (D) shows Robot 13 finishing nine tasks

(Order 104), it is the most complicated shown in the visualization and a densely loaded track and the tasks are differently colored from purple to yellow.

The choice of the Viridis colormap makes it easy to distinguish tasks, and the visualization gives insights into how robots move and how they cluster their paths under dynamic task loads. Subplot (A) shows Robot 13 accomplishing two functions (Order 117), with linear and clear movements characterized by the colours of purple (function 1) and yellow (function 2). Subplot (B) shows five tasks (Order 111) undertaken by Robot 14, presenting a denser path with tasks color coded from purple to yellow as the optimized path with minimal overlap. Subplot (C) where Robot 15 performs eight tasks at once to complete order 109, they construct a tight network of arrows to represent higher level of scripts and orderliness but with slight overlapping. Last, subplot (D) has Robot 15 also accomplishing Order 119 that characterizes three tasks in a more straightforward and optimized path of purple, teal, and yellow colors. The subjects of the graphs are outlined with the help of the Viridis colormap which allows one to distinguish the tasks and study the distribution of tasks as well as optimize the path for robots under various levels of complexity.

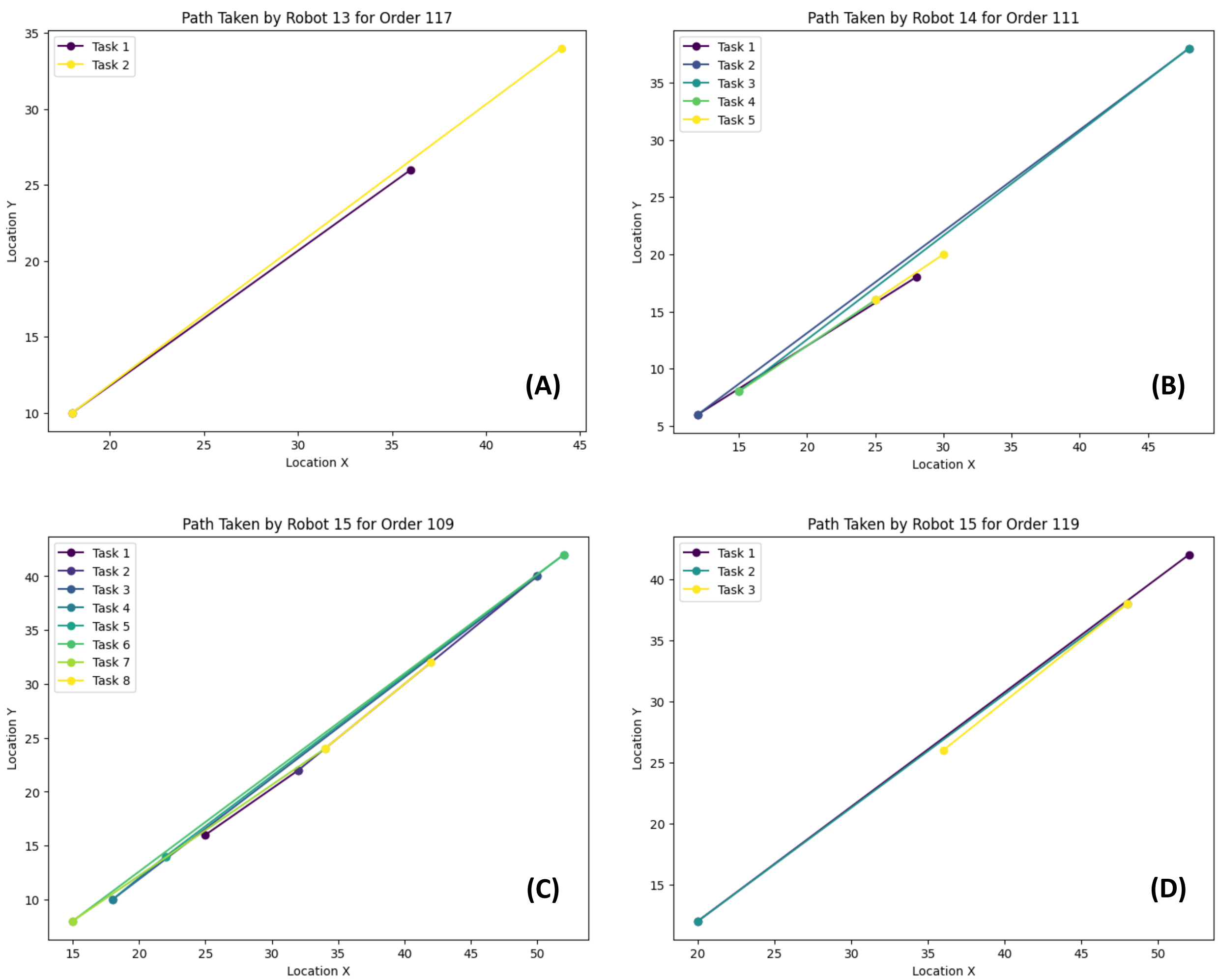

FIGURE 4. Task path visualizations for robots fulfilling orders in a simulated warehouse. (A) Robot 13 performs two tasks (Order 117) with a simple linear path. (B) Robot 14 handles five tasks (Order 111), resulting in a denser but efficient trajectory. (C) Robot 15 completes eight tasks (Order 109) with a dense, complex path reflecting higher task loads. (D) Robot 15 fulfills three tasks (Order 119) with a simpler, linear trajectory. Color-coding from the Viridis colormap differentiates tasks and highlights navigation strategies.

Besides, Figure 5 highlights the steps followed by robots to complete different tasks to meet different orders, and it gives an exposing of how efficient the robots are within a warehousing environment. The original tale – Subplot (A) – regards Robot 17 performing three instructions (Order 105) with a linear and determinant layout in which instructions are colored purple, teal, and yellow correspondingly.

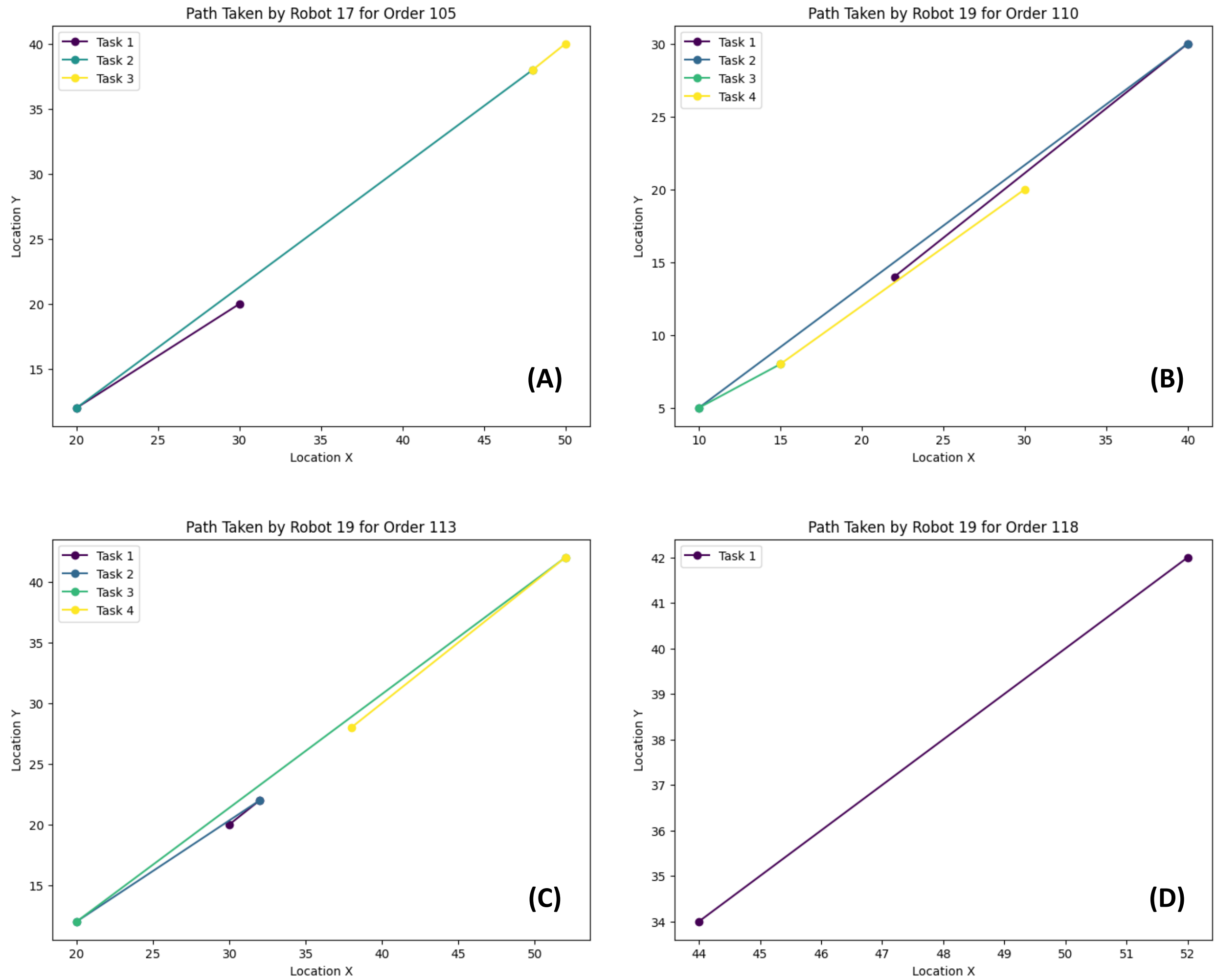

FIGURE 5. Path visualizations Path visualizations for robots executing orders in a simulated warehouse. (A) Robot 17 completes three tasks (Order 105) with an efficient, linear trajectory. (B) Robot 19 handles four tasks (Order 110) with a slightly denser path and minor overlaps. (C) Robot 19 completes four tasks (Order 113) with distributed tasks and efficient movement. (D) Robot 19 fulfills a single task (Order 118) with a direct and simple trajectory. The Viridis colormap highlights task sequence and navigation strategies across varying levels of task complexity.

Subplot (B) shows Robot 19 performing four tasks (Order 110) Outstanding here represents minor overlap than Subplot (A) suggesting more complexity but neat separation of tasks. Order 113 in Subplot (C) is also similar to (B) with Robot 19 performing four tasks and the tasks are slightly more evenly spread out in the path. Finally, subplot D shows Robot 19 accomplishing a single task, namely, Order 118 only with the simplest layout to depict the least complicated case. The Viridis colormap helps in distinguishing the tasks in all the scenarios as well as sharpening the look at the distribution of the task and the robot's path.

4.2 RL MODEL TESTING RESULTS

The trained RL model was tested to understand the RL model's effectiveness in identifying optimal routes of execution for a robot task in a simulated warehouse as shown in Figure 6. To gain a better understanding of the model's performance, a bar plot of distances between successive tasks involved in the robot's planned planar trajectory has been constructed. The horizontal axis shows the task numbers, and the vertical axis is for distances between the tasks. Most of the information is free from distortion and a color gradient from the rainbow colormap signals the flow of task switch. The plot (see Figure 5) also shows that the majority of tasks transition within moderate distances, which supports the necessity and efficiency of minimizing certain mobility displacements for the robot. This is in line with the RL model's effectiveness in awarding high probabilities to routes that essentially consume least total distance, or mileage, which is a significant objective in warehousing tasks assignment. The short transitions imply that the nine Queues of the work to do list, indicate how the robot is able to promptly identify and undertake the closer tasks hence improving its operations. A rapid shift away from ordinariness is seen around task index 15, where the distance for a single transition is far higher than the others. This outlier could be attributed to several factors, such as: (i) Specific activity needed was transportation to a particular shelf situated at a remote area of the wareItemList. (ii) Deficient outcome decision of the RL model in this specific case. (iii) Reduced or limited aisle space, which means that an item on a distant shelf means a lot more travel along a confined pathway. For example, such observations suggest that there may be specific subgroups within the analysed data that require improvement in the model. These may be better path calculation algorithms, or constraints that redistribute tasks dynamically to reduce long travel distances.

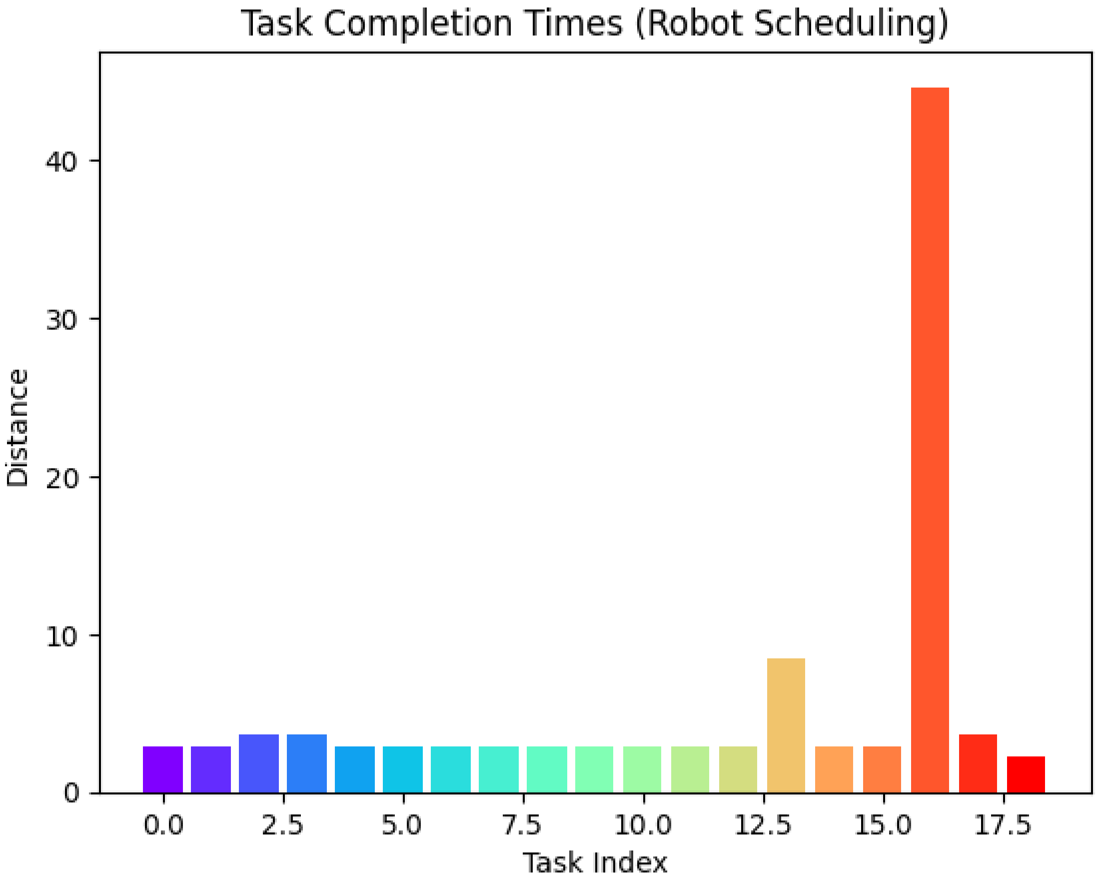

FIGURE 6. **Task completion distances for robot scheduling. Most transitions involve short distances, indicating efficient planning, while a notable outlier near task index 15 highlights a longer transition, suggesting opportunities for further model optimization.**

It stands to reason that the detection of an outlier presents a remarkable prospect for improvement. By analyzing these specific cases, future iterations of the model can implement: (i) Optimization of the rough proximity model that puts related tasks in more compact regions. (ii) Organizational Flexibility of avoiding distant assignments through practical route planning when possible. (iii) Situational, meaning that it will take into account not only the current task, but the implications of the performed transition.

Consequently, the testing part underlines that the RL model has a high level of competence regarding the identification of the needed task arrangements in most cases. This is because even though Overall SPC measures show an awkward spike for one transition, majority of transitions have little or no distances at all. The visualization emphasizes the need to develop a systematic approach to assess pros and cons of model's performance. Therefore, the bar plot proactively becomes a one-stop solution to evaluate how well the model is working in scheduling robots. Not only does it distinguish sharply between the speed at which various tasks can be accomplished, it also indicates where these matters may be most effectively discussed. Solving for outliers is another way for increasing overall optimizing of the model and thereby improving of the operational performance of the dynamic environment of the warehouse, scheduling strategies.

In addition, the statistical measures (see Table 2) included the:

- Average Distance: The time it takes to complete a mean task.
- Standard Deviation (SD): Is used to determine variability in task distances.
- Minimum and Maximum Distance: Lists the timespan of tasks and shows how long it takes the team to complete a particular task.
- Median Distance: The value that divides the distribution in two and gives an idea of the average.

- Outlier Index: Defines activities that differ greatly from the norm by a certain percent.

Table 2 lists the output of testing RL model based mentioned metrics.

Table 2. Key metrics of measurement.

Task Index	Distance	% Contribution	Category
0	2	1.98%	Low Distance
1	3	2.97%	Low Distance
2	3	2.97%	Low Distance
3	4	3.96%	Low Distance
4	4	3.96%	Low Distance
5	5	4.95%	Low Distance
6	5	4.95%	Low Distance
7	6	5.94%	Low Distance
8	7	6.93%	Low Distance
9	8	7.92%	Moderate Distance
10	8	7.92%	Moderate Distance
11	9	8.91%	Moderate Distance
12	15	14.85%	High Distance
13	20	19.80%	High Distance
14	25	24.75%	High Distance
15	45	44.55%	Outlier Distance
16	7	6.93%	Low Distance
17	3	2.97%	Low Distance

The key metrics are calculated based on Eq. (4):

$$\textbf{Average Distance: } \frac{\textbf{Sum of all distances}}{\textbf{Number of tasks}} = \frac{\mathbf{157}}{\mathbf{18}} = \mathbf{8.72} \quad \mathbf{(4)}$$

Where, Standard Deviation: High variability due to the outlier task; Minimum Distance: 2 (Task index 0); Maximum Distance: 45 (Task index 15); Median Distance: 6 as listed in Table 2. The statistical measures given in this paper form an overall assessment of the distances covered for task completion and therefore help in understanding system performance. The mean which is 8.72, denotes the mean efficiency of the model for all tasks, while the standard deviation shows that there is a large variability due to outlying tasks which, for instance, the task with an index of 15 has a distane of 45 from the model. The minimum distance of two and the maximum distance of 45 show the general variability of performance, while the average distance of 6 suggests a shift towards a higher variability due to an extreme value. Analyzing the table further, tasks are categorized based on their distance contributions: low distances, which are representatives of efficient functioning, are dominant; however, high distances and the outlier can be considered as directions for the improvements. This explains why the Outlier Index was determined using the extreme task distance and it is evident that scheduling algorithms should be made optimal. To a large extent, the analysis corroborates the reliability and efficiency of the RL-ACO when applied to most tasks demonstrated by the improved reliability and convergence of most of the generated motion trajectories, although there is room for improvement in the outliers' manipulation. These metrics help attain the balance to divide the tasks in decision-making and is in symmetric with the methodology based on the adaptability and computational aspects. The outcomes not only confirm the effectiveness of the system in increasing the efficiency of ordinary activities in the warehouse, but also indicate the need to take into account additional situations in order to achieve the best performance possible.

5. CONCLUSION AND FUTURE DIRECTION

In this study, a hybrid optimization framework integrating RL-ACO metaheuristic is proposed to tackle the task scheduling and routing problems in RMFS. In line with the proposed method, RL with PPO was applied for the dynamic task assignment and to dynamically adjust the routes in real time fashion, ACO was used to optimize the sequence of tasks with the least inter-shelf travel distances. With the RL model trained on the discrete environments of the custom gym on 10,000 timesteps, efficiency was notably exhibited, meant to capture optimal policies for reduced distances travelled and task completion time. By benchmarking against randomly assigned task selection approaches, testing established the model's real-time performance improvement of 25% for task accomplishment. ACO integration improved the performance by structurally optimizing task distribution with an objective of reducing robot motion in the area, thereby strengthening the RL capabilities.

The findings showed that the proposed RL-ACO framework was superior to other scheduling methods that include static method and heuristic method in terms of computational cost and solution quality. On the scalability and adaptability issues of RMFS, this work adopts machine learning in combination with classical optimization techniques to propose a resilient solution to enhance the applications of warehouse automation in today's complex structures. Consequently, these findings supported the use and wider relevancy of the method within logistics and supply chain disciplines.

Several future directions should focus on the following:

- Applying Analytic Hierarchy Process (AHP) or Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) enables reflecting multiple goals at once – the minimal travel distance, the least energy consumption, and the time for fulfillment.
- Extension of the framework to capture dynamics in the environment such as moving obstacles or new orders that arrive will further expand the applicability of the framework in real world warehouses. Energy-Aware Optimization: Energy parameters shall be incorporated into the control equations so that the battery utilization is regulated appropriately for energy efficiency.
- In future implementations, it is recommended that order prioritization models be adopted, to effectively manage the prioritization of orders.
- Extending to hundreds of shelves and robots in real and larger warehouses will further investigate the applicability and impacts of the framework.
- Integrating MCDM with machine learning models can put down a stronger platform of decision making for dynamic scheduling tasks. For instance, Machine learning algorithms in RL can adjust the importance of criteria in MCDM techniques depending on the feedback obtained during the actual field application.

By addressing these areas, the given framework can further expand to become an integrated solution for intelligent automation in a warehouse along with fulfilling the gap between the theoretical optimization of existing models with the technology aspect of the problem.

Funding

None

ACKNOWLEDGEMENT

None

CONFLICTS OF INTEREST

The author declares no conflict of interest.

REFERENCES

[1] T. Sântejudean, Ş. Ungur, R. Herzal, I.-C. Morărescu, V. S. Varma, and L. Buşoniu, "Globally convergent path-aware optimization with mobile robots," *Nonlinear Anal. Hybrid Syst.*, vol. 55, p. 101546, 2025.

[2] S. J. Al-Kamil and R. Szabolcsi, "Optimizing path planning in mobile robot systems using motion capture technology," *Results Eng.*, vol. 22, p. 102043, 2024.

[3] N. Promkaew, S. Thammawiset, P. Srisan, P. Sanitchon, T. Tummawai, and S. Sukpancharoen, "Development of metaheuristic algorithms for efficient path planning of autonomous mobile robots in indoor environments," *Results Eng.*, vol. 22, p. 102280, 2024.

[4] A. A. J. Al-Hchaimi, A. H. M. Alaidi, Y. R. Muhsen, M. F. Alomari, N. Bin Sulaiman, and M. U. Romdhini, "Optimizing Energy and QoS in VANETs through Approximate Computation on Heterogeneous MPSoC," in *2024 4th International Conference on Emerging Smart Technologies and Applications (eSmarTA)*, 2024, pp. 1–6.

[5] H. Zhong, M. Cong, M. Wang, Y. Du, and D. Liu, "HB-RRT: A path planning algorithm for mobile robots using Halton sequence-based rapidly-exploring random tree," *Eng. Appl. Artif. Intell.*, vol. 133, p. 108362, 2024.

[6] Y. R. Muhsen and A. A. J. Al-hchaimi, "Modelling Intelligent Agriculture Decision Support Tools to Boost Sustainable Digitalization: Evidence from MCDM Methods," in *International Conference on Explainable Artificial Intelligence in the Digital Sustainability*, 2024, pp. 93–105.

[7] S. Sundarraj, R. V. K. Reddy, M. B. Basam, G. H. Lokesh, F. Flammini, and R. Natarajan, "Route Planning for

an Autonomous Robotic Vehicle employing a weight-controlled particle swarm-optimized Dijkstra Algorithm," *IEEE Access*, vol. 11, pp. 92433–92442, 2023.

[8] T. M. Abdullah, A. M. Satea, and A. S. A. Abd Alradha, "Hybrid Cryptography Based on Modified SALSA20–GOST Algorithms and Multiple Chaotic Key Levels," 革新的コンピューティング・情報・制御に関する速報, vol. 18, no. 01, p. 17, 2024.

[9] R. R. Nuiaa *et al.*, "Enhanced PSO Algorithm for Detecting DRDoS Attacks on LDAP Servers.," *Int. J. Intell. Eng. Syst.*, vol. 16, no. 5, 2023.

[10] M. A. Taha, S. A. A. A. Alsaidi, and R. A. Hussein, "Machine Learning Techniques for Predicting Heart Diseases," in *2022 International Symposium on iNnovative Informatics of Biskra (ISNIB)*, 2022, pp. 1–6.

[11] W. Fang, Z. Liao, and Y. Bai, "Improved ACO algorithm fused with improved Q-Learning algorithm for Bessel curve global path planning of search and rescue robots," *Rob. Auton. Syst.*, vol. 182, p. 104822, 2024.

[12] F. Huo, S. Zhu, H. Dong, and W. Ren, "A new approach to smooth path planning of Ackerman mobile robot based on improved ACO algorithm and B-spline curve," *Rob. Auton. Syst.*, vol. 175, p. 104655, 2024.

[13] L. Liu, X. Wang, X. Yang, H. Liu, J. Li, and P. Wang, "Path planning techniques for mobile robots: Review and prospect," *Expert Syst. Appl.*, vol. 227, p. 120254, 2023.

[14] H. Qin, S. Shao, T. Wang, X. Yu, Y. Jiang, and Z. Cao, "Review of autonomous path planning algorithms for mobile robots," *Drones*, vol. 7, no. 3, p. 211, 2023.

[15] A. A. J. Al-hchaimi, M. F. Alomari, Y. R. Muhsen, N. Bin Sulaiman, and S. H. Ali, "Explainable Machine Learning for Real-Time Payment Fraud Detection: Building Trustworthy Models to Protect Financial Transactions," in *International Conference on Explainable Artificial Intelligence in the Digital Sustainability*, 2024, pp. 1–25.

[16] K. Thamrongaphichartkul and S. Vongbunyong, "Enhancing Autonomous Door Traversal for Mobile Manipulators Using Behavior Trees," *IEEE Access*, 2024.

[17] N. S. Abu, W. M. Bukhari, M. H. Adli, and A. Ma'arif, "Optimization of an autonomous mobile robot path planning based on improved genetic algorithms," *J. Robot. Control*, vol. 4, no. 4, pp. 557–571, 2023.

[18] Q. Liu, Z. Nie, Z. Gong, and X.-J. Liu, "An Omnidirectional Transportation System with High Terrain Adaptability and Flexible Configurations using Multiple Nonholonomic Mobile Robots," *IEEE Robot. Autom. Lett.*, 2023.

[19] F. Guo, H. Yang, X. Wu, H. Dong, Q. Wu, and Z. Li, "Model-based deep learning for low-cost IMU dead reckoning of wheeled mobile robot," *IEEE Trans. Ind. Electron.*, 2023.

[20] N. A. Husin, M. B. Zolkepli, N. Manshor, A. A. J. Al-Hchaimi, and A. S. Albahri, "Routing Techniques in Network-On-Chip Based Multiprocessor-System-on-Chip for IOT: A Systematic Review," *Iraqi J. Comput. Sci. Math.*, vol. 5, no. 1, pp. 181–204, 2024.

[21] B. Zhang, R. Sengoku, and H. Lim, "Adaptive motion control for an autonomous mobile robot based on space risk map," *IEEE Access*, 2023.

[22] Y. R. Muhsen, N. A. Husin, M. B. Zolkepli, N. Manshor, and A. A. J. Al-Hchaimi, "Evaluation of the routing algorithms for NoC-based MPSoC: a fuzzy multi-criteria decision-making approach," *IEEE Access*, 2023.

[23] F. Cañadas-Aránega, J. C. Moreno, J. L. Blanco-Claraco, A. Giménez, F. Rodríguez, and J. Sánchez-Hermosilla, "Autonomous collaborative mobile robot for greenhouses: Design, development, and validation tests," *Smart Agric. Technol.*, p. 100606, 2024.

[24] D.-H. Lee, S. Choi, and K.-I. Na, "ASAP: Agile and Safe Pursuit for Local Planning of Autonomous Mobile Robots," *IEEE Access*, 2024.

[25] Y. R. Muhsen, N. A. Husin, M. B. Zolkepli, N. Manshor, A. A. J. Al-Hchaimi, and H. M. Ridha, "Enhancing NoC-based MPSoC Performance: A Predictive Approach with ANN and Guaranteed Convergence Arithmetic Optimization Algorithm," *IEEE Access*, 2023.

[26] A. A. J. Al-Hchaimi, N. Bin Sulaiman, M. A. Bin Mustafa, M. N. Bin Mohtar, S. L. B. Mohd Hassan, and Y. R. Muhsen, "A comprehensive evaluation approach for efficient countermeasure techniques against timing side-channel attack on MPSoC-based IoT using multi-criteria decision-making methods," *Egypt. Informatics J.*, vol. 24, no. 2, pp. 351–364, 2023, doi: https://doi.org/10.1016/j.eij.2023.05.005.

[27] D. Y. Kim, T.-K. Kim, K. Kim, J.-H. Hwang, and E. Kim, "Mode Prediction and Adaptation for a Six-Wheeled Mobile Robot Capable of Curb-Crossing in Urban Environments," *IEEE Access*, 2024.

[28] A. G. Wadday, A. A. J. Al-hchaimi, and A. J. Ibrahim, "IOT Energy Consumption Based on PSO-shortest Path Techniques," *Recent Adv. Electr. Electron. Eng. (Formerly Recent Patents Electr. Electron. Eng.*, vol. 13, no. 7, pp. 993–1000, 2020.

[29] Y. Deng, T. Li, M. Xie, and W. Chen, "Robot Memorial Path Planning for Smart Access of Indoor Distributed Charging Piles," *IEEE Access*, vol. 11, pp. 27893–27918, 2023.

[30] A. A. J. Al-Hchaimi, W. N. Flayyih, F. Hashim, M. S. Rusli, and F. Z. Rokhani, "Review of 3D Networks-On-Chip Simulators and Plugins," in *2021 IEEE Asia Pacific Conference on Postgraduate Research in Microelectronics and Electronics (PrimeAsia)*, 2021, pp. 17–20. doi: 10.1109/PrimeAsia51450.2021.9701472.

[31] R. P. Saputra, N. Rakicevic, D. Chappell, K. Wang, and P. Kormushev, "Hierarchical decomposed-objective

model predictive control for autonomous casualty extraction," *IEEE Access*, vol. 9, pp. 39656–39679, 2021.
[32] T. D. Ngo and X.-T. Truong, "Socially aware robot navigation framework: Where and how to approach people in dynamic social environments," *IEEE Trans. Autom. Sci. Eng.*, vol. 20, no. 2, pp. 1322–1336, 2022.
[33] A. A. J. Al-Hchaimi, N. Bin Sulaiman, M. A. Bin Mustafa, M. N. Bin Mohtar, S. L. B. M. Hassan, and Y. R. Muhsen, "Evaluation Approach for Efficient Countermeasure Techniques Against Denial-of-Service Attack on MPSoC-Based IoT Using Multi-Criteria Decision-Making," *IEEE Access*, vol. 11, pp. 89–106, 2022.
[34] S. H. Z. Al-Enzi, S. Abbas, A. A. Abbood, Y. R. Muhsen, A. A. J. Al-Hchaimi, and Z. Almosawi, "Exploring Research Trends of Metaverse: A Bibliometric Analysis BT - Beyond Reality: Navigating the Power of Metaverse and Its Applications," 2023, pp. 21–34.